Electronic Colloquium on Computational Complexity, Report No. 30 (2012)

Optimal bounds for monotonicity and Lipschitz testing over the hypercube

Deeparnab Chakrabarty
dechakr@microsoft.com
Microsoft Research, Bangalore

C. Seshadhri
scomand@sandia.gov
Sandia National Labs, Livermore*

Abstract

The problem of monotonicity testing of the boolean hypercube is a classic well-studied, yet unsolved question in property testing. We are given query access to $f : \{0,1\}^n \mapsto R$ (for some ordered range R). The boolean hypercube $\mathcal{B}^n$ has a natural partial order, denoted by $\prec$ (defined by the product of coordinate-wise ordering). A function is *monotone* if all pairs $x \prec y$ in $\mathcal{B}^n$, $f(x) \leq f(y)$. The distance to monotonicity, ε_f, is the minimum fraction of values of f that need to be changed to make f monotone. It is known that the edge tester using $O(n \log |R|/\varepsilon)$ samples can distinguish a monotone function from one where $\varepsilon_f > \varepsilon$. On the other hand, the best lower bound for monotonicity testing is $\min(|R|^2, n)$. This leaves a quadratic gap in our knowledge, since $|R|$ can be 2^n.

We prove that the edge tester only requires $O(n/\varepsilon)$ samples (regardless of R), resolving this question. Our technique is quite general, and we get optimal edge testers for the Lipschitz property. We prove a very general theorem showing that edge testers work for a class of "bounded-derivative" properties, which contains both monotonicity and Lipschitz.

1 Introduction

Given a function $f : \{0,1\}^n \mapsto R$, what can we learn about the properties of f without reading all of f? The field of property testing [RS96, GGR98] formalizes this question by dealing with relaxed decision problems. Conventionally, the *distance* $\Delta(f,g)$ between two functions f and g is defined to be the fraction of domains points where f and g differ. Formally, $\Delta(f,g) = |\{x | f(x) \neq g(x)\}|/2^n$, the Hamming distance between these functions. Consider some property $\mathcal{P}$ of functions over the boolean hypercube, which is some subset of the functions over the boolean hypercube with range R. We define the distance between f and $\mathcal{P}$, denoted by $\varepsilon_{f,\mathcal{P}}$ to be $\min_{g \in \mathcal{P}} \Delta(f,g)$. Essentially, this is the minimum "amount" by which f must be changed to have the property $\mathcal{P}$. Given a parameter $\varepsilon \in (0,1)$ and query access to the function f, the classic property testing question is to design a randomized algorithm for the following problem. If $\varepsilon_{f,\mathcal{P}} = 0$ (meaning $f \in \mathcal{P}$) "accept", and if $\varepsilon_{f,\mathcal{P}} > \varepsilon$ "reject". If $\varepsilon_{f,\mathcal{P}} \in (0, \varepsilon)$, then any answer is allowed. The aim is to get a running time of $\mathsf{poly}(n/\varepsilon)$ (or at least, $\mathsf{poly}(n)$ for constant ϵ). Such an algorithm is called a *property tester for* $\mathcal{P}$.

An important property studied in this framework is that of *monotonicity*. Let x_i be the ith coordinate of a point $x \in \{0,1\}^n$. We define a partial order: $x \preceq y$, if $\forall i \in [n], x_i \leq y_i$. Assume that the range R is totally ordered, so we can think of $R \subseteq \mathbb{R}$. A function is *monotone* if it satisfies $\forall x \prec y, f(x) \leq f(y)$. An intriguing feature of this property is that it *cannot* be tested with a

*Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

ISSN 1433-8092

constant number of samples. Monotonicity is often studied in the context of different ranges. The tester of choice is usually the *edge tester*. This simply samples a uniform random point x, flips a uniform random coordinate to get y, and checks if $f(x), f(y)$ violate monotonicity. How many pairs do we need to sample to get a bonafide monotonicity tester? When the range is boolean, Goldreich et al [GGL+00] prove that $O(n/\varepsilon)$ samples suffice. For an arbitrary range R, Dodis et al [DGL+99] shows that $O(n \log |R|/\varepsilon)$ samples are enough for a tester. In the worst case, $R = 2^n$, so the running time is $O(n^2/\varepsilon)$. In a recent breakthrough, Blais, Brody, and Matulef [BBM11] prove that $\Omega(\min(n, |R|^2))$ samples are required to test monotonicity. (This holds even for adaptive, two-sided testers.) The main question is to give an optimal bound for monotonicity testing over the hypercube. This is widely regarded as one of the outstanding open problems in property testing, which has stood unsolved for almost a decade.

We resolve this question here, and show the edge tester is truly optimal (when $|R| \geq \sqrt{n}$).

Theorem 1. *Let $f : \{0,1\}^n \mapsto \mathbb{R}$. The edge tester with query complexity $O(n/\varepsilon)$ is a valid property tester for monotonicity.*

Our techniques are quite general and also apply to problem of testing *Lipschitz functions*, introduced by Jha and Raskhodnikova [JR11]. A function $f : \{0,1\}^n \mapsto R$ is called *c-Lipschitz* if for all $x, y, |f(x) - f(y)| \leq c\|x - y\|_1$. The edge tester for this property queries an adjacent x and y and checks if $|f(x) - f(y)| \leq c$. It was shown that for the range $R = \delta\mathbb{Z}$, the edge tester runs in $O(n^2/(\delta\varepsilon))$[1]. Our methods provide an optimal bound for Lipschitz-testing for all ranges. A lower bound in [JR11] shows that this cannot be improved.

Theorem 2. *Let $f : \{0,1\}^n \mapsto \mathbb{R}$. The edge tester with query complexity $O(n/\varepsilon)$ is a valid property tester for c-Lipschitz.*

When functions of the form $f : \{0,1\}^n \mapsto R$ are considered, distance is considered only with respect to functions having this range. We are considering a larger range $\mathbb{R}$, so how does that affect our theorems? Note that the distance to a property only becomes *smaller* on considering a larger range. The edge tester is insensitive to the range of the function, so our theorems are most certainly valid when we measure distance with respect to the smaller range R.

1.1 Edge testers for the generalized Lipschitz property

We state our main technical result in this section. Let $\mathcal{B}^n := \{0,1\}^n$ and $\mathsf{Hyp}^n = (\mathcal{B}^n, H)$ be the undirected graph where $H = \{(x,y) : ||x - y||_1 = 1\}$. Given $x, y \in \mathsf{B}$, we say $x \preceq y$ if $x_i \leq y_i$ for all $1 \leq i \leq n$. We will be working with functions $f : \mathsf{B} \mapsto \mathbb{R}$ defined on the n-dimensional hypercube.

Definition 1. *Let $\beta > \alpha$ be in $\mathbb{R}$. A function $f : \mathcal{B}^n \mapsto \mathbb{R}$ is* (α, β)-Lipschitz *if: $\forall (x,y) \in H$, where $x \prec y$, $\alpha \leq f(y) - f(x) \leq \beta$. The set (or alternately, the* property*) of (α, β)-Lipschitz functions is denoted by $\mathcal{L}_{\alpha,\beta}$.*

This is a general class of properties: monotonicity is precisely the $(0, \infty)$-Lipschitz property, and the usual definition of c-Lipschitz is that of $(-c, +c)$-Lipschitz. (If the reader is uncomfortable with the choice of β as ∞, β can be thought of as much larger than any value in f.) We now give a laundry list of fairly standard property testing definitions that make it convenient to express our main result.

Definition 2.
- *The* distance to being (α, β)-Lipschitz *is* $\min_{g \in \mathcal{L}_{\alpha,\beta}} \Delta(f, g)$. *We use $\varepsilon_{\alpha,\beta,f}$ to denote this quantity.*

[1] One can get a better bound of $O(nD/(\delta\varepsilon))$, where D is a bound on range of values that f takes.

- *A* violated edge *for* $\mathcal{L}_{\alpha,\beta}$ *is an edge* $(x,y) \in H$ $(x \prec y)$ *such that* $f(y) - f(x) \notin [\alpha, \beta]$.
- *The* $\mathcal{L}_{\alpha,\beta}$ edge tester *queries (the endpoints of) a uniform random edge of* Hyp^n *and rejects* f *if the edge is violated.*

Our main result can now be succinctly stated as follows.

Theorem 3. *Let* $f : \{0,1\}^n \mapsto \mathbb{R}$. *There are at least* $\varepsilon_{\alpha,\beta,f} 2^{n-1}$ *violated edges for* $\mathcal{L}_{\alpha,\beta}$.

A standard corollary of this theorem gives an optimal property tester for $\mathcal{L}_{\alpha,\beta}$. We provide a proof for completeness. Theorem 1 and Theorem 2 follow directly by setting the property $\mathcal{L}_{\alpha,\beta}$ appropriately.

Corollary 4. *The* $\mathcal{L}_{\alpha,\beta}$ *edge tester independently invoked* $O(n/\varepsilon)$ *times will always accept a function* $f \in \mathcal{L}_{\alpha,\beta}$ *and, with probability* $> 2/3$, *will reject* f *such that* $\varepsilon_{\alpha,\beta,f} > \varepsilon$.

Proof. The $\mathcal{L}_{\alpha,\beta}$ edge tester never rejects a function in $\mathcal{L}_{\alpha,\beta}$. Suppose $\varepsilon_{\alpha,\beta,f} > \varepsilon$. By Theorem 3, there are at least $\varepsilon 2^n$ violated edges for $\mathcal{L}_{\alpha,\beta}$ in f. The total number of edges in Hyp^n is $n2^{n-1}$, so the fraction of violated edges is at least $\varepsilon/2n$. The probability that one invocation of the edge tester rejects is at least $\varepsilon/2n$. The probability that $4n/\varepsilon$ independent invocations of the edge tester does *not* reject is at most $(1 - \varepsilon/2n)^{4n/\varepsilon} < 1/3$. □

1.2 Related work

The field of property testing has a long history, starting from the seminal papers of Rubinfeld and Sudan [RS96] and Goldreich, Goldwasser, and Ron [GGR98]. The property of linearity is arguably the first property over the hypercube studied in this context [BLR93]. We refer the reader to surveys [Fis01, Ron09]. We go into detail here on monotonicity results only over the boolean hypercube. The study of monotonicity was initiated by Goldreich et al [GGL+00], who proved that edge tester only needs $O(n/\varepsilon)$ samples to test monotonicity on the boolean range. Using a clever range reduction technique, Dodis et al [DGL+99] showed that this tester worked for general range R, albeit with a running time of $O(n(\log R)/\varepsilon)$.

Since then, it has been open whether any tester can beat this bound, and specifically whether the edge tester can be shown to be better. This problem has been fairly well studied [GGL+00, DGL+99, LR01, FLN+02, Bha, BCGSM10, BBM11]. The best lower bound for the boolean range is $\Omega(\sqrt{n}/\varepsilon)$ proven by Fischer et al[FLN+02]. This was improved to $\Omega(n/\varepsilon)$ by Briët et al [BCGSM10] for large ranges, when the tester is non-adaptive with one-side error. Blais, Brody, and Matulef [BBM11] prove an $\Omega(\min(n, |R|^2))$ using communication complexity arguments for general testers. Other posets (like the total order) have been studied in these as well as many other papers [EKK+00, AC06, Fis04, HK08, PRR06, ACCL06, BRW05, BGJ+09].

Jha and Raskhodnikova [JR11] initiate the study of Lipschitz properties from a tester perspective. They showed that the edge tester works when the domain is discrete, giving a bound of $O(n^2/(\delta\varepsilon))$ (in the worst case). A lower bound of $\Omega(n)$, using communication complexity ideas, was also shown.

1.3 Main ideas

The general challenge of property testing is to relate the tester behavior to the distance to the property. Consider monotonicity. We want to show that a large distance to monotonicity implies many violated edges. Most current analyses of the edge tester go via what we could call the *contrapositive route*. If there are few violated edges in f, then we wish to show the distance to monotonicity is small. Since there are few violated edges, let us actually modify f and make it monotone. If we can charge our changes to the violated edges, then we have a proof. There is an inherently "constructive" viewpoint to this: the proof specifies a method to convert non-monotone functions to monotone ones.

Implementing this becomes difficult as the range becomes large, and bounds degrade with R. A non-constructive approach may give more power, but how do we get a handle on the distance? The *violation graph* provides a method. The violation graph has an edge between any pair of comparable hypercube vertices (x, y) $(x \prec y)$ if $f(x) > f(y)$. The size of the minimum vertex cover is exactly $\varepsilon_f 2^n$, and maximal matchings in the graph have size at least $\varepsilon_f 2^{n-1}$ [DGL+99]. Can this imply that there are many violated edges? Lehman and Ron [LR01] look at this view. The monotonicity testing problem is reduced to very interesting routing problems on the hypercube. Briët et al [BCGSM10] prove lower bounds for these routing problems showing that this approach has fundamental limitations. In these reductions, the function values are altogether ignored, so some of the structure of monotonicity is given up for the sake of clean combinatorial problems.

Our proof is intimately connected with the actual function values and is non-constructive. The key insight is to consider a *weighted* violation graph. The weight of edge (x, y) $(x \prec y)$ is $f(x) - f(y)$. This can be thought of as a measure of the magnitude of this violation. We now look at the maximum weighted matching M in the violation graph. Naturally, this is maximal, so we know it has at least $\varepsilon_f 2^{n-1}$ edges.

Assume that all function values are unique. Each of the pairs in M will be uniquely identified with a violated edge (not quite, but it is not far from the truth). Consider a pair in M (x, y) that "crosses" the r-th dimension. This means that x and y differ in their rth bit. Let us try to find a violated edge in the rth dimension associated with it. This will be done by trying to increase the matching weight by replacing pairs. Since this is not possible, we will gain structural information about these pairs.

Now for the magic. Let y' be obtained by flipping the rth bit of y_r (set $x \prec y$, so $y'_r = 0$). We have $x \prec y'$. If $f(y') > f(y)$, we are done. Suppose not, so $f(y') < f(y)$ and $f(x) - f(y') > f(x) - f(y)$. If we could match (x, y') instead of (x, y), the matching weight would go up! Because M has maximum weight, y' itself must be present in a matched pair (y', y''). Furthermore, we can show that $y' \succ y''$. If not, then $f(y') - f(y'') > 0$ (since (y', y'') is a violation). So $f(x) - f(y'') > [f(x) - f(y)] + [f(y') - f(y'')]$. We can replace (x, y) and (y', y'') in the matching by the single pair (x, y'') and increase the weight, contradicting the maximality of M. Observe how the maximality of M allows us to make many arguments about these pairs and incident edges.

So we have pairs (x, y), (y', y''), where $y'_r = y''_r = 0$ (and $y'' \prec y'$). We now flip the rth bit of y'' to get z, where $z_r = 1$. We can show that $z \prec y$. (The interested reader is recommended to prove this, to get a feel for the argument.) So we could try to match (x, y') and (y'', z), and gain some more properties of these pairs. And so the argument proceeds. We keep alternately following pairs in M and edges crossing the rth dimension, and we show that eventually a violated edge is encountered. Furthermore, starting from a different pair of M, we prove that a different violated edge is reached.

But what about the generalized Lipschitz property? It turns out the basic ideas still work, despite the fact that monotonicity has an inherent directionality, making for easier proofs. The weights of the violation graph measure how much pairs violates the (α, β)-Lipschitz condition. The charging of pairs of M crossing the rth-dimension to violated edges goes along the similar lines, with a lot more notation. Many arguments that were somewhat trivial or easy for monotonicity require more work now. These also involve some monotonous case analyses, thereby showing us that monotonicity is a fundamental aspect of these proofs.

1.4 Some preliminaries

We will focus on a fixed dimension size n and a fixed property $\mathcal{L}_{\alpha,\beta}$. For ease of notation, we will drop all these sub/superscripts. We use B for $\mathcal{B}^n$, Hyp for Hyp^n, and H for the edges of Hyp. It will be convenient to assume that $\alpha + \beta \geq 0$. This is no loss of generality. The (α, β)-Lipschitz property asserts that for every $(x, y) \in H$, $x \prec y$, $f(y) - f(x) \in [\alpha, \beta]$. This is equivalent to $f(x) - f(y) \in [-\beta, -\alpha]$. This means that $\mathcal{L}_{\alpha,\beta}$ is the same as $\mathcal{L}_{-\beta,-\alpha}$ on the "reversed" version of Hyp (where edges are directed in the opposite direction).

2 A pseudo-distance for $\mathcal{L}_{\alpha,\beta}$

We begin by defining a weighted graph $\mathsf{G} = (\mathsf{B}, E)$. This is just a bi-directional version of Hyp, so E contains directed edges of the form (x, y), where $\|x - y\|_1 = 1$. The length of edge (x, y) is gives as follows. If $x \prec y$, the length is $-\alpha$. If $x \succ y$, the length is β.

Definition 3. *The pseudo-distance* $\mathsf{d}(x, y)$ *between* $x, y \in \mathsf{B}$ *is the shortest path length from* x *to* y *in* G.

Even though edges have negative lengths, we will shortly show that this is well-defined. This function is asymmetric, meaning that $\mathsf{d}(x, y)$ and $\mathsf{d}(y, x)$ are possibly different. Furthermore, $\mathsf{d}(x, y)$ can be negative, so this does not truly qualify to be a distance (in the usual parlance of metrics). Nonetheless, $\mathsf{d}(x, y)$ has many useful properties, which can be proven by expressing it in a more convenient form. Given any $x, y \in \mathsf{B}$, we define $\mathsf{hcd}(x, y)$ to be the $z \in \mathsf{B}$ maximizing $||z||_1$ such that $x \succ z$ and $y \succ z$. That is, z is the highest common descendant of x and y. Note that if $x \succ y$ then $\mathsf{hcd}(x, y) = y$.

Claim 5. *For any* $x, y \in \mathsf{B}$, $\mathsf{d}(x, y) = \beta||x - \mathsf{hcd}(x, y)||_1 - \alpha||y - \mathsf{hcd}(x, y)||_1$.

Proof. Let us partition the coordinate set $[n] = A \sqcup B \sqcup C$ with the following property. For all $i \in A$, $x_i = 1$, $y_i = 0$. For all $i \in B$, $x_i = 0$, $y_i = 1$, and for all $i \in C$, $x_i = y_i$. Any path in G can be thought of as sequence of coordinate increments and decrements. Any path from x to y must finally increment all coordinates in A, decrement all coordinates in B, and preserve coordinates in C. Furthermore, any increments adds $-\alpha$ to the path length, and a decrement adds β.

Fix a path, and let I_i and D_i denote the number of increments and decrements in dimension. For $i \in A$, $D_i = I_i + 1$, for $i \in B$, $I_i = D_i + 1$, and for $i \in C$, $I_i = D_i$. The path length is given by

$$\begin{aligned} & \sum_{i \in A} (\beta D_i - \alpha I_i) + \sum_{i \in B} (\beta D_i - \alpha I_i) + \sum_{i \in C} (\beta D_i - \alpha I_i) \\ = \; & \sum_{i \in A} [\beta + I_i(\beta - \alpha)] + \sum_{i \in B} [-\alpha + D_i(\beta - \alpha)] + \sum_{i \in C} I_i(\beta - \alpha) \\ \geq \; & \beta|A| - \alpha|B| \end{aligned}$$

For the inequality, we use the fact that $\beta \geq \alpha$. Hence, $\mathsf{d}(x, y) \geq \beta|A| - \alpha|B|$. Let $z = \mathsf{hcd}(x, y)$. Note that for $i \in A \cup B$, $z_i = 0$, and for $i \in C$, $z_i = x_i = y_i$. Consider the path from x that only decrements to reach z, and then only increments to reach y. The length of this path is exactly $\beta|A| - \alpha|B|$. Furthermore, $|A| = \|x - z\|_1$ and $|B| = \|y - z\|_1$. This completes the proof. $\square$

It is instructive to keep in mind what this distance translates to in the case of monotonicity and Lipschitz. In the case of monotonicity (when $\alpha = 0, \beta = \infty$), we get $\mathsf{d}(x, y) = \infty$ unless $x \prec y$

in which case $\mathsf{d}(x,y) = 0$. In the case of Lipschitz, the distance $d(x,y)$ is precisely the Hamming distance $\mathsf{d}(x,y) = \mathsf{hcd}(x,y)$.

The next two claims establish some properties of the pseudo-distance.

Claim 6. *(Linearity) If $x \succ z \succ y$ or $x \prec z \prec y$, $\mathsf{d}(x,y) = \mathsf{d}(x,z) + \mathsf{d}(z,y)$.*
(Triangle Inequality) For any $x,y,z \in \mathsf{B}$, $\mathsf{d}(x,y) \le \mathsf{d}(x,z) + \mathsf{d}(z,y)$.
(Projection)Let $x,y \in \mathsf{B}$ such that $x_r = y_r$. For $x' = x \oplus e_r$ and $y' = y \oplus e_r$, $\mathsf{d}(x,y) = \mathsf{d}(x',y')$.
(Positivity) If $\mathsf{d}(x,y) = 0$, then $\mathsf{d}(y,x) > 0$.

Proof. The linearity property follows from Claim 5. Suppose $x \succ z \succ y$. We have $\mathsf{hcd}(x,y) = y$, $\mathsf{hcd}(x,z) = z$, and $\mathsf{hcd}(y,z) = y$. Hence, $\mathsf{d}(x,y) = \beta||x-y||_1 = \beta(||x-z||_1 + ||z-y||_1) = \mathsf{d}(x,z) + \mathsf{d}(z,y)$. The other case is analogous.

The triangle inequality follows because $\mathsf{d}(x,y)$ is a shortest path length. For the projection property, let $z = \mathsf{hcd}(x,y)$ and let $z' = \mathsf{hcd}(x',y')$. Note that z and z' also differ only in the rth coordinate. Thus, $||x-z||_1 = ||x'-z'||_1$ and $||y-z||_1 = ||y'-z'||_1$. We have $\mathsf{d}(x,y) = \beta||x-z||_1 - \alpha||y-z||_1 = \beta||x'-z'||_1 - \alpha||y'-z'||_1 = \mathsf{d}(x',y')$. Suppose the positivity property does not hold. So $\mathsf{d}(x,y) = 0$ and $\mathsf{d}(y,x) \le 0$. Hence, $\beta||x-z||_1 = \alpha||y-z||_1$ and $\beta||y-z||_1 \le \alpha||x-z||_1$. Adding, we get $\beta \le \alpha$, a contradiction. □

We also have a generalization of the linearity property that will be useful to state explicitly. We will make use of this in one of our main proofs.

Claim 7. *Suppose x and y differ in the rth coordinate. Let $\widehat{x} = x \oplus e_r$.*
1. *If $x_r = 0$: $\mathsf{d}(x,y) - \mathsf{d}(\widehat{x},y) = -\alpha$ and $\mathsf{d}(y,x) - \mathsf{d}(y,\widehat{x}) = \beta$.*
2. *If $x_r = 1$: $\mathsf{d}(x,y) - \mathsf{d}(\widehat{x},y) = \beta$ and $\mathsf{d}(y,x) - \mathsf{d}(y,\widehat{x}) = -\alpha$.*

Proof. First, let us assume that $x_r = 0$ (so $y_r = 1$). Let $z = \mathsf{hcd}(x,y)$ and $\widehat{z} = \mathsf{hcd}(\widehat{x},y)$. The coordinates of z and $\widehat{z}$ are the same except for the rth one. Note that the rth coordinate of z must be zero, but that of $\widehat{z}$ is 1. We have $||x-z||_1 = ||\widehat{x}-\widehat{z}||_1$ but $||y-z||_1 = ||y-\widehat{z}||_1 + 1$. Hence, $\mathsf{d}(x,y) - \mathsf{d}(\widehat{x},y) = -\alpha$ and $\mathsf{d}(y,x) - \mathsf{d}(y,\widehat{x}) = \beta$.

When $x_r = 1$, $||x-z||_1 = ||\widehat{x}-\widehat{z}||_1 + 1$ but $||y-z||_1 = ||y-\widehat{z}||_1$. So, $\mathsf{d}(x,y) - \mathsf{d}(\widehat{x},y) = \beta$ and $\mathsf{d}(y,x) - \mathsf{d}(y,\widehat{x}) = -\alpha$. □

The following lemma connects the distance to the property $\mathcal{L}_{\alpha,\beta}$.

Lemma 8. *A function is (α,β)-Lipschitz iff for all $x,y \in \mathsf{B}$, $f(x) - f(y) - \mathsf{d}(x,y) \le 0$.*

Proof. Suppose the function satisfied the inequality for all x,y. If x and y differ in one-coordinate with $x \succ y$, we get $f(x) - f(y) \le \mathsf{d}(x,y) = \beta$ and $f(y) - f(x) \le -\alpha$ implying f is (α,β)-Lipschitz. Conversely, suppose f is (α,β)-Lipschitz. Setting $z = \mathsf{hcd}(x,y)$ (for $x,y \in \mathsf{B}$), we get $f(x) - f(z) \le \beta||x-z||_1$ and $\alpha||y-z||_1 \le f(y) - f(z)$. Summing these, $f(x) - f(y) \le \beta||x-z||_1 - \alpha||y-z||_1 = \mathsf{d}(x,y)$. □

The next lemma is a generalization of a standard argument for monotonicity testing. We construct a "violation graph" and argue that the size of a minimum vertex cover is exactly $\varepsilon_f 2^n$. A similar statement is also known for the Lipschitz property, and we prove this for generalized Lipschitz functions. We crucially use the triangle inequality for $\mathsf{d}(x,y)$.

We define an undirected weighted clique K on B. Given a function f, we define the weight $w(x,y)$ (for any $x,y \in \mathsf{B}$) as follows:

$$w(x,y) \;:=\; \max\Big(f(x) - f(y) - \mathsf{d}(x,y), \quad f(y) - f(x) - \mathsf{d}(y,x)\Big) \tag{1}$$

Note that although the distance d is asymmetric, the weights are defined on an undirected graph. Lemma 8 shows that a function is (α, β)-Lipschitz iff all $w(x, y) \leq 0$. Once again, it is instructive to understand the special cases of monotonicity and Lipschitz. For monotonicity, we get that $w(x, y) = f(x) - f(y)$ when $x \prec y$ and $-\infty$ otherwise. For Lipschitz, we get $w(x, y) = |f(x) - f(y)| - ||x - y||_1$. We define the *violation graph* as $VG_f = (\mathsf{B}, V_f)$ where $V_f = \{(x, y) : w(x, y) > 0\}$. The violation graph is unweighted.

Lemma 9. *The size of a minimum vertex cover in VG_f is exactly $\varepsilon_f 2^n$.*

Proof. Let U be a minimum vertex cover in VG_f. Since each edge in VG_f is a violation, the points at which the function is modified must intersect all edges, and therefore should form a vertex cover. Thus, $\varepsilon_f 2^n \geq |U|$. We now show how to modify the function values at U to get a function f' with no violations. We invoke the following claim with $V = \mathsf{B} - U$, and $f'(x) = f(x), \forall x \in V$.

Claim 10. *Consider partial function f' defined on a subset $V \subseteq \mathsf{B}$, such that for all $\forall x, y \in V$, $f'(x) - f'(y) \leq \mathsf{d}(x, y)$. It is possible to fill in the remaining values such that $\forall x, y \in \mathsf{B}$, $f'(x) - f'(y) \leq \mathsf{d}(x, y)$.*

Proof. We prove by backwards induction on the size of V. If $|V| = 0$, this is trivially true. Now for the induction step. It suffices to just define f' for some $u \notin V$. We need to set $f'(u)$ so that $f'(u) - f'(y) \leq \mathsf{d}(u, y)$ and $f'(x) - f'(u) \leq \mathsf{d}(x, u)$ for all $x, y \in V$. Let us first argue that

$$m := \max_{x \in V} (f(x) - \mathsf{d}(x, u)) \;\leq\; \min_{y \in V} (f(y) + \mathsf{d}(u, y)) =: M$$

Suppose not, so for some $x, y \in V$, $f'(x) - \mathsf{d}(x, u) > f'(y) + \mathsf{d}(u, y)$. That implies that $f'(x) - f'(y) > \mathsf{d}(x, u) + \mathsf{d}(u, y) \geq \mathsf{d}(x, y)$ (using triangle inequality). That violates the condition, so $m \leq M$. We can therefore set $f(u) \in [m, M]$ and ensure that $\forall x, y \in V \cup \{u\}$, $f'(x) - f'(y) \leq \mathsf{d}(x, y)$. ☐

This gives a function f' such that $\Delta(f, f') = |U|/2^n$. By Lemma 8, f' is (α, β)-Lipschitz, and $|U| \geq \varepsilon_f 2^n$. Hence, $|U| = \varepsilon_f 2^n$. ☐

The following is a simple corollary of the previous lemma; it follows since the endpoints of any maximal matching forms a vertex cover.

Corollary 11. *The size of any maximal matching in VG_f is $\geq \varepsilon_f 2^{n-1}$.*

In the next section, we exhibit a maximal matching of VG_f whose size is *at most* the number of violated edges. Before moving on, we make a technical claim that allows for easier arguments about w. Essentially, by a perturbation argument, we can assume that $w(x, y)$ is never exactly zero.

Claim 12. *For any function f, there exists a function f' with the following properties. Both f and f' have the same number of violated edges, $\varepsilon_f = \varepsilon_{f'}$, and for all $x, y \in \mathsf{B}$, $w_{f'}(x, y) \neq 0$.*

Proof. We will construct a function f' such that $w_{f'}(x, y)$ has the same sign as $w_f(x, y)$. When $w_f(x, y) = 0$, then $w_{f'}(x, y) < 0$. Since exactly the same pairs have a strictly positive weight, their violation graphs are identical. Both functions have the same number of violated edges and by Lemma 9, $\varepsilon_f = \varepsilon_{f'}$.

Set $f'(x) = (1 - \eta_f) f(x) + \sigma_f \|x\|_1$, where η_f and σ_f are very small (say, $\eta_f = \frac{1}{2^{2L}}$, and $\sigma_f = \frac{1}{2^{3L}}$ where L is the precision of f). We have $f'(x) - f'(y) = (1 - \eta_f)(f(x) - f(y)) + \sigma_f(\|x\|_1 - \|y\|_1)$.

If $f(x) \neq f(y)$, then $f'(x) - f'(y) - \mathsf{d}(x, y)$ has the same sign as $f(x) - f(y) - \mathsf{d}(x, y)$. Under this circumstance, when $w_f(x, y) \neq 0$ $w_{f'}(x, y)$ has the same sign.

Suppose $f(x) = f(y)$. If $w_f(x, y)$ is non-zero, then (since σ_f is so small) $w_{f'}(x, y)$ maintains the sign. So assume that $w_f(x, y) = 0$. Wlog, $\mathsf{d}(x, y) = 0$, so by Claim 5, $\mathsf{d}(y, x) > 0$. Setting $z = \mathsf{hcd}(x, y)$, we get $\beta||x - z||_1 = \alpha||y - z||_1$ and $\alpha||x - z||_1 < \beta||y - z||_1$. Adding and using the fact that $\alpha + \beta > 0$, $||x - z||_1 < ||y - z||_1$. Hence $\|x\|_1 - \|y\|_1 < 0$, and $f'(x) - f'(y) < 0$. Therefore, $w_{f'}(x, y) < 0$. □

Henceforth, we will just assume that $w_f(x, y) \neq 0$ for any x, y.

3 Violated edges through weighted matchings

We begin with some notation regarding matchings. For every $1 \leq r \leq n$, let $\mathsf{B}_r^0 := \{x \in \mathsf{B} : x_r = 0\}$ and $\mathsf{B}_r^1 := \{x \in \mathsf{B} : x_r = 1\}$ be the two $(n-1)$-dimensional hypercubes generated by dimension r. The edge set H of Hyp can be partitioned as $H_1 \sqcup H_2 \cdots \sqcup H_n$, where $H_r = \{(x, y) \in H : x_r \neq y_r\}$ are the edges crossing the rth dimension. Note that H_r is a perfect matching between B_r^0 and B_r^1. Let $C_r := V_f \cap H_r$ denote the set of edges in the violation graph VG_f with one endpoint each in B_r^0 and B_r^1. These edges are called the *r-crossing violated edges*.

Let M be a maximum weight matching in K where the weights are given by $w(x, y)$, as in (1). Note that M must be a maximal matching in VG_f which contains all the positive weight edges. From Corollary 11, we get the following.

Claim 13. $|M| \geq \varepsilon_f 2^{n-1}$.

Proof. Naturally, M will not contain any edge with negative weight, and there are no edges with weight 0. Hence, M is completely contained in VG_f. The matching M must also be a maximal matching in VG_f. By Lemma 9 and the fact that the endpoints of a maximal matching form a vertex cover, $2|M| \geq \varepsilon_f 2^n$. □

For $1 \leq r \leq n$, let M_r be the pairs of M with one endpoint each in B_r^0 and B_r^1. These are *r-cross pairs of M*. We use the notation $M(u)$ to denote the vertex v if $(u, v) \in M$; else $M(u)$ is undefined. The main lemma of our paper is the following. It shows that the number of r-crossing violated edges is at least the number of r-cross pairs of M.

Lemma 14 (Main Lemma)**.** *For all $1 \leq r \leq n$, $|M_r| \leq |C_r|$.*

We first show that this lemma implies Theorem 3.

Proof of Theorem 3: From Lemma 14, $|M| \leq \sum_{r=1}^{n} |M_r| \leq \sum_{r=1}^{n} |C_r|$, where the first inequality followed from the fact that any pair $(x, y) \in M$ must be r-crossing for some $1 \leq r \leq n$. The final sum is just the total number of violating edges, since the C_r's form a partition of this set. By Claim 13, $|M| \geq \varepsilon_f 2^{n-1}$, completing the proof. □

We use the remainder of the paper to prove Lemma 14. This will require some technical set up, performed in the next subsection. We will fix some r. This will allow us to reframe Lemma 14 in terms of matchings and a special sequence $\mathbf{S}_x$.

3.1 Alternating Paths and the Sequence $\mathbf{S}_x$

Both M and H_r are matchings in B; in fact the latter is a perfect matching. Hence, the symmetric difference is a collection of alternating paths and cycles. Let X_r be the endpoints of M_r that are present in these paths and cycles. (Note that if a pair in M_r is actually an edge in H_r, then the endpoints of M_r are *not* present in the alternating paths/cycles.) We will denote the set of alternating paths/cycles that contain some vertex of X_r by $\mathcal{A}$.

We define the sequence $\mathbf{S}_x$ for all $x \in X_r$ as follows.

1. The first term $\mathbf{S}_x(0)$ is x.
2. For even i, $\mathbf{S}_x(i+1) = H_r(\mathbf{S}_x(i))$.
3. For odd i: if $\mathbf{S}_x(i)$ is in X_r or is M-unmatched (so $M(\mathbf{S}_x(i))$ is undefined), then $\mathbf{S}_x$ terminates. Otherwise, $\mathbf{S}_x(i+1) = M(\mathbf{S}_x(i))$.

An intuitive way of understanding $\mathbf{S}_x$ is by looking at what happens in $\mathcal{A}$. All the paths/cycles of $\mathcal{A}$ containing points of X_r can be partitioned into contiguous sequences. Pick any vertex in $x \in X_r$ and start walking along the H_r-edge incident to it. (Since H_r is a perfect matching, this edge always exists.) We stop when we reach a vertex in X_r. We keep repeating this procedure until all paths/cycles in $\mathcal{A}$ are subpartitioned into the sequences.

Observe that any cycle containing some point of $x \in X_r$ also contains $M(x) \in X_r$. Hence, this decomposition breaks the cycle into a collection of paths with the following property. The first and last vertices these paths are in X_r, and all internal vertices in the path are not in X_r. The starting and ending edges are in H_r. (The paths are undirected, so the label of start and end is quite arbitrary.) Every vertex in X_r is the start or end of some path. The sequence $\mathbf{S}_x$ is simply the ordered list of vertices (starting from x) in the path containing x.

We list out these basic properties of $\mathbf{S}_x$. We use $T(x)$ to denote the last vertex in $\mathbf{S}_x$.

Proposition 15.

- *Every $\mathbf{S}_x$ terminates.*
- *$T(x)$ is either M-unmatched or $T(x) \in X_r$. In the latter case, $\mathbf{S}_{T(x)}$ is just the reverse of $\mathbf{S}_x$ and $T(T(x)) = x$.*
- *For $x, y \in X_r$, either $y = T(x)$ or $\mathbf{S}_x$ and $\mathbf{S}_y$ are disjoint.*

We will need another simple claim about the sub-hypercubes that the vertices of $\mathbf{S}_x$ lie in. For $\mathbf{S}_x(i)$, the class $i \pmod 4$ is very important for our proof. Many properties will follow a regular pattern depending on $i \pmod 4$, and the following is merely the first.

Proposition 16. *$\mathbf{S}_x(i) \in \mathsf{B}_r^{x_r}$ for $i \equiv 0, 3 \pmod 4$ and $\mathbf{S}_x(i) \in \mathsf{B}_r^{1-x_r}$ for $i \equiv 1, 2 \pmod 4$.*

Proof. We start with $\mathbf{S}_x(0) = x \in \mathsf{B}_r^{x_r}$. Suppose i is even. Since $\mathbf{S}_x(i+1) = H_r(\mathbf{S}_x(i))$, $\mathbf{S}_x(i)$ and $\mathbf{S}_x(i+1)$ lie on opposite sides of the rth-dimension. For odd i, $\mathbf{S}_x(i+1) = M(\mathbf{S}_x(i))$, where $\mathbf{S}_x(i) \notin X_r$. The pair $(\mathbf{S}_x(i), \mathbf{S}_x(i+1)) \in M$ does not cross the rth-dimension, and so both these vertices are on the same side of the rth-dimension. □

Our main charging lemma tells us that every sequence $\mathbf{S}_x$ contains a violated r-cross edge.

Lemma 17 (Charging Lemma). *$\forall x \in X_r$, there exists an even i such that $(\mathbf{S}_x(i), \mathbf{S}_x(i+1)) \in C_r$.*

Proof of Lemma 14: Let $W_1 = M_r \cap H_r$ and $W_2 = M_r \setminus W_1$. The set W_1 is simply the set of matched pairs that are also (violated) edges in Hyp. Hence, $W_1 \subseteq C_r$. Note that these edges cannot appear in $\mathcal{A}$, since this is contained in the symmetric difference of M and H_r. All endpoints of W_2 pairs are present in $\mathcal{A}$, and this is exactly the set X_r.

Let Y_r be the set of endpoints of W_1. Define a mapping between $X_r \cup Y_r$ (vertices) to C_r (edges). For any endpoint of W_1, map it to the edge of C_r containing it. By Lemma 17, for every $x \in X_r$, there exists an edge $(\mathbf{S}_x(i), \mathbf{S}_x(i+1)) \in C_r$. We map x to this edge. Prop. 15 tells us that for $x, y \in X_r$, the only way they can both be mapped to the same edge $e \in C_r$ is if $y = T(x)$. Furthermore, the endpoints of e cannot be in Y_r, since e belongs to the symmetric difference of M and H_r. Since W_1 is a matching, exactly two vertices in Y_r map to a single edge.

All in all, we map at most 2 vertices in $X_r \cup Y_r$ to C_r. So, $|X_r \cup Y_r| \leq 2|C_r|$. The proof ends by observing that $|X_r \cup Y_r| = 2|M_r|$. □

One can take everything up to this point as merely a preamble for the main proof. In the authors' opinion, the arguments made in next subsection are really the main contribution. We show here *why* many r-cross pairs in M_r imply many violated edges along the rth-dimension.

3.2 Proof of the Charging Lemma (Lemma 17)

The proof is technical and heavy (maybe excessively so) on notation. As a warmup, we illustrate the main ideas by sketching a proof for monotonicity (that is, $\alpha = 0, \beta = \infty$). A complete proof requires a case analysis that introduces some (non-intuitive and not very helpful) notation. Here, we will give a proof for one of these cases; all the other cases do follow analogously and are practically equivalent.

3.2.1 Warmup: Monotonicity

Recall for monotone functions, $w(x, y) = f(x) - f(y)$ if $x \prec y$, and negative for all other pairs. Fix $x \in X_r$. Assume (for the other case is similar) that $x \in \mathsf{B}_r^0$. Let $y = M(x)$; $y \in \mathsf{B}_r^1$. For brevity's sake we let s denote the sequence $\mathbf{S}_x$ and use s_i, for $i \geq 0$, to denote $\mathbf{S}_x(i)$. We also define $s_{-1} := y$. By Prop. 16, $s_i \in \mathsf{B}_r^0$ whenever $i \equiv 0, 3 \pmod 4$, and $s_i \in \mathsf{B}_r^1$ otherwise. For contradiction's sake, we assume $(s_{i-1}, s_i) \notin C_r$ for any odd i. Thus,

$$f(s_{i-1}) - f(s_i) > 0,\ \forall i \equiv 3 \pmod 4; \qquad f(s_i) - f(s_{i-1}) > 0,\ \forall i \equiv 1 \pmod 4 \tag{$*$}$$

will be assumed to hold throughout. For odd i, the pair (s_i, s_{i+1}) lies in M, but a priori we do not know which of these two is the ancestor and which is the descendant. The following lemma characterizes this.

Lemma 18. *For odd i, suppose $s_{i+1} = M(s_i)$ is defined. Then,*

$$\forall i \equiv 1 \pmod 4,\ s_{i+1} \succ s_i; \quad \forall i \equiv 3 \pmod 4,\ s_i \succ s_{i+1}$$

Proof. The proof is by induction on i. Assume the claim is true for all odd $j < i$, for some $i \equiv 1 \pmod 4$. The proof for the other case is similar. Suppose for contradiction, $s_i \succ s_{i+1}$. We now construct a matching M' of larger weight than M as follows. Delete the set of M-edges $E_- := \{(s_j, s_{j+1}) : j \text{ odd }, -1 \leq j \leq i\}$, and add the set of edges

$$E_+ := (s_{-1}, s_1) \cup \{(s_{j-1}, s_{j+2}) : j \text{ odd }, 1 \leq j \leq i-4\} \cup (s_{i-3}, s_{i+1})$$

Check that $M - E_- + E_+$ is a valid matching which leaves s_i, s_{i-1} unmatched. Now we consider the weights. The weight of E_-, by induction, is $W_- =$

$$[f(s_0) - f(s_{-1})] + [f(s_1) - f(s_2)] + [f(s_4) - f(s_3)] + \cdots + [f(s_{i-1}) - f(s_{i-2})] + [f(s_{i+1}) - f(s_i)]$$

Observe the signs changing from term to term due to induction hypothesis, except for the last term which is assumed for the sake of contradiction. Also by induction, and since $(s_{j-1}, s_{j+2}) = (s_j \oplus e_r, s_{j+1} \oplus e_r)$, we get that whenever $1 \leq j \equiv 1 \pmod 4$, $s_{j+2} \succ s_{j-1}$ and whenever $(i-2) \geq j \equiv 3 \pmod 4$, $s_{j-1} \succ s_{j+2}$. By the assumption, we get $s_{i-3} \succ s_i \succ s_{i+1}$. Using this, we get the weight of E_+ is precisely $W_+ =$

$$[f(s_1) - f(s_{-1})] + [f(s_0) - f(s_3)] + \cdots + [f(s_{i-5}) - f(s_{i-2})] + [f(s_i) - f(s_{i-3})]$$

Thus, we get the weight of the new matching is precisely $w(M) - W_- + W_+ = w(M) + f(s_i) - f(s_{i-1})$. By ($*$), we get that $f(s_i) > f(s_{i-1})$ contradicting the maximality of M. □

Armed with this handle on the ancestor-descendant relationships, we can show that every odd s_i belongs to a matching pair.

Lemma 19. *For odd i, if s_i exists then $s_{i+1} = M(s_i)$ exists.*

Proof. Suppose not. Then as in the proof of the previous lemma we can find a better matching. Once again, assume $i \equiv 1 \pmod 4$. We delete the set of edges $E_- := \{(s_j, s_{j+1}) : j \text{ odd }, -1 \leq j \leq i-2\}$ and add the set of edges $E_+ = (s_{-1}, s_1) \cup \{(s_{j-1}, s_{j+2}) : j \text{ odd }, 1 \leq j \leq i-3\}$. Lemma 18 shows that $M - E_- + E_+$ is a valid matching whose weight is, as before, $w(M) + f(s_i) - f(s_{i-1}) > w(M)$ by $(*)$. □

Lemma 20. *For odd i, if s_i exists then $s_i \notin X_r$.*

Proof. This is really just a corollary of Lemma 18. Suppose $i \equiv 1 \pmod 4$. Then, by Prop. 16, $s_i \in \mathsf{B}^1_r$. By Lemma 18, $M(s_i) = s_{i+1} \succ s_i$, and so $s_{i+1} \in B^1_r$. Hence, (s_i, s_{i+1}) is not an r-cross pair, and $s_i \notin X_r$. If $i \equiv 3 \pmod 4$, then $s_i \in \mathsf{B}^0_r$ and $M(s_i) \prec s_i$. Again, $s_i \notin X_r$. □

We conclude that for any $x \in X_r$, if Condition $(*)$ holds, then $\mathbf{S}_x$ can never terminate. This is because for odd i, $M(s_i)$ exists and $s_i \notin X_r$ implying $s_{i+1} = M(s_i)$. The non-termination contradictions Prop. 15, and therefore $(*)$ must be violated.

3.2.2 The General Proof

As in the warm-up, we fix $x \in X_r$. Suppose for contradiction that $\mathbf{S}_x$ does not contain an edge of C_r. We will show that $\mathbf{S}_x$ cannot terminate which contradics Prop. 15. The proof crucially uses that fact that M is a maximum weight matching. We start of with some new notation and technical definitions.

Preliminaries. Let $y = M(x)$, and thus $(y, x) \in M_r$ is a cross pair. The weight $w(y, x)$ is given by $\max(f(x) - f(y) - \mathsf{d}(x, y), f(y) - f(x) - \mathsf{d}(y, x))$. (Since $w(x, y) = w(y, x)$, it will be convenient for later calculations to choose the later.) To abstract out these two cases cleanly, we define the following.

- The functions d_{-1} and d_1: The functions d_1 and d_{-1} based on the order of arguments. We set $\mathsf{d}_1(x, y) = \mathsf{d}(x, y)$ and $\mathsf{d}_{-1}(x, y) = \mathsf{d}(y, x)$.
- The marker bit $\mathfrak{b}$: If $w(y, x) = f(y) - f(x) - \mathsf{d}(y, x)$, then $\mathfrak{b} = 1$. Otherwise, it is -1.
- The function $\sigma(y', x', b)$: $\sigma(y', x', \mathfrak{b}) = \mathfrak{b}(f(y') - f(x')) - \mathsf{d}_{\mathfrak{b}}(y, x)$.

Note that $w(y, x) = \mathfrak{b}(f(y) - f(x)) + \mathsf{d}_{\mathfrak{b}}(y, x)$ and $w(y, x) = \sigma(y, x, \mathfrak{b})$. For the sake of brevity, we let s denote the sequence $\mathbf{S}_x$, and let s_i denote $\mathbf{S}_x(i)$. Also let $s_{-1} := y$ and $s_t = T(x)$. We will always assume index i to be odd. Hence, $s_{i-1} = H_r(s_i)$.

Condition $(**)$: We assume, for the sake of contradiction, that $(s_i, s_{i-1}) \notin C_r$ for any odd i. Therefore, $f(s_i) - f(s_{i-1})$ must satisfy the (α, β)-Lipschitz condition. By the perturbation of Claim 12, there is always some slack. To express this cleanly, we use the indicator μ_i defined as

$$\text{for odd } i, \quad \mu_i = \begin{cases} 1 & \text{if } i \equiv 1 \pmod 4 \\ 0 & \text{if } i \equiv 3 \pmod 4 \end{cases}$$

By Prop. 16, if $i \equiv 3 \pmod 4$, $s_i \in \mathsf{B}^{x_r}_r$. Otherwise, $s_i \in \mathsf{B}^{1-x_r}_r$. Hence, when $\mu_i = 0, x_r = 1$ or $\mu_i = 1, x_r = 0$, $s_i \succ s_{i-1}$. When $\mu_i = 0, x_r = 0$ or $\mu_i = 1, x_r = 1$, $s_{i-1} \prec s_i$. We have the following, referred to as Condition $(**)$, where the strict inequality is because of the slack. For all odd i:

$$\alpha < (-1)^{\mu_i + x_r}(f(s_{i-1}) - f(s_i)) < \beta \qquad (**)$$

Pair sets E_- and E_+: One of the main aspects of the argument is modifying the matching M by deleting and inserting some pairs. Since M is a maximum weight matching, the total weight cannot decrease. This will lead to various inequalities involving f and d-values. The matching M is modified by removing all pairs incident to $\mathbf{S}_x$, up to (but not including) s_i. What does these pairs look like? The pairs are (y, x), (s_1, s_2), (s_3, s_4), …, (s_{i-2}, s_{i-1}). (For $i = 1$, this is just (y, x). As long as $M(s_j)$ is defined for $1 \leq j < i$, this sequence of edges is well-defined.) This leads us to define the subset $E_-(i) \subseteq M$. The minus is to denote pairs to be removed. For later convenience, we split the union in two groups.

$$\begin{aligned} E_-(i) &= \{(y,x)\} \cup \{(s_j, s_{j+1}) : j \text{ is odd }, 1 \leq j \leq i-2\} \\ &= \{(y,x)\} \cup \{(s_{4\ell+1}, s_{4\ell+2}) : 0 \leq \ell \leq \lfloor i/4 \rfloor - \mu_i\} \cup \{(s_{4\ell+3}, s_{4\ell+4}) : 0 \leq \ell \leq \lfloor i/4 \rfloor - 1\} \end{aligned}$$

The pairs added will depend on the statement we wish to prove. Nonetheless, there is a core set of common pairs. The aim is to select a set whose weight can be compared to $w(E_-(i))$. We will prove shortly that this weight (sort of) looks like $\sigma(y, x, 1) + \sigma(s_1, s_2, -1) + \sigma(s_3, s_4, 1) + \sigma(s_5, s_6, -1) \ldots$. In other words, the bit argument keeps switching. Let us focus on the $f(\cdot)$ terms in $w(E_-)$. We have $[f(y) - f(x)]$ $+[f(s_2) - f(s_1)]$ $+[f(s_3) - f(s_4)]$ $+[f(s_6) - f(s_5)] + [f(s_7) - f(s_8)] \ldots$. We wish to pair these up differently but maintain the same "weight structure". We will always pair terms with odd and even indices together (except for y). We start with (y, s_1). Now, $x = s_0$ needs to paired with an odd indexed s_j with $f(s_j)$ with a negative coefficient. So we get (s_0, s_3). The next to be paired is s_2, which we manage by (s_2, s_5). Then we get (s_4, s_7). We want to stay on vertices used in $E_-(i)$, so we will not involve s_i. Formally,

$$\begin{aligned} E_+(i) &= \{(y, s_1)\} \cup \{(s_j, s_{j+3}) : j \text{ is even}, 0 \leq j \leq i-5\} \\ &= \{(y, s_1)\} \cup \{(s_{4\ell}, s_{4\ell+3}) : 0 \leq \ell \leq \lfloor i/4 \rfloor - 1\} \cup \{(s_{4\ell-2}, s_{4\ell+1}) : 1 \leq \ell \leq \lfloor i/4 \rfloor - \mu_i\} \end{aligned}$$

Proposition 21. *The pairs in $E_-(i)$ exactly involve all vertices in $\{s_j : -1 \leq j \leq i-2\}$. The pairs in $E_+(i)$ exactly involve vertices in $E_-(i) \setminus s_{i-3}$.*

As in the monotonicity case, we need to understand the weights of the pairs of M in $\mathbf{S}_x$. For odd i, we know that $w(s_i, s_{i+1})$ is $\max(\sigma(s_i, s_{i+1}, -1), \sigma(s_i, s_{i+1}, 1))$, but which value does it take? To execute the argument described above, we need to know this. It turns out that this is exactly decided by μ_i, and therefore has a very consistent behavior. This is analogous to Lemma 18 from the warmup.

Lemma 22. *For odd i, suppose $s_{i+1} = M(s_i)$ is defined. Then, $w(s_i, s_{i+1}) = \sigma(s_i, s_{i+1}, (-1)^{\mu_i}\mathfrak{b})$.*

Proof. The proof is by induction over i. For $i = -1$, we are looking at $w(s_{-1}, s_0) = w(y, x)$. The parameter $\mathfrak{b}$ was chosen so that $w(y, x) = \sigma(y, x, \mathfrak{b})$. We now perfom the induction step. For an odd i, suppose the claim is true for all odd $j < i$. To simply the case analysis, let us set bit $(-1)^{\mu_i}\mathfrak{b}$ to be $\mathfrak{b}$ if $i \equiv 3 \pmod 4$ and $-\mathfrak{b}$ otherwise. Our aim is to show that $w(s_i, s_{i+1}) = \sigma(s_i, s_{i+1}, (-1)^{\mu_i}\mathfrak{b})$.

We will prove by contradiction, so let $w(s_i, s_{i+1}) = \sigma(s_i, s_{i+1}, (-1)^{\mu_i+1}\mathfrak{b})$. As explained earlier, we will define a set of M-pairs E_{rem} and another set of pairs E_{add} defined on the same set as E_{rem}. We choose $E_{rem} := E_-(i) \cup \{(s_i, s_{i+1})\}$. The set of new pairs, E_{add} is defined as $E_+(i) \cup \{(s_{i-3}, s_{i+1})\}$. Observe that $E_+(i)$ does not involve s_{i-3} (the largest even index involved is $i-5$), so this is a valid set of matched pairs.

We now compute the weights of edges in these sets. The following definition and claim are motivated by this.

Definition 4. *For odd i suppose s_i exists. Define sums $W_+(i)$ and $W_-(i)$ as follows.*

$$W_-(i) = \sigma(y, x, \mathfrak{b}) + \sum_{\ell=0}^{\lfloor i/4 \rfloor - \mu_i} \sigma(s_{4\ell+1}, s_{4\ell+2}, -\mathfrak{b}) + \sum_{\ell=0}^{\lfloor i/4 \rfloor - 1} \sigma(s_{4\ell+3}, s_{4\ell+4}, \mathfrak{b})$$

$$W_+(i) = \sigma(y, s_1, \mathfrak{b}) + \sum_{\ell=0}^{\lfloor i/4 \rfloor - 1} \sigma(s_{4\ell}, s_{4\ell+3}, -\mathfrak{b}) + \sum_{\ell=1}^{\lfloor i/4 \rfloor - \mu_i} \sigma(s_{4\ell-2}, s_{4\ell+1}, \mathfrak{b})$$

By induction, note that we have $w(E_-(i)) = W_-(i)$ and therefore, we get

$$w(E_{rem}) = W_-(i) + \sigma(s_i, s_{i+1}, (-1)^{\mu_i+1}\mathfrak{b})$$

We can also *lower bound* $w(E_{add})$ as follows:

$$\begin{aligned} w(E_{add}) &\geq \sigma(y, s_1, \mathfrak{b}) + \sum_{\ell=0}^{\lfloor i/4 \rfloor - 1} \sigma(s_{4\ell}, s_{4\ell+3}, -\mathfrak{b}) + \sum_{\ell=1}^{\lfloor i/4 \rfloor + \mu_i} \sigma(s_{4\ell-2}, s_{4\ell+1}, \mathfrak{b}) + \sigma(s_{i-3}, s_{i+1}, (-1)^{\mu_i+1}\mathfrak{b}) \\ &= W_+(i) + \sigma(s_{i-3}, s_{i+1}, (-1)^{\mu_i+1}\mathfrak{b}) \end{aligned}$$

The following technical claim, which we prove afterwards, gives the difference between W_+ and W_-.

Claim 23. *For odd i,*

$$W_+(i) - W_-(i) = (-1)^{\mu_i+1}\mathfrak{b}(f(s_{i-1}) - f(s_{i-3})) - \mathsf{d}_\mathfrak{b}(y, s_1) + \mathsf{d}_\mathfrak{b}(y, x) + \mathsf{d}_{(-1)^{\mu_i+1}\mathfrak{b}}(s_{i-2}, s_{i-1})$$

Combining with Claim 23,

$$\begin{aligned} w(E_{add}) - w(E_{rem}) &\geq (-1)^{\mu_i+1}\mathfrak{b}(f(s_{i-1}) - f(s_{i-3})) + (-1)^{\mu_i}\mathfrak{b}f(s_i) + (-1)^{\mu_i+1}\mathfrak{b}f(s_{i-3}) \\ &\quad - \mathsf{d}_\mathfrak{b}(y, s_1) + \mathsf{d}_\mathfrak{b}(y, x) + \mathsf{d}_{(-1)^{\mu_i+1}\mathfrak{b}}(s_{i-2}, s_{i-1}) - \mathsf{d}_{(-1)^{\mu_i+1}\mathfrak{b}}(s_{i-3}, s_i) \end{aligned} \quad (2)$$

By Prop. 21, we can remove E_{rem} and add E_{add} to get a valid matching. Because M is a maximum weight matching, $w(E_{rem}) \geq w(E_{add})$. This can be used to get a bound on f-value difference between two adjacent vertices as follows. First,

Claim 24. *For odd i,*

$$(-1)^{\mu_i}\mathfrak{b}(f(s_{i-1}) - f(s_i)) \geq \mathsf{d}_\mathfrak{b}(y, x) - \mathsf{d}_\mathfrak{b}(y, s_1).$$

Proof. We do a case analysis based on i. Suppose $i \equiv 1 \pmod 4$. Hence, $(-1)^{\mu_i}\mathfrak{b} = -\mathfrak{b}$.Substituting in (2),

$$\begin{aligned} w(E_{add}) - w(E_{rem}) &\geq \mathfrak{b}f(s_{i-1}) - \mathfrak{b}f(s_{i-3}) - \mathfrak{b}f(s_i) + \mathfrak{b}f(s_{i-3}) \\ &\quad - \mathsf{d}_\mathfrak{b}(y, s_1) + \mathsf{d}_\mathfrak{b}(y, x) + \mathsf{d}_\mathfrak{b}(s_{i-2}, s_{i-1}) - \mathsf{d}_\mathfrak{b}(s_{i-3}, s_i) \end{aligned}$$

By the projection property, $\mathsf{d}_\mathfrak{b}(s_{i-2}, s_{i-1}) = \mathsf{d}_\mathfrak{b}(s_{i-3}, s_i)$. Now we use that M is a maximum weight matching. Since $w(E_{add}) - w(E_{rem}) \leq 0$, $\mathfrak{b}(f(s_i) - f(s_{i-1})) \geq \mathsf{d}_\mathfrak{b}(y, x) - \mathsf{d}_\mathfrak{b}(y, s_1)$.

Suppose $i \equiv 3 \pmod 4$. The proof is analogous. We have $(-1)^{\mu_i}\mathfrak{b} = \mathfrak{b}$. Substituting in (2),

$$\begin{aligned} w(E_{add}) - w(E_{rem}) &\geq \mathfrak{b}f(s_{i-3}) - \mathfrak{b}f(s_{i-1}) + \mathfrak{b}f(s_i) - \mathfrak{b}f(s_{i-3}) \\ &\quad - \mathsf{d}_\mathfrak{b}(y, s_1) + \mathsf{d}_\mathfrak{b}(y, x) + \mathsf{d}_{-\mathfrak{b}}(s_{i-2}, s_{i-1}) - \mathsf{d}_{-\mathfrak{b}}(s_{i-3}, s_i) \end{aligned}$$

We use projection and the fact that M is a maximum weight matching to conclude that $\mathfrak{b}(f(s_{i-1}) - f(s_i)) \geq \mathsf{d}_\mathfrak{b}(y, x) - \mathsf{d}_\mathfrak{b}(y, s_1)$. ☐

Claim 25. *If* $(-1)^{\mu_i}\mathfrak{b}(f(s_{i-1})-f(s_i)) \geq \mathsf{d}_{\mathfrak{b}}(y,x)-\mathsf{d}_{\mathfrak{b}}(y,s_1)$*, condition* $(**)$ *is violated.*

Proof. The fact that (x,y) is an r-cross pair is now put to use. Since $s_1 = H_r(x)$, we will use Claim 7 to deal with $\mathsf{d}_{\mathfrak{b}}(y,x)-\mathsf{d}_{\mathfrak{b}}(y,s_1)$.

By Claim 7, if $\mathfrak{b}=1$ and $x_r=0$, $\mathsf{d}_{\mathfrak{b}}(y,x)-\mathsf{d}_{\mathfrak{b}}(y,s_1) = \mathsf{d}(y,x)-\mathsf{d}(y,s_1)=\beta$. If $\mathfrak{b}=-1$ and $x_r=1$, then also $\mathsf{d}_{\mathfrak{b}}(y,x)-\mathsf{d}_{\mathfrak{b}}(y,s_1)=\beta$. So, $(-1)^{\mu_i}\mathfrak{b}(f(s_{i-1})-f(s_i)) \geq \beta$. Note that $\mathfrak{b}=(-1)^{x_r}$, so $(-1)^{\mu_i+x_r}\mathfrak{b}(f(s_{i-1})-f(s_i)) \geq \beta$. This contradicts $(**)$.

On the other hand, if $\mathfrak{b}=1, x_r=1$, or $\mathfrak{b}=-1, x_r=0$, then $\mathsf{d}_{\mathfrak{b}}(y,x)-\mathsf{d}_{\mathfrak{b}}(y,s_1)=-\alpha$. Since $\mathfrak{b}=-(-1)^{x_r}$, we get $(-1)^{\mu_i+x_r}\mathfrak{b}(f(s_{i-1})-f(s_i)) \leq \alpha$. Again, contrary to $(**)$. □

Therefore, assuming $(**)$, we have proved the lemma. All that remains is the proof of Claim 23 which we provide next. □

Proof of Claim 23: We expand out the function σ to get longer (but similar) expressions for $W_-(i)$ and $W_+(i)$.

$$\begin{aligned}
W_-(i) &= \mathfrak{b}(f(y)-f(x)) - \mathsf{d}_{\mathfrak{b}}(y,x) + \sum_{\ell=0}^{\lfloor i/4\rfloor+\mu_i} [-\mathfrak{b}(f(s_{4\ell+1})-f(s_{4\ell+2})) - \mathsf{d}_{-\mathfrak{b}}(s_{4\ell+1},s_{4\ell+2})] \\
&\quad + \sum_{\ell=0}^{\lfloor i/4\rfloor-1} [\mathfrak{b}(f(s_{4\ell+3})-f(s_{4\ell+4})) - \mathsf{d}_{\mathfrak{b}}(s_{4\ell+3},s_{4\ell+4})] \\
&= \mathfrak{b}\left[f(y) - \sum_{\ell=0}^{\lfloor i/4\rfloor} f(s_{4\ell}) - \sum_{\ell=0}^{\lfloor i/4\rfloor+\mu_i} f(s_{4\ell+1}) + \sum_{\ell=0}^{\lfloor i/4\rfloor+\mu_i} f(s_{4\ell+2}) + \sum_{\ell=0}^{\lfloor i/4\rfloor-1} f(s_{4\ell+3})\right] \\
&\quad -\mathsf{d}_{\mathfrak{b}}(y,x) - \sum_{\ell=0}^{\lfloor i/4\rfloor+\mu_i} \mathsf{d}_{-\mathfrak{b}}(s_{4\ell+1},s_{4\ell+2}) - \sum_{\ell=0}^{\lfloor i/4\rfloor-1} \mathsf{d}_{\mathfrak{b}}(s_{4\ell+3},s_{4\ell+4})
\end{aligned}$$

$$\begin{aligned}
W_+(i) &= \mathfrak{b}(f(y)-f(s_1)) - \mathsf{d}_{\mathfrak{b}}(y,s_1) + \sum_{\ell=0}^{\lfloor i/4\rfloor-1} [-\mathfrak{b}(f(s_{4\ell})-f(s_{4\ell+3})) - \mathsf{d}_{-\mathfrak{b}}(s_{4\ell},s_{4\ell+3})] \\
&\quad + \sum_{\ell=1}^{\lfloor i/4\rfloor+\mu_i} [\mathfrak{b}(f(s_{4\ell-2})-f(s_{4\ell+1})) - \mathsf{d}_{\mathfrak{b}}(s_{4\ell-2},s_{4\ell+1})] \\
&= \mathfrak{b}\left[f(y) - \sum_{\ell=0}^{\lfloor i/4\rfloor-1} f(s_{4\ell}) - \sum_{\ell=0}^{\lfloor i/4\rfloor+\mu_i} f(s_{4\ell+1}) + \sum_{\ell=0}^{\lfloor i/4\rfloor-1+\mu_i} f(s_{4\ell+2}) + \sum_{\ell=0}^{\lfloor i/4\rfloor-1} f(s_{4\ell+3})\right] \\
&\quad -\mathsf{d}_{\mathfrak{b}}(y,s_1) - \sum_{\ell=0}^{\lfloor i/4\rfloor-1} \mathsf{d}_{-\mathfrak{b}}(s_{4\ell},s_{4\ell+3}) - \sum_{\ell=1}^{\lfloor i/4\rfloor+\mu_i} \mathsf{d}_{\mathfrak{b}}(s_{4\ell-2},s_{4\ell+1})
\end{aligned}$$

We use the projection property of d (Claim 6). Note that this property also holds for d_b (for any marker b). Hence, $\mathsf{d}_{-\mathfrak{b}}(s_{4\ell},s_{4\ell+3}) = \mathsf{d}_{-\mathfrak{b}}(H_r(s_{4\ell}),H_r(s_{4\ell+3})) = \mathsf{d}_{-\mathfrak{b}}(s_{4\ell+1},s_{4\ell+2})$. Similarly, $\mathsf{d}_{\mathfrak{b}}(s_{4\ell-2},s_{4\ell+1}) = \mathsf{d}_{\mathfrak{b}}(s_{4\ell-1},s_{4\ell})$. In the second summation of the very last line for $W_+(i)$, we can use projection, modify indices, and replace by $\sum_{\ell=0}^{\lfloor i/4\rfloor+\mu_i-1} \mathsf{d}_{\mathfrak{b}}(s_{4\ell+3},s_{4\ell+4})$

We subtract these bounds.

$$\begin{aligned}
W_+(i)-W_-(i) &= \mathfrak{b}(f(s_{4\hat{i}}) - f(s_{4\hat{i}+4\mu_i+2})) \\
&\quad -\mathsf{d}_{\mathfrak{b}}(y,s_1) + \mathsf{d}_{\mathfrak{b}}(y,x) + (1+\mu_i)\mathsf{d}_{-\mathfrak{b}}(s_{4\hat{i}+1},s_{4\hat{i}+2}) - \mu_i\mathsf{d}_{\mathfrak{b}}(s_{4\hat{i}-1},s_{4\hat{i}})
\end{aligned}$$

If $i \equiv 1 \pmod 4$, $\mu_i = -1$ and $4\lfloor i/4 \rfloor = i-1$. If $i \equiv 3 \pmod 4$, $\mu_i = 0$ and $4\lfloor i/4 \rfloor = i-3$. Substitution completes the proof. □

Using the previous lemma, we can show that $\mathbf{S}_x$ will never terminate. We start by showing that if s_i exists (for odd i), then s_i belongs to a pair of M.

Lemma 26. *For odd i, if s_i exists, then $M(s_i)$ exists.*

Proof. Suppose $M(s_i)$ does not exist. We can now involve s_i in a new matching. Set $E_{rem} = E_-(i)$. Set $E_{add} = E_+(i) \cup \{s_{i-3}, s_i\}$. We can remove E_{rem} and add E_{add} to get a valid matching. Note that this is possible because s_i does not participate in a pair of M. So $w(E_{add}) - w(E_{rem}) \leq 0$. By Lemma 22 and Claim 23,

$$\begin{aligned}
0 \geq w(E_{add}) - w(E_{rem}) &= w(E_+(i)) - w(E_-(i)) + w(s_{i-3}, s_i) \\
&\geq W_+(i) - W_-(i) + \sigma(s_{i-3}, s_i, c(-1)^{\mu_i+1}\mathfrak{b}) \\
&= (-1)^{\mu_i+1}\mathfrak{b}(f(s_{i-1}) - f(s_{i-3})) + (-1)^{\mu_i+1}\mathfrak{b}(f(s_{i-3}) - f(s_i)) \\
&\quad -\mathsf{d}_{\mathfrak{b}}(y, s_1) + \mathsf{d}_{\mathfrak{b}}(y, x) + \mathsf{d}_{(-1)^{\mu_i+1}\mathfrak{b}}(s_{i-2}, s_{i-1}) - \mathsf{d}_{(-1)^{\mu_i+1}\mathfrak{b}}(s_{i-3}, s_i) \\
&= (-1)^{\mu_i+1}\mathfrak{b}(f(s_{i-1}) - f(s_i)) - \mathsf{d}_{\mathfrak{b}}(y, s_1) + \mathsf{d}_{\mathfrak{b}}(y, x) \\
\Longrightarrow (-1)^{\mu_i}\mathfrak{b}(f(s_{i-1}) - f(s_i)) &\geq \mathsf{d}_{\mathfrak{b}}(y, x) - \mathsf{d}_{\mathfrak{b}}(y, s_1)
\end{aligned}$$

The last equality follows from projection. The final conclusion contradicts Condition (*), by Claim 25. □

Now for the last leg of our proof. For odd i, if s_i exists, then s_{i+1} will also exists. The only possible way that $\mathbf{S}_x$ can terminate is if (for some i) $s_i \in X_r$. We show that cannot happen. Since s_1 certainly exists (being just $M(x)$), we are forced to conclude that $\mathbf{S}_x$ will never terminate. Our contradiction is finally complete.

Lemma 27. *For odd i, if s_i exists, then $s_i \notin X_r$.*

Proof. By Lemma 26, $M(s_{i+1})$ exists. For contradiction's sake, suppose $(s_i, s_{i+1}) \in M_r$. We set $E_{rem} = E_-(i+2) = E_-(i) \cup \{(s_i, s_{i+1})\}$. We set $E_{add} = E_+(i+2) \cup \{s_{i-1}, s_{i+1}\}$. By Prop. 21, the largest index among vertices in E_{rem} is i. Furthermore, $s_{i-1} \notin E_+(i+2)$, so E_{add} is a valid matching. Removing E_{rem} and adding E_{add} results in a valid matching. We use Claim 23 and substitute $(-1)^{\mu_{i+2}+1} = (-1)^{\mu_i}$.

$$\begin{aligned}
0 \geq w(E_{add}) - w(E_{rem}) &= w(E_+(i+2)) - w(E_-(i+2)) + w(s_{i-1}, s_{i+1}) \\
&\geq W_+(i+2) - W_-(i+2) + \sigma(s_{i-1}, s_{i+1}, (-1)^{\mu_i}\mathfrak{b}) \\
&\geq (-1)^{\mu_i}\mathfrak{b}(f(s_{i+1}) - f(s_{i-1})) + (-1)^{\mu_i}\mathfrak{b}(f(s_{i-1}) - f(s_{i+1})) \\
&\quad -\mathsf{d}_{\mathfrak{b}}(y, s_1) + \mathsf{d}_{\mathfrak{b}}(y, x) + \mathsf{d}_{(-1)^{\mu_i}\mathfrak{b}}(s_i, s_{i+1}) - \mathsf{d}_{(-1)^{\mu_i}\mathfrak{b}}(s_{i-1}, s_{i+1}) \\
&= \mathsf{d}_{\mathfrak{b}}(y, x) - \mathsf{d}_{\mathfrak{b}}(y, s_1) + \mathsf{d}_{(-1)^{\mu_i}\mathfrak{b}}(s_i, s_{i+1}) - \mathsf{d}_{(-1)^{\mu_i}\mathfrak{b}}(s_{i-1}, s_{i+1})
\end{aligned}$$

We will show that this expression is exactly $\beta - \alpha$, showing that $\alpha \geq \beta$, a contradiction. The key is that since s_i and s_{i+1} differ in the rth coordinate and $s_{i-1} = H_r(s_i)$, we can invoke Claim 7 to bound this expression.

We perform a case analysis. Suppose $\mu_i = 0$. The above expression is $\mathsf{d}_{\mathfrak{b}}(y, x) - \mathsf{d}_{\mathfrak{b}}(y, s_1) + \mathsf{d}_{\mathfrak{b}}(s_i, s_{i+1}) - \mathsf{d}_{\mathfrak{b}}(s_{i-1}, s_{i+1})$. By Prop. 16, $s_i \in \mathsf{B}_r^{x_r}$. We use Claim 7 now. If $\mathfrak{b} = 1$ and $x_r = 0$, $\mathsf{d}_{\mathfrak{b}}(y, x) - \mathsf{d}_{\mathfrak{b}}(y, s_1) = \beta$ and $\mathsf{d}_{\mathfrak{b}}(s_i, s_{i+1}) - \mathsf{d}_{\mathfrak{b}}(s_{i-1}, s_{i+1}) = -\alpha$. If $\mathfrak{b} = 1$ and $x_r = 1$, $\mathsf{d}_{\mathfrak{b}}(y, x) - \mathsf{d}_{\mathfrak{b}}(y, s_1) = -\alpha$ and $\mathsf{d}_{\mathfrak{b}}(s_i, s_{i+1}) - \mathsf{d}_{\mathfrak{b}}(s_{i-1}, s_{i+1}) = \beta$. An identical argument holds when $\mathfrak{b} = 0$.

Suppose $\mu_i = -1$, so the expression is $\mathsf{d}_{\mathfrak{b}}(y, x) - \mathsf{d}_{\mathfrak{b}}(y, s_1) + \mathsf{d}_{\mathfrak{b}}(s_{i+1}, s_i) - \mathsf{d}_{\mathfrak{b}}(s_{i+1}, s_{i-1})$. (We switched the order of arguments in the latter two terms.) By Prop. 16, $s_i \in \mathsf{B}_r^{1-x_r}$. If $\mathfrak{b} = 1$ and $x_r = 0$, we again get that this expression is $\beta - \alpha$. (The remaining cases are also just applications of Claim 7.) □

References

[AC06] N. Ailon and B. Chazelle. Information theory in property testing and monotonicity testing in higher dimension. *Information and Computation*, 204(11):1704–1717, 2006. 3

[ACCL06] N. Ailon, B. Chazelle, S. Comandur, and D. Liu. Estimating the distance to a monotone function. *Random Structures and Algorithms*, 31(3):1704–1711, 2006. 3

[BBM11] E. Blais, J. Brody, and K. Matulef. Property testing lower bounds via communication complexity. In *Proceedings of 26th Annual IEEE Conference on Computational Complexity (CCC)*, pages 210–220, 2011. 2, 3

[BCGSM10] J. Briët, S. Chakraborty, D. García-Soriano, and A. Matsliah. Monotonicity testing and shortest-path routing on the cube. In *Proceedings of 13th international conference on Approximation (RANDOM/APPROX)*, pages 462–475, 2010. 3, 4

[BGJ+09] A. Bhattacharyya, E. Grigorescu, K. Jung, S. Raskhodnikova, and D. Woodruff. Transitive-closure spanners. In *Proceedings of the 18th Annual Symposium on Discrete Algorithms (SODA)*, pages 531–540, 2009. 3

[Bha] A. Bhattacharyya. A note on the distance to monotonicity of boolean functions. Technical report. 3

[BLR93] Manuel Blum, Michael Luby, and Ronitt Rubinfeld. Self-testing/correcting with applications to numerical problems. *Journal of Computer and System Sciences*, 47(3):549–595, 1993. 3

[BRW05] T. Batu, R. Rubinfeld, and P. White. Fast approximate *PCP*s for multidimensional bin-packing problems. *Information and Computation*, 196(1):42–56, 2005. 3

[DGL+99] Y. Dodis, O. Goldreich, E. Lehman, S. Raskhodnikova, D. Ron, and A. Samorodnitsky. Improved testing algorithms for monotonicity. *Proceedings of the 3rd International Workshop on Randomization and Approximation Techniques in Computer Science (RANDOM)*, pages 97–108, 1999. 2, 3, 4

[EKK+00] F. Ergun, S. Kannan, R. Kumar, R. Rubinfeld, and M. Viswanathan. Spot-checkers. *Journal of Computer Systems and Sciences (JCSS)*, 60(3):717–751, 2000. 3

[Fis01] E. Fischer. The art of uninformed decisions: A primer to property testing. *Bulletin of EATCS*, 75:97–126, 2001. 3

[Fis04] E. Fischer. On the strength of comparisons in property testing. *Information and Computation*, 189(1):107–116, 2004. 3

[FLN+02] E. Fischer, E. Lehman, I. Newman, S. Raskhodnikova, and R. Rubinfeld. Monotonicity testing over general poset domains. *Proceedings of the 34th Annual ACM Symposium on the Theory of Computing (STOC)*, pages 474–483, 2002. 3

[GGL+00] O. Goldreich, S. Goldwasser, E. Lehman, D. Ron, and A. Samordinsky. Testing monotonicity. *Combinatorica*, 20:301–337, 2000. 2, 3

[GGR98] O. Goldreich, S. Goldwasser, and D. Ron. Property testing and its connection to learning and approximation. *Journal of the ACM*, 45(4):653–750, 1998. 1, 3

[HK08] S. Halevy and E. Kushilevitz. Testing monotonicity over graph products. *Random Structures and Algorithms*, 33(1):44–67, 2008. 3

[JR11] M. Jha and S. Raskhodnikova. Testing and reconstruction of lipschitz functions with applications to data privacy. In *Proceedings of the 52nd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 433–442, 2011. 2, 3

[LR01] E. Lehman and D. Ron. On disjoint chains of subsets. *Journal of Combinatorial Theory, Series A*, 94(2):399–404, 2001. 3, 4

[PRR06] M. Parnas, D. Ron, and R. Rubinfeld. Tolerant property testing and distance approximation. *Journal of Computer and System Sciences*, 6(72):1012–1042, 2006. 3

[Ron09] D. Ron. Algorithmic and analysis techniques in property testing. *Foundations and Trends in Theoretical Computer Science*, 5(2):73–205, 2009. 3

[RS96] R. Rubinfeld and M. Sudan. Robust characterization of polynomials with applications to program testing. *SIAM Journal of Computing*, 25:647–668, 1996. 1, 3

ECCC ISSN 1433-8092

http://eccc.hpi-web.de