Distance-Sensitive Property Testing Lower Bounds

Joshua Brody  
Swarthmore College  
joshua.e.brody@gmail.com

Pooya Hatami  
University of Chicago  
pooya@cs.uchicago.edu

May 11, 2014

Abstract

In this paper, we consider several property testing problems and ask how the query complexity depends on the distance parameter $\epsilon$. We achieve new lower bounds in this setting for the problems of testing whether a function is monotone and testing whether the function has low Fourier degree. For monotonicity testing, our lower bound matches the recent upper bound of Chakrabarty and Seshadhri [12].
1 Introduction

Property Testing is a subfield which seeks to understand what can be learned about a large object given limited access to the object itself. In a typical setup, a property tester is a randomized algorithm that, given a large object as input, must (i) accept with probability $2/3$ if the object has a certain property, and (ii) reject with probability $2/3$ if the object is far from having said property. We charge the tester for the number of queries it makes to the object, and hope that it runs in time significantly sublinear in the size of the object. The query complexity of a property $P$, denoted $Q(P)$, is the minimum number of queries needed to test an object for $P$.

Property testing has been considered for many different properties on many classes of objects, including testing properties of graphs, probability distributions, and functions. See surveys by Goldreich [22, 21] and Ron [27] for comprehensive development. We focus on testing properties of functions. A property of functions $f : D \to R$ is a subset $P \subseteq \{f : D \to R\}$. A query to $f$ is $f(x)$ for some $x \in D$. We say that $f$ is $\epsilon$-far from $P$ if $\Pr[f(x) \neq g(x)] \geq \epsilon$ for all $g \in P$.

Since the seminal work of Rubinfeld and Sudan [30], several properties have been considered, including testing linearity [9], junta testing [18, 5, 6], testing whether a function is isomorphic to a given function [8, 1, 14], and testing whether a function can be computed by various weak models of computation, including size-$s$ decision trees [15] and small-width OBDDs [20, 23, 28].

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 are known for different property testing problems, very few general techniques are known beyond the use of Yao’s minimax lemma. Recently, Blais et. al. [7] came up with a new technique to prove property testing lower bounds, using known lower bounds for randomized communication complexity problems. In particular, they show how to reduce certain communication complexity problems to testing problems, thus showing that communication lower bounds imply lower bounds for property testing. They show that this technique is indeed powerful by applying it on several testing problems and improving on some previous known lower bounds for testing $k$-linearity, $k$-juntas, Fourier degree $\leq k$, etc. It has not been obvious how to come up with lower bounds with dependence on the distance parameter $\epsilon$ using this technique. In this work, we extend the technique of Blais et al. and prove testing lower bounds that depend on $\epsilon$.

1.1 Our Results

Monotinicity Testing: Our first result is a new lower bound for monotonicity testers. A function $f : \{-1,1\}^n \to R$ is monotone if $f(x) \leq f(y)$ whenever $x \leq y$.\footnote{We say that $x \leq y$ if $x_i \leq y_i$ for all $1 \leq i \leq n$.} Let MONO be the property that a function is monotone. Monotonicity testing on various domains has been extensively studied [3, 17, 23, 16, 19, 4, 10]. However, for functions on the boolean hypercube, progress remained elusive until very recently, and gaps between known upper and lower bounds remain. For large range sizes, recent progress has closed this gap considerably: the query complexity lies between $O(n/\epsilon)$ (Chakrabarty and Seshadhri [12]) and $\Omega(\min\{n, |R|^2\})$ (Blais et al. [7]). In this work, we give a new lower bound that completely closes this gap for large $|R|$.

Informal Statement (Theorem 3.5). Testing MONO requires $\Omega\left(\min\{n/\epsilon, |R|^2/\epsilon\}\right)$ queries.

Note that for $|R| = \Omega(\sqrt{n})$, Theorem 3.5 is tight, even for subconstant $\epsilon$. Establishing lower bounds sensitive to the distance parameter is not just a trivial pursuit. Indeed, recent work suggests that for monotonicity testing of boolean functions, understanding how the distance parameter...
affects the query complexity is key to understanding the overall difficulty of monotonicity testing. In another very recent work, Chakrabarty and Seshadhri [11] give a new tester for Boolean functions. Their tester is nonadaptive, has one-sided error, and makes $\tilde{O}(n^{5/6} \epsilon^{-5/3})$ queries. Their result is surprising and somewhat counterintuitive, because their tester is a path tester, and an earlier lower bound of [10] states that $\Omega(n/\epsilon)$ queries are required. This lower bound crucially assumes a linear dependence on $\epsilon$ in the distance parameter. In this way, focusing on bounds sensitive to the distance parameter $\epsilon$ appears key to obtaining tight bounds.

**Testing Fourier Degree:** Our second result is for testing whether a Boolean function $f : \{-1,1\}^n \rightarrow \{-1,1\}$ has low Fourier degree. $f$ has Fourier degree $k$ if all nonzero Fourier coefficients have degree at most $k$. Let $\text{FOURIER}_k$ denote this property.

Upper bounds of $2^{O(k)/\epsilon^2}$ are known for testing whether a Boolean function has Fourier degree $\leq k$ or is $\epsilon$-far from any Boolean function with Fourier degree $k$ [13] [15]. The best lower bound known on this problem has been $\Omega(\min\{k,n-k\})$ [7] [14], which holds for any $\epsilon \leq 1/2$. In this paper we show a lower bound of $\Omega(\min\{k,n-k\}/\sqrt{\epsilon})$ for testing Fourier degree.

**Informal Statement** (Theorem 4.1). Testing $\text{FOURIER}_k$ requires $\Omega(\min\{k,n-k\}/\sqrt{\epsilon})$ queries.

To our knowledge, this is the first lower bound for $\text{FOURIER}_k$ sensitive to $\epsilon$. While the bound is far from tight (in terms of both $k$ and $\epsilon$) even this analysis requires heavier machinery than our lower bound for MONO testers.

Our final result is a distance-sensitive lower bound for approximate Fourier degree testing. In this problem, the goal is to distinguish functions of Fourier degree at most $k$ from those that are far from any function with Fourier degree at most $n - k$.

**Informal Statement** (Theorem 4.4). Any nonadaptive tester for approximate Fourier degree requires $\Omega(1/\epsilon)$ queries.

Chakrabarty et al. [13] gave a lower bound of $\Omega(k)$ for the same problem. Thus, the combined lower bound is $\Omega(k + 1/\epsilon)$.

### 1.2 Outline

In Section 2, we formalize notation and describe the tools we use. Section 3 develops the lower bound for monotonicity testers, and Section 4 gives our lower bounds on Fourier degree.

### 2 Preliminaries and Notation

We use $[n]$ to denote the set $\{1, \ldots, n\}$. The Hamming weight of a string $x \in \{-1,1\}^n$, denoted $|x|$, is the number of $i$ such that $x_i = 1$. We occasionally abuse notation and associate a string $x \in \{-1,1\}^n$ with the corresponding set $\{i : x_i = +1\}$. When working with multiple strings $x_1, \ldots, x_t \in \{-1,1\}^n$, we use double subscripts $x_{ij}$ to denote the $j$-th bit of $x_i$. This paper analyzes univariate functions $f : \{-1,1\}^n \rightarrow \{-1,1\}$, but in our proof of the monotonicity testing lower bound it will often be useful to think of $f$ as a bivariate function. Hence, we abuse notation somewhat and use $f(t,z)$ to refer to a univariate function whose input is the concatenation of $t$ and $z$. 


2.1 Communication Complexity and Property Testing Lower Bounds

In this section we will give a brief introduction to communication complexity, and state known lower bounds for the famous set disjointness problem. The two-party communication model was introduced by Andrew C. Yao [31] in 1979. In this paper, we are chiefly concerned with (public-coin) randomized communication complexity. In a typical communication protocol, there are two players, Alice and Bob. Alice receives an input $x$; Bob receives an input $y$, and they wish to communicate to jointly compute some function $f(x, y)$ of their inputs. In a randomized protocol, Alice and Bob have access to a shared random string $R$ which they can use to select what messages to send. Furthermore, players are allowed a small amount of error. The communication cost of a protocol is the worst-case maximum number of bits sent, taken over all possible inputs and values of $R$. The communication complexity of a function $f$ is the minimal communication cost of a protocol computing $f$.

Definition 2.1. The bounded-error randomized communication complexity of $f$, denoted by $R_\epsilon(f)$ is the minimum cost of a randomized communication protocol that on all inputs $(x, y)$ computes $f$ with success probability $1 - \epsilon$. We set $R(f) := R_{1/3}(f)$.

Of particular interest to us is the set-disjointness problem.

Definition 2.2 (Set Disjointness). Alice and Bob are given $x$ and $y$, $x, y \in \{-1, 1\}^n$, and need jointly to compute

$$\text{DISJ}_n(x, y) = \bigvee_{i=1}^n (x_i \land y_i),$$

where $(a \land b) = 1$ if $a = b = 1$, and $-1$ otherwise.

We drop the subscript when it is clear from context. SET-DISJOINTNESS is perhaps the primary communication problem used to prove lower bounds in other areas of computer science. Obtaining a tight lower bound on its communication complexity was an important problem first solved by Kalyanasundaram and Schnitger [25], then simplified by Razborov [26] and later by Bar-Yossef et al. [2] using information complexity.

Theorem 2.3 ([25, 26, 2]). $R(\text{DISJ}_n) = \Omega(n)$.

We will also be interested in the SPARSE-SET-DISJOINTNESS problem, denoted $k$-DISJ. In this version, we are promised that the inputs each have Hamming weight $k$ ($|x| = |y| = k$) and that $|x \cap y| \leq 1$; i.e., the intersection, if it exists, is unique. This “unique-intersection” promise is implicit in the lower bounds of Kalyanasundaram and Schnitger and of Razborov. The following lower bound from [7] is a straightforward reduction from DISJ.

Lemma 2.4 ([7]). $R(k\text{-DISJ}_n) = \Omega(\min\{2k, n - 2k\})$.

This tightens a folklore $\Omega(k)$ lower bound, which holds as long as $k \leq n/3$. Note that this is essentially tight, as shown in a clever protocol by Håstad and Wigderson [24].

Theorem 2.5 ([24]). $R(k\text{-DISJ}_n) = O(k)$.

It will be convenient for our applications to consider the following direct-sum variants of DISJ.

Definition 2.6. The functions OR-DISJ$^\ell_m$, OR-$k$-DISJ$^\ell_m : \{-1, 1\}^\ell \times \{-1, 1\}^{\ell m} \to \{-1, 1\}$ are defined as

$$\text{OR-DISJ}_m^\ell(x_1, \ldots, x_\ell, y_1, \ldots, y_\ell) := \bigvee_{i=1}^\ell \text{DISJ}_m(x_i, y_i).$$
Lemma 2.7. The following direct-sum properties hold:

1. \( R(\text{OR-disj}_m^\ell) = \Omega(\ell m) \).

2. \( R(\text{OR-k-disj}_m^\ell) = \Omega(\min\{\ell k, \ell(m - 2k)\}) \).

Proof. We include the proof for \( \text{OR-k-disj}_m^\ell \); the proof for \( \text{OR-disj}_m^\ell \) is similar. Let \( x_1, \ldots, x_\ell \) and \( y_1, \ldots, y_\ell \) be inputs to \( \text{OR-k-disj}_m^\ell \), and let \( x \) and \( y \) be the concatenation of \( x_1, \ldots, x_\ell \) and \( y_1, \ldots, y_\ell \) respectively. Note that \( x \) and \( y \) each have Hamming weight \( \ell k \), and furthermore \( x \) and \( y \) intersect iff \( x_i \) and \( y_i \) intersect for some \( i \). Thus, we have

\[
\text{OR-k-disj}_m^\ell(x_1, \ldots, x_\ell, y_1, \ldots, y_\ell) = \bigvee_{i=1}^{\ell} \text{k-disj}_m^\ell(x_i, y_i) = \bigvee_{i=1}^{\ell} \bigvee_{m, j=1} x_{ij} \land y_{ij} = (\ell k)\text{-DISJ}_{\ell m}(x, y).
\]

Thus, any protocol for \( \text{OR-k-disj}_m^\ell \) is also a protocol for \( (\ell k)\text{-DISJ}_{\ell m} \). From Lemma 2.4 it follows that \( R(\text{OR-k-disj}_m^\ell) = R((\ell k)\text{-DISJ}_{\ell m}) = \Omega(\min\{2\ell k, \ell m - 2\ell k\}) \).

Next, we summarize the terminology and main lemma for proving testing lower bounds via communication complexity, reformulating the notation in a way convenient for our results. For more details, consult the work of Blais et al. [7].

Definition 2.8 (Combining Operator). A combining operator \( \psi \) takes as input two functions \( f, g : \{-1, 1\}^n \to \{-1, +1\} \) and returns a function \( h : \{-1, 1\}^n \to R \).

A combining operator is simple if for all \( f, g \) and for all \( x \in \{-1, 1\}^n \), \( h(x) \) can be computed given only \( x \) and the queries \( f(x) \) and \( g(x) \).

For a property \( P \) and combining operator \( \psi \), let \( C^P_\psi \) denote the communication problem where Alice and Bob receive \( f \) and \( g \) respectively and wish to determine if \( \psi(f, g) \) has property \( P \) or is far from having \( P \). The connection between property testing and this communication game is captured in the following lemma.

Lemma 2.9 (Main Reduction Lemma ([7], Lemma 2.4)). For any simple combining operator \( \psi \) and any property \( P \), we have

\[ R(C^P_\psi) \leq 2Q(P). \]

2.2 Fourier Analysis of Boolean Functions

Our result on testing Fourier degree uses Fourier Analysis. We briefly present basic definitions and results here. Consider the \( 2^n \)-dimensional vector space of all functions \( f : \{-1, 1\}^n \to R \). An inner product on this space can be defined as follows

\[ \langle f, g \rangle = \frac{1}{2^n} \sum_{x \in \{-1, 1\}^n} f(x)g(x) = \mathbb{E}_x[f \cdot g], \]

where the latter expectation is taken uniformly over all \( x \in \{-1, 1\}^n \). This defines the \( l_2 \)-norm

\[ ||f||_2 = \sqrt{\langle f, f \rangle} = \sqrt{\mathbb{E}_x[f^2]}. \]
Definition 2.1. For $S \subseteq [n]$, the character $\chi_S : \{-1,1\}^n \rightarrow \{-1,1\}$ is defined as

$$\chi_S(x) = \prod_{i \in S} x_i.$$

The set of characters forms an orthonormal basis for the inner product space. Hence, every function $f : \{-1,1\}^n \rightarrow \mathbb{R}$ can be written uniquely as

$$f = \sum_S \langle f, \chi_S \rangle \cdot \chi_S.$$  

The above equation is referred to as the Fourier expansion of $f$, and the Fourier coefficient of $f$ corresponding to set $S$ is defined as

$$\hat{f}(S) = \langle f, \chi_S \rangle.$$

Parseval’s Identity states that

$$\|f\|_2^2 = \sum_{S \subseteq [n]} \hat{f}(S)^2. \tag{1}$$

Definition 2.2. The Fourier degree of a Boolean function $f : \{-1,1\}^n \rightarrow \{-1,1\}$ is equal to the maximum $k > 0$ such that there exists $S \subseteq [n]$, $|S| = k$, for which $\hat{f}(S) \neq 0$.

3 Testing Monotonicity

Before getting into the full proof of our lower bound, we give a high-level description. We use the technique of Blais et al. and reduce from OR-DISJ. The function $h : \{-1,1\}^n \rightarrow \mathbb{R}$ we create uses the first $\log(1/\epsilon)$ bits as an index $t$. Then, the value of $h(t, z)$ is essentially the function used by Blais et al. in their $\Omega(n)$ lower bound for testing monotonicity, evaluated on the $t$-th pair of inputs for OR-DISJ. If $x_t$ and $y_t$ intersect in, say, the $i$-th bit, then $h$ will violate monotonicity on edges in the $i$-th direction, but only when the value of the index is $t$. This happens with probability $\epsilon$ (taken over a random input to $h$). Therefore, the overall distance to monotonicity is $\Omega(\epsilon)$.

There is one final complication. We need to ensure that monotonicity is not spuriously violated in the $i$-th direction when $i$ is one of the coordinates that defines $t$. To manage this, we increase the value of $h$ by $\Omega(|t|)$. This ensures that monotonicity is not violated in a direction corresponding to one of the bits that make up $t$, no matter what happens with the rest of the input to $h$.

Modulo a few technical complications and adjustment of variables, this completes the proof that testing monotonicity for functions with high range size requires $\Omega(n/\epsilon)$ queries. For smaller range sizes, we adopt some range reduction tricks from Blais et al. [7] One technical complication arises, but conceptually, the reductions are the same.

Before proving the lower bound for MONO, we define some functions and demonstrate some related basic facts that will be useful for our proof. For $i \in \mathbb{N}$ and $x \in \{-1,1\}^*$, let $x^i+$ and $x^i-$ denote the strings obtained from $x$ by setting $x_i = +1$ and $x_i = -1$ respectively.

The following fact says that when flipping the $i$th bit of a string, the value of the character changes only when $i \in S$.

Fact 3.1. For any $S \subseteq [n]$ and $x \in \{-1,1\}^n$, $\chi_S(x^{i+}) = \chi_S(x^{i-})$ if and only if $i \notin S$.

In Section 3.1 we will need to manipulate sums of certain character functions. The corollary below easily follows from Fact 3.1.

Corollary 3.2. For any $S, T \subseteq [n], i \in [n]$, and $x \in \{-1,1\}^n$, we have
Claim 3.6. directly from the following claims: Claim 3.7. There exists a constant $\Omega(\min \frac{\ell}{\epsilon})$ requires Theorem 3.5.

Next, we generalize character functions to operate on lists of strings. Recall that we associate strings $x_j \in \{-1, 1\}^m$ with the corresponding set $\{r : x_{jr} = 1\}$.

Definition 3.3. Let $x = (x_1, \ldots, x_t) \in \{-1, 1\}^{tm}$. The function $f_x : \{-1, 1\}^\log(\ell) + m \rightarrow \{-1, 1\}$ is defined as

$$f_x(j, z) := \chi_{x_j}(z).$$

We adopt the convention of mapping $x \rightarrow f_x$ and $y \rightarrow g_y$. The following fact is an analogue of Corollary 3.2 and specifies what happens to the sum of (generalized) character functions when we flip an input bit from −1 to 1.

Fact 3.4. For any $x, y \in \{-1, 1\}^{\ell m}, t \in \{-1, 1\}^{\log(\ell)}$, and any $z \in \{-1, 1\}^m$, the following hold.

1. For all $i \in [\log(\ell)]$, we have

$$f_x(t^{i+}, z) + g_y(t^{i+}, z) - f_x(t^{i-}, z) - g_y(t^{i-}, z) \geq -4.$$

2. For all $i \in [m]$, we have

$$f_x(t, z^{i+}) + g_y(t, z^{i+}) - f_x(t, z^{i-}) - g_y(t, z^{i-}) = -4,$$

if $i \in x_t \cap y_t$ and $f_x(t, z^{i+}) = g_y(t, z^{i+}) = -1$. Otherwise, we have

$$f_x(t, z^{i+}) + g_y(t, z^{i+}) - f_x(t, z^{i+}) - g_y(t, z^{i+}) \geq -2.$$

3.1 The Monotonicity Lower Bound

Theorem 3.5. Fix $\epsilon$ such that $2^{-n/2} < \epsilon < 1/10$. Then, testing $h : \{0, 1\}^n \rightarrow R$ for monotonicity requires $\Omega(\min\{n/\epsilon, |R|^2/\epsilon\})$ queries.

Define $\ell := 1/8\epsilon$, and assume $\log(\ell)$ is an integer. Let $m := n - \log(\ell)$. Theorem 3.5 follows directly from the following claims:

Claim 3.6. If $|R| = \Omega(n)$ then testing $h : \{0, 1\}^n \rightarrow R$ for monotonicity requires $\Omega(n/\epsilon)$ queries.

Claim 3.7. There exists a constant $c$ such that if $|R| \geq c\sqrt{m}$ then testing $h : \{0, 1\}^n \rightarrow R$ for monotonicity requires $\Omega(n/\epsilon)$ queries.

Claim 3.8. If $R = O(\sqrt{m})$ then testing $h : \{0, 1\}^n \rightarrow R$ for monotonicity requires $\Omega(|R|^2/\epsilon)$ queries.

In the rest of this section we prove the above claims.

Proof of Claim 3.7. We reduce from OR-DIST$^m_n$. Let $\psi$ be the combining operator that, given functions $f, g : \{-1, 1\}^n \rightarrow \{-1, 1\}$, returns the function $h : \{-1, 1\}^n \rightarrow \mathbb{Z}$ defined by

$$h(t, z) := 4|t| + 2|z| + f(t, z) + g(t, z).$$
Define $C^\text{MONO}_\psi$ to be the communication game where Alice and Bob are given functions $f$ and $g$ respectively and must test whether $h$ is monotone or $\epsilon$-far from monotone. By Lemma 2.9 and Lemma 2.7 we have

$$2Q(\text{MONO}) \geq R(C^\text{MONO}_\psi) \quad \text{and} \quad R(\text{OR-DISJ}_m^\ell) = \Omega(\ell m) = \Omega(n/\epsilon) .$$

We complete the proof by showing that $R(C^\text{MONO}_\psi) \geq R(\text{OR-DISJ}_m^\ell)$.  Given OR-DISJ$_m^\ell$ inputs $x = (x_1, \ldots, x_\ell)$ and $y = (y_1, \ldots, y_\ell)$, Alice and Bob construct functions $f_x$ and $g_y$ respectively. We claim that when $x_t$ and $y_t$ are disjoint for all $t \in [\ell]$, $h$ is monotone, and conversely when $x_t$ and $y_t$ intersect for some $t$, $h$ is $\epsilon$-far from monotone. To see this, fix an index $t$ and a string $z$.

Our first task is to show that monotonicity cannot be violated in a direction corresponding to the index $t$. Suppose that $i \in [\log(\ell)]$, and consider $h(t^+, z) - h(t^-, z)$. We have

$$h(t^+, z) - h(t^-, z) = 4|t^+| + 2|z| + f_x(t^+, z) + g_y(t^+, z) - 4|t^-| - 2|z| - f_x(t^-, z) - g_y(t^-, z) = 4 + f_x(t^+, z) + g_y(t^+, z) - f_x(t^-, z) - g_y(t^-, z) \geq 0 ,$$

where the inequality holds by Fact 3.4. This shows that monotonicity is never violated in the direction of an index coordinate.

Our next task is to show that if $i$ is a coordinate corresponding to $z$, then monotonicity is violated if and only if $x_t$ and $y_t$ intersect. Formally, let $i \in [m]$, and consider $h(t, z^+) - h(t, z^-)$. We have

$$h(t, z^+) - h(t, z^-) = 4|t| + 2|z^+| + f_x(t, z^+) + g_y(t, z^+) - 4|t| - 2|z^-| - f_x(t, z^-) - g_y(t, z^-) = 2 + f_x(t, z^+) + g_y(t, z^+) - f_x(t, z^-) - g_y(t, z^-) .$$

By Fact 3.4, this is negative if and only if $i \in x_t \cap y_t$ and $f_x(t, z^+) = g_y(t, z^-) = -1$. Furthermore, this latter condition happens with probability $1/4$, where the probability is over a random $z$.

Together, these cases show that $h$ is monotone when $x_t \cap y_t = \emptyset$ for all $t$. On the other hand, if $x_t$ and $y_t$ intersect for some $t$, fix $i \in x_t \cap y_t$. For a random $z$, $h(t, z^+) - h(t, z^-) < 0$ with probability $1/4$; thus, a $(1/4)$-fraction of edges in the $i$th direction are violated when the index equals $t$, and a random index equals $t$ with probability $1/\ell = 8\epsilon$. To attain a monotone function, we must change at least one endpoint of each the violating edges in the $i$th coordinate. Therefore, we must change at least $2^n \cdot (1/4) \cdot 1/\ell \cdot (1/2) = \epsilon \cdot 2^n$ points to arrive at a monotone function. Hence, $h$ is $\epsilon$-far from monotone.

**Proof of Claim 3.7** Fix a constant $c'$ such that $|z| - m/2 \leq c' \sqrt{m}$ with probability $15/16$. Using $h$ as defined in Claim 3.6, we define $h' : \{-1, 1\}^n \to R$ by

$$h'(t, z) = \begin{cases} +\infty & \text{if } |z| \geq m/2 + c' \sqrt{m} , \\ -\infty & \text{if } |z| \leq m/2 - c' \sqrt{m} , \\ h(t, z) & \text{otherwise.} \end{cases}$$

Note that $|R| = c\sqrt{m}$ for some constant $c$ depending on $c'$. Recall that a $(1/4)$-fraction of edges in the $i$-th direction are violated when $i \in x_t \cap y_t$ and the index equals $t$, and therefore we need to
change at least \((1/8)\) of \(h(t, z)\) to get a monotone function. By our choice of \(c', h(t, z) \neq h'(t, z)\) with probability at most \(1/16\) (over a random \(z\) and any fixed \(t\)).

It follows that for each \(t\) where \(x_t\) and \(y_t\) intersect, we need to change \(h'(t, z)\) for at least a \((1/16)\)-fraction of \(z\) to get a monotone function. Overall, the distance to monotonicity is at least \(1/(16\ell)\) = \(\epsilon/2\). Rescaling \(\epsilon\) completes the proof. \(\square\)

**Proof of Claim 3.8** We use a claim from \([7]\).

**Claim 3.9** ([7], Claim 4.2). If there is a \(q\)-query algorithm that tests \(f : \{0, 1\}^n \rightarrow R\) for monotonicity when \(|R| = O(\sqrt{n})\) then there is a \(q\)-query algorithm that tests \(g : \{0, 1\}^m \rightarrow R\) for monotonicity where \(m = \Omega(|R|^2)\).

Claim 3.7 shows that testing \(g\) requires \(\Omega(m/\epsilon)\) queries. Thus, testing \(f\) requires \(\Omega(m/\epsilon) = \Omega(|R|^2/\epsilon)\) queries. \(\square\)

4 Testing Fourier Degree

In this section we present our lower bound for testing whether a given function has low Fourier degree. For convenience, in the context of Fourier analysis we consider the Boolean function to be of the form \(f : \{-1, 1\}^n \rightarrow \{-1, 1\}\).

Diakonikolas et al. \([15]\) considered the problem of testing whether a Boolean function \(f\) has Fourier degree at most \(k\). They proved a general lower bound of \(\Omega(\log k)\), and a lower bound of \(\tilde{\Omega}(\sqrt{k})\) for the non-adaptive tester with any \(\epsilon \leq 1/2\). They also present an algorithm with \(\tilde{O}(2^{6k}/\epsilon^2)\) query complexity for this problem. Chakraborty et al. \([14]\) and later Blais et al. \([7]\) improved the lower bound to \(\Omega(\min\{k, n-k\})\), for any \(0 \leq \epsilon \leq 1/2\). In this section we show how to use the communication complexity technique to prove a lower bound of \(\Omega(\min\{k, n-k\}/\sqrt{\epsilon})\) on testing whether a Boolean function is of Fourier degree at most \(k\).

**Theorem 4.1.** Let \(\epsilon \geq 2^{-k-1}\). Testing whether a Boolean function \(f : \{-1, 1\}^n \rightarrow \{-1, 1\}\) has Fourier degree \(\leq k\) or \(\epsilon\)-far from any Boolean function with Fourier degree \(\leq k+1\), requires \(\Omega(\min\{k, n-k\}/\sqrt{\epsilon})\) queries.

**Remark.** Note that the case when \(\epsilon \leq 2^{-k-1}\) is at least as hard as when \(\epsilon = 2^{-k-1}\) and thus an exponential lower bound of \(\Omega(2^k)\) for such \(\epsilon\) follows immediately from the above theorem.

In order to prove our lower bound we first need to construct Boolean functions with certain properties.

4.1 Our Constructions of Functions

In this section we give a method how to construct functions which are of Fourier degree \(\leq k\) and functions that are far from having Fourier degree at most \(k\).

**Definition 4.2** (Functions Defined by Index Selectors). Let \(\ell > 0\) be an integer. Call a map

\[
C^{\ell} : \{-1, 1\}^\ell \rightarrow \mathcal{P}([n]\{1, \ldots, \ell\})
\]

to be an index selector, where \(\mathcal{P}(\cdot)\) denotes the power set. Given an index selector \(C^{\ell}\) one can define a Boolean function \(f^{\ell} : \{-1, 1\}^n \rightarrow \{-1, 1\}\) as following

\[
f^{\ell}(x_1, \ldots, x_n) = \chi_{C(x_1, \ldots, x_\ell)}(x_1, \ldots, x_n),
\]

where \(\chi_A(x_1, \ldots, x_n) := \prod_{i \in A} x_i\) for \(A \subseteq [n]\). We call \(f^{\ell}\) an index selector function.
In the next three propositions we show how the cardinalities of the sets \( \mathcal{C}^\ell(a_1, \ldots, a_\ell) \) can lead \( f^\ell \) to be of low Fourier degree, or to be far from any Boolean function with low Fourier degree. For the sake of simplicity we will often avoid the superscript \( \ell \) in \( \mathcal{C}^\ell(a_1, \ldots, a_\ell) \) and simply write \( \mathcal{C}(a_1, \ldots, a_\ell) \).

**Proposition 4.1.** The Boolean function \( f^\ell : \{-1, 1\}^n \to \{-1, 1\} \) as described above, is of Fourier degree \( m + \ell \) if

\[
\forall (a_1, \ldots, a_\ell) \in \{-1, 1\}^\ell, |\mathcal{C}(a_1, \ldots, a_\ell)| \leq m.
\]

**Proof.** We have to prove that for any \( (a_1, \ldots, a_\ell) \in \{-1, 1\}^\ell \), \( |\mathcal{C}(a_1, \ldots, a_\ell)| \leq m \).

The last equality follows from the fact that

\[
\sum_{x_{\ell+1}, \ldots, x_n \in \{-1, 1\}} \chi_{\mathcal{C}(x_1, \ldots, x_\ell)}(x_1, \ldots, x_\ell) \cdot \chi_{\mathcal{C}(x_1, \ldots, x_n)}(x_1, \ldots, x_n) = 0,
\]

since

\[
\exists i \in S : i \notin \mathcal{C}(x_1, \ldots, x_\ell) \cup \{1, \ldots, \ell\},
\]

because \( |S| \geq m + \ell + 1 \).

**Proposition 4.2.** The Boolean function \( f^\ell \) is \( 1/2^{\ell+1} \)-far from any Boolean function of Fourier degree \( \leq m - 1 \) if for only one \((b_1, \ldots, b_\ell) \in \{-1, 1\}^\ell\), \( |\mathcal{C}(b_1, \ldots, b_\ell)| \geq m \), and

\[
\forall (a_1, \ldots, a_\ell) \neq (b_1, \ldots, b_\ell) : |\mathcal{C}(a_1, \ldots, a_\ell)| \leq m - 1.
\]

**Proof.** We first prove that for any \( U \subseteq \{1, \ldots, \ell\} \), the Fourier coefficient of \( |f^\ell| \) at \( S := U \cup \mathcal{C}(b_1, \ldots, b_\ell) \) is equal to \( 1/2^\ell \).

\[
\hat{f}^\ell(S) = \langle f^\ell, \chi_{U \cup \mathcal{C}(b_1, \ldots, b_\ell)} \rangle = 2^{-n} \sum_{x \in \{-1, 1\}^n} \chi_{x_\ell}(x) \cdot \chi_{U \cup \mathcal{C}(b_1, \ldots, b_\ell)}(x)
\]

\[
= 2^{-n} \sum_{x_\ell \in \{-1, 1\}^\ell} \left( \prod_{i \in U} x_i \right) \sum_{x_{\ell+1}, \ldots, x_n} \chi_{x_\ell}(x) \chi_{\mathcal{C}(b_1, \ldots, b_\ell)}(x)
\]

\[
= 2^{-\ell} \prod_{i \in U} b_i + 2^{-n} \sum_{x_\ell \neq (b_1, \ldots, b_\ell)} \left( \prod_{i \in U} x_i \right) \sum_{x_{\ell+1}, \ldots, x_n} \chi_{x_\ell}(x) \cdot \chi_{\mathcal{C}(b_1, \ldots, b_\ell)}(x)
\]

\[
= 2^{-\ell} \prod_{i \in U} b_i.
\]

The last equality follows from the fact that if \((a_1, \ldots, a_\ell) \neq (b_1, \ldots, b_\ell)\) then \( |\mathcal{C}(b_1, \ldots, b_\ell)| > |\mathcal{C}(a_1, \ldots, a_\ell)| \), and

\[
\sum_{x_{\ell+1}, \ldots, x_n} \chi_{x_\ell}(x) \cdot \chi_{\mathcal{C}(b_1, \ldots, b_\ell)}(x) = 0.
\]
Let \( g : \{-1,1\}^n \to \{-1,1\} \) be a Boolean function with Fourier degree at most \( m - 1 \), namely we can write
\[
g(x) = \sum_{S \subseteq [n]} \hat{g}(S) \chi_S(x).
\]
Notice that the distance between the two functions \( f \) and \( g \) with range \( \{-1,1\} \) can be formulated as \( \frac{1}{2} ||f - g||_2^2 = \frac{1}{2} \mathbb{E}[(f - g)^2] \). Finally Parseval Identity implies that
\[
||f - g||_2^2 = \sum_{S \subseteq [n]} |S| \leq m - 1 (\hat{f}(S) - \hat{g}(S))^2 + \sum_{S \subseteq [n]} |S| \geq m \hat{f}(S)^2 \geq \sum_{S \subseteq [n]} \sum_{U \subseteq [\ell]} (2^{-\ell} \prod_{i \in U} b_i)^2 = 2^{-\ell}.
\]

\[\square\]

**Proposition 4.3.** The Boolean function \( f^\ell \) is \( 1/2^{2\ell+1} \)-far from any Boolean function of Fourier degree \( \leq m + \ell - 1 \) if for only one \((b_1, \ldots, b_\ell) \in \{-1,1\}^\ell, |C(b_1, \ldots, b_\ell)| \geq m \), and
\[
\forall (a_1, \ldots, a_\ell) \neq (b_1, \ldots, b_\ell) : |C(a_1, \ldots, a_\ell)| \leq m - 1.
\]

**Proof.** The proof is similar to the proof of Proposition 4.2 with the difference that we only use the fact that \( \hat{f}((\ell] \cup C(b_1, \ldots, b_\ell)) = 2^{-\ell} \), and thus \( f^\ell \) is \( 2^{-2\ell-1} \) far from any function with Fourier degree \( m + \ell - 1 \).

\[\square\]

### 4.2 Proof of Theorem 4.1

In this section, we show how to use the communication complexity technique to prove a lower bound of \( \Omega(\min\{k,n-k\}/\sqrt{\epsilon}) \) on testing whether a Boolean function is of Fourier degree at most \( k \).

**Proof of Theorem 4.1.** Let \( \ell \) be the largest integer such that \( \epsilon < 2^{-2\ell-1} \). Notice that \( \epsilon \geq 2^{-k-1} \) implies \( \ell \leq \frac{k}{2} \). Also, let \( m := n - \ell \). Assume that \( n-k \) is even, and let \( d := (n-k)/2 \). We prove that \( \Omega(2^\ell \cdot \min\{k,m-k\}) \) queries are required to test whether a Boolean function has Fourier degree \( \leq k \) or is \( \epsilon \)-far from any Boolean function with degree \( \leq k + 1 \) by reducing from OR-

Let \( \psi \) be the combining operator that, given functions \( f, g : \{-1,1\}^n \to \{-1,1\} \), returns the function \( h \) defined as \( h := f \cdot g \cdot \chi_{[n]\setminus[\ell]} \). Define \( C^\text{FOURIER}_{k,\psi} \) to be the communication game where Alice and Bob are given functions \( f, g \) respectively and must test whether \( h \) has Fourier degree at most \( k \) or is \( \epsilon \)-far from all functions of Fourier degree at most \( k \). By Lemma 2.9 and Lemma 2.7 we have
\[
2Q(C^\text{FOURIER}_{k,\psi}) \geq R(C^\text{FOURIER}_{k,\psi}) \quad \text{and} \quad R(\text{OR-\text{DISJ}}_{m}^{2^\ell}) = \Omega(2^\ell \min\{d,m-d\}) = \Omega \left( \frac{\min\{k,n-k\}}{\sqrt{\epsilon}} \right).
\]

We complete the proof by showing that \( R(C^\text{FOURIER}_{k,\psi}) \geq R(\text{OR-\text{DISJ}}_{m}^{2^\ell}) \).

Given inputs \( x = (x_1, \ldots, x_2^\ell) \) and \( y = (y_1, \ldots, y_2^\ell) \) to \( \text{OR-\text{DISJ}}_{m}^{2^\ell} \), Alice and Bob create functions \( f, g \) by using index selectors. Specifically, Alice defines the index selector \( C_x \) in the natural way—for any \( t \in \{-1,1\}^\ell \), set \( C_x(t) := x_t \). Bob similarly builds an index selector \( C_y \) from \( y \). Let \( f_x \) and \( g_y \) be the functions defined by \( C_x \) and \( C_y \), as described in Definition 4.2. We claim
that the combined function $h$ is also an index selector function. To see this, fix $a \in \{-1,1\}^\ell$, and notice that $h(t, z) = \chi_{R_1}(z) \cdot \chi_{R_2}(z) \cdot \chi_{[m]}(z) = \chi_T(z)$, where $T = [m] \setminus (x_t \Delta y_t)$. Thus, $h$ describes an index selector function for the index selector $D_{xy}$ defined by $D_{xy}(t) := [m] \setminus (x_t \Delta y_t)$.

The index selector $D_{xy}$ is highly structured. In particular,

$$|D_{xy}(t)| = \begin{cases} 
  k - \ell & \text{if } x_t \cap y_t = \emptyset \\
  k + 2 - \ell & \text{if } |x_t \cap y_t| = 1
\end{cases}$$

By Proposition 4.1 and Proposition 4.3 we have that $h$ has Fourier degree $k$ if $x_t \cap y_t = \emptyset$ for all $t$, and that $h$ is $\epsilon$-far from any Boolean function of degree $k + 1$ when $x_t, y_t$ intersect for a unique $t$. Thus, an answer to $C^\psi_{\text{FOURIER}k}$ gives an answer to $\text{OR-}d\text{-DISJ}^k_{m}$.

4.3 Approximate Fourier degree testing

Chakraborty et al. [13] proved that testing whether a Boolean function has Fourier degree at most $k$ or it is far from any Boolean function with Fourier degree $n - \Theta(1)$ requires $\Omega(k)$ queries. Here we prove an $\Omega(1/\epsilon)$ lower bound for the non-adaptive tester, using Yao’s minimax principle. For this we introduce two distributions $D_+$ and $D_-$ on Boolean functions where $D_+$ is a distribution supported only on Boolean functions with Fourier degree $\leq k$ and $D_-$ is only supported on Boolean functions that are $\epsilon$-far from any Boolean function with Fourier degree $\leq n - 2k$. This combined with Chakraborty et al.’s result gives an $\Omega(k + \frac{1}{\epsilon})$ lower bound for non-adaptively approximate testing the Fourier degree.

**Theorem 4.4.** Let $\epsilon \geq 2^{-(k/2 - 1)}$. Non-adaptively Testing whether a Boolean function $f : \{-1,1\}^n \to \{-1,1\}$ has Fourier degree $\leq k$ or it is $\epsilon$-far from any Boolean function with Fourier degree $\leq n - k$ requires $\Omega(\frac{k}{\epsilon})$ queries.

**Proof.** Let $\ell$ be the largest integer such that $\epsilon < 2^{-\ell - 1}$. We prove that $\Omega(2^\ell)$ queries are required in order to non-adaptively test whether a Boolean function has Fourier degree $\leq k$ or is $\epsilon$-far from any Boolean function with degree $\leq n - k$. Notice that since $\epsilon \geq 1/2^{\frac{k}{2} - 1}$ thus $\ell \leq \frac{k}{2} - 1$.

Let $D_+$ be the distribution obtained by the following random process. Let $C^\ell(a_1, ..., a_\ell)$, for every $(a_1, ..., a_\ell) \in \{-1, +1\}^\ell$, be a uniformly chosen subset of size $k/2$ of $\{\ell + 1, ..., n\}$. Let $f^\ell$ defined by $C^\ell$ as described in Section 4.1. Proposition 4.1 immediately implies that $f^\ell$ has Fourier degree $\leq k$.

Let $D_-$ be the distribution obtained by the following process. We choose $(b_1, ..., b_\ell) \in \{-1, 1\}^\ell$ uniformly at random and choose $C^\ell(b_1, ..., b_\ell)$ to be a previously fixed subset of cardinality $n - k + 1$ of $\{\ell + 1, ..., n\}$. Also for any $(a_1, ..., a_\ell) \in \{-1, 1\}^\ell$, where $(a_1, ..., a_\ell) \neq (b_1, ..., b_\ell)$, we choose $C^\ell(a_1, ..., a_\ell)$ uniformly to be a subset of cardinality $k/2$ of $\{\ell + 1, ..., n\}$. Finally, construct $f^\ell$ according to $C^\ell$. Proposition 4.2 immediately implies that $f^\ell$ is $2^{-\ell - 1}$-far from any Boolean function with Fourier degree $\leq n - k$.

Let our final distribution $D$ be that with probability $1/2$ we draw $f^\ell$ from $D_+$ and with probability $1/2$ we draw $f^\ell$ from $D_-$. Now by Yao’s minimax principle if we prove that any deterministic algorithm with less than $2^\ell/6$ queries makes a mistake with constant probability, this implies that the original testing problem with constant probability of error requires $\frac{2^\ell}{6} = \Omega(\frac{k}{\epsilon})$ queries.

For any deterministic set of queries to the function on $d$ inputs $x^1, ..., x^{d'}$, with $d \leq \frac{2^\ell}{6}$,

$$|\{(a_1, ..., a_\ell) | \exists 1 \leq i \leq d \exists j | x^i_j = (a_1, ..., a_\ell)\}| \leq d \leq \frac{2^\ell}{6}.$$
Therefore the measure of the set of functions from support of $D_-$ for which the deterministic tester has not yet queried any input from the high degree subcube is at least

$$\frac{1}{2} \cdot \frac{2^\ell - 2^\ell/6}{2^\ell} = \frac{5}{12} \geq \frac{1}{3}.$$

Since the same is true for $D_+$, the deterministic tester will make an error with probability at least $\frac{1}{3}$.

\[\square\]

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] Ziv Bar-Yossef, T. S. Jayram, Ravi Kumar, and D. Sivakumar. An information statistics approach to data stream and communication complexity. In Proc. 43rd Annual IEEE Symposium on Foundations of Computer Science, pages 209–218, 2002.

[3] 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.

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

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

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

[7] Eric Blais, Joshua Brody, and Kevin Matulef. Property testing lower bounds via communication complexity. Computational Complexity, 2012.

[8] 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.

[9] 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.

[10] 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.

[11] Deeparnab Chakrabarty and C. Seshadhri. An $o(n)$ monotonicity tester for boolean functions over the hypercube. In Proc. 45th Annual ACM Symposium on the Theory of Computing, 2013.

[12] Deeparnab Chakrabarty and C. Seshadhri. Optimal bounds for monotonicity and lipschitz testing over hypercubes and hypergrids. In Proc. 45th Annual ACM Symposium on the Theory of Computing, 2013.

[13] Sourav Chakraborty, David García-Soriano, and Arie Matsliah. Efficient sample extractors for juntas with applications. In ICALP (1), pages 545–556, 2011.

[14] 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.

[15] 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.
[16] 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.

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

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

[19] 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.

[20] 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.

[21] Oded Goldreich. A brief introduction to property testing. In Studies in Complexity and Cryptography, pages 465–469. 2011.

[22] Oded Goldreich. Introduction to testing graph properties. In Studies in Complexity and Cryptography, pages 470–506. 2011.

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

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

[25] Bala Kalyanasundaram and Georg Schnitger. The probabilistic communication complexity of set intersection. SIAM J. Disc. Math., 5(4):547–557, 1992.

[26] Alexander Razborov. On the distributional complexity of disjointness. In Proc. 17th International Colloquium on Automata, Languages and Programming, pages 249–253, 1990.

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

[28] Dana Ron and Gilad Tsur. Testing computability by width two obdds. In 13th International Workshop on Randomization and Computation (RANDOM), 2009.

[29] Dana Ron and Gilad Tsur. Testing computability by width-2 obdds where the variable order is unknown. In 7th International Conference on Algorithms and Complexity, 2010.

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

[31] Andrew C. Yao. Some complexity questions related to distributive computing. In Proc. 11th Annual ACM Symposium on the Theory of Computing, pages 209–213, 1979.