Monotonicity Testing and Shortest-Path Routing on the Cube

Jop Briët* Sourav Chakraborty* David García-Soriano* Arie Matsliah*

Abstract

We study the problem of monotonicity testing over the hypercube. As previously observed in several works, a positive answer to a natural question about routing properties of the hypercube network would imply the existence of efficient monotonicity testers. In particular, if any ℓ source-sink pairs on the directed hypercube (with all the sources and all the sinks dinstinct) can be connected with edge-disjoint paths, then monotonicity of functions $f : \{0,1\}^n \to \mathcal{R}$ can be tested with $O(n)$ queries, for *any* totally ordered range $\mathcal{R}$. More generally, if at least an $\alpha(n)$ fraction of the pairs can always be connected with edge-disjoint paths then the query-complexity is $O(n/\alpha(n))$.

We construct a family of instances of $\ell = \Omega(2^n)$ pairs in n-dimensional hypercubes such that no more than roughly a $\frac{1}{\sqrt{n}}$ fraction of the pairs can be simultaneously connected with edge-disjoint paths. This answers an open question of Lehman and Ron [LR01], and suggests that the aforementioned appealing combinatorial approach for deriving query-complexity upper bounds from routing properties cannot yield, by itself, query-complexity bounds better than $\approx n^{3/2}$. Additionally, our construction can also be used to obtain a strong counterexample to Szymanski's conjecture on routing in the hypercube. In particular, we show that for any $\delta > 0$, the n-dimensional hypercube is not $n^{\frac{1}{2}-\delta}$-realizable with shortest paths, while previously it was only known that hypercubes are not 1-realizable with shortest paths.

We also prove a lower bound of $\Omega(n/\epsilon)$ queries for one-sided non-adaptive testing of monotonicity over the n-dimensional hypercube, as well as additional bounds for specific classes of functions and testers.

*Centrum Wiskunde & Informatica, Amsterdam. Email: {`jop.briet,sourav,david,ariem`}`@cwi.nl`.

1 Background

Testing monotonicity of functions [DGL$^+$99, GGL$^+$00, EKK$^+$00, FLN$^+$02, Fis04, AC06, Ras99, Bha08, HK08] is one of the oldest and most studied problems in Property Testing. The problem is defined as follows. Let $\mathcal{D}$ be a partially ordered set (poset) and let $\mathcal{R} \subseteq \mathbb{Z}$. A function $f : \mathcal{D} \to \mathcal{R}$ is monotone if for every (comparable) pair $x, y \in \mathcal{D}$, $x \leq y$ implies $f(x) \leq f(y)$. A function f is ϵ-far from monotone if it has to be changed on at least an ϵ-fraction of the domain $\mathcal{D}$ to become monotone. A (q, ϵ)-monotonicity tester for domain $\mathcal{D}$ and range $\mathcal{R}$ is a probabilistic algorithm that, given oracle access to a function $f : \mathcal{D} \to \mathcal{R}$, satisfies the following: (a) it makes at most q queries to f; (b) it accepts with probability at least $2/3$ if f is monotone; (c) it rejects with probability at least $2/3$ if f is ϵ-far from monotone.

The simplest monotonicity testers are those that specify all their queries in advance (non-adaptively) and reject if and only if they reveal a violation, i.e. if $f(x) > f(y)$ for some comparable pair $x \leq y$ of points queried from $\mathcal{D}$. These *non-adaptive* testers with *one-sided error* are the only ones considered in this paper, unless explicitly stated otherwise. We note that nearly all known monotonicity testers are non-adaptive and have one-sided error. Furthermore, it is also known that if $\mathcal{D}$ is totally ordered then non-adaptive testers with one-sided error are as powerful (in terms of query complexity) as general ones [Fis04].

For general domains $\mathcal{D}$, Fischer et al. [FLN$^+$02] proved that testing monotonicity is equivalent to several natural problems, including testing certain graph properties and testing assignments for Boolean formulae. Domains of the form $\{0, 1, \ldots, m\}^n$, however, received most of the attention [DGL$^+$99, EKK$^+$00, GGL$^+$00, Fis04, Ras99, Bha08, BGJ$^+$09]. Here a function $f : \{0, \ldots, m\}^n \to \mathcal{R}$ is monotone if for every $x, y \in \{0, \ldots, m\}^n$ such that $x_i \leq y_i$ for all $i \in [n]$, $f(x) \leq f(y)$ holds. In this paper we focus on a well-studied special subcase of the above, where $m = 1$ and $\mathcal{R} \subseteq \mathbb{Z}$.

1.1 Preliminaries

Every $x \in \{0, 1\}^n$ is identified with the subset $support(x) = \{i \in [n] : x_i = 1\}$ as usual. With a slight abuse of notation, we interpret binary strings as sets (and vice-versa). E.g., we write $x \subseteq y$ (or $x \leq y$) for two strings $x, y \in \{0, 1\}^n$ such that $support(x) \subseteq support(y)$.

The *directed n-dimensional hypercube* (or simply *n-cube*) is a directed graph $H_n = (V_n, E_n)$ with $V_n = \{0, 1\}^n$ and $E_n = \{(x, y) : x \subseteq y \text{ and } |y| = |x| + 1\}$. The h-th *layer* (or *level*) of H_n contains all $x \in V_n$ with $|x| = h$.

Definition 1.1 *A set $\mathcal{P} \subseteq V_n \times V_n$ of ℓ pairs $\{(s^i, t^i)\}_{i=1}^{\ell}$ is called a* source-sink pairing *(of size ℓ), with* sources $s^1, \ldots, s^\ell$ *and sinks $t^1, \ldots, t^\ell$, if*

- *$s^i \subset t^i$ for all $i \in [\ell]$ and*
- *$s^i \neq s^j$, $s^i \neq t^j$ and $t^i \neq t^j$ for all $i, j \in [\ell]$, $i \neq j$.*

$\mathcal{P}$ is aligned *if in addition $|s^i| = |s^j|$ and $|t^i| = |t^j|$ for all $i, j \in [\ell]$.*

Notice that $\mathcal{P}$ is a source-sink pairing if and only if it forms a (partial) matching in the transitive closure of H_n. Throughout this paper we denote by $\mathcal{P}$ only sets of pairs that form a source-sink pairing, even when it is not explicitly mentioned.

A (directed) path in H_n is called a *$\mathcal{P}$-path* if it connects some source s^i from $\mathcal{P}$ to its sink t^i. A subset $C \subseteq E_n$ is called a *$\mathcal{P}$-cut* if every $\mathcal{P}$-path in H_n uses at least one edge from C. Similarly, a subset $S \subseteq V_n$ is called a *$\mathcal{P}$-vertex-cut* if every $\mathcal{P}$-path uses at least one vertex from S. We write $\mathsf{maxflow}(\mathcal{P})$ for the size of the largest set of edge-disjoint $\mathcal{P}$-paths, $\mathsf{mincut}(\mathcal{P})$ for the size of the smallest $\mathcal{P}$-cut and $\mathsf{minvertexcut}(\mathcal{P})$ for the size of the smallest $\mathcal{P}$-vertex-cut. Clearly $\mathsf{mincut}(\mathcal{P})$ is an upper bound on both $\mathsf{minvertexcut}(\mathcal{P})$ and $\mathsf{maxflow}(\mathcal{P})$.[1]

We define the terms *sparsity* and *meagerness* as in [RL05, ABY08, AHJ$^+$06]. The *sparsity* of $\mathcal{P}$ is the ratio $\mathsf{mincut}(\mathcal{P})/|\mathcal{P}|$, and the *vertex-sparsity* of $\mathcal{P}$ is the ratio $\mathsf{minvertexcut}(\mathcal{P})/|\mathcal{P}|$. The *sparsity* and the *vertex-sparsity* of H_n are defined as $\min_{\mathcal{P}}\{\mathsf{mincut}(\mathcal{P})/|\mathcal{P}|\}$ and $\min_{\mathcal{P}}\{\mathsf{minvertexcut}(\mathcal{P})/|\mathcal{P}|\}$, respectively. The definitions of *meagerness* and *vertex-meagerness* are similar, except for the stronger requirement that the corresponding cuts disconnect *all* sources s^i from *all* sinks t^j.

Observe that (1) sparsity $\geq$ vertex-sparsity; (2) meagerness $\geq$ vertex-meagerness; (3) meagerness $\geq$ sparsity and (4) vertex-meagerness $\geq$ vertex-sparsity.

Given $\mathcal{R} \subseteq \mathbb{Z}$ and a function $f : \{0,1\}^n \to \mathcal{R}$, we say that a pair $(x,y) \in V_n \times V_n$ is *violated* by f if $x \leq y$ and $f(x) > f(y)$. If in addition $(x,y) \in E_n$, we call it a *violated edge*. We denote by $\mathsf{Viol}(f)$ the set of all pairs (x,y) violated by f, and by $\mathsf{EdgeViol}(f)$ the set of all *edges* violated by f. Thus, f is monotone if and only if $\mathsf{Viol}(f) = \mathsf{EdgeViol}(f) = \emptyset$.

We denote by $\epsilon_M(f) \in [0,1]$ the relative distance of f from being monotone, i.e. the minimum of $\Pr_x[f(x) \neq g(x)]$ taken over all monotone functions $g : \{0,1\}^n \to \mathcal{R}$. By $\delta_M(f) \in [0,1]$ we denote the fraction $|\mathsf{EdgeViol}(f)|/|E_n| = |\mathsf{EdgeViol}(f)|/(n2^{n-1})$ of edges violated by f.

2 Our results and related work

2.1 Monotonicity testers via sparsity lower bounds

One of the earliest upper bounds on the query-complexity of monotonicity testing on the hypercube used an approach based on the concepts of meagerness and sparsity [GGLR98]. In particular, [GGLR98] observed that if the meagerness of H_n is at least 1, then monotonicity of Boolean functions can be tested with $O(n/\epsilon)$ queries. Then they proved that vertex-meagerness (and hence meagerness too) is at least 1 if the possible pairings $\mathcal{P}$ are restricted to *aligned* sets, satisfying $|s^i| = |s^j|$ and $|t^i| = |t^j|$ for all i,j (see also [LR01] for a detailed proof). This sufficed to derive an upper-bound of $O(n^2)$ queries for any constant $\epsilon > 0$.

While a lower bound on meagerness implies query-complexity upper bounds for Boolean functions, a lower bound on sparsity implies query-complexity upper bounds for functions with general range (see Section 3.1 for details). In particular, if the sparsity of H_n is at least $\mu = \mu(n)$, then monotonicity of functions with any linearly ordered range can be tested with $O(n/(\epsilon\mu))$ queries. In [LR01] the authors ask whether the sparsity of any $\mathcal{P}$ (or even just of the aligned ones) is at least 1, noting that this would imply efficient monotonicity testing as well as progress on some long-standing questions regarding routing in the hypercube network. We prove that the answer to these questions is *no*.

Theorem 2.1 *The sparsity of H_n is at most $n^{-\frac{1}{2}+o(1)}$. Furthermore, this upper bound on the sparsity can be demonstrated with both aligned sets and $\Omega(2^n)$-sized sets:*

- *for any $\delta > 0$ and large enough n there is an* aligned *set $\mathcal{P}$ in H_n with sparsity at most $n^{-\frac{1}{2}+\delta}$;*
- *for any $\delta > 0$ there is $\epsilon > 0$, such that for large enough n there is a set $\mathcal{P}$ in H_n of size $|\mathcal{P}| \geq \epsilon 2^n$ with sparsity at most $n^{-\frac{1}{2}+\delta}$.*

[1] The quantities $\mathsf{mincut}(\mathcal{P})$ and $\mathsf{maxflow}(\mathcal{P})$ are duals of each other, when written in linear integer programming formulations [SM90]. Unlike the case with a single pair in $\mathcal{P}$, there is a gap between the integer and real-valued solutions, and these two quantities need not coincide.

Theorem 2.1 is proved in Section 3.2. In Appendix A we also show that the vertex-sparsity of H_n is $O(1/n)$.

2.2 Routing in the hypercube and Szymanski's conjecture

The hypercube is a natural and well-studied architecture for multi-processor systems and networks. The ability to route arbitrary permutations on it models flow of information in a network of processors. In this context, a doubly-directed version of H_n is usually considered, where each edge in E_n is replaced with a pair of anti-parallel edges. Let us denote the doubly-directed version of H_n by $H_n^{\uparrow\downarrow}$. A permutation π of V_n is 1-*realizable* if there exist pairwise edge-disjoint paths in $H_n^{\uparrow\downarrow}$ that connect every v with $\pi(v)$. A permutation π is k-*realizable* if there exist paths connecting every v with $\pi(v)$ such that each edge is used by at most k paths. Szymanski [Szy89] conjectured that any permutation π of V_n is 1-realizable with *shortest paths*. It was proved that the conjecture holds up to dimension 3, but later Lubiw [Lub90] provided a counterexample in dimension 5 that is not 1-realizable using shortest paths. While it is still unknown whether or not every permutation is 1-realizable *without* requiring shortest paths[2], the fact that any permutation is 2-realizable follows from the classical work of Beneš [Ben65] (see [Lub90] for details). In contrast, we prove that if we insist on the shortest-path condition, there are permutations that are not k-realizable for any k significantly smaller than $\sqrt{n}$. Specifically, the construction in Theorem 2.1 can be used (see Section 3.3) to prove the following.

Theorem 2.2 *For any $\delta > 0$ and large enough n, there are permutations on V_n that cannot be $n^{\frac{1}{2}-\delta}$-realized in $H_n^{\uparrow\downarrow}$ with shortest paths.*

Remark 2.3 *Any upper bound $\mu(n)$ on the sparsity of H_n can be used to show that $H_n^{\uparrow\downarrow}$ is not $1/\mu(n)$-realizable with shortest paths. However, the opposite is not true. In particular, the counterexample from [Lub90] does not imply that the sparsity of H_5 is less than 1.*

2.3 New bounds on testing monotonicity

Currently the best known query-complexity bounds for testing monotonicity (non-adaptively with one-sided error) of functions $f : \{0,1\}^n \to \mathcal{R}$ are:

- an upper bound of $O(\frac{n}{\epsilon}\log|\mathcal{R}|)$ for any range $\mathcal{R}$ [DGL+99];
- a lower bound of $\Omega(\sqrt{n}/\epsilon)$ for Boolean ranges (and hence larger ranges too) [FLN+02].

The tester used in the upper bound of [DGL+99] is perhaps the most natural one: it picks an edge $(x, y) \in E_n$ uniformly at random, and rejects if $f(x) > f(y)$. Let us call this test an *edge-test*. [DGL+99] prove that the probability that a single execution of an edge-test rejects is $\Omega(\frac{\epsilon_M(f)}{n\log|\mathcal{R}|})$, by relating the distance of a function from monotone to the number of edges that it violates.

It is an interesting open question whether the general upper bound of [DGL+99] can be improved into one that is independent of $|\mathcal{R}|$ (or at least has a better dependence on it). Since we can assume without loss of generality that $|\mathcal{R}| \le 2^n$, any upper bound of $o(n^2/\epsilon)$ queries would be an improvement. We make a small step in this direction. Call a function $f : \{0,1\}^n \to \mathcal{R}$ *dist-k monotone* if $f(y) \ge f(x)$ for every $y > x$ with $|y| > |x| + k$. In this terminology dist-0 monotone is simply monotone. In Section 3.4 we prove that given a dist-3 monotone function f, we can test if f is monotone with $O(n^{3/2}/\epsilon)$ queries. We actually prove the following stronger claim:

[2] Since the original conjecture was shown to be false, the weaker version that does not require shortest paths is now called Szymanski's conjecture.

Theorem 2.4 *Let $\epsilon > 0$, $\mathcal{R} \subseteq \mathbb{Z}$ and let $f : \{0,1\}^n \to \mathcal{R}$ be a dist-3 monotone function. If f is ϵ-far from being monotone then* $|\,\mathsf{EdgeViol}(f)| \geq \Omega(\frac{2^n}{\epsilon\sqrt{n}})$.

The upper bound on the query complexity follows using the edge-tests described above.

We remark that the case dist-3 is considered here because it is the first non-trivial one (it is easy to see that both dist-1 and dist-2 monotone functions can be tested in $O(n/\epsilon)$ queries).

In Section 3.5 we also extend the lower bound of $\Omega(\sqrt{n}/\epsilon)$ of [FLN+02] to $\Omega(n/\epsilon)$, for large enough $|\mathcal{R}|$. Using the "Range-Reduction Lemma" of [DGL+99], the new bound implies an improved lower bound of $\Omega(n/(\epsilon \log n))$ for the Boolean range, in the special case of pair-tests (tests that pick a set of comparable pairs (x, y) according to some distribution, and reject if and only if one of them forms a violation). We note that pair-tests are not overly restricted: essentially all known query-complexity upper bounds for monotonicity-testing use (or can be easily converted into ones that use) pair-tests. Furthermore, the new lower-bound almost matches the aforementioned upper-bound of $O(n/\epsilon)$ achieved by the edge-test (a special case of pair-tests).

3 Proofs

3.1 From sparsity to monotonicity testers

The basic combinatorial interpretation of $\epsilon_M(f)$ is given in the following lemma:

Lemma 3.1 *[DGL+99, FLN+02, GGL+00] Let $f : \{0,1\}^n \to \mathcal{R}$ be a function, and define the* violation graph *of f as the undirected graph $G = (\{0,1\}^n, E)$, where $\{x, y\} \in E$ if either (x, y) or (y, x) is in* $\mathsf{Viol}(f)$. *Then $\epsilon_M(f)2^n$ is exactly the size of a minimum vertex cover of G. Consequently, there is a matching in G of size at least $\epsilon_M(f)2^{n-1}$.*

An important observation is that since G is a subgraph of the transitive closure of H_n, the matching of violated pairs in Lemma 3.1 forms a source-sink pairing $\mathcal{P}$ (see Definition 1.1) of size $\epsilon_M(f)2^{n-1}$.

As we mentioned earlier, the best known upper bounds for testing monotonicity over hypercubes are obtained by a simple edge-tester, which picks a set of edges from H_n uniformly at random, queries f on their endpoints, and rejects if one of them is violated. Recall that $\delta_M(f)$ denotes the fraction of edges in H_n that are violated by f; thus the success probability of the edge-tester is determined by $\delta_M(f)$. Goldreich et al prove the following:

Theorem 3.2 *[GGLR98, GGL+00] For any $f : \{0,1\}^n \to \{0,1\}$, $\delta_M(f) \geq \frac{\epsilon_M(f)}{n}$.*

More generally, [DGL+99] use their range-reduction lemma to conclude that for any $f : \{0,1\}^n \to \mathcal{R}$, $\delta_M(f) \geq \frac{\epsilon_M(f)}{n \log |\mathcal{R}|}$. Since without loss of generality $|\mathcal{R}| \leq 2^n$, this gives an upper bound of $O(n^2/\epsilon)$ queries for testing monotonicity of all functions $f : \{0,1\}^n \to \mathcal{R}$.

Clearly, obtaining better lower bounds on $\delta_M(f)$ is sufficient for improving the upper bounds on the query complexity of testing monotonicity. (It may even be the case that Theorem 3.2 holds for any $\mathcal{R}$). The next lemma states that this can also be done by proving lower bounds on the sparsity of H_n.

Lemma 3.3 *Let $\mu(n)$ denote the sparsity of H_n. For any $\epsilon > 0$ and $\mathcal{R} \subseteq \mathbb{Z}$, monotonicity of functions $f : \{0,1\}^n \to \mathcal{R}$ can be tested with $O(\frac{n}{\epsilon\mu(n)})$ queries.*

Proof: Let $\epsilon > 0$ and let $f : \{0,1\}^n \to \mathcal{R}$ be ϵ-far from monotone. Let $\mathcal{P}$ be the set of $\epsilon_M(f)2^{n-1} \geq \epsilon 2^{n-1}$ vertex-disjoint violated pairs promised to exist by Lemma 3.1. By definition, $\mathcal{P}$ is a source-sink pairing. Notice that since every $(s^i, t^i) \in \mathcal{P}$ is violated, we have that every path from s^i to t^i must contain at least one violated edge. That is, every $\mathcal{P}$-path contains at least one violated edge. It follows that the set $\mathsf{EdgeViol}(f)$ is a $\mathcal{P}$-cut and that we have $|\mathsf{EdgeViol}(f)|/|\mathcal{P}| \geq \mu(n)$. Hence, we can lower bound $\delta_M(f)$ by $\delta_M(f) = \frac{|\mathsf{EdgeViol}(f)|}{|E_n|} \geq \frac{\epsilon\mu(n)}{2n}$. We can thus conclude that $O\big(\frac{n}{\epsilon\mu(n)}\big)$ edge queries suffice to find an edge-violation with constant probability. ■

3.2 Proof of Theorem 2.1

We use a number of properties of the parity-check matrix of Hamming codes, which we now describe. For an integer $k \geq 1$, let the strings $y \in \{0,1\}^k \backslash \{0\}$ represent the indices of bit positions of binary strings of length $n = 2^k - 1$. The Hamming code consists of the n-bit strings $x \in \{0,1\}^n$ that, for every $i \in [k]$, have an even number of positions y for which $y_i = 1$ and $x_y = 1$. The columns of its $k \times n$ parity check matrix p are all possible non-zero k-bit vectors y; this matrix represents a linear map $p : \{0,1\}^n \to \{0,1\}^k$, with arithmetic done modulo 2. Therefore, for any unit vector e_y (i.e., the vector having 1 at position y and 0 elsewhere), $p(e_y) = y$. In addition, for all x, y, $p(x \oplus e_y) = p(x) \oplus y$.

Codewords of the Hamming code correspond to strings satisfying $p(x) = 0$ (here and in what follows we use 0 to denote the all-zero vector of the appropriate size). The k bit positions of the form 2^i (i.e., $1, 2, 4, \ldots, (n+1)/2$) can be viewed as the parity bits of the code; in a codeword they are determined by the remaining $n - k$ bits.

3.2.1 Warm-up

To showcase the main ideas in the construction, we first show that the sparsity of the hypercube is at most $O(\frac{1}{n^{1/3}})$; better bounds are derived in Section 3.2.2.

Proposition 3.4 *Let $k > 0$ be a multiple of three, and $n = 2^k - 1$. There is a pairing $\mathcal{P} \subseteq V_n \times V_n$ in H_n of size $|\mathcal{P}| = \Omega(2^n)$ having a $\mathcal{P}$-cut $C \subseteq E_n$ of size $|C| = O(2^n/n^{1/3})$.*

Proof: For $a \in \{0,1\}^n$, consider the k parity bits $p(a)$ and divide them into three groups of size $k/3$ each, denoted $x(a), y(a)$ and $z(a)$. For convenience, we write (v_1, v_2, v_3) to denote the concatenation of three vectors $v_1, v_2, v_3 \in \{0,1\}^{k/3}$, and whenever no confusion may arise, we interpret every $v \in \{0,1\}^k$ as an element of $\{0\} \cup [n]$. With this convention, we have $p(a) = (x(a), y(a), z(a))$, and if one of v_1, v_2 or v_3 is non-zero, then $(v_1, v_2, v_3) \in [n]$.

The following set S is the set of sources of $\mathcal{P}$:

$$S = \Big\{ s \in \{0,1\}^n : \Big(x(s) \neq 0 \wedge y(s) \neq 0 \wedge z(s) \neq 0 \Big) \wedge \Big(s_{(x(s),y(s),0)} = s_{(x(s),0,z(s))} = s_{(0,y(s),z(s))} = 0 \Big) \Big\}.$$

For each source $s \in S$, we define the sink $t = s \cup \{(x(s), y(s), 0), (x(s), 0, z(s)), (0, y(s), z(s))\}$. That is, the three directions leading from s to t are $(x(s), y(s), 0), (x(s), 0, z(s))$ and $(x(s), 0, z(s))$. The first three conditions on a member s of S ensure that all three directions are (1) distinct; (2) proper (i.e. non-zero); and (3) have a k-bit binary representation with Hamming weight greater than one. The last condition ensures that the relevant bits of s are set to zero.

The pairing $\mathcal{P}$ will be given by all pairs (s, t) defined in this way. Clearly $s \subseteq t$ and $|t - s| = 3$. It is also easy to verify that $|S| = \left(2^{k/3} - 1\right)^3 2^{n-k-3} = \Omega(2^n)$, since none of the directions used corresponds to a parity bit, namely, none of them is a power of 2.

To prove that $\mathcal{P}$ is a pairing, it remains to show that all sources are distinct, and that no source is also a sink. Because of the properties of map p, after flipping e.g. bit $(x, y, 0)$ from a source s with parity (x, y, z), we reach a vertex with parity $(0, 0, z)$. If we continue the computation, we see that the parities of the eight vertices in the cube from s to t are:

- Level 3 (sink): (x, y, z).
- Level 2: $(x, 0, 0), (0, y, 0), (0, 0, z)$.
- Level 1: $(0, 0, z), (0, y, 0), (x, 0, 0)$.
- Level 0 (source): (x, y, z).

Notice that the parities at level 1 are distinct, as are the parities at level 2.

Since the three directions from s to t are determined by $p(s) = (x, y, z) = p(t)$, it follows immediately that the set of sinks is disjoint from the set of sources (these bits already belong to t, so $t \notin S$). Likewise, if two different sources s_1 and s_2 were associated with the same sink t, we would get $p(s_1) = p(t) = p(s_2)$, so the three directions from s_1 to t are the same as from s_2 to t, implying $s_1 = s_2$. Hence $\mathcal{P}$ is indeed a pairing.

Let $Q \subseteq V_n$ be the set of vertices that are at level 1 or 2 for some pair $(s, t) \in \mathcal{P}$ (that is, they lie on a path from s to t, and are neither s nor t). All vertices in Q have parities of one of the forms $(0, 0, z), (0, y, 0), (x, 0, 0)$, hence $|Q| = O(2^n/n^{2/3})$. Now take the set $C \subseteq E_n$ of all edges of H_n with both endpoints in Q; it is clearly a $\mathcal{P}$-cut. Furthermore, each vertex of Q is incident with at most $3 \cdot 2^{k/3} = O(n^{1/3})$ edges from C. This follows from the fact that every $v \in Q$ with parity vector, say, $(x, 0, 0)$, can be incident only with those edges in C that have directions corresponding to vectors of the form $(x, y, 0)$, $(x, 0, z)$ or $(x', 0, 0)$, for various $y, z, x' \in \{0,1\}^{k/3}$. Therefore, $|C| = O(2^n/n^{1/3})$, concluding the proof. ■

3.2.2 Improved bounds

In the main construction, we divide the length-k strings into m equally-sized parts, we let d be the distance between pairs in the pairing and w be the number of non-zero length-(k/m) parts of the parity strings of the direction vectors. The main tool is the following lemma about certain sets of vectors used to generalize the proof in the warm-up. The reader should keep in mind that an example of such a set of vectors for $m = 3$, $d = 3$, $w = 2$, is $V = \{110, 101, 011\}$, and was used implicitly in the previous proof.

For our purposes, all parameters involved except k and n should be thought of as constants, although the constants hidden in the Big-O notation are absolute.

Lemma 3.5 *Suppose $V \subseteq \{0,1\}^m$, $d = |V|$, and $w \in \mathbb{N}$ are such that:*

1. $2 \leq |v| \leq w$ *for all* $v \in V$,
2. $\bigoplus_{v \in V} v = 0$, *and*
3. *For all $W \subseteq V$ of size $|W| = \lfloor d/2 \rfloor$,* $|\bigoplus_{v \in W} v| \geq \lceil m/2 \rceil$

Let k be a non-zero multiple of m and $n = 2^k - 1$. Then there is a pairing $\mathcal{P} \subseteq V_n \times V_n$ of vertices of H_n of size $|\mathcal{P}| = \Omega(2^{n-d})$ that has a $\mathcal{P}$-cut $C \subseteq E_n$ of size $|C| = O\left(\frac{2^n}{\sqrt{n}}\, n^{w/m}\, d2^d\right)$ and with the additional property that each source in $\mathcal{P}$ is at distance exactly d from its sink.

Proof: Divide $[k]$ into m disjoint subsets $G_1, \ldots, G_m \subseteq [k]$ of size k/m; e.g. $G_i = \{(i-1)k/m + 1, \ldots, ik/m\}$. For $a \in \{0,1\}^n$, consider the k parity bits $p(a) \in \{0,1\}^k$ of a, and split them into m blocks according to $G_1, \ldots, G_m$ [3]; let us call each of the corresponding k/m-bit substrings $x_1(a), \ldots, x_m(a)$. Thus, $p(a)$ is the concatenation of $x_1(a), x_2(a), \ldots, x_m(a)$.

For a subset $v \subseteq [m]$, let $Z_v = \bigcup_{i \in A} G_i$. Given $v \subseteq [m]$ and $p \subseteq [k]$, define the *projection* of p on v to be $\Pi_v(p) = p \cap Z_v$, (remember that p and $\Pi_v(x)$ can be interpreted as strings in $\{0,1\}^k$ as well). For example, in the preceding subsection, $\Pi_{110}((x,y,z)) = (x,y,0)$. Consider the set

$$S = \{a \in \{0,1\}^n : \forall_{i \in [m]}\ x_i(a) \neq 0 \text{ and } \forall_{v \in V}\ a_{\Pi_v(p(a))} = 0\}.$$

This will be set of sources in $\mathcal{P}$. Note that the expression $a_{\Pi_v(p(a))}$, referring to bit number $\Pi_v(p(a))$ of a, is well-defined, because the condition $\forall_i\ x_i(a) \neq 0$, along with $v \neq 0$, implies $\Pi_v(p(a)) \neq 0$.

The set of d directions between a source s and the corresponding sink t will be determined by the parity of s alone, in the following way: for $p \in \{0,1\}^k$, let $D(p) = \bigcup_{v \in V} \{\Pi_v(p)\}$. Condition 1 of the hypothesis of the lemma implies that if $s \in S$, $|D(p)| = |V| = d$, and all elements of $D(p)$ have weight at least 2.

For each source $s \in S$, we define the sink $t = s \cup D(p(s))$; by construction $s \subseteq t$, and $t - s = |D(p(s))| = d$. $\mathcal{P}$ will be defined as the union of all such ordered pairs (s,t): $\mathcal{P} = \cup_{s \in S}\{(s, s \cup D(p(s)))\}$. Observe that $|\mathcal{P}| = |S| = \left(2^{k/m} - 1\right)^m 2^{n-k-d} = \Omega(2^{n-d})$.

We prove now that $\mathcal{P}$ forms a pairing: the set of sinks is disjoint from the set of sources, and no two different sources have the same sink. Because of the aforementioned properties of the parity check p, for any source-sink pair (s,t) we have $p(t) = p(s) \oplus \bigoplus_{v \in V} \Pi_v(p(s)) = p(s) \oplus \Pi_{\bigoplus_V v}(p(s)) = p(s)$ (where we used the second property of V and simple properties of the projection operator). Since for every $d \in D(p)$, $d \notin s$ but $d \in t$, it follows that no sink is a source too. Likewise, if two sinks t_1 and t_2 (corresponding to sources s_1 and s_2) were the same ($t_1 = t_2$), we would have $p(s_1) = p(s_2)$, which implies $D(p(s_1)) = D(p(s_2))$ and therefore $s_1 = s_2$.

To conclude, we only need to bound the size of a smallest $\mathcal{P}$-cut. Consider the set of vertices halfway between a source and a sink:

$$Q = \{x \in \{0,1\}^n : \text{there exists } (s,t) \in \mathcal{P} \text{ such that } s \subseteq x \subseteq t \text{ and } |x - s| = \lfloor d/2 \rfloor\}$$

(notice the slightly different definition of Q, compared to the one in 3.2.1).

Due to the third property of V and the definition of $D(p(s))$, it follows that $b \in Q$ implies that at least half of $x_1(b), \ldots, x_m(b)$ are zero. So the set $\{p(b) : b \in Q\}$ has size at most $\binom{d}{d/2}(2^{k/m}-1)^{m/2}$, and does not contain unit vectors; therefore $|Q| \leq \frac{2^n}{n+1}\binom{d}{d/2}(2^{k/m}-1)^{m/2} = O(\frac{2^n}{\sqrt{n}} \frac{2^d}{\sqrt{d}})$.

An edge cut is given by $C = \{(b,c) \in E_n : b \in Q \wedge c - b \in D(p(S))\}$, where $D(p(S)) = \bigcup_{s \in S}\{D(p(s))\}$. Thus, $|C| \leq |Q||D(p(S))|$. The claim follows since $|D(p(S))| \leq d(2^{k/m}-1)^w$. ■

Proof of Theorem 2.1: We prove a strengthening of the second part of the theorem that implies the first as well. To be precise, we show that, for every $1 > \delta > 0$, there exist $\epsilon > 0$ and d such that, for all large enough n, there is a pairing $\mathcal{P}$ in H_n of size $|\mathcal{P}| \geq \epsilon 2^n$, sparsity at most $n^{-1/2+\delta}$ and with the additional property that all pairs in $\mathcal{P}$ have distance exactly d. By partitioning the pairs in $\mathcal{P}$ according the level modulo d of their source, and applying a simple averaging argument, we conclude that there must exist an *aligned* pairing in H_n with sparsity at most $n^{-1/2+\delta}$.

[3] Strictly speaking, in order to do this we first impose an arbitrary ordering on the elements of each G_i.

First note that, whatever our choice of m, w, d, we can assume without loss of generality that m divides n. Otherwise, if $n > m$ take the largest multiple $n'(n,m)$ of m that is at most n, find the set $\mathcal{P}$ guaranteed by the lemma and notice that $H_{n'}$ can be embedded into H_n. This only affects the relevant ratios by at most a constant. This can be counteracted by choosing any smaller $\delta' < \delta$ and a large enough n to start with, and then taking the set $\mathcal{P}$ for $H_{n'(n,m)}$. Similarly, we can assume n is of the form $n = 2^k - 1$.

Let $w = \lceil 1/\delta \rceil$, $m = w^2$, $d = 2w$. It only remains to show that sets with parameters m, d, w, as in the hypotheses of Lemma 3.5, exist.

Arrange the w^2 elements of $[m]$ into a square matrix $A \in \{0,1\}^{w \times w}$. Associate one vector with each row and each column of A ($2w$ vectors in total). The i-th row is associated with the subset (or vector in $\{0,1\}^w$) $R_i = \{r \in [m] : (i-1)w < r \leq iw\}$; the j-th column will correspond to the subset $C_j = \{r \in [m] : (r-1) \bmod w = j-1\}$. Let $V = \bigcup_{i \in [w]} \{R_i, S_i\}$. Clearly, $|V| = 2w$ and for all $v \in V$, we have $|v| = w > 1$. It is also apparent that $\oplus_{v \in V} v = 0$, because any $k \in [m]$ belongs to exactly two vectors in V, namely R_i and C_j, where $k = (i-1)w + j$ with $i, j \in [w]$.

Finally, we show that, for any $W \subseteq V$ with $|V| = d/2 = w$, $| \oplus_{v \in W} v| \geq \frac{m}{2} = \frac{w^2}{2}$. Suppose W contains a row elements R_i and $w - a$ column elements C_j; then it is easy to see that $| \oplus_{v \in W} v| = a^2 + (w-a)^2 \geq \frac{w^2}{2}$ by the quadratic mean-arithmetic mean inequality. ∎

3.3 Proof of Theorem 2.2

Let $\mathcal{P}$ and C be the pairing and the cut constructed in the proof of Theorem 2.1. Let π be any permutation on V_n that maps each source in $\mathcal{P}$ to its sink. Notice that any shortest path in $H_n^{\uparrow\downarrow}$ that connects a source of $\mathcal{P}$ to its sink must also be a directed path in H_n. Hence, any realization of $\mathcal{P}$ with shortest paths must use some edge in C at least $|\mathcal{P}|/|C| = \Omega(n^{1/2-\delta})$ times. ∎

3.4 Proof of Theorem 2.4

Let $\epsilon > 0$, $\mathcal{R} \subseteq \mathbb{Z}$ and let $f : \{0,1\}^n \to \mathcal{R}$ be a dist-3 monotone function. If f is ϵ-far from being monotone, then by Lemma 3.1 there is a set $\mathcal{P}$ of $\epsilon 2^{n-1}$ vertex disjoint pairs in H_n that are violated by f. Furthermore, since f is dist-3 monotone, for every $(s^i, t^i) \in \mathcal{P}$ we have $|t^i| \leq |s^i| + 3$. To prove Theorem 2.4 we show that the sparsity of such $\mathcal{P}$ must be $\Omega(1/\sqrt{n})$.

Let C be a smallest $\mathcal{P}$-cut, and let us prove that $|C|/|\mathcal{P}| \geq \Omega(1/\sqrt{n})$. First we note that it is possible to assume that C has *no* edges that are incident with any source s^i or sink t^j from $\mathcal{P}$ (and in particular, this will mean that no pair in $\mathcal{P}$ has distance 1 or 2): Let $p > 0$ be the number of edges in C that are incident to some source or sink of a pair in $\mathcal{P}$. If $p \geq |\mathcal{P}|/4$ then we are done, since clearly $|C| \geq p$. Otherwise, removing these p edges from C and the corresponding pairs from $\mathcal{P}$ leaves a set C' of size $|C| - p$ that cuts a subset $\mathcal{P}' \subseteq \mathcal{P}$ of at least $|\mathcal{P}| - 2p$ pairs. This is due to the fact that the pairs in $\mathcal{P}$ are disjoint, and hence each edge can be incident with at most two pairs. Since $p \leq |\mathcal{P}|/4$, we have $\frac{|C|-p}{|\mathcal{P}|-2p} \leq 2\frac{C}{|\mathcal{P}|}$, so it is enough to prove the claim for $C \triangleq C'$ and $\mathcal{P} \triangleq \mathcal{P}'$.

For $0 \leq h \leq n-3$, let $\mathcal{P}^h \subseteq \mathcal{P}$ be the set of pairs $(s^i, t^i) \in \mathcal{P}$ with $|s^i| = h$ (and $|t^i| = h+3$). Clearly C is a $\mathcal{P}^h$-cut for every h. Let $C^h \subseteq C$ denote the set of edges in C that lie on some $\mathcal{P}^h$-path. Since C^h has no edges incident to any s^i or t^j, in order to cut $\mathcal{P}^h$ we must use exactly those edges between levels $h+1$ and $h+2$ that lie on some $\mathcal{P}^h$-path. So the sets C^h, $0 \leq h \leq n-3$, are in fact disjoint. Therefore it is sufficient to prove that $C^h/|\mathcal{P}^h| \geq \Omega(1/\sqrt{n})$ for all h.

Fix h, and for clarity let us redefine $\mathcal{P} \triangleq \mathcal{P}^h$ and $C \triangleq C^h$. Each pair $(s^i, t^i) \in \mathcal{P}$ defines a sub-cube of dimension 3, which we will denote by H_3^i, that contains all vertices and edges that belong to one of the six possible paths from s^i to t^i.

Observation 3.6 *For any two pairs* $(s^i,t^i),(s^j,t^j)\in\mathcal{P}$, $|E(H_3^i)\cap E(H_3^j)|\le 1$.

Proof: Assume that $|E(H_3^i)\cap E(H_3^j)|\ge 2$ for some $i\ne j$, and let $e=(a,b)$ and $e'=(a',b')$ be two edges in $E(H_3^i)\cap E(H_3^j)$. Since the pairs (s^i,t^i) and (s^j,t^j) are disjoint, both e and e' should lie between layers $h+1$ and $h+2$. Therefore, $a=a'=s^i\cup s^j$ and $b=b'=t^i\cap t^j$, contradicting the assumption that $e\ne e'$. ■

Consider the directed graph $G=(V,E)$ with $V=\bigcup_{(s^i,t^i)\in\mathcal{P}}V(H_3^i)$ and $E=\bigcup_{(s^i,t^i)\in\mathcal{P}}E(H_3^i)$. Since every s^i has out-degree 3 in G (and in-degree 0), the number of edges between layers h and $h+1$ of H_n that belong to G is exactly $3|\mathcal{P}|$. Let $A=a_1,\dots,a_k$ be the vertices in layer $h+1$ of H_n that belong to G, let $\alpha_1,\dots,\alpha_k$ denote their in-degrees and let $\beta_1,\dots,\beta_k$ denote their out-degrees in G. We have that $\sum_{i\in[k]}\alpha_i=3|\mathcal{P}|$, and our goal is to prove that $|C|\equiv\sum_{i\in[k]}\beta_i=\Omega(|\mathcal{P}|/\sqrt{n})$.

Consider vertex a_i. For every pair $(s^j,t^j)\in\mathcal{P}$ such that $a_i\in V(H_3^j)$ there are two edges in H_3^j going out of a_i. Since for any two pairs $(s^j,t^j),(s^{j'},t^{j'})\in\mathcal{P}$ we have $|E(H_3^j)\cap E(H_3^{j'})|\le 1$, it follows that $\binom{\beta_i}{2}\ge\alpha_i$. So $\beta_i>\sqrt{\alpha_i}$ for all i and hence $|C|=\sum_{i\in[k]}\beta_i>\sum_{i\in[k]}\sqrt{\alpha_i}=\sum_{i\in[k]}\frac{\alpha_i}{\sqrt{\alpha_i}}\ge\frac{3|\mathcal{P}|}{\sqrt{n}}$, as $\alpha_i\le n$.

3.5 An $\Omega(n/\epsilon)$ lower bound for general functions

Theorem 3.7 *Let* $\mathcal{R}\subseteq\mathbb{Z}$, $|\mathcal{R}|=\Omega(\sqrt{n})$. *Testing monotonicity of functions* $f\{0,1\}^n\to\mathcal{R}$ *(non-adaptively with one-sided error) requires* $\Omega(n/\epsilon)$ *queries.*

Proof: A non-adaptive q-query monotonicity tester with one-sided error queries f on a set Q of at most q vertices and rejects if and only if one of the comparable pairs in Q is violated. Hence, it is sufficient to show a family $\mathcal{F}_n$ of functions $f:\{0,1\}^n\to\mathcal{R}$ that are ϵ-far from monotone (for a fixed $\epsilon>0$ and all n) and such that, for any fixed set $Q\subseteq\{0,1\}^n$ of size $o(n)$, a random $f\sim_U\mathcal{F}_n$ induces a violated pair in Q with probability less than 1/3.

For every n, we will define a family $\mathcal{F}_n=\{f_1,\dots,f_n\}$ of n functions $f_i:\{0,1\}^n\to\mathcal{R}$ with the following properties:

- every f_i is ϵ-far from monotone, for some absolute constant $\epsilon>0$;
- for any set $Q\subseteq\{0,1\}^n$, $\Pr_{i\sim_U[n]}[(Q\times Q)\cap\mathsf{Viol}(f_i)\ne\emptyset]\le\frac{|Q|-1}{n}$.

This will imply that any tester making fewer than $\frac{2n}{3}$ queries will fail with probability at least 1/3.

Similarly to [FLN+02], each $f_i\in\mathcal{F}_n$ will violate some pairs that differ in the i'th coordinate. But here we will make sure that only the actual *edges* of H_n are violated, making it more difficult to catch violated pairs.

We now formally define $\mathcal{F}_n$. Let $\mathcal{R}=\{0,1,\dots,2\sqrt{n}\}$, and let $h(x)\triangleq|x|-n/2+\sqrt{n}$ for all $x\in\{0,1\}^n$. For each $i\in[n]$ we define $f_i:\{0,1\}^n\to\mathcal{R}$ as follows:

$$f_i(x)=\begin{cases}0, & h(x)<0\\ 2\sqrt{n}, & h(x)>2\sqrt{n}\\ h(x), & h(x)\in\mathcal{R}\text{ and }x_i\ne h(x)\bmod 2\\ h(x)+(-1)^{x_i}, & h(x)\in\mathcal{R}\text{ and }x_i=h(x)\bmod 2\end{cases}$$

Notice that for every $i\in[n]$, $\mathsf{Viol}(f_i)=\mathsf{EdgeViol}(f_i)$, and the edges in $\mathsf{EdgeViol}(f_i)$ are vertex disjoint. So by Lemma 3.1, the functions $f_i\in\mathcal{F}_n$ are ϵ-far from monotone (for some fixed $\epsilon>0$) if $|\mathsf{EdgeViol}(f_i)|\ge\epsilon 2^n$. Indeed, $|\mathsf{EdgeViol}(f_i)|$ equals the number of points $x\in\{0,1\}^n$ such that:

$h(x) \in \mathcal{R}$, $h(x) = 0 \pmod 2$ and $x_i = 0$. Notice that for $n > 10$, these constitute roughly a quarter of all points $y \in \{0,1\}^n$ with $h(y) \in \mathcal{R}$. On the other hand, it follows from Chernoff bounds that for some constant $\rho > 0$ and for all $n > 10$, the number of points $y \in \{0,1\}^n$ with $h(y) \in \mathcal{R}$ is at least $\rho 2^n$. Setting $\epsilon = \rho/5$, we conclude that all functions $f_i \in \mathcal{F}_n$ are ϵ-far from monotone.

Now we prove that $\Pr_{i \sim_U [n]}[(Q \times Q) \cap \mathsf{Viol}(f_i) \neq \emptyset] \leq \frac{|Q|-1}{n}$. Fix Q and consider the *undirected* graph $G = (V, E)$, where $V = Q$ and $E = \{\{x,y\} \in Q \times Q : (x,y) \in E_n\}$. In other words, G is the undirected skeleton of the subgraph of H_n induced on Q. For $x, y \in \{0,1\}^n$ we write $x = y^{(j)}$ if x equals y in all coordinates except j. Let $T \subseteq [n]$ be a set of directions spanned by E, namely, $T = \{j : \text{there exists } \{x,y\} \in E \text{ such that } x = y^{(j)}\}$. Clearly, the success probability of the test is bounded by $|T|/n$. To finish the proof, we show that $|T| \leq |Q| - 1$.

Consider a minimal subgraph G' of G that spans all directions in T. Clearly, $|E(G')| = |T|$. Therefore G' is acyclic, since any cycle in the undirected skeleton of H_n must travel in every direction even number of times. So $|T| = |E(G')| \leq |V(G')| - 1 = |Q| - 1$. ■

Remark 3.8 *We proved a lower bound of $\Omega(n)$ queries for some constant $\epsilon > 0$. To get a lower bound of $\Omega(n/\epsilon)$ for any $\epsilon = \epsilon(n)$ we need to compose our lower bound with a simple "hiding" procedure. Namely, we define a distribution $\mathcal{F}'_n$ that fools any deterministic tester with $o(n/\epsilon)$ queries as follows: first, partition H_n into disjoint subcubes, each of size $\epsilon 2^n$ (for simplicity we assume that $1/\epsilon$ is a power of 2); then pick a random subcube C in this partition, and value it with a random $f_i \in \mathcal{F}_{n-\log 1/\epsilon}$; value the other subcubes so that there are no violations outside C. Now for any fixed set Q of $o(n/\epsilon)$ queries, the expected number of queries that hit C is $o(n)$, and we know that with $o(n)$ queries it is impossible to find a violation in a random f_i.*

Notice that the range $\mathcal{R}$ of the functions f_i is of size $O(\sqrt{n})$ - much smaller than the 2^n different values a function on the hypercube may have. Therefore, applying the range-reduction lemma[4] of [DGL+99] we get that the query complexity of testing monotonicity of *Boolean* functions with pair-testers is $\Omega(n/(\epsilon \log n))$, which is tight up to the $\log n$ factor.

4 Concluding remarks

We suggest three open problems related to this line of work:

First, is it true that the best testers for monotonicity over H_n are in fact pair-testers? The question is of interest even just for Boolean-range functions, since a positive answer coupled with our $\Omega(\frac{n}{\epsilon \log n})$ lower bound for pair testers would give an almost-tight lower bound.

Another challenge is to find better upper bounds for the special case of testing monotonicity of dist-k monotone functions, for some $k \geq 3$. As we saw in Section 3.2.1, non-trivial sparsity upper bounds can be found even if we restrict ourselves to pairings in which all pairs are at distance 3. This seems to indicate, in our opinion, that a better understanding of the small-distance situations will yield new insights that may be applicable in the general case.

Finally, recall from Section 3.4 that for $k \leq 3$, dist-k monotonicity can be tested with $O(n^{3/2})$ queries; on the other hand, the construction in Section 3.2.1 shows that sparsity considerations alone will never yield upper bounds better than this. In view of these results, it is natural to ask whether these two measures need to coincide for larger k; that is, whether the complexity of edge-testers may be better than the values derived from sparsity upper-bounds.

[4] The range-reduction lemma of [DGL+99] says roughly that if T is a pair-tester (i.e. T picks pairs of comparable vertices according to some distribution and rejects if one of them is violated) that rejects any ϵ-far Boolean function with probability p, then T rejects any ϵ-far $f : \{0,1\}^n \to \mathcal{R}$ with probability at least $p \log^{-1} |\mathcal{R}|$.

References

[ABY08] Ali Al-Bashabsheh and Abbas Yongaçoglu. On the k-pairs problem. *CoRR*, abs/0805.0050, 2008.

[AC06] Nir Ailon and Bernard Chazelle. Information theory in property testing and monotonicity testing in higher dimension. *Inf. Comput.*, 204(11):1704–1717, 2006.

[AHJ+06] Micah Adler, Nicholas J. A. Harvey, Kamal Jain, Robert D. Kleinberg, and April Rasala Lehman. On the capacity of information networks. In *SODA*, pages 241–250, 2006.

[Ben65] V.E. Benes. Mathematical theory of connecting networks and telephone traffic. *New York : Academic Press*, 1965.

[BGJ+09] Arnab Bhattacharyya, Elena Grigorescu, Kyomin Jung, Sofya Raskhodnikova, and David P. Woodruff. Transitive-closure spanners. In *SODA '09: Proceedings of the Nineteenth Annual ACM -SIAM Symposium on Discrete Algorithms*, pages 932–941, Philadelphia, PA, USA, 2009. Society for Industrial and Applied Mathematics.

[Bha08] Arnab Bhattacharyya. A note on the distance to monotonicity of boolean functions. *Electronic Colloquium on Computational Complexity (ECCC)*, 15(012), 2008.

[DGL+99] Yevgeniy Dodis, Oded Goldreich, Eric Lehman, Sofya Raskhodnikova, Dana Ron, and Alex Samorodnitsky. Improved testing algorithms for monotonicity. In *RANDOM-APPROX*, pages 97–108, 1999.

[EKK+00] Funda Ergün, Sampath Kannan, Ravi Kumar, Ronitt Rubinfeld, and Mahesh Viswanathan. Spot-checkers. *J. Comput. Syst. Sci.*, 60(3):717–751, 2000.

[Fis04] Eldar Fischer. On the strength of comparisons in property testing. *Inf. Comput.*, 189(1):107–116, 2004.

[FLN+02] Eldar Fischer, Eric Lehman, Ilan Newman, Sofya Raskhodnikova, Ronitt Rubinfeld, and Alex Samorodnitsky. Monotonicity testing over general poset domains. In *STOC*, pages 474–483, 2002.

[GGL+00] Oded Goldreich, Shafi Goldwasser, Eric Lehman, Dana Ron, and Alex Samorodnitsky. Testing monotonicity. *Combinatorica*, 20(3):301–337, 2000.

[GGLR98] Oded Goldreich, Shafi Goldwasser, Eric Lehman, and Dana Ron. Testing monotonicity. In *FOCS*, pages 426–435, 1998.

[HK08] Shirley Halevy and Eyal Kushilevitz. Testing monotonicity over graph products. *Random Struct. Algorithms*, 33(1):44–67, 2008.

[LR01] Eric Lehman and Dana Ron. On disjoint chains of subsets. *J. Comb. Theory, Ser. A*, 94(2):399–404, 2001.

[Lub90] Anna Lubiw. Counterexample to a conjecture of szymanski on hypercube routing. *Inf. Process. Lett.*, 35(2):57–61, 1990.

[Ras99] Sofya Raskhodnikova. Monotonicity testing. Master's thesis, Department of Electrical Engineering and Computer Science, MIT, Cambridge, MA, 1999.

[RL05] A. Rasala-Lehman. *Network coding.* PhD thesis, Department of Electrical Engineering and Computer Science, MIT, Cambridge, MA, 2005.

[SM90] Farhad Shahrokhi and D. W. Matula. The maximum concurrent flow problem. *J. ACM*, 37(2):318–334, 1990.

[Szy89] Ted H. Szymanski. On the permutation capability of a circuit-switched hypercube. In *ICPP (1)*, pages 103–110, 1989.

A Better bound on the vertex-sparsity of H_n

Theorem A.1 *The vertex-sparsity of H_n is $O(1/n)$.*

Proof: Let $n \geq 8$ be a multiple of 4. We construct an aligned source-sink pairing $\mathcal{P} \subseteq V_n \times V_n$ of $2^{n/2}(n/4-1)$ disjoint pairs (s^i, t^i), such that for all i, j, $|t^j| = |s^i| + 2$. Then we show a $\mathcal{P}$-vertex-cut M of size $2^{n/2}$.

Consider the following set of pairs: $\mathcal{P}_0 = \{(0001, 1011), (1000, 1101), (0010, 0111), (0100, 1110)\}$. Any $\mathcal{P}_0$-path (of length 2) goes through one of the following vertices: $M_0 = \{1001, 0011, 1100, 0110\}$.

Using this small example, we construct the large one recursively. For $i \geq 1$, we set $\mathcal{P}_i = \{(01a, 01b) : (a, b) \in \mathcal{P}_{i-1}\} \cup \{(10a, 10b) : (a, b) \in \mathcal{P}_{i-1}\} \cup \{(00a, 11a) : a \in M_{i-1}\}$. Let M_i denote the set of all internal vertices that lie on some $\mathcal{P}_i$-path. Notice that $M_i = \{01a : a \in M_{i-1}\} \cup \{10a : a \in M_{i-1}\}$. So, we have:

- $|\mathcal{P}_i| = 2|\mathcal{P}_{i-1}| + |M_{i-1}|$;
- $|M_i| = 2|M_{i-1}|$.

Solving these recurrence equations we get $|M_i| = 2^i|M_0|$ and $|\mathcal{P}_i| = 2^i|\mathcal{P}_0| + i2^{i-1}|M_0|$. Let $\mathcal{P} \triangleq \mathcal{P}_{(n-4)/2}$ and $M \triangleq M_{(n-4)/2}$. Since M is a $\mathcal{P}$-vertex-cut by definition, we only need to show that the sizes of M and $\mathcal{P}$ are as advertised. Indeed, $|M| = 2^{(n-4)/2}|M_0| = 2^{n/2}$ and $|\mathcal{P}| = 2^{(n-4)/2}|\mathcal{P}_0| + 2^{(n-4)/2-1}\left(\frac{n-4}{2}\right)|M_0| = 2^{n/2}(n/4-1)$. ■