Electronic Colloquium on Computational Complexity, Revision 2 of Report No. 30 (2012)

Optimal bounds for monotonicity and Lipschitz testing over hypercubes and hypergrids

D. CHAKRABARTY, Microsoft Research
C. SESHADHRI, Sandia National Laboratories, Livermore

The problem of monotonicity testing over the hypergrid and its special case, the hypercube, is a classic, well-studied, yet unsolved question in property testing. We are given query access to $f : [k]^n \mapsto \mathbf{R}$ (for some ordered range $\mathbf{R}$). The hypergrid/cube has a natural partial order given by coordinate-wise ordering, denoted by $\prec$. A function is *monotone* if for all pairs $x \prec y$, $f(x) \leq f(y)$. The distance to monotonicity, ε_f, is the minimum fraction of values of f that need to be changed to make f monotone. For $k = 2$ (the boolean hypercube), the usual tester is the *edge tester*, which checks monotonicity on adjacent pairs of domain points. It is known that the edge tester using $O(\varepsilon^{-1} n \log |\mathbf{R}|)$ samples can distinguish a monotone function from one where $\varepsilon_f > \varepsilon$. On the other hand, the best lower bound for monotonicity testing over general $\mathbf{R}$ is $\Omega(n)$. We resolve this long standing open problem and prove that $O(n/\varepsilon)$ samples suffice for the edge tester. For hypergrids, existing testers require $O(\varepsilon^{-1} n \log k \log |\mathbf{R}|)$ samples. We give a (non-adaptive) monotonicity tester for hypergrids running in $O(\varepsilon^{-1} n \log k)$ time, recently shown to be optimal. Our techniques lead to optimal property testers (with the same running time) for the natural *Lipschitz property* on hypercubes and hypergrids. (A c-Lipschitz function is one where $|f(x) - f(y)| \leq c\|x - y\|_1$.) In fact, we give a general unified proof for $O(\varepsilon^{-1} n \log k)$-query testers for a class of "bounded-derivative" properties that contains both monotonicity and Lipschitz.

Categories and Subject Descriptors: F.2.2 [**Analysis of algorithms and problem complexity**]: Nonnumerical Algorithms and Problems—*Computations on discrete structures*; G.2.1 [**Discrete Mathematics**]: Combinatorics—*Combinatorial algorithms*

General Terms: Theory

Additional Key Words and Phrases: Property Testing, Monotonicity, Lipschitz functions

1. INTRODUCTION

Monotonicity testing over hypergrids [Goldreich et al. 2000] is a classic problem in property testing. We focus on functions $f : \mathbf{D} \mapsto \mathbf{R}$, where the domain, $\mathbf{D}$, is the hypergrid $[k]^n$ and the range, $\mathbf{R}$, is a total order. The hypergrid/hypercube defines the natural coordinate-wise partial order: $x \preceq y$, iff $\forall i \in [n], x_i \leq y_i$. A function f is *monotone* if $f(x) \leq f(y)$ whenever $x \preceq y$. The *distance to monotonicity*, denoted by ε_f, is the minimum fraction of places at which f must be changed to have the property $\mathcal{P}$. Formally, if $\mathcal{M}$ is the set of all monotone functions, $\varepsilon_f \triangleq \min_{g \in \mathcal{M}} (|\{x | f(x) \neq g(x)\}|/|\mathbf{D}|)$. Given a parameter $\varepsilon \in (0, 1)$, the aim is to design a randomized algorithm for the following problem. If $\varepsilon_f = 0$ (meaning f is monotone), the algorithm must accept with probability $> 2/3$, and if $\varepsilon_f > \varepsilon$, it must reject with probability $> 2/3$. If $\varepsilon_f \in (0, \varepsilon)$, then any answer is allowed. Such an algorithm is called a *monotonicity tester*. The quality of a tester is determined by the number of queries to f. A *one-sided tester* accepts with probability 1 if the function is monotone. A *non-adaptive tester* decides all of its queries in advance, so the queries are

A preliminary version of this result appeared as [Chakrabarty and Seshadhri 2013a].

DOI 10.1145/0000000.0000000 http://doi.acm.org/10.1145/0000000.0000000

ISSN 1433-8092

independent of the answers it receives. Monotonicity testing has been studied extensively in the past decade [Ergun et al. 2000; Goldreich et al. 2000; Dodis et al. 1999; Lehman and Ron 2001; Fischer et al. 2002; Ailon and Chazelle 2006; Fischer 2004; Halevy and Kushilevitz 2008; Parnas et al. 2006; Ailon et al. 2006; Batu et al. 2005; Bhattacharyya et al. 2009; Briët et al. 2012; Blais et al. 2012]. Of special interest is the hypercube domain, $\{0,1\}^n$. [Goldreich et al. 2000] introduced the *edge tester*. Let $\mathbf{H}$ be the pairs that differ in precisely one coordinate (the edges of the hypercube). The edge tester picks a pair in $\mathbf{H}$ uniformly at random and checks if monotonicity is satisfied by this pair. For boolean range, [Goldreich et al. 2000] prove $O(n/\varepsilon)$ samples suffice to give a bonafide montonicity tester. [Dodis et al. 1999] subsequently showed that $O(\varepsilon^{-1}n\log|\mathbf{R}|)$ samples suffice for a general range $\mathbf{R}$. In the worst case, $|\mathbf{R}| = 2^n$, and so this gives a $O(n^2/\varepsilon)$-query tester. The best known general lower bound is $\Omega(\min(n, |\mathbf{R}|^2))$ [Blais et al. 2012]. It has been an outstanding open problem in property testing (see Question 5 in the Open Problems list from the Bertinoro Workshop [Ber 2011]) to give an optimal bound for monotonicity testing over the hypercube. We resolve this by showing that the edge tester is indeed optimal (when $|\mathbf{R}| \geq \sqrt{n}$).

THEOREM 1.1. *The edge tester is a $O(n/\varepsilon)$-query non-adaptive, one-sided monotonicity tester for functions $f : \{0,1\}^n \mapsto \mathbf{R}$.*

For general hypergrids $[k]^n$, [Dodis et al. 1999] give a $O(\varepsilon^{-1}n\log k\log|\mathbf{R}|)$-query monotonicity tester. Since $|\mathbf{R}|$ can be as large as k^n, this gives a $O(\varepsilon^{-1}n^2\log^2 k)$-query tester. In this paper, we give a $O(\varepsilon^{-1}n\log k)$-query monotonicity tester on hypergrids that generalizes the edge tester. This tester is also a uniform pair tester, in the sense it defines a set $\mathbf{H}$ of pairs, picks a pair uniformly at random from it, and checks for monotonicity among this pair. The pairs in $\mathbf{H}$ also differ in exactly one coordinate, as in the edge tester.

THEOREM 1.2. *There exists a non-adaptive, one-sided $O(\varepsilon^{-1}n\log k)$-query monotonicity tester for functions $f : [k]^n \mapsto \mathbf{R}$.*

Remark 1.3. Subsequent to the conference version of this work, the authors proved a $\Omega(\varepsilon^{-1}n\log k)$-query lower bound for monotonicity testing on the hypergrid for any (adaptive, two-sided error) tester [Chakrabarty and Seshadhri 2013b]. Thus, both the above theorems are optimal.

A property that has been studied recently is that of a function being *Lipschitz*: a function $f : [k]^n \mapsto \mathbf{R}$ is called c-Lipschitz if for all $x, y \in [k]^n, |f(x) - f(y)| \leq c\|x - y\|_1$. The Lipschitz testing question was introduced by [Jha and Raskhodnikova 2011], who show that for the range $\mathbf{R} = \delta\mathbb{Z}$, $O((\delta\varepsilon)^{-1}n^2)$ queries suffice for Lipschitz testing. For general hypergrids, [Awasthi et al. 2012] recently give an $O((\delta\varepsilon)^{-1}n^2k\log k)$-query tester for the same range. [Blais et al. 2014] prove a lower bound of $\Omega(n\log k)$ queries for non-adaptive monotonicity testers (for sufficiently large $\mathbf{R}$). We give a tester for the Lipschitz property that improves all known results and matches existing lower bounds. Observe that the following holds for arbitrary ranges.

THEOREM 1.4. *There exists a non-adaptive, one-sided $O(\varepsilon^{-1}n\log k)$-query c-Lipschitz tester for functions $f : [k]^n \mapsto \mathbf{R}$.*

Our techniques apply to a class of properties that contains monotonicity and Lipschitz. We call it the bounded derivative property, or more technically, the (α,β)-Lipschitz property. Given parameters α, β, with $\alpha < \beta$, we say that a function $f : [k]^n \mapsto \mathbf{R}$ has the (α,β)-Lipschitz property if for any $x \in [k]^n$, and y obtained by increasing exactly one coordinate of x by exactly 1, we have $\alpha \leq f(y) - f(x) \leq \beta$. Note that when $(\alpha = 0, \beta = \infty)$[1], we get monotonicity. When $(\alpha = -c, \beta = +c)$, we get c-Lipschitz.

THEOREM 1.5. *There exists a non-adaptive, one-sided $O(\varepsilon^{-1} n \log k)$-query (α, β)-Lipschitz tester for functions $f : [k]^n \mapsto \mathbf{R}$, for any $\alpha < \beta$. There is no dependence in the running time on α and β.*

Although Theorem 1.5 implies all the other theorems stated above, we prove Theorem 1.1 and Theorem 1.2 before giving a whole proof of Theorem 1.5. The final proof is a little heavy on notation, and the proof of the monotonicity theorems illustrates the new techniques.

1.1. Previous work

We discuss some other previous work on monotonicity testers for hypergrids. For the total order (the case $n = 1$), which has been called the monotonicity testing problem on the *line*, [Ergun et al. 2000] give a $O(\varepsilon^{-1} \log k)$-query tester, and this is optimal [Ergun et al. 2000; Fischer 2004]. Results for general posets were first obtained by [Fischer et al. 2002]. The elegant concept of 2-TC spanners introduced by [Bhattacharyya et al. 2009] give a general class of monotonicity testers for various posets. It is known that such constructions give testers with polynomial dependence of n for the hypergrid [Bhattacharyya et al. 2012]. For constant n, [Halevy and Kushilevitz 2008; Ailon and Chazelle 2006] give a $O(\varepsilon^{-1} \log k)$-query tester (although the dependency on n is exponential). From the lower bound side, [Fischer et al. 2002] first prove an $\Omega(\sqrt{n})$ (non-adaptive, one-sided) lower bound for hypercubes. [Briët et al. 2012] give an $\Omega(n/\varepsilon)$-lower bound for non-adaptive, one-sided testers, and a breakthrough result of [Blais et al. 2012] prove a general $\Omega(\min(n, |\mathbf{R}|^2)$ lower bound. Testing the Lipschitz property is a natural question that arises in many applications. For instance, given a computer program, one may like to test the robustness of the program's output to the input. This has been studied before, for instance in [Chaudhuri et al. 2011], however, the solution provided looks into the code to detect if the program satisfies Lipschitz or not. The property testing setting is a black-box approach to the problem. [Jha and Raskhodnikova 2011] also provide an application to differential privacy; a class of mechanisms known as Laplace mechanisms proposed by [Dwork et al. 2006] achieve privacy in the process of outputting a function by adding a noise proportional to the Lipschitz constant of the function. [Jha and Raskhodnikova 2011] gave numerous results on Lipschitz testing over hypergrids. They give a $O(\varepsilon^{-1} \log k)$-query tester for the line, a general $\Omega(n)$-query lower bound for the Lipschitz testing question on the hypercube, and a non-adaptive, 1-sided $\Omega(\log k)$-query lower bound on the line.

2. THE PROOF ROADMAP

The challenge of property testing is to relate the tester behavior to the distance of the function to the property. Consider monotonicity over the hypercube. To argue about the edge tester, 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 they show the distance to monotonicity is small. This is done by modifying f to make it monotone, and bounding the number of changes as a function of the number of violated edges. There is an inherently "constructive" viewpoint to this: it specifies a method to convert non-monotone functions to monotone ones. Implementing this becomes difficult when the range is large, and existing bounds degrade with $\mathbf{R}$. For the Lipschitz property, this route becomes incredibly complex. A non-constructive approach may give more power, but how does one get a handle on the distance? The *violation graph* provides a method. The violation graph has $[k]^n$ as the vertex set and

[1] If the reader is uncomfortable with the choice of β as ∞, β can be thought of as much larger than any value in f.

an edge between any pair of comparable domain vertices (x, y) $(x \prec y)$ if $f(x) > f(y)$. The following theorem can be found as Corollary 2 in [Fischer et al. 2002].

THEOREM 2.1 ([FISCHER ET AL. 2002]). *The size of the minimum vertex cover of the violation graph is exactly $\varepsilon_f|\mathbf{D}|$. As a corollary, the size of any maximal matching in the violation graph is at least $\frac{1}{2}\varepsilon_f|\mathbf{D}|$.*

Can a large matching in the violated graph imply there are many violated edges? [Lehman and Ron 2001] give an approach by reducing the monotonicity testing problem on the hypercube to routing problems. For any k source-sink pairs on the *directed* hypercube, suppose $k\mu(k)$ edges need to be deleted in order to pairwise separate them. Then $O(n/\varepsilon\mu(n))$ queries suffice for the edge tester. Therefore, if $\mu(n)$ is at least a constant, one gets a linear query monotonicity tester on the cube. Lehman and Ron [Lehman and Ron 2001] explicitly ask for bounds on $\mu(n)$. [Briët et al. 2012] show that $\mu(n)$ could be as small as $1/\sqrt{n}$, thereby putting an $\Omega(n^{3/2}/\varepsilon)$ bottleneck to the above approach. In the reduction above, the function values are altogether ignored. More precisely, once one moves to the combinatorial routing question on source-sink pairs, the fact that they are related by actual function values is lost. Our analysis crucially uses the value of the functions to argue about the structure of the maximal matching in the violation graph.

2.1. It's all about matchings

The key insight is to move to a *weighted* violation graph. The weight of violation (x, y) depends on the property at hand; for now it suffices to know that for monotonicity, the weight of (x, y) $(x \prec y)$ is $f(x) - f(y)$. This can be thought of as a measure of the magnitude of the violation. (Violation weights were also used for Lipschitz testers [Jha and Raskhodnikova 2011].) We now look at a maximum *weighted* matching $\mathbf{M}$ in the violation graph. Naturally, this is maximal as well, so $|\mathbf{M}| \geq \frac{1}{2}\varepsilon_f|\mathbf{D}|$. All our algorithms pick a pair uniformly at random from a predefined set $\mathbf{H}$ of pairs, and check the property on that pair. For the hypercube domain, $\mathbf{H}$ is the set of all edges of the hypercube. Our analysis is based on the construction of a one-to-one mapping from pairs in $\mathbf{M}$ to *violating* pairs in $\mathbf{H}$. This mapping implies the number of violated pairs in $\mathbf{H}$ is at least $|\mathbf{M}|$, and thus the uniform pair tester succeeds with probability $\Omega(\varepsilon_f|\mathbf{D}|/|\mathbf{H}|)$, implying $O(|\mathbf{H}|/\varepsilon_f|\mathbf{D}|)$ queries suffice to test monotonicity. For the hypercube, $|\mathbf{H}| = n2^{n-1}$ and $|\mathbf{D}| = 2^n$, giving the final bound of $O(n/\varepsilon_f)$. To obtain this mapping, we first decompose $\mathbf{M}$ into sets $M_1, M_2, \ldots, M_t$ such that each pair in $\mathbf{M}$ is in at least one M_i. Furthermore, we *partition* $\mathbf{H}$ into perfect matchings $H_1, H_2, \ldots, H_t$. In the hypercube case, M_i is the collection of pairs in $\mathbf{M}$ whose ith coordinates differ, and H_i is the collection of hypercube edges differing *only* in the ith coordinate; for the hypergrid case, the partitions are more involved. We map each pair in M_i to a unique violating pair in H_i. For simplicity, let us ignore subscripts and call the matchings M and H. We will assume in this discussion that $M \cap H = \emptyset$. Consider the *alternating paths and cycles* generated by the symmetric difference of $\mathbf{M} \setminus M$ and H. Take a point x involved in a pair of M, and note that it can only be present as the endpoint of an alternating path, denoted by $\mathbf{S}_x$. Our main technical lemma shows that each such $\mathbf{S}_x$ contains a violated H-pair.

2.2. Getting the violating H-pairs

Consider M, the pairs of $\mathbf{M}$ which differ on the ith coordinate, and H is the set of edges in the dimension cut along this coordinate. Let $(x, y) \in M$, and say $x[i] = 0$ giving us $x \prec y$. (We denote the ath coordinate of x by $x[a]$.) Recall that the weight of this violation is $f(x) - f(y)$. It is convenient to think of $\mathbf{S}_x$ as follows. We begin from x and take the incident H-edge to reach s_1 (note that that $s_1 \prec y$). Then we take the $(\mathbf{M} \setminus M)$-pair containing s_1 to get s_2. But what if no such pair existed? This can be possible in two ways: either s_1 was $\mathbf{M}$-unmatched or s_1 is M-matched. If s_1 is $\mathbf{M}$-unmatched, then delete

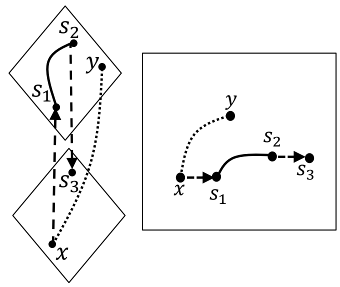

Fig. 1: The alternating path: the dotted lines connect pairs of M, the solid curved lines connect pairs of $\mathbf{M} \setminus M$, and the dashed lines are H-pairs.

(x, y) from $\mathbf{M}$ and add (s_1, y) to obtain a new matching. If (x, s_1) was not a violation, and therefore $f(x) < f(s_1)$[2], we get $f(s_1) - f(y) > f(x) - f(y)$. Thus the new matching has strictly larger weight, contradicting the choice of $\mathbf{M}$. If s_1 was M-matched, then let $(s_1, s_2) \in M$. First observe that $s_1 \succ s_2$. This is because $s_1[i] = 1$ (since $s_1[i] \neq x[i]$) and since $(s_1, s_2) \in M$ they must differ on the ith coordinate implying $s_2[i] = 0$. This implies $s_2 \prec y$, and so we could replace pairs (x, y) and (s_2, s_1) in $\mathbf{M}$ with (s_2, y). Again, if (x, s_1) is not a violation, then $f(s_2) - f(y) > [f(s_2) - f(s_1)] + [f(x) - f(y)]$, contradicting the maximality of $\mathbf{M}$. Therefore, we can taje a $(\mathbf{M} \setminus M)$-pair to reach s_2. With care, this argument can be carried over till we find a violation, and a detailed description of this is given in §5. Let us demonstrate a little further (refer to the left of Fig. 1). Start with $(x, y) \in M$, and $x[i] = 0$. Following the sequence $\mathbf{S}_x$, the first term s_1 is x projected "up" dimension cut H. The second term is obtained by following the $\mathbf{M} \setminus M$-pair incident to s_1 to get s_2. Now we claim that $s_2 \succ s_1$, for otherwise one can remove (x, y) and (s_1, s_2) and add (x, s_1) and (s_2, y) to increase the matching weight. (We just made the argument earlier; the interested reader may wish to verify.) In the next step, s_2 is projected "down" along H to get s_3. By the nature of the dimension cut H, $x \prec s_3$ and $s_1 \prec y$. So, if s_3 is unmatched and (s_2, s_3) is not a violation, we can again rearrange the matching to improve the weight. We alternately go "up" and "down" H in traversing $\mathbf{S}_x$, because of which we can modify the pairs in $\mathbf{M}$ and get other matchings in the violation graph. The maximality of $\mathbf{M}$ imposes additional structure, which leads to violating edges in H. In general, the spirit of all our arguments is as follows. Take an endpoint of M and start walking along the sequence given by the alternating paths generated by $\mathbf{M} \setminus M$ and H. Naturally, this sequence must terminate somewhere. If we never encounter a violating pair of H during the entire sequence, then we can rewire the matching $\mathbf{M}$ and increase the weight. Contradiction! Observe the crucial nature of alternating up and down movements along H. This happens because the first coordinate of the points in $\mathbf{S}_x$ switches between the two values of 0 and 1 (for $k = 2$). Such a reasoning does not hold water in the hypergrid domain. The structure of $\mathbf{H}$ needs to be more complex, and is not as simple as a partition of the edges of the hypergrid. Consider the extreme case of the line $[k]$. Let 2^r be less than k. We break $[k]$ into contiguous pieces of length 2^r. We can now match the first part to the second, the third to the fourth, etc. In other words, the pairs look like $(1, 2^r + 1)$, $(2, 2^r + 2)$, ..., $(2^r, 2^{r+1})$, then $(2^{r+1} + 1, 2^{r+1} + 2^r + 1)$, $(2^{r+1} + 2, 2^{r+1} + 2^r + 2)$, etc. We can construct such matchings for all powers of 2 less than k, and these will be our H_i's. Those familiar with existing proofs for monotonicity on $[k]$ will not be surprised by this set of matchings. All methods need to cover all "scales" from 1 to k (achieved by making them all powers of 2 up to k). It can

[2]We are assuming here that all function values are distinct; as we show in Claim 8.9 this is without loss of generality.

also be easily generalized to $[k]^n$. What about the choice of $\mathbf{M}$? Simply choosing $\mathbf{M}$ to be a maximum weight matching and setting up the sequences $\mathbf{S}_x$ does not seem to work. It suffices to look at $[k]^2$ and the matching H along the first coordinate where $r = 0$, so the pairs are $\{(x, x')|x[1] = 2i-1, x'[1] = 2i, x[2] = x'[2]\}$. A good candidate for the corresponding M is the set of pairs in $\mathbf{M}$ that connect lower endpoints of H to higher endpoints of H. Let us now follow $\mathbf{S}_x$ as before. Refer to the right part of Fig. 1. Take $(x, y) \in M$ and let $x \prec y$. We get s_1 by following the H-edge on x, so $s_1 \succ x$. We follow the $\mathbf{M} \setminus M$-pair incident to s_1 (suppose it exists) to get s_2. It could be that $s_2 \succ s_1$. It is in s_3 that we see a change from the hypercube. We could get $s_3 \succ s_2$, because there is no guarantee that s_2 is at the higher end of an H-pair. This could not happen in the hypercube. We could have a situation where s_3 is unmatched, we have not encountered a violation in H, and yet we cannot rearrange $\mathbf{M}$ to increase the weight. For a concrete example, consider the points as given in Fig. 1 with function values $f(x) = f(s_1) = f(s_3) = 1$, $f(y) = f(s_2) = 0$. Some thought leads to the conclusion that s_3 must be less than s_2 for any such rearrangement argument to work. The road out of this impasse is suggested by the two observations. First, the difference in 1-coordinates between s_1 and s_2 must be odd. Next, we could rearrange and match (x, s_2) and (s_1, y) instead. The weight may not increase, but this matching might be more amenable to the alternating path approach. We could start from a maximum weight matching that also maximizes the number of pairs where coordinate differences are even. Indeed, the insight for hypergrids is the definition of a *potential* Φ for $\mathbf{M}$. The potential Φ is obtained by summing for every pair $(x, y) \in \mathbf{M}$ and every coordinate a, the largest power of 2 dividing the difference $|x[a] - y[a]|$. We can show that a maximum weight matching that also maximizes Φ does not end up in the bad situation above. With some addition arguments, we can generalize the hypercube proof. We describe this in §7.

2.3. Attacking the generalized Lipschitz property

One of the challenges in dealing with the Lipschitz property is the lack of direction. The Lipschitz property, defined as $\forall x, y, |f(x) - f(y)| \leq \|x - y\|_1$, is an undirected property, as opposed to monotonicity. In monotonicity, a point x only "interacts" with the subcube above and below x, while in Lipschitz, constraints are defined between all pairs of points. Previous results for Lipschitz testing require very technical and clever machinery to deal with this issue, since arguments analogous to monotonicity do not work. The alternating paths argument given above for monotonicity also exploits this directionality, as can be seen by heavy use of inequalities in the informal calculations. Observe that in the monotonicity example for hypergrids in Fig. 1, the fact that $s_3 \succ s_2$ (as opposed to $s_3 \prec s_2$) required the potential Φ (and a whole new proof). A subtle point is that while the property of Lipschitz is undirected, violations to Lipschitz are "directed". If $|f(x) - f(y)| > \|x - y\|_1$, then either $f(x) - f(y) > \|x - y\|_1$ or $f(y) - f(x) > \|x - y\|_1$, but never both. This can be interpreted as a direction for violations. In the alternating paths for monotonicity (especially for the hypercube), the partial order relation between successive terms follow a fixed pattern. This is crucial for performing the matching rewiring. As might be guessed, the weight of a violation (x, y) becomes $\max(f(x) - f(y) - \|x - y\|_1, f(y) - f(x) - \|x - y\|_1)$. For the generalized Lipschitz problem, this is defined in terms of a pseudo-distance over the domain. We look at the maximum weight matching as before (and use the same potential function Φ). The notion of "direction" takes the place of the partial order relation in monotonicity. The main technical arguments show that these directions follow a fixed pattern in the corresponding alternating paths. Once we have this pattern, we can perform the matching rewiring argument for the generalized Lipschitz problem.

3. THE ALTERNATING PATHS FRAMEWORK

The framework of this section is applicable for all (α, β)-Lipschitz properties over hypergrids. We begin with two objects: $\mathbf{M}$, the matching of violating pairs, and H, a matching of $\mathbf{D}$.

The pairs in H will be aligned along a fixed dimension (denote it by r) with the same ℓ_1 distance, called the H-distance. That is, each pair (x, y) in H will differ only in one coordinate and the difference will be the same for all pairs. We now give some definitions.

— $L(H), U(H)$: Each pair $(x, y) \in H$ has a "lower" end x and an "upper" end y depending on the value of the coordinate at which they differ. We use $L(H)$ (resp. $U(H)$) to denote the set of lower (resp. upper) endpoints. Note that $L(H) \cap U(H) = \emptyset$.
— H-straight pairs, $st_H(\mathbf{M})$: All pairs $(x, y) \in \mathbf{M}$ with both ends in $L(H)$ or both in $U(H)$.
— H-cross pairs, $cr_H(\mathbf{M})$: All pairs $(x, y) \in \mathbf{M} \setminus H$ such that $x \in L(H)$, $y \in U(H)$, and the H-distance divides $|y[r] - x[r]|$.
— H-skew pairs, $sk_H(\mathbf{M}) = \mathbf{M} \setminus (st_H(\mathbf{M}) \cup cr_H(\mathbf{M}))$.
— X: A set of lower endpoints in $cr_H(\mathbf{M}) \setminus H$.

Consider the domain $\{0, 1\}^n$. We set H to be (say) the first dimension cut. $st_H(\mathbf{M})$ is the set of pairs in $(x, y) \in \mathbf{M}$ where $x[1] = y[1]$. All other pairs $(x, y) \in \mathbf{M}$ $(x \prec y)$ are in $cr_H(\mathbf{M})$ since $x[1] = 0$ and $y[1] = 1$. There are no H-skew pairs. The set X will be chosen differently for the applications. We require the following technical definition of *adequate matchings*. This arises because we will use matchings that are not necessarily perfect. A perfect matching H is always adequate.

Definition 3.1. A matching H is *adequate* if for every violation (x, y), both x and y participate in the matching H.

We will henceforth assume that H is adequate. The symmetric difference of $st_H(\mathbf{M})$ and H is a collection of alternating paths and cycles. Because H is adequate and $st_H(\mathbf{M}) \cap cr_H(\mathbf{M}) = \emptyset$, any point in $x \in X$ is the endpoint of some alternating path (denoted by $\mathbf{S}_x$). Throughout the paper, i denotes an even index, j denotes an odd index, and k is an arbitrary index.

(1) The first term $\mathbf{S}_x(0)$ is x.
(2) For even i, $\mathbf{S}_x(i+1) = H(\mathbf{S}_x(i))$.
(3) For odd j: if $\mathbf{S}_x(j)$ is $st_H(\mathbf{M})$-matched, $\mathbf{S}_x(j+1) = \mathbf{M}(\mathbf{S}_x(j))$. Otherwise, terminate.

We start with a simple property of these alternating paths.

PROPOSITION 3.2. *For $k \equiv 0, 3 \pmod 4$, $s_k \in L(H)$. For non-negative $k \equiv 1, 2 \pmod 4$, $s_k \in U(H)$.*

PROOF. If k is even, then $(s_k, s_{k+1}) \in H$. Therefore, either $s_k \in L(H)$ and $s_{k+1} \in U(H)$ or vice versa. If k is odd, (s_k, s_{k+1}) is a straight pair. So s_k and s_{k+1} lie in the same sets. Starting with $s_0 \in L(H)$, a trivial induction completes the proof. □

The following is a direct corollary of Prop. 3.2.

COROLLARY 3.3. *If $i \equiv 0 \pmod 4$, $s_i \prec s_{i+1}$. If $i \equiv 2 \pmod 4$, $s_{i+1} \prec s_i$.*

We will prove that every $\mathbf{S}_x$ contains a violated H-pair. Henceforth, our focus is entirely on some fixed sequence $\mathbf{S}_x$.

3.1. The sets $E_-(i)$ and $E_+(i)$

Our proofs are based on matching rearrangements, and this motivates the definitions in this subsection. For convenience, we denote $\mathbf{S}_x$ by $x = s_0, s_1, s_2, \ldots$. We also set $s_{-1} = y$. Consider the sequence $s_{-1}, s_0, s_1, \ldots, s_i$, for even $i > 1$. We define

$$E_-(i) = (s_{-1}, s_0), (s_1, s_2), (s_3, s_4), \ldots, (s_{i-1}, s_i) = \{(s_j, s_{j+1}) : j \text{ odd}, -1 \leq j < i\}$$

This is simply the set of $\mathbf{M}$-pairs in $\mathbf{S}_x$ up to s_i. We now define $E_+(i)$. Think of this as follows. We first pair up (s_{-1}, s_1). Then, we go in order of $\mathbf{S}_x$ to pair up the rest. We pick the first unmatched s_k and pair it to the first term of *opposite* parity. We follow this till s_{i+1} is paired. These sets are illustrated in Fig. 2.

$$\begin{aligned} E_+(i) &= (s_{-1}, s_1), (s_0, s_3), (s_2, s_5), \ldots, (s_{i-4}, s_{i-1}), (s_{i-2}, s_{i+1}) \\ &= \{(s_{-1}, s_1)\} \cup \{(s_{i'}, s_{i'+3}) : i' \text{ even}, 0 \le i' \le i-2\} \end{aligned}$$

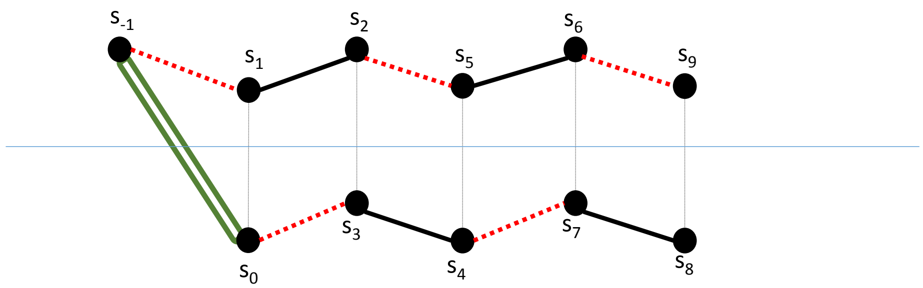

Fig. 2: Illustration for $i = 8$. The light vertical edges are H-edges. The dark black ones are $st_H(\mathbf{M})$-pairs. The green, double-lined one on the left is the starting M-pair. The dotted red pairs form $E_+(8)$. All points alove the horizonatal line are in $U(H)$, the ones below are in $L(H)$.

PROPOSITION 3.4. *$E_-(i)$ involves $s_{-1}, s_0, \ldots, s_i$, while $E_+(i)$ involves $s_{-1}, s_0, \ldots, s_{i-1}, s_{i+1}$.*

4. THE STRUCTURE OF $\mathbf{S}_x$ FOR MONOTONICITY

We now focus on monotonicity, and show that $\mathbf{S}_x$ is highly structured. (The proof for general Lipschitz will also follow the same setup, but requires more definitions.) The weight of a pair (x, y) is defined to be $f(x)-f(y)$ if $x \prec y$, and is $-\infty$ otherwise. We will assume that all function values are distinct. This is without loss of generality although we prove it formally later in Claim 8.9. Thus violating pairs have positive weight. We choose a maximum weight matching $\mathbf{M}$ of pairs. Note that every pair in $\mathbf{M}$ is a violating pair. We remind the reader that for even k, $(s_k, s_{k+1}) \in H$ and for odd k, $(s_k, s_{k+1}) \in st_H(\mathbf{M})$.

4.1. Preliminary observations

PROPOSITION 4.1. *For all $x, y \in L(H)$ (or $U(H)$), $x \prec y$ iff $H(x) \prec H(y)$. Consider pair $(x, y) \in cr_H(\mathbf{M})$ such that $x \prec y$. Then $H(x) \prec y$ and $x \prec H(y)$.*

PROOF. For any point in $x \in L(H)$, $H(x)$ is obtained by adding the H-distance to a specific coordinate. This proves the first part. The H-distance divides $|[y[r] - x[r]|$ (where H is aligned in dimension r) and $(x, y), x \prec y$ is a cross pair. Hence $y[r] - x[r]$ is at least the H-distance. Note that $H(x)$ is obtained by simply adding this distance to the r coordinate of x, so $H(x) \prec y$. $\square$

PROPOSITION 4.2. *All pairs in $E_-(i)$ and $E_+(i)$ are comparable. Furthermore, $s_1 \prec s_{-1}$ and for all even $0 \le k \le i-2$, $s_k \prec s_{k+3}$ iff $s_{k+1} \prec s_{k+2}$.*

PROOF. All pairs in $E_-(k)$ are in $\mathbf{M}$, and hence comparable. Consider pair $(s_{-1}, s_1) \in E_+(k)$. Since $s_1 = H(s_0)$ and (s_0, s_1) is a cross-pair, by Prop. 4.1, $s_1 \prec s_{-1}$. Consider pair (s_k, s_{k+3}), where k is even. (Refer to Fig. 2.) The pair $(H(s_k), H(s_{k+3})) = (s_{k+1}, s_{k+2})$ is in $st_H(\mathbf{M})$. Hence, the points are comparable and both lie in $L(H)$ or $U(H)$. By Prop. 4.1, s_k, s_{k+3} inherit their comparability from s_{k+1}, s_{k+2}. $\square$

For some even i, suppose (s_i, s_{i+1}) is a not a violation. Corollary 3.3 implies

$$\begin{aligned} &\text{If } i \equiv 0 \pmod 4, f(s_{i+1}) - f(s_i) > 0. \\ &\text{If } i \equiv 2 \pmod 4, f(s_i) - f(s_{i+1}) > 0. \end{aligned} \tag{$*$}$$

We will also state an *ordering* condition on the sequence.

$$\begin{aligned} &\text{If } i \equiv 0 \pmod 4,\ \ s_i \prec s_{i-1}. \\ &\text{If } i \equiv 2 \pmod 4,\ \ s_i \succ s_{i-1}. \end{aligned} \tag{$**$}$$

Remember these conditions and Corollary 3.3 together as follows. If $i \equiv 0 \pmod 4$, s_i is on smaller side, otherwise it is on the larger side. In other words, if $i \equiv 0 \pmod 4$, s_i is smaller than its "neighbors" in $\mathbf{S}_x$. For $i \equiv 2 \pmod 4$, it is bigger. For condition $(*)$, if $i \equiv 0 \pmod 4$, $f(s_i) < f(s_{i-1})$.

4.2. The structure lemmas

We will prove a series of lemmas that prove structural properties of $\mathbf{S}_x$ that are intimately connected to conditions $(*)$ and $(**)$. These proofs are where much of the insight lies.

LEMMA 4.3. *Consider some even index i such that s_i exists. Suppose conditions $(*)$ and $(**)$ held for all even indices $\leq i$. Then, s_{i+1} is $\mathbf{M}$-matched.*

PROOF. The proof is by contradiction, so assume that $\mathbf{M}(s_{i+1})$ does not exist. Assume $i \equiv 0 \pmod 4$. (The proof for the case $i \equiv 2 \pmod 4$ is similar and omitted.) Consider sets $E_-(i)$ and $E_+(i)$. Note that $s_{-1}, s_0, s_1, \ldots, s_{i+1}$ are all distinct. By Prop. 3.4, $\mathbf{M}' = \mathbf{M} - E_-(i) + E_+(i)$ is a valid matching. We will argue that $w(\mathbf{M}') > w(\mathbf{M})$, a contradiction. By condition $(**)$,

$$\begin{aligned} w(E_-(i)) \ = \ & [f(s_0) - f(s_{-1})] + [f(s_1) - f(s_2)] + [f(s_4) - f(s_3)] + \cdots \\ & \cdots + [f(s_{i-3}) - f(s_{i-2})] + [f(s_i) - f(s_{i-1})] \end{aligned} \tag{1}$$

By the second part of Prop. 4.2 (for even k, $s_k \prec s_{k+3}$ iff $s_{k+1} \prec s_{k+2}$) and condition $(**)$, we know the comparisons for all pairs in $E_+(i)$.

$$\begin{aligned} w(E_+(i+2)) \ = \ & [f(s_1) - f(s_{-1})] + [f(s_0) - f(s_3)] + [f(s_5) - f(s_2)] + \cdots \\ & \cdots + [f(s_{i-4}) - f(s_{i-1})] + [f(s_{i+1}) - f(s_{i-2})] \end{aligned} \tag{2}$$

Note that the coefficients of common terms in $w(E_+(i))$ and $w(E_-(i))$ are identical. The only terms not involves (by Prop. 3.4) are $f(s_{i+1})$ in $w(E_+(i))$ and $f(s_i)$ in $w(E_-(i))$. The weight of the new matching is precisely $w(\mathbf{M}) - W_- + W_+ = w(\mathbf{M}) + f(s_{i+1}) - f(s_i)$. By $(*)$ for i, this is strictly greater than $w(\mathbf{M})$, contradicting the maximality of $\mathbf{M}$. $\square$

So, under the condition of Lemma 4.3, s_{i+1} is $\mathbf{M}$-matched. We can also specify the comparison relation of s_{i+1}, $\mathbf{M}(s_{i+1})$ (as condition $(**)$) using an almost identical argument. Abusing notation, we will denote $\mathbf{M}(s_{i+1})$ as s_{i+2}. (This is no abuse if $(s_{i+1}, \mathbf{M}(s_{i+1}))$ is a straight pair.)

LEMMA 4.4. *Consider some even index i such that s_i exists. Suppose conditions $(*)$ and $(**)$ held for all even indices $\leq i$. Then, condition $(**)$ holds for $i+2$.*

Before we prove this lemma, we need the following distinctness claim.

CLAIM 4.5. *Consider some odd j such that s_j and $\mathbf{M}(s_j)$ exist. Suppose condition $(*)$ and $(**)$ held for all even $i < j$. Then the sequence $s_{-1}, s_0, s_1, \ldots, s_j, \mathbf{M}(s_j)$ are distinct.*

PROOF. (If $(s_j, \mathbf{M}(s_j)) \in st_H(\mathbf{M})$, this is obviously true. The challenge is when $\mathbf{S}_x$ terminates at s_j.) The sequence from s_0 to s_j is an alternating path, so all terms are distinct. If $s_j \neq y$, then the claim holds. Suppose $s_j = y$. Note that $j > 1$, since $(x, y) \notin H$.

Since $y \in U(H)$, by Prop. 3.2, $j \equiv 1 \pmod 4$. Condition ($**$) holds for $j-1$, so $s_{j-1} \prec s_j = y$ and by Corollary 3.3, $s_{j-1} \prec s_{j-2}$. Note that $(s_{j-1}, s_j) \in H$ and (x, s_j) is a cross pair. By Prop. 4.1, $x \prec s_{j-1}$ and thus $x \prec s_{j-2}$. We replace pairs $A = \{(x,y), (s_{j-2}, s_{j-1})\} \in \mathbf{M}$ with (x, s_{j-2}), and argue that the weight has increased. We have $w(A) = [f(x) - f(y)] + [f(s_{j-1}) - f(s_{j-2})] = [f(x) - f(s_{j-2})] - [f(y) - f(s_{j-1})]$. By condition ($*$) on i, $f(y) = f(s_j) > f(s_{j-1})$, contradicting the maximality of $\mathbf{M}$. $\square$

Proof. (of Lemma 4.4) By Lemma 4.3, $\mathbf{M}(s_{i+1})$ exists. Assume $i \equiv 0 \pmod 4$ (the other case is analogous and omitted). The proof is again by contradiction, so we assume condition ($**$) does not hold for $i+2$. This means $s_{i+2} = \mathbf{M}(s_{i+1}) \prec s_{i+1}$. Consider sets $E_-(i+2)$ and $E' = E_+(i-2) \cup (s_{i-2}, s_{i+2})$. By Claim 4.5, $s_{-1}, s_0, s_1, \ldots, s_{i+2}$ are distinct. So $\mathbf{M}' = \mathbf{M} - E_-(i) + E'$ is a valid matching and we argue that $w(\mathbf{M}') > w(\mathbf{M})$. By condition ($**$) for even $i' < i+2$ and the assumption $s_{i+2} \prec s_{i+1}$.

$$\begin{aligned} w(E_-(i+2)) &= [f(s_0) - f(s_{-1})] + [f(s_1) - f(s_2)] + [f(s_4) - f(s_3)] + \cdots \\ &\quad \cdots + [f(s_{i-3}) - f(s_{i-2})] + [f(s_i) - f(s_{i-1})] + [f(s_{i+2}) - f(s_{i+1})] \end{aligned}$$

Observe how the last term in the summation differs from the trend. All comparisons in $E_+(i-2)$ are determined by Prop. 3.4, just as we argued in the proof of Lemma 4.3. The expression for $w(E_+(i-2))$ is basically given in (2). It remains to deal with (s_{i-2}, s_{i+2}). By condition ($**$) for i, $s_i \prec s_{i-1}$. Thus, by Prop. 3.4, $s_{i+1} \prec s_{i-2}$. Combining with the assumption of $s_{i+2} \prec s_{i+1}$, we deduce $s_{i+2} \prec s_{i-2}$.

$$\begin{aligned} w(E_+(i+2)) &= [f(s_1) - f(s_{-1})] + [f(s_0) - f(s_3)] + [f(s_5) - f(s_2)] + \cdots \\ &\quad \cdots + [f(s_{i-3}) - f(s_{i-6})] + [f(s_{i-4}) - f(s_{i-1})] + [f(s_{i+2}) - f(s_{i-2})] \end{aligned}$$

The coefficients are identical, except that $f(s_i)$ and $f(s_{i+1})$ do not appear in $w(E_+(i+2))$. We get $w(\mathbf{M}) - W_- + W_+ = w(\mathbf{M}) + f(s_{i+1}) - f(s_i)$. By ($*$) for i, we contradict the maximality of $\mathbf{M}$. $\square$

A direct combination of the above statements yields the main structure lemma.

Lemma 4.6. *Suppose $\mathbf{S}_x$ contains no violated H-pair. Let the last term by s_j (j is odd). For every even $i \leq j+1$, condition ($**$) holds, and s_j belongs to a pair in $sk_H(\mathbf{M})$.*

Proof. We prove the first statement by contradiction. Consider the smallest even $i \leq j+1$ where condition ($**$) does not hold. Note that for $i = 0$, the condition does hold, so $i \geq 2$. We can apply Lemma 4.4 for $i-2$, since all even indices at most $i-2$ satisfy ($*$) and ($**$). But condition ($**$) holds for i, completing the proof. Now apply Lemma 4.3 and Lemma 4.4 for $j-1$. Conditions ($*$) and ($**$) hold for all relevant even indices. Hence, s_j must be $\mathbf{M}$-matched and condition ($**$) holds for $j+1$. Since $\mathbf{S}_x$ terminates at s_j, s_j cannot be $st_H(\mathbf{M})$-matched. Suppose s_j was $cr_H(\mathbf{M})$ matched. Let $j \equiv 1 \pmod 4$. By Prop. 3.2, $s_j \in U(H)$, so $s_{j+1} = \mathbf{M}(s_j) \prec s_j$, violating condition ($**$). A similar argument holds when $j \equiv 3 \pmod 4$. Hence, s_j must be $sk_H(\mathbf{M})$-matched. $\square$

5. MONOTONICITY ON BOOLEAN HYPERCUBE

We prove Theorem 1.1. Since $\mathbf{M}$ is also is a maximal family of disjoint violating pairs, and therefore, $|\mathbf{M}| \geq \frac{1}{2}\varepsilon_f \cdot 2^n$. We denote the set of all edges of the hypercube as $\mathbf{H}$. We partition $\mathbf{H}$ into $H_1, \ldots, H_n$ where H_r is the collection of hypercube edges which differ in the rth coordinate. Each H_r is a perfect matching and is adequate. Note that $st_{H_r}(\mathbf{M})$ is the set of $\mathbf{M}$-pairs which do not differ in the rth coordinate. The H-distance is trivially 1, so $cr_{H_r}(\mathbf{M})$ is the set of $\mathbf{M}$-pairs that differ in the rth coordinate. Importantly, $sk_{H_r}(\mathbf{M}) = \emptyset$.

Lemma 5.1. *For all $1 \leq r \leq n$, the number of violating H_r-edges is at least $cr_{H_r}(\mathbf{M})/2$.*

PROOF. Feed in $\mathbf{M}$ and H_r to the alternating path machinery. Set X to be the set of all lower endpoints of $cr_{H_r}(\mathbf{M}) \setminus H_r$, so $|X| = |cr_{H_r}(\mathbf{M}) \setminus H_r|/2$. Since $sk_{H_r}(\mathbf{M}) = \emptyset$, by Lemma 4.6, all sequences $\mathbf{S}_x$ must contain a violated H_r-edge. The total number of violated H_r-edges is at least $|X| + |cr_{H_r}(\mathbf{M}) \cap H_r|$. □

The above lemma proves Theorem 1.1. Observe that every pair in $\mathbf{M}$ belongs to some set $cr_{H_r}(\mathbf{M})$. The edge tester only requires $O(n/\varepsilon)$ queries, since the success probability of a single test is at least

$$\frac{1}{|\mathbf{H}|}\sum_{r=1}^{n} cr_{H_r}(\mathbf{M})/2 \geq |\mathbf{M}|/(n2^{n-2}) \geq \varepsilon/2n.$$

6. SETTING UP FOR HYPERGRIDS

We setup the framework for hypergrid domains. The arguments here are property independent. Consider domain $[k]^n$ and set $\ell = \lceil \lg k \rceil$. We define $\mathbf{H}$ to be pairs that differ in exactly one coordinate, and furthermore, the difference is a power of 2. The tester chooses a pair in $\mathbf{H}$ uniformly at random, and checks the property on this pair. We partition $\mathbf{H}$ into $n(\ell+1)$ sets $H_{a,b}$, $1 \leq a \leq n$, $0 \leq b \leq \ell$. $H_{a,b}$ consists of pairs (x, y) which differ only in the ath coordinate, and furthermore $|y[a] - x[a]| = 2^b$. Unfortunately, $H_{a,b}$ is not a matching, since each point can participate in potentially two pairs in $H_{a,b}$. To remedy this, we further partition $H_{a,b}$ into $H^0_{a,b}$ and $H^1_{a,b}$. For any pair $(x, y) \in H_{a,b}$, exactly one among $x[a] \pmod{2^{b+1}}$[3] and $y[a] \pmod{2^{b+1}}$ is $> 2^b$ and one is $\leq 2^b$. We put $(x, y) \in H_{a,b}$ with $x \prec y$ in $H^0_{a,b}$ if $y[a] \pmod{2^{b+1}} > 2^b$, and in the set $H^1_{a,b}$ if $1 \leq y[a] \pmod{2^{b+1}} \leq 2^b$. For example, $H_{1,0}$ has all pairs that only differ by $2^0 = 1$ in the first coordinate. We partition these pairs depending on whether the higher endpoint has even or odd first coordinate. Note that each $H^0_{a,b}$ and $H^1_{a,b}$ are matchings. We have $L(H^0_{a,b}) = \{x | x[a] \pmod{2^{b+1}} \leq 2^b\}$ and $U(H^0_{a,b}) = \{y | y[a] \pmod{2^{b+1}} > 2^b\}$. The sets are exactly switched for $H^1_{a,b}$. Because of the matchings are not perfect, we are forced to introduce the notion of adequacy of matchings. A matching H is adequate if for every violation (x, y), both x and y participate in the matching H (Definition 3.1). We will eventually prove the following theorem.

THEOREM 6.1. *Let k be a power of 2. Suppose for every violation (x, y) and every coordinate a, $|y[a] - x[a]| \leq 2^c$ (for some c). Furthermore, suppose that for $b \leq c$, all matchings $H^0_{a,b}, H^1_{a,b}$ are adequate. Then there exists a maximal matching $\mathbf{M}$ of the violation graph such that the number of violating pairs in $\mathbf{H}$ is at least $|\mathbf{M}|/2$.*

We reduce to this special case using a simple padding argument. The following theorem implies Theorem 1.2.

THEOREM 6.2. *Consider any function $f : [k]^n \mapsto R$. At least an $\varepsilon_f/(4n(\lceil \log k \rceil + 1)$-fraction of pairs in $\mathbf{H}$ are violations.*

PROOF. Let $\hat{k} = 2^\ell$ be the smallest power of 2 larger than $4k$. Let us construct a function $\hat{f} : [\hat{k}]^n \mapsto \mathbf{R} \cup \{-\infty, +\infty\}$. Let $\mathbf{1}$ denote the n-dimensional vector all 1s vector. For x such that all $x_i \in [\hat{k}/4+1, \hat{k}/4+k-1]$, we set $\hat{f}(x) = f(x - \frac{\hat{k}\cdot\mathbf{1}}{4})$. (We will refer to this region as the "original domain".) If any coordinate of x is less than $\hat{k}/4$, we set $\hat{f}(x) = -\infty$. Otherwise, we set $f(x) = +\infty$. All violations are contained in the original domain. For any violation (x, y) and coordinate a, $|y[a] - x[a]| \leq k < 2^{\ell-2}$. Let $\hat{\mathbf{H}}$ be the corresponding set of pairs in domain $[\hat{k}]^n$. For $b \leq \ell - 2$ (and every a), every point in the original domain participates in

[3]We abuse notation and define $p \pmod{2^{b+1}}$ to be 2^{b+1} (instead of 0) if $2^{b+1} \mid p$.

all matchings in $\hat{\mathbf{H}}$. So, each of these matchings is adequate. Since every maximal matching of the violation graph has size at least $\varepsilon_f k^n/2$, by Theorem 6.1, the number of violating pairs in $\hat{\mathbf{H}}$ is at least $\varepsilon_f k^n/2$. The matching $\mathbf{H}$ is exactly the set of pairs of $\hat{\mathbf{H}}$ completely contained in the original domain. All violating pairs in $\hat{\mathbf{H}}$ are contained in $\mathbf{H}$. The total size of $\mathbf{H}$ is at most $nk^n(\lceil \log k \rceil + 1)$. The proof is completed by dividing $\varepsilon_f k^n/4$ by the size of $\mathbf{H}$. $\square$

Henceforth, we will assume that $k = 2^\ell$ and that all matchings $H^0_{a,b}, H^1_{a,b}$ are adequate (for $b \leq c$, where 2^c is an upper bound on the coordinate difference for any violation).

6.1. The potential Φ

Define $\mathtt{msd}(a)$ of a non-negative integer a to be the largest power of 2 which divides a. That is, $\mathtt{msd}(a) = p$ implies $2^p \mid a$ but $2^{p+1} \nmid a$. We define $\mathtt{msd}(0) := \ell + 1$. For any $x \in \mathbb{Z}^n$, define $\Phi(x) = \sum_{c=1}^{n} \mathtt{msd}(|x[c]|)$. Now given a matching $\mathbf{M}$, define the following potential.

$$\Phi(\mathbf{M}) := \sum_{(x,y)\in\mathbf{M}} \Phi(x-y) = \sum_{(x,y)\in\mathbf{M}} \sum_{c=1}^{n} \mathtt{msd}(|y[c]-x[c]|). \tag{3}$$

We will choose maximum weighted matchings that also maximize $\Phi(\mathbf{M})$. To give some intuition for the potential, note that it is aligned towards picking pairs which differ in as few coordinates as possible (since $\mathtt{msd}(0)$ is large). Furthermore, divisibility by powers of 2 is favored.

7. MONOTONICITY ON HYPERGRIDS

In this section, we prove Theorem 1.2. As in the hypercube case, the weight of a pair (x, y) is defined to be $f(x) - f(y)$ if $x \prec y$, and $-\infty$ otherwise. We set $\mathbf{M}$ to be a maximum weighted matching that maximizes $\Phi(\mathbf{M})$. So $|\mathbf{M}| \geq \varepsilon_f k^n/2$. Fix $H^r_{a,b}$. It is instructive to explicitly see the pairs in $st_{H^r_{a,b}}(\mathbf{M})$ and $cr_{H^r_{a,b}}(\mathbf{M})$. Consider a pair (x, y), $x \prec y$ in these sets.

— $st_{H^r_{a,b}}(\mathbf{M})$: $x[a], y[a] \pmod{2^{b+1}} \leq 2^b$, or $x[a], y[a] \pmod{2^{b+1}} > 2^b$.

— $cr_{H^r_{a,b}}(\mathbf{M})$: $\mathtt{msd}(|y[a]-x[a]|) = b$, $x \in L(H^r_{a,b})$ (thus $y \in U(H^r_{a,b})$).

Now we do have skew pairs, and the potential Φ was designed specifically to handle such pairs. Note that every pair in $\mathbf{M}$ belongs to some $cr_{H^r_{a,b}}(\mathbf{M})$. There exists some a, b such that $\mathtt{msd}(|y[a]-x[a]|) = b$. If $x[a] \pmod{2^{b+1}} \leq 2^b$, then $(x,y) \in cr_{H^0_{a,b}}(\mathbf{M})$, otherwise $(x, y) \in cr_{H^1_{a,b}}(\mathbf{M})$. Therefore, the following lemma directly implies Theorem 6.1.

LEMMA 7.1. *For all r, a, b, the number of violated $H^r_{a,b}$-pairs is at least $|cr_{H^r_{a,b}}(\mathbf{M})|/2$.*

PROOF. We assume that $H^r_{a,b}$ is adequate. Feed in $H^r_{a,b}$ and $\mathbf{M}$ to the alternating paths machinery, with X as the set of lower endpoints in $cr_{H^r_{a,b}}(\mathbf{M}) \setminus H^r_{a,b}$. By Lemma 4.6, if a sequence $\mathbf{S}_x$ does not contain a violating $H^r_{a,b}$-pair, then the last term s_j must belong to $sk_{H^r_{a,b}}(\mathbf{M})$. By Lemma 7.2, $\mathtt{msd}(|s_j[a]-\mathbf{M}(s_j)[a]|) > b$. But then both s_j and $\mathbf{M}(s_j)$ belong to $L(H^r_{a,b})$ or $U(H^r_{a,b})$, implying $(s_j, \mathbf{M}(s_j)) \in st_H(\mathbf{M})$. Contradiction. Every sequence $\mathbf{S}_x$ contains a violating $H^r_{a,b}$-pair, and the calculation in Lemma 5.1 completes the proof. $\square$

The main technical work is in the proof of Lemma 7.2. Fix a, b, r. For convenience, we lose all superscripts and subscripts.

LEMMA 7.2. *Suppose $\mathbf{S}_x$ contains no violated H-pair. Let the last term be s_j (j is odd). Then $\mathtt{msd}(|s_j[a]-\mathbf{M}(s_j)[a]|) > b$.*

PROOF. For convenience, we denote $s_{j+1} = \mathbf{M}(s_j)$. We prove by contradiction, so $\texttt{msd}(|s_j[a] - s_{j+1}[a]|) \leq b$. By Lemma 4.6, for all even $i \leq j+1$, condition ($**$) holds and s_j belongs to an H-skew pair. We will rewire $\mathbf{M}$ to $\mathbf{M}'$ such that weight remains the same but the potential increases. We will remove the set $E_-(j+1)$ from $\mathbf{M}$ and add the set $\hat{E} = E_+(j-1) \cup (s_{j-1}, s_{j+1})$. Observe that both $E_-(j+1)$ and $\hat{E}$ involve all terms in $s_{-1}, \ldots, s_{j+1}$. We will assume that $j \equiv 1 \pmod 4$ (the other case is analogous and omitted). By ($**$),

$$w(E_-(j+1)) = [f(s_0)-f(s_{-1})]+[f(s_1)-f(s_2)]+[f(s_4)-f(s_3)]+\cdots+[f(s_{j-1})-f(s_{j-2})]+[f(s_j)-f(s_{j+1})]$$

Now for $w(\hat{E})$, all pairs other than (s_{j-1}, s_{j+1}) have their order decided by Prop. 3.4. By ($**$) for $j-1$ and Corollary 3.3 for $j+1$, $s_{j-1} \prec s_j \prec s_{j+1}$.

$$w(\hat{E}) = [f(s_1)-f(s_{-1})]+[f(s_0)-f(s_3)]+[f(s_5)-f(s_2)]+\cdots+[f(s_j)-f(s_{j-3})]+[f(s_{j-1})-f(s_{j+1})]$$

We get $w(E_-(j+1)) = w(\hat{E})$, so the weight stays the same. It remains the argue that the potential has increased, as argued in Claim 7.3 $\square$

CLAIM 7.3. *Suppose* $\texttt{msd}(|s_j[a] - s_{j+1}[a]|) \leq b$. *Then* $\Phi(\hat{E}) > \Phi(E_-(j+1))$.

PROOF. Consider $(s_{j'}, s_{j'+1})$ for odd $-1 < j' < j$. Both these terms are either in $L(H)$ or $U(H)$. Hence, $\Phi(s_{j'} - s_{j'+1}) = \Phi(H(s_{j'}) - H(s_{j'+1})) = \Phi(s_{j'-1} - s_{j'+2})$. So most quantities in $\Phi(E_-(j+1))$ and $\Phi(\hat{E})$ are identical.

$$\Phi(\hat{E}) - \Phi(E_-(j+1)) = \Phi(s_{-1} - s_1) + \Phi(s_{j+1} - s_{j-1}) - [\Phi(s_{-1} - s_0) + \Phi(s_j - s_{j+1})]$$

Since $s_1 = H(s_0)$, the points $s_{-1} - s_1$ and $s_{-1} - s_0$ only differ in the ath coordinate. A similar argument works for the remaining terms. Using $|\cdot|_a$ to denote the absolute value of the ath coordinate,

$$\Phi(\hat{E})-\Phi(E_-(j+1)) = \texttt{msd}(|s_{-1}-s_1|_a)+\texttt{msd}(|s_{j+1}-s_{j-1}|_a)-[\texttt{msd}(|s_{-1}-s_0|_a)+\texttt{msd}(|s_j-s_{j+1})|_a]$$

Note that $\texttt{msd}(|s_{-1} - s_0|_a) = b$, by definition, since it lies in $cr_{H^0_{a,b}}(\mathbf{M})$. Furthermore $|s_{-1} - s_1|_a = |s_{-1} - H(s_0)|_a = |s_{-1} - s_0|_a - 2^b$, so $\texttt{msd}(|s_{-1} - s_1|_a) > b$. (Note the strict inequality.) It suffices to show that $\texttt{msd}(|s_{j+1} - s_{j-1}|_a) \geq \texttt{msd}(|s_j - s_{j+1}|_a)$. Because $s_{j-1} = H(s_j)$, $|s_{j+1} - s_{j-1}|_a$ is either $|2^b + |s_j - s_{j+1}|_a|$ or $|2^b - |s_j - s_{j+1}|_a|$. In either case, the assumption $\texttt{msd}(|s_j - s_{j+1}|_a) \leq b$ implies $\texttt{msd}(|s_{j+1} - s_{j-1}|_a) \geq \texttt{msd}(|s_j - s_{j+1}|_a)$. $\square$

8. A PSEUDO-DISTANCE FOR (α, β)-LIPSCHITZ

A key concept that unifies Lipschitz and monotonicity is a pseudo-distance defined on $\mathbf{D}$. The challenge faced in the final proof is tweezing out all the places in the previous argument where the distance function is "hidden". We define a weighted directed graph $\mathsf{G} = (\mathbf{D}, E)$ where $\mathbf{D}$ is the hypergrid $[k]^n$. 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 8.1. The function $\mathsf{d}(x, y)$ between $x, y \in \mathbf{D}$ is the shortest path length from x to y in G.

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 is not 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 $x, y \in \mathbf{D}$, we define $\mathsf{hcd}(x, y)$ to be the $z \in \mathbf{D}$ maximizing $||z||_1$ such that $x \succ z$ and $y \succ z$. Note that if $x \succ y$ then $\mathsf{hcd}(x, y) = y$.

CLAIM 8.2. *For any* $x, y \in \mathbf{D}$, $\mathsf{d}(x, y) = \beta||x - \mathsf{hcd}(x, y)||_1 - \alpha||y - \mathsf{hcd}(x, y)||_1$.

PROOF. Let us partition the coordinate set $[n] = A \sqcup B \sqcup C$ with the following property. For all $i \in A$, $x_i > y_i$. For all $i \in B$, $x_i < y_i$, 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 decrement all coordinates in A, increment all coordinates in B, and preserve coordinates in C. Furthermore, increments add $-\alpha$ to the path length, and decrements add β. Fix a path, and let I_i and D_i denote the number of increments and decrements in dimension i. For $i \in A$, $D_i = I_i + |x_i - y_i|$, for $i \in B$, $I_i = D_i + |x_i - y_i|$, and for $i \in C$, $I_i = D_i$. The path length is given by

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

For the inequality, we use $\beta \geq \alpha$. Let $z = \mathsf{hcd}(x, y)$. Note that $z_i = \min(x_i, y_i)$. 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 $\beta \sum_{i \in A}(x_i - y_i) - \alpha \sum_{i \in B}(y_i - x_i)$. $\square$

It is instructive see the distance for monotonicity and Lipschitz. In the case of monotonicity (when $\alpha = 0, \beta = \infty$), $\mathsf{d}(x, y) = 0$ if $x \prec y$ and $\mathsf{d}(x, y) = \infty$ otherwise. In the case of Lipschitz, $\mathsf{d}(x, y) = ||x - y||_1$. The next two claims establish some properties of the pseudo-distance.

CLAIM 8.3.
— *(Triangle equality) Fix x, y. Suppose z has the property that for all coordinates a, $z[a]$ lies in $[x[a], y[a]]$ or $[y[a], x[a]]$ (whichever is valid). Then, $\mathsf{d}(x, y) = \mathsf{d}(x, z) + \mathsf{d}(z, y)$.*
— *(Triangle inequality) $\mathsf{d}(x, y) \leq \mathsf{d}(x, z) + \mathsf{d}(z, y)$.*
— *(Projection)Let v be a vector with a single non-zero coordinate. Let $x' = x + v$ and $y' = y + v$. Then $\mathsf{d}(x, y) = \mathsf{d}(x', y')$.*
— *(Positivity) Consider a "cycle" of distinct points $x_1, x_2, \ldots, x_s, x_{s+1} = x_1$ Then $\sum_{c=1}^{s} \mathsf{d}(x_c, x_{c+1}) > 0$.*

PROOF. The triangle equality property follows from Claim 8.2. 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 (say) the ath coordinate by the same amount v_a. Thus, $||x - z||_1 = ||x' - z'||_1$ and $||y - z||_1 = ||y' - z'||_1$, implying $\mathsf{d}(x, y) = \mathsf{d}(x', y')$. For positivity, note that $\mathsf{d}(s_c, s_{c+1})$ is the length of a path in G. So $\sum_{c=1}^{s} \mathsf{d}(x_c, x_{c+1})$ is length of a non-trivial cycle in G. Each coordinate increment adds $-\alpha$ to the length, and a decrement adds β. The number of increments and decrements are the same, so the length is a strictly positive multiple of $\beta - \alpha$, a strictly positive quantity. $\square$

The following lemma connects the distance to the (α, β)-Lipschitz property.

LEMMA 8.4. *A function is (α, β)-Lipschitz iff for all $x, y \in \mathbf{D}$, $f(x) - f(y) - \mathsf{d}(x, y) \leq 0$.*

PROOF. Suppose the function satisfied the inequality for all x, y. If x and y differ in one-coordinate by 1 with $x \prec y$, we get $f(y) - f(x) \leq \beta = \mathsf{d}(y, x)$ and $f(y) - f(x) \geq \alpha = -\mathsf{d}(x, y)$ implying f is (α, β)-Lipschitz. Conversely, suppose f is (α, β)-Lipschitz. Setting $z = \mathsf{hcd}(x, y)$, $f(x) - f(z) \leq \beta||x - z||_1$ and $\alpha||y - z||_1 \leq f(y) - f(z)$. Summing these, $f(x) - f(y) \leq \beta||x - z||_1 - \alpha||y - z||_1 = \mathsf{d}(x, y)$. $\square$

We give a simple, but important fact about distances related to the function values.

CLAIM 8.5. $\min(f(x) - f(y) - \mathsf{d}(x, y), f(y) - f(x) - \mathsf{d}(y, x)) < 0$.

PROOF. Suppose not. Then $f(x) - f(y) - \mathsf{d}(x, y) + f(y) - f(x) - \mathsf{d}(y, x) \geq 0$, implying $\mathsf{d}(x, y) + \mathsf{d}(y, x) \leq 0$. This violates the positivity of Claim 8.3. $\square$

The next lemma is a generalization of Theorem 2.1, which argued that the size of a minimum vertex cover is exactly $\varepsilon_f|\mathbf{D}|$. We crucially use the triangle inequality for $\mathsf{d}(x, y)$. We define an undirected weighted clique on $\mathbf{D}$. Given a function f, we define the weight $w(x, y)$ (for any $x, y \in \mathbf{D}$) as follows:

$$w(x, y) \;:=\; \max\Big(f(x) - f(y) - \mathsf{d}(x, y),\;\; f(y) - f(x) - \mathsf{d}(y, x)\Big) \tag{4}$$

Note that although the distance d is asymmetric, the weight is symmetric. Lemma 8.4 shows that a function is (α, β)-Lipschitz iff all $w(x, y) \leq 0$. Once again, consider the special cases of monotonicity and Lipschitz. For monotonicity, $w(x, y) = f(x) - f(y)$ when $x \prec y$ and $-\infty$ otherwise. For Lipschitz, $w(x, y) = |f(x) - f(y)| - ||x - y||_1$. We define the unweighted *violation graph* as $VG_f = (\mathbf{D}, E)$ where $E = \{(x, y) : w(x, y) > 0\}$. The following lemma generalizes Theorem 2.1 from [Fischer et al. 2002].

LEMMA 8.6. *The size of a minimum vertex cover in VG_f is exactly $\varepsilon_f|\mathbf{D}|$.*

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, $\varepsilon_f|\mathbf{D}| \geq |U|$. We show how to modify the function values at U to get a function f' with no violations. We invoke the following claim with $V = \mathbf{D} - U$, and $f'(x) = f(x), \forall x \in V$. This gives a function f' such that $\Delta(f, f') = |U|/|\mathbf{D}|$. By Lemma 8.4, f' is (α, β)-Lipschitz, and $|U| \geq \varepsilon_f|\mathbf{D}|$. Hence, $|U| = \varepsilon_f|\mathbf{D}|$.

CLAIM 8.7. *Consider partial function f' defined on a subset $V \subseteq \mathbf{D}$, 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 $\forall x, y \in \mathbf{D}$, $f'(x) - f'(y) \leq \mathsf{d}(x, y)$.*

PROOF. We prove by backwards induction on the size of V. If $|V| = |\mathbf{D}|$, this is trivially true. Now for the induction step. It suffices define f' for some $u \notin V$. We need to define $f'(u)$ so that $f'(u) - f'(y) \leq \mathsf{d}(u, y)$ and $f'(x) - f'(u) \leq \mathsf{d}(x, u)$ for all $x, y \in V$. It suffices to 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). Contradiction, so $m \leq M$. $\square$

The following is a simple corollary of the previous lemma.

COROLLARY 8.8. *The size of any maximal matching in VG_f is at least $\frac{1}{2}\varepsilon_f|\mathbf{D}|$.*

By a perturbation argument, we can assume that $w(x, y)$ is never exactly zero. This justifies the strict inequalities used in the monotonicity proofs.

CLAIM 8.9. *For any function f, there exists a function f' with the following properties. Both f and f' have the same set of violated pairs, $\varepsilon_f = \varepsilon_{f'}$, and for all $x, y \in \mathbf{D}$, $w_{f'}(x, y) \neq 0$.*

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. By Lemma 8.6, $\varepsilon_f = \varepsilon_{f'}$. Construct the following digraph T on $\mathbf{D}$. For every x, y such that $f(x) - f(y) - \mathsf{d}(x,y) = 0$, put a directed edge from y to x. Suppose there is a cycle $x_1, x_2, \ldots, x_s, x_{s+1} = x_1$ in this digraph. Then $\sum_{c=1}^{s}[f(s_c) - f(s_{c+1}) - \mathsf{d}(s_c, s_{c+1})] = -\sum_{c=1}^{s} \mathsf{d}(s_c, s_{c+1}) = 0$. This violates the positivity of Claim 8.3, so T is a DAG. Pick a sink s. For any x, $f(x) - f(s) - \mathsf{d}(x,s)$ is non-zero. Infinitesimally decrease $f(s)$ (call the new function f'). For all x, $w_{f'}(x,s)$ has the same sign as $w_f(x,s)$ and is strictly negative if $w_f(x,s) = 0$. By iterating in this manner, we generate the desired function f'. $\square$

9. GENERALIZED LIPSCHITZ TESTING ON HYPERGRIDS

In this section, we prove Theorem 1.5. With the distance $\mathsf{d}(x,y)$ in place, the basic spirit of the monotonicity proofs can be carried over. The final proof requires manipulations of the distance function. We do not explicitly have the "directed" behavior of monotonicity that allows for many of rewiring arguments. The matching $\mathbf{H}$ is the same as in §6. The generalized Lipschitz tester picks a pair $(x,y) \in \mathbf{H}$ at random. We choose $\mathbf{M}$ to be the maximum weight matching that also maximizes $\Phi(M)$ (as defined by (3)). We again set up the alternating paths as in §3, by fixing some matching $H^r_{a,b}$ and taking alternating paths with $st_{H^r_{a,b}}(\mathbf{M})$. We have a minor change that aids in some case analysis. By Claim 8.5, either $f(x) - f(y) > \mathsf{d}(x,y)$ or $f(y) - f(x) > \mathsf{d}(y,x)$, but not both. We will show that it suffices to consider only one of these cases. To that effect, define the set X as follows.

$$X = \{x | (x,y) \in cr_{H^r_{a,b}}(\mathbf{M}) \setminus H^r_{a,b}, x \in L(H^r_{a,b}), f(x) - f(y) > \mathsf{d}(x,y)\}$$

(For monotonicity, the last condition is redundant.) As before, the main lemma is the following.

LEMMA 9.1. *For all $x \in X$, $\mathbf{S}_x$ contains a violated $H^r_{a,b}$-pair.*

We apply some symmetry arguments to show the next lemma, which proves Theorem 6.1. For convenience, we drop the sub/superscripts in $H^r_{a,b}$. (Note that we do not lose the 2 factor here, as compared to Lemma 5.1.)

LEMMA 9.2. *The number of violations in H is at least $cr_H(\mathbf{M})$.*

PROOF. We can classify the endpoints of $cr_H(\mathbf{M}) \setminus H^r_{a,b}$ into the following sets. Consider a generic $(x,y) \in cr_H(\mathbf{M})$ where $x \in L(H)$. If $f(x) - f(y) > \mathsf{d}(x,y)$, we put x in X and y in Y. Otherwise, $f(y) - f(x) > \mathsf{d}(y,x)$, and we put x in X' and y in Y'. By Lemma 9.1, for $x \in X$, $\mathbf{S}_x$ has a violated H-pair. Consider $x' \in X'$. Take the function $\hat{f} = -f$ and the $(-\beta, -\alpha)$-Lipschitz property. By Claim 8.2, the new distance satisfies $\hat{\mathsf{d}}(u,v) = \mathsf{d}(v,u)$. If $f(u) - f(v) > \mathsf{d}(u,v)$, then $\hat{f}(v) - \hat{f}(u) > \hat{\mathsf{d}}(v,u)$ (and vice versa). Hence, the violation graphs, the weights, $\mathbf{M}$, and the alternating paths are identical. Take $x' \in X'$, so it belongs to some $(x', y') \in cr_H(\mathbf{M})$. We have $\hat{f}(x) - \hat{f}(y) > \hat{\mathsf{d}}(x,y)$. Applying Lemma 9.1 to $\hat{f}$ for the $(-\beta, -\alpha)$-Lipschitz property, $\mathbf{S}_{x'}$ has a violated H-pair. All in all, for any $x \in X \cup X'$, $\mathbf{S}_x$ contains a violated H-pair. To deal with $Y \cup Y'$, we will first reverse the entire domain, by switching the direction of all edges in the hypergrid. (Represent this transformation by $\Psi : [k]^n \to [k]^n$, and note that $\Psi^{-1} = \Psi$.) By the shortest path definition of d, the new distance satisfies $\hat{\mathsf{d}}(u,v) = \mathsf{d}(\Psi(v), \Psi(u))$. Hence, we are looking at the $(-\beta, -\alpha)$-Lipschitz property. The matching H remains the same, but the identities of $L(H)$ and $U(H)$ have switched. Construct function $\hat{f}(x) = -f(\Psi(x))$. If in the original domain $f(u) - f(v) > \mathsf{d}(u,v)$, then $\hat{f}(\Psi(v)) - \hat{f}(\Psi(u)) > \hat{\mathsf{d}}(\Psi(v), \Psi(u))$ (and vice versa). Again, the alternating path structure is identical. Consider in the original domain $(x,y) \in cr_H(\mathbf{M})$ where $x \in L(H)$. In the new domain, $\Psi(y) \in L(H)$. Hence, we can apply the conclusion of the previous paragraph for all points in $y \in \Psi(Y \cup Y')$, and deduce that $\mathbf{S}_y$ contains a violated H-pair. Finally, we conclude that every alternating path with an endpoint of $cr_H(\mathbf{M}) \setminus H^r_{a,b}$ contains a violated pair. There are at least $|cr_H(\mathbf{M}) \setminus H^r_{a,b}|$ such (disjoint) alternating paths. $\square$

9.1. Preliminary setup

All the propositions of §3 hold, since they were independent of the property at hand. We start by generalizing the monotonicity-specific setup done in §4. We fix some matching $H^r_{a,b}$, and drop all super/subscripts for ease of notation.

PROPOSITION 9.3. *Consider the pairs in $E_-(i)$ and $E^+(i)$. For all even $0 \le j \le i-2$, $\mathsf{d}(s_j, s_{j+3}) = \mathsf{d}(s_{j+1}, s_{j+2})$ and $\mathsf{d}(s_{j+3}, s_j) = \mathsf{d}(s_{j+2}, s_{j+1})$.*

PROOF. By Prop. 3.2, s_j and s_{j+3} both lie in $L(H)$ or $U(H)$. Hence, $s_{j+1} = H(s_j)$ and $s_{j+2} = H(s_{j+3})$ are both obtained by adding or subtracting 2^b from the ath coordinate. By the projection property, $\mathsf{d}(s_j, s_{j+3}) = \mathsf{d}(s_{j+1}, s_{j+2})$ and $\mathsf{d}(s_{j+3}, s_j) = \mathsf{d}(s_{j+2}, s_{j+1})$. $\square$

Our aim is to generalize the conditions ($*$) and ($**$). The former condition is obtained by assuming that (s_i, s_{i+1}) is not a violation. For monotonicity, this implies a single inequality, but here, there are two inequalities. It turns out that because we are in the setting where $w(x,y) = f(x) - f(y) - \mathsf{d}(x,y) > 0$, only one of these is necessary. For even i, if (s_i, s_{i+1}) is not a violation, Corollary 3.3 implies

$$\begin{aligned} &\text{If } i \equiv 0 \pmod 4, f(s_{i+1}) - f(s_i) > \alpha 2^b. \\ &\text{If } i \equiv 2 \pmod 4, f(s_i) - f(s_{i+1}) > \alpha 2^b. \end{aligned} \tag{$\circ$}$$

Nowe we generalize ($**$). The pair (s_{i-1}, s_i) is a violation, but we do not know whether $w(s_{i-1}, s_i)$ is $f(s_{i-1}) - f(s_i) - \mathsf{d}(s_{i-1}, s_i)$ or $f(s_i) - f(s_{i-1}) - \mathsf{d}(s_i, s_{i-1})$. The following is the equivalent of the ordering condition of ($**$).

$$\begin{aligned} &\text{If } i \equiv 0 \pmod 4, f(s_i) - f(s_{i-1}) > \mathsf{d}(s_i, s_{i-1}). \\ &\text{If } i \equiv 2 \pmod 4, f(s_{i-1}) - f(s_i) > \mathsf{d}(s_{i-1}, s_i). \end{aligned} \tag{$\circ\circ$}$$

9.2. The structure lemmas

This lemma is the direct analogue of Lemma 4.3. The proof is also along similar lines.

LEMMA 9.4. *Consider some even index i such that s_i exists. Suppose conditions ($\circ$) and ($\circ\circ$) held for all even indices $\le i$. Then, s_{i+1} is $\mathbf{M}$-matched.*

PROOF. The proof is by contradiction. Assume $i \equiv 0 \pmod 4$. (The proof for the case $i \equiv 2 \pmod 4$ is similar and omitted.) As in the proof of Lemma 4.3, we argue that $w(\mathbf{M}') > w(\mathbf{M})$, where $\mathbf{M}' = \mathbf{M} - E_-(i) + E_+(i)$. By condition ($**$),

$$\begin{aligned} w(E_-(i)) \;=\; & [f(s_0) - f(s_{-1}) - \mathsf{d}(s_0, s_{-1})] + [f(s_1) - f(s_2) - \mathsf{d}(s_1, s_2)] \\ & + [f(s_4) - f(s_3) - \mathsf{d}(s_4, s_3)] + [f(s_5) - f(s_6) - \mathsf{d}(s_5, s_6)] + \cdots \\ & + [f(s_{i-3}) - f(s_{i-2}) - \mathsf{d}(s_{i-3}, s_{i-2})] + [f(s_i) - f(s_{i-1}) - \mathsf{d}(s_i, s_{i-1})] \end{aligned} \tag{5}$$

For $w(E_+(i))$, it suffices to find a lower bound. Since (for any $u, v \in \mathbf{D}$) $w(u,v)$ is the maximum of two expressions, we can choose the expression to match $w(E_-(i))$ as much as possible. For a pair (s_k, s_{k+3}) in $E_+(i)$, we bound the weight by $f(s_k) - f(s_{k+3}) - \mathsf{d}(s_k, s_{k+3})$ if $j \equiv 0 \pmod 4$ and by $f(s_{k+3}) - f(s_k) - \mathsf{d}(s_{k+3}, s_k)$ if $j \equiv 2 \pmod 4$. This ensure that the coefficients of $f(\cdot)$ are identical to those in (5).

$$\begin{aligned} w(E_+(i)) \;\ge\; & [f(s_1) - f(s_{-1}) - \mathsf{d}(s_1, s_{-1})] + [f(s_0) - f(s_3) - \mathsf{d}(s_0, s_3)] \\ & + [f(s_5) - f(s_2) - \mathsf{d}(s_5, s_2)] + [f(s_4) - f(s_7) - \mathsf{d}(s_4, s_7)] + \cdots \\ & + [f(s_{i-4}) - f(s_{i-1}) - \mathsf{d}(s_{i-4}, s_{i-1})] + [f(s_{i+1}) - f(s_{i-2}) - \mathsf{d}(s_{i+1}, s_{i-2})] \end{aligned} \tag{6}$$

Note that only $w(E_+(i))$ involves $f(s_{i+1})$ and only $w(E_-(i))$ involves $f(s_i)$, but all other $f(\cdot)$ terms have identical coefficients. To deal with the difference of the distances, we use Prop. 9.3. All the distance terms in (6) except for the first cancel out with an equivalent term in (5).

$$w(E_+(i)) - w(E_-(i)) \;\geq\; f(s_{i+1}) - f(s_i) - \mathsf{d}(s_1, s_{-1}) + \mathsf{d}(s_0, s_{-1})$$

Since (s_0, s_{-1}) is a cross pair and $s_1 = H(s_0)$, we can use triangle equality to deduce that $\mathsf{d}(s_0, s_{-1}) - \mathsf{d}(s_1, s_{-1}) = \mathsf{d}(s_0, s_1) = -\alpha 2^b$. Combining, $w(E_+(i)) - w(E_-(i)) \geq f(s_{i+1}) - f(s_i) - \alpha 2^b$. By condition ($\circ$) for i, the RHS is strictly positive. Contradiction. $\square$

Now for analogue of Lemma 4.4 and Claim 4.5. We will prove the latter first.

LEMMA 9.5. *Consider some even index i such that s_i exists. Suppose conditions ($\circ$) and ($\circ\circ$) held for all even indices $\leq i$. Then, condition ($\circ\circ$) holds for $i + 2$.*

CLAIM 9.6. *Let j be the last index of $\mathbf{S}_x$. Suppose conditions ($\circ$) and ($\circ\circ$) hold for all even $i < j$. Then the sequence $s_{-1}, s_0, s_1, \ldots, s_j, \mathbf{M}(s_j)$ are distinct.*

PROOF. By the arguments in Claim 4.5, it suffices to get a contradiction assuming $s_j = y$. Since $y \in U(H)$, by Prop. 3.2, $j \equiv 1 \pmod 4$. Note that $s_{j-1} = H(y)$ and (x, y) is a cross pair. Therefore, we have the triangle equality $\mathsf{d}(x, y) = \mathsf{d}(x, s_{j-1}) + \mathsf{d}(s_{j-1}, y) = \mathsf{d}(x, s_{j-1}) - \alpha 2^b$. We will replace pairs $A = \{(x, y), (s_{j-1}, s_{j-2})\} \in \mathbf{M}$ with (x, s_{j-2}), and argue that the weight has increased. Applying condition ($\circ\circ$) for $j - 1$,

$$\begin{aligned}
w(A) &= [f(x) - f(y) - \mathsf{d}(x, y)] + [f(s_{j-1}) - f(s_{j-2}) - \mathsf{d}(s_{j-1}, s_{j-2})] \\
&= f(x) - f(y) + f(s_{j-1}) - f(s_{j-2}) - \mathsf{d}(x, s_{j-1}) + \alpha 2^b - \mathsf{d}(s_{j-1}, s_{j-2}) \\
&\leq f(x) - f(y) + f(s_{j-1}) - f(s_{j-2}) - \mathsf{d}(x, s_{j-2}) + \alpha 2^b \qquad \text{(triangle inequality)} \\
&= [f(x) - f(s_{j-2}) - \mathsf{d}(x, s_{j-2})] - [f(y) - f(s_{j-1}) - \alpha 2^b] \\
&\leq w(x, s_{j-2}) - [f(y) - f(s_{j-1}) - \alpha 2^b]
\end{aligned}$$

The second term is strictly positive (by condition ($\circ$) for $j - 1 \equiv 0 \pmod 4$), contradicting the maximality of $\mathbf{M}$. $\square$

PROOF. (of Lemma 9.5) Assume $i \equiv 0 \pmod 4$. (The proof for the case $i \equiv 2 \pmod 4$ is similar and omitted.) By Lemma 9.4, $\mathbf{M}(s_{i+1})$ exists, and is denoted by s_{i+2}. The proof is by contradiction, so assume condition ($\circ\circ$) does not hold for $i + 2 \equiv 2 \pmod 4$. This means $f(s_{i+1}) - f(s_{i+2}) \leq \mathsf{d}(s_{i+1}, s_{i+2})$. Since (s_{i+1}, s_{i+2}) is a violation, this implies $w(s_{i+1}, s_{i+2}) = f(s_{i+}) - f(s_{i+1}) - \mathsf{d}(s_{i+2}, s_{i+1})$. We set $E' = E_+(i - 2) \cup (s_{i-2}, s_{i+2})$. We argue that $w(\mathbf{M}') > w(\mathbf{M})$, where $\mathbf{M}' = \mathbf{M} - E_-(i + 2) + E'$. By Prop. 3.4 and Claim 9.6, $\mathbf{M}'$ is a valid matching. By condition ($\circ\circ$) for even $k < i + 2$ and the above conclusion on $w(s_{i+1}, s_{i+2})$, we get almost the same expression as (5).

$$\begin{aligned}
w(E_-(i + 2)) \;=\; & [f(s_0) - f(s_{-1}) - \mathsf{d}(s_0, s_{-1})] + [f(s_1) - f(s_2) - \mathsf{d}(s_1, s_2)] \\
& + [f(s_4) - f(s_3) - \mathsf{d}(s_4, s_3)] + [f(s_5) - f(s_6) - \mathsf{d}(s_5, s_6)] + \cdots \\
& + [f(s_{i-3}) - f(s_{i-2}) - \mathsf{d}(s_{i-3}, s_{i-2})] + [f(s_i) - f(s_{i-1}) - \mathsf{d}(s_i, s_{i-1})] \\
& + [f(s_{i+2}) - f(s_{i+1}) - \mathsf{d}(s_{i+2}, s_{i+1})] \qquad (7)
\end{aligned}$$

For $w(E')$, we follow the same pattern in (6).

$$\begin{aligned}
w(E') \;\geq\; & [f(s_1) - f(s_{-1}) - \mathsf{d}(s_1, s_{-1})] + [f(s_0) - f(s_3) - \mathsf{d}(s_0, s_3)] \\
& + [f(s_5) - f(s_2) - \mathsf{d}(s_5, s_2)] + [f(s_4) - f(s_7) - \mathsf{d}(s_4, s_7)] + \cdots \\
& + [f(s_{i-3}) - f(s_{i-6}) - \mathsf{d}(s_{i-3}, s_{i-6})] + [f(s_{i-4}) - f(s_{i-1}) - \mathsf{d}(s_{i-4}, s_{i-1})] \\
& + [f(s_{i+2}) - f(s_{i-2}) - \mathsf{d}(s_{i+2}, s_{i-2})] \qquad (8)
\end{aligned}$$

By Prop. 9.3, all distance terms in (6) barring the first and last are identical to an equivalent term in (5).

$$\begin{aligned} w(E_+(i+2)) - w(E_-(i+2)) \ \geq \ & f(s_{i+1}) - f(s_i) \\ & -\mathsf{d}(s_1, s_{-1}) - \mathsf{d}(s_{i+2}, s_{i-2}) + \mathsf{d}(s_0, s_{-1}) + \mathsf{d}(s_i, s_{i-1}) + \mathsf{d}(s_{i+2}, s_{i+1}) \end{aligned}$$

As in the proof of Lemma 9.4, $\mathsf{d}(s_0, s_{-1}) - \mathsf{d}(s_1, s_{-1}) = \mathsf{d}(s_0, s_1) = -\alpha 2^b$. Furthermore,

$$\begin{aligned} -\mathsf{d}(s_{i+2}, s_{i-2}) + \mathsf{d}(s_i, s_{i-1}) + \mathsf{d}(s_{i+2}, s_{i+1}) \ \geq \ & \mathsf{d}(s_i, s_{i-1}) - \mathsf{d}(s_{i+1}, s_{i-2}) \quad \text{(triangle inequality)} \\ = \ & 0 \quad \text{(Prop. 9.3)} \end{aligned}$$

Combining, $w(E') - w(E_-(i+2)) \geq f(s_{i+1}) - f(s_i) - \alpha 2^b$. This is strictly positive, by condition ($\circ$) for i. Contradiction. □

We proceed to the analogue of Lemma 4.6. Because of the use of distances and potentials, we require a much simpler statement.

LEMMA 9.7. *Suppose $\mathbf{S}_x$ contains no violated H-pair. Let the last term by s_j (j is odd). For every even $i \leq j+1$, condition ($\circ\circ$) holds. Furthermore, s_j is $\mathbf{M} \setminus st_H(\mathbf{M})$-matched.*

PROOF. The first part is identical to that of Lemma 4.6. Condition ($\circ\circ$) holds for $i = 0$, and applications of Lemma 9.5 complete the proof. By Lemma 9.4 s_j is $\mathbf{M}$-matched, but being the last term cannot be $st_H(\mathbf{M})$-matched. □

9.3. The existence of a violated edge in $\mathbf{S}_x$

We show the existence of a violated H-edge in $\mathbf{S}_x$, proving Lemma 9.1. Suppose $\mathbf{S}_x$ has no violated H-pair. By Lemma 9.7, s_j is $\mathbf{M} \setminus st_H(\mathbf{M})$-matched. By the following lemma (analogue of Lemma 7.2) asserts $\mathtt{msd}(s_j[a] - s_{j+1}[a]) > b$, implying s_j is $st_H(\mathbf{M})$-matched.

LEMMA 9.8. *Suppose $\mathbf{S}_x$ contains no violated H-pair. Let the last term by s_j (j is odd). Then $\mathtt{msd}(s_j[a] - s_{j+1}[a]) > b$.*

PROOF. The proof is analogous to that of Lemma 7.2. By Lemma 9.7, for all even $i \leq j+1$, condition ($\circ\circ$) holds. By Claim 9.6, $s_{-1}, s_0, s_1, \ldots, s_j, \mathbf{M}(s_j) = s_{j+1}$ are all distinct. We rewire $\mathbf{M}$ to $\mathbf{M}'$ by removing $E_-(j+1)$ from $\mathbf{M}$ and adding the set $\hat{E} = E_+(j-1) \cup (s_{j-1}, s_{j+1})$. We will assume that $j \equiv 1 \pmod 4$ (the other case is analogous and omitted). By ($\circ\circ$), we can exactly express $w(E_-(j+1))$.

$$\begin{aligned} w(E_-(j+1)) \ = \ & [f(s_0) - f(s_{-1}) - \mathsf{d}(s_0, s_{-1})] + [f(s_1) - f(s_2) - \mathsf{d}(s_1, s_2)] \\ & +[f(s_4) - f(s_3) - \mathsf{d}(s_4, s_3)] + [f(s_5) - f(s_6) - \mathsf{d}(s_5, s_6)] + \cdots \\ & +[f(s_{j-1}) - f(s_{j-2}) - \mathsf{d}(s_{j-1}, s_{j-2})] + [f(s_j) - f(s_{j+1}) - \mathsf{d}(s_j, s_{j+1})] \end{aligned}$$

We get a lower bound for $w(\hat{E})$ that matches the f terms exactly.

$$\begin{aligned} w(\hat{E}) \ \geq \ & [f(s_1) - f(s_{-1}) - \mathsf{d}(s_1, s_{-1})] + [f(s_0) - f(s_3) - \mathsf{d}(s_0, s_3)] \\ & +[f(s_5) - f(s_2) - \mathsf{d}(s_5, s_2)] + [f(s_4) - f(s_7) - \mathsf{d}(s_4, s_7)] + \cdots \\ & +[f(s_j) - f(s_{j-3}) - \mathsf{d}(s_j, s_{j-3})] + [f(s_{j-1}) - f(s_{j+1}) - \mathsf{d}(s_{j-1}, s_{j+1})] \end{aligned}$$

By Prop. 9.3, the distance terms $\mathsf{d}(s_c, s_{c+3})$ and $\mathsf{d}(s_{c+3}, s_c)$ can be matched to equivalent terms. In the following, we use the equality $\mathsf{d}(s_0, s_{-1}) - \mathsf{d}(s_1, s_{-1}) = -\alpha 2^b$.

$$\begin{aligned} w(\hat{E}) - w(E_-(j+1)) \ \geq \ & -\mathsf{d}(s_1, s_{-1}) - \mathsf{d}(s_{j-1}, s_{j+1}) + \mathsf{d}(s_0, s_{-1}) + \mathsf{d}(s_j, s_{j+1}) \\ \geq \ & -\alpha 2^b - \mathsf{d}(s_{j-1}, s_j) \quad \text{(triangle inequality)} \\ = \ & -\alpha 2^b - (-\alpha 2^b) = 0 \quad \text{(By Prop. 3.2, } j \equiv 1 \pmod 4\text{, so } s_j \in U(H).) \end{aligned}$$

So $\mathbf{M}'$ is also a maximum weight matching. Observe that the potential Φ is independent of the property at hand. Claim 7.3 only uses the basic structure of the alternating paths and is applicable here. It asserts that if $\mathtt{msd}(s_j[a] - s_{j+1}[a]) \leq b$, then $\Phi(\mathbf{M}') > \Phi(\mathbf{M})$, contradicting the choice of $\mathbf{M}$. $\square$

10. ACKNOWLEDGEMENTS

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. CS is grateful for the support received from the Early Career LDRD program at Sandia National Laboratories.

REFERENCES

2011. Open problems in data streams, property testing, and related topics. http://sublinear.info/files/bertinoro2011_kanpur2009.pdf. 2

AILON, N. AND CHAZELLE, B. 2006. Information theory in property testing and monotonicity testing in higher dimension. *Information and Computation 204,* 11, 1704–1717. 2, 3

AILON, N., CHAZELLE, B., COMANDUR, S., AND LIU, D. 2006. Estimating the distance to a monotone function. *Random Structures and Algorithms 31,* 3, 1704–1711. 2

AWASTHI, P., JHA, M., MOLINARO, M., AND RASKHODNIKOVA, S. 2012. Testing Lipschitz functions on hypergrid domains. In *Proceedings of the International Workshop on Randomization and Computation (RANDOM)*. 2

BATU, T., RUBINFELD, R., AND WHITE, P. 2005. Fast approximate *PCP*s for multidimensional bin-packing problems. *Information and Computation 196,* 1, 42–56. 2

BHATTACHARYYA, A., GRIGORESCU, E., JHA, M., JUNG, K., RASKHODNIKOVA, S., AND WOODRUFF, D. 2012. Lower bounds for local monotonicity reconstruction from transitive-closure spanners. *SIAM Journal of Discrete Math 26,* 2, 618–646. Conference version in RANDOM 2010. 3

BHATTACHARYYA, A., GRIGORESCU, E., JUNG, K., RASKHODNIKOVA, S., AND WOODRUFF, D. 2009. Transitive-closure spanners. In *Proceedings of the 18th Annual Symposium on Discrete Algorithms (SODA)*. 531–540. 2, 3

BLAIS, E., BRODY, J., AND MATULEF, K. 2012. Property testing lower bounds via communication complexity. *Computational Complexity 21,* 2, 311–358. 2, 3

BLAIS, E., RASKHODNIKOVA, S., AND YAROSLAVTSEV, G. 2014. Lower bounds for testing properties of functions on hypergrid domains. In *Conference on Computational Complexity*. 2

BRIËT, J., CHAKRABORTY, S., GARCÍA-SORIANO, D., AND MATSLIAH, A. 2012. Monotonicity testing and shortest-path routing on the cube. *Combinatorica 32,* 1, 35–53. 2, 3, 4

CHAKRABARTY, D. AND SESHADHRI, C. 2013a. Optimal bounds for monotonicity and Lipschitz testing over hypercubes and hypergrids. In *Proceedings of Symposium on Theory of Computing (STOC)*. 1

CHAKRABARTY, D. AND SESHADHRI, C. 2013b. An optimal lower bound for monotonicity testing over hypergrids. In *Proceedings of the International Workshop on Randomization and Computation (RANDOM)*. 2

CHAUDHURI, S., GULWANI, S., LUBLINERMAN, R., AND NAVIDPOUR, S. 2011. Proving programs robust. In *Joint Meeting of the European Software Engineering Conference and the ACM SIGSOFT Symposium on the Foundations of Software Engineering (ESEC/FSE)*. 3

DODIS, Y., GOLDREICH, O., LEHMAN, E., RASKHODNIKOVA, S., RON, D., AND SAMORODNITSKY, A. 1999. Improved testing algorithms for monotonicity. *Proceedings of the 3rd International Workshop on Randomization and Approximation Techniques in Computer Science (RANDOM)*, 97–108. 2

DWORK, C., MCSHERRY, F., NISSIM, K., AND SMITH, A. 2006. Calibrating noise to sensitivity in private data analysis. In *Proceedings of the Theory of Cryptography Conference (TCC)*. 3

ERGUN, F., KANNAN, S., KUMAR, R., RUBINFELD, R., AND VISWANATHAN, M. 2000. Spot-checkers. *Journal of Computer Systems and Sciences (JCSS) 60,* 3, 717–751. 2, 3

FISCHER, E. 2004. On the strength of comparisons in property testing. *Information and Computation 189,* 1, 107–116. 2, 3

FISCHER, E., LEHMAN, E., NEWMAN, I., RASKHODNIKOVA, S., RUBINFELD, R., AND SAMORODNITSKY, A. 2002. Monotonicity testing over general poset domains. In Proceedings of the 34th Annual ACM Symposium on the Theory of Computing (STOC). *Proceedings of the 34th Annual ACM Symposium on the Theory of Computing (STOC)*, 474–483. 2, 3, 4, 15

Goldreich, O., Goldwasser, S., Lehman, E., Ron, D., and Samorodnitsky, A. 2000. Testing monotonicity. *Combinatorica 20*, 301–337. 1, 2

Halevy, S. and Kushilevitz, E. 2008. Testing monotonicity over graph products. *Random Structures and Algorithms 33,* 1, 44–67. 2, 3

Jha, M. and Raskhodnikova, S. 2011. Testing and reconstruction of Lipschitz functions with applications to data privacy. In *Proceedings of the 52nd Annual Symposium on Foundations of Computer Science (FOCS)*. 433–442. 2, 3, 4

Lehman, E. and Ron, D. 2001. On disjoint chains of subsets. *Journal of Combinatorial Theory, Series A 94,* 2, 399–404. 2, 4

Parnas, M., Ron, D., and Rubinfeld, R. 2006. Tolerant property testing and distance approximation. *Journal of Computer and System Sciences 6,* 72, 1012–1042. 2

ECCC ISSN 1433-8092

http://eccc.hpi-web.de