Query-efficient computation in property testing and learning theory

García Soriano, D.

Citation for published version (APA):
Chapter 8

Monotonicity testing and shortest-path routing

The content of this chapter is based on the paper


8.1 Introduction

Testing monotonicity of functions \cite{DGL+99, Ras99, GGL+00, EKK+00, Fis04, FLN+02, AC06, 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 finite, 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 $f(x) \leq f(y)$. By the standard definitions, a function $f$ is \( \varepsilon \)-far from monotone if it has to be changed on at least an \( \varepsilon \)-fraction of the domain $\mathcal{D}$ to become monotone. A one-sided $(q, \varepsilon)$-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 \( \varepsilon \)-far from monotone.

The simplest monotonicity testers are those which specify all their queries in advance (non-adaptively) and reject if and only if the responses reveal a violation, i.e., if $f(x) > f(y)$ for some comparable pair $x \leq y$ of points from $\mathcal{D}$. These non-adaptive testers with one-sided error are the only ones considered in this chapter, 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
Chapter 8. Monotonicity testing and shortest-path routing

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+09a, BGJ+09b]. Here the order relation \( x \leq y \) is defined to hold for \( x, y \in \{0, 1\}^n \) when \( x_i \leq y_i \) for all \( i \in [n] \). In this chapter we focus on a well-studied subcase of the above, where \( m = 1 \): integer-valued functions with domain \( \{0, 1\}^n \).

8.2 Preliminaries

Recall from Section 1.4 that \( H_n = (V_n, E_n) \) is the graph of the directed \( n \)-dimensional hypercube.

Given \( \mathcal{R} \subseteq \mathbb{Z} \), a finite poset \((\mathcal{D}, \leq)\), and a function \( f : \mathcal{D} \to \mathcal{R} \), we say that a pair \((x, y) \in \mathcal{D} \times \mathcal{D}\) is violated by \( f \) if \( x \leq y \) and \( f(x) > f(y) \). An edge is a pair \((x, y) \in \mathcal{D} \times \mathcal{D}\) with \( x < y \) and such that there is no \( z \) with \( x < z < y \); when \((\mathcal{D}, \leq) = (V_n, \subseteq)\), this is tantamount to saying that \((x, y) \in E_n\).

The set of all violated pairs of \( f \) is denoted \( \text{Viol}(f) \), and the set of all violated edges is denoted \( \text{EdgeViol}(f) \). Clearly, the function \( f \) is monotone if and only if \( \text{Viol}(f) = \text{EdgeViol}(f) = \emptyset \).

We denote by \( \varepsilon_M(f) \in [0, 1] \) the relative distance of \( f \) from being monotone, i.e., the minimum of \( \mathbb{P}_{x \in \mathcal{D}}[f(x) \neq g(x)] \) taken over all monotone functions \( g : \mathcal{D} \to \mathcal{R} \) (the minimum exists even if \( \mathcal{R} \) is infinite, but we shall not need this). Let \( \delta_M(f) \in [0, 1] \) denote the fraction of edges violated by \( f \); for the hypercube poset this is \(|\text{EdgeViol}(f)|/|E_n| = |\text{EdgeViol}(f)|/(n2^{n-1})\).

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

- \( s^i \subseteq t^i \) 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 chapter we denote by \( \mathcal{P} \) only sets of pairs that form a source-sink pairing, even when not explicitly mentioned.

A (directed) path in \( H_n \) is called a \( \mathcal{P}-\text{path} \) if it connects some source \( s^i \) from \( \mathcal{P} \) to its sink \( t^i \). A subset \( C \subseteq E_n \) is called a \( \mathcal{P}-\text{cut} \) 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 \( \mathcal{P}-\text{vertex-cut} \) if every \( \mathcal{P} \)-path uses at least one vertex from \( S \). We write \( \text{maxflow}(\mathcal{P}) \) for the size of the
largest set of edge-disjoint $P$-paths, $\text{mincut}(P)$ for the size of the smallest $P$-cut and $\text{minvertexcut}(P)$ for the size of the smallest $P$-vertex-cut. Clearly $\text{mincut}(P)$ is an upper bound on both $\text{minvertexcut}(P)$ and $\text{maxflow}(P)$. Unlike the case with a single pair in $P$, the quantities $\text{mincut}(P)$ and $\text{maxflow}(P)$ need not coincide.

We define the terms sparsity and meagerness as in [RL05], [ABY08], [HKL06]. The sparsity of $P$ is the ratio $\frac{\text{mincut}(P)}{|P|}$, and the vertex sparsity of $P$ is the ratio $\frac{\text{minvertexcut}(P)}{|P|}$. In other words, sparsity is the average number of edges per source-sink pair that one has to remove to disconnect every source from its sink, whereas vertex sparsity is the average number of vertices per source-sink pair that one has to remove to disconnect every source from its sink. 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 $t_j$.

The sparsity and the vertex sparsity of $H_n$ are defined as $\min_{P}\{\text{mincut}(P)/|P|}\}$ and $\min_{P}\{\text{minvertexcut}(P)/|P|}\}$, respectively (where $P$ ranges over all pairings $P \subseteq V_n \times V_n$).

Observe that
\begin{enumerate}
  \item sparsity $\geq$ vertex sparsity;
  \item meagerness $\geq$ vertex meagerness;
  \item meagerness $\geq$ sparsity;
  \item vertex meagerness $\geq$ vertex sparsity.
\end{enumerate}

### 8.3 From sparsity bounds to monotonicity testers

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. In particular, Goldreich et al. [GGLR98] observed that if the meagerness of $H_n$ is at least 1, then monotonicity of boolean functions would be testable with $O(n/\varepsilon)$ queries. They reasoned as follows. It can be shown that a function $f$ which is $\Omega(1)$-far from monotone induces a pairing $P$ of violated pairs of cardinality $\Omega(2^n)$. Being a boolean function, each of these pairs $(s^i, t^i)$ must then satisfy $f(s^i) = 1$ and $f(t^i) = 0$. By transitivity, if $s^i \subseteq t^i$, then there must be some violated edge in any path from $s^i$ to $t^i$, even if $i \neq j$. If the meagerness of $P$ is at least one, then there are at least $|P|$ edges witnessing the non-monotonicity of $f$: at least one per path in the optimal set of paths that disconnect all sources from all sinks in $P$. This would mean that a random edge in $H_n$ belongs to $\text{Viol}(f)$ with probability at least $|P|/|E_n| = \Omega(1/n)$. Then the following simple algorithm would be a one-sided tester of monotonicity: pick an edge from $E_n$ at random, reject if it is violated, and repeat $O(n)$ times.

What they proved is that vertex meagerness (and hence meagerness too) is 1 if the possible pairings $P$ are restricted to aligned sets, satisfying $|s^i| = |s^j|$ and
According to the ordering of violations between $s_i$ and $t_i$ (but not between $s_i$ and $t_j$. In particular, we will presently see that 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 $O(n/(\varepsilon \mu))$ queries. In [LR01], Lehman and Ron asked whether the sparsity of any $P$ (or even just of the aligned ones) is at least 1, noting that this would imply the existence of efficient monotonicity testers as well as progress on some long-standing questions regarding routing on the hypercube network. As they wrote, it appears that a counterexample must be sizable, if one exists at all. We prove that a counterexample does exist and the answer to both of their questions is no.

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

8.3.1. Lemma ([FLN+02, Corollary 2];[DGL+99, Lemma 7]). Let $f: D \to \mathcal{R}$ be a function, and define the violation graph of $f$ as the undirected graph $G = (D, E)$, where $(x, y) \in E$ if either $(x, y)$ or $(y, x)$ is in $\text{Viol}(f)$. Then $\varepsilon_M(f) 2^n$ is exactly the size of a minimum vertex cover of $G$. Consequently, there is a matching in $G$ of size at least $\varepsilon_M(f) 2^{n-1}$.

Proof. Let $g$ be monotone and $\varepsilon_M(f)$-close to $f$, and write $T \triangleq \{ x \in D \mid f(x) \neq g(x) \}$; we have $|T| = \varepsilon_M(f) 2^n$. Let $C$ be a minimum vertex cover of the violation graph $G$. We show that $|C| = |T|$; the “consequently” part then follows from the easy fact that the size of any maximal matching in a graph is at least half the size of the minimum vertex cover, as the endpoints of any maximal matching form a vertex cover.

Clearly $T$ must be a vertex cover of $G$, otherwise $g$ wouldn’t be monotone. Hence $|T| \geq |C|$. To prove $|C| \leq |T|$, we show how that if $S$ is a vertex cover of the violation graph of $f$ (not necessarily smallest), then $f$ can be made monotone by redefining it on $S$. We proceed by induction on the size of $S$. The base case, $S = \emptyset$, is trivial. If $S$ is nonempty, take any minimal element $x \in S$ (according to the ordering of $D$). Consider the sets $x_\prec \triangleq \{ y \in D \mid y < x \}$ and $x_\succ \triangleq \{ y \in D \mid y > x \}$. Modifying $f(x)$ only affects the violations occurring among elements of $x_\prec \cup \{ x \} \cup x_\succ$. Let us define $\tilde{f}$ to be equal to $f$ except on input $x$, where we let

$$\tilde{f}(x) \triangleq \max\{ f(y) \mid y \in x_\prec \},$$

1 If the function is boolean, something stronger holds. In this case $G$ is bipartite because violations only occur among pairs $x, y \in D$ with $x \leq y$, $f(x) = 1$ and $f(y) = 0$. Therefore, by König’s Theorem [Die05, Theorem 2.1.1], the size of the maximum matching equals $|C|$, the size of the minimum cover.
where we adopt the convention that the maximum of the empty set is the smallest
element of the range \( \mathcal{R} \). By definition there are no violations in the new function \( \hat{f} \)
between the elements of \( x_\prec \) and the input \( x \). Nor are there any violations between
\( x_\prec \) and \( x_\succ \), because \( \hat{f} \) equals \( f \) on them and \( S \) is a vertex cover of the violation
graph of \( f \) that does not intersect \( x_\prec \). There cannot be a violation either between
\( x \) and \( x_\succ \) because by definition, \( f(x) = f(y) \) for some \( y \in x_\prec \), which means there
would be a violation between \( x_\prec \) and \( x_\succ \) too. Hence \( S - \{ x \} \) is a vertex cover of
the violation graph of \( \hat{f} \), and we are done by induction.

Regarding functions defined on the hypercube, an important observation is
that since \( G \) is a subgraph of the transitive closure of \( H_n \), the matching of violated
pairs in Lemma 8.3.1 forms a source-sink pairing \( P \) (see Definition 8.2.1) of size
\( \varepsilon M(f) 2^{n-1} \).

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:

8.3.2. Theorem. \([\text{GGLR98, GGL}^+00]\) For any \( f: \{0,1\}^n \rightarrow \{0,1\} \),
\[ \delta_M(f) \geq \frac{\varepsilon M(f)}{n}. \]

More generally, \([\text{DGL99}]\) use their range-reduction lemma to conclude that for
any \( f: \{0,1\}^n \rightarrow \mathcal{R} \), \( \delta_M(f) \geq \frac{\varepsilon M(f)}{n \log |\mathcal{R}|} \). Since without loss of generality \(|\mathcal{R}| \leq 2^n \),
this gives an upper bound of \( O(n^2/\varepsilon) \) queries for testing monotonicity of all
functions \( f: \{0,1\}^n \rightarrow \mathcal{R} \).

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 8.3.2 holds for any \( \mathcal{R} \).) The next lemma states that this
can also be done by proving lower bounds on the sparsity of \( H_n \).

8.3.3. Lemma. Let \( \mu(n) \) denote the sparsity of \( H_n \). For any \( \varepsilon > 0 \) and \( \mathcal{R} \subseteq \mathbb{Z} \),
monotonicity of functions \( f: \{0,1\}^n \rightarrow \mathcal{R} \) can be tested with \( O\left(\frac{n}{\varepsilon \mu(n)}\right) \) queries.

Proof. Let \( \varepsilon > 0 \) and let \( f: \{0,1\}^n \rightarrow \mathcal{R} \) be \( \varepsilon \)-far from monotone. Let \( \mathcal{P} \) be
the set of \( \varepsilon M(f) 2^{n-1} \geq \varepsilon 2^{n-1} \) vertex-disjoint violated pairs promised by Lemma
8.3.1. By definition, \( \mathcal{P} \) is a source-sink pairing. Notice that since every \((s^i, t^i) \in \mathcal{P}\)
is violated, every path from \( s^i \) to \( t^i \) must contain at least one violated edge. It follows that the set \( \text{EdgeViol}(f) \) is a \( \mathcal{P} \)-cut and \( |\text{EdgeViol}(f)|/|\mathcal{P}| \geq \mu(n) \). Hence
\[ \delta_M(f) = \frac{|\text{EdgeViol}(f)|}{|E_n|} \geq \frac{\varepsilon \mu(n)}{n}. \] We can thus conclude that \( O\left(\frac{n}{\varepsilon \mu(n)}\right) \) edge queries
suffice to find an edge-violation with constant probability. \( \square \)
8.4 Upper bounds on sparsity

We prove the following theorem.

8.4.1. Theorem (Briët et al. [BCGM12]). The sparsity of $H_n$ is at most $n^{-\frac{1}{2}+o(1)}$. Furthermore, this upper bound on sparsity can be demonstrated both with aligned sets and with $\Omega(2^n)$-sized sets:

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

We use a number of properties of the structure of perfect binary Hamming codes (see, e.g., [MS77, Lin98]), which we now describe. For an integer $k \geq 1$, let the strings $y \in \{0, 1\}^k \setminus \{0\}^k$ represent indices to bit positions of binary strings of length $n = 2^k - 1$. The Hamming code is a linear code consisting 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 $x_y = 1$. We are, however, more interested in the properties of the parity check matrix $p$ of the code. This is a $k \times n$ binary matrix whose columns are all possible nonzero $k$-bit vectors $y$; it represents a linear map $p: \{0, 1\}^n \rightarrow \{0, 1\}^k$ over GF(2). Therefore, for any unit vector $e_y$ (i.e., the element of $\{0, 1\}^n$ having 1 at position $y$ and 0 elsewhere), $p(e_y) = y$. Consequently, 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). We refer to the $k$ positions of the form $2^i$ (i.e., 1, 2, 4, $\ldots$, $(n + 1)/2$, corresponding to vectors of the form $e_i = 0^{i-1}10^{n-i}$), as the redundancy bits of the code; in a codeword $x$, the $k$ values $x_{e_i}, i = 1 \ldots k$ are determined by the remaining $n - k$ bits of $x$. Moreover, for general $a \in \{0, 1\}^n$, they are determined by the remaining $n - k$ bits and the parity vector $p(a)$.

8.4.1 Warm-up

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

8.4.2. Proposition. Let $k > 0$ be a multiple of three, and $n = 2^k - 1$. There is a pairing $P \subseteq V_n \times V_n$ in $H_n$ of size $|P| = \Omega(2^n)$ that admits a $P$-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 will write $(v_1, v_2, v_3)$ to denote the concatenation of three vectors $v_1, v_2, v_3 \in \{0, 1\}^{k/3}$,
8.4. Upper bounds on sparsity

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 at least one of \( v_1, v_2, v_3 \in \{0, 1\}^{k/3} \) is nonzero, then \((v_1, v_2, v_3) \in [n] \).

The set \( S \) of sources of \( \mathcal{P} \) is the set of all \( s \in \{0, 1\}^n \) that satisfy

\[
(x(s) \neq 0 \land y(s) \neq 0 \land z(s) \neq 0) \land (s_{(x(s), y(s), 0)} = s_{(x(s), 0, z(s))} = s_{(0, y(s), z(s))} = 0).
\]

For each source \( s \in S \), we define its sink \( t = t(s) \) as

\[
t = s \cup \{(x(s), y(s), 0), (x(s), 0, z(s)), (0, y(s), z(s))\}.
\]

Note \( t \) is at distance 3 from \( s \), and 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 clause in the conditions on a member \( s \) of \( S \) ensures that all three directions are (1) distinct; and (2) have a \( k \)-bit binary representation with Hamming weight strictly greater than one (in particular they represent proper directions, i.e., they are nonzero). The second clause ensures that the relevant bits of \( s \) take the value 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 easy to verify that

\[
|S| = (2^{k/3} - 1)^3 2^{n-k-3} = 2^{n-3} \frac{(2^{k/3} - 1)^3}{2^k} = \Omega(2^n),
\]

as follows. There are \((2^{k/3} - 1)^3\) ways to pick \( x, y, z \in \{0, 1\}^{k/3} - \{0\} \). For each such choice, we can construct a source \( s \) by letting \( s_{(x,y,0)} = s_{(x,0,z)} = s_{(0,y,z)} = 0 \) and setting the remaining \( n-k-3 \) non-redundant bits arbitrarily; there are \( 2^{n-k-3} \) ways to accomplish this. (Note none of the directions used corresponds to a redundancy bit, i.e., none of them is a power of 2 because their binary representations have at least two ones.) Finally, the values \( s \) takes on the redundancy bits are now determined by the values already set and the condition \( p(s) = (x, y, z) \). The last equality also implies that different choices of \( x, y, z \), along with the remaining non-redundant bits, lead to different sources; putting all together we get the equality stated on the size of \( S \).

To prove that \( \mathcal{P} \) is a pairing, it remains to show that all sinks are distinct, and that no source is also a sink. Recall that one of the properties of map \( p \) is that \( p(a \oplus e_{(x,y,z)}) = p(a) \oplus (x, y, z) \). So after flipping, e.g., bit \((x, y, 0)\) from a source \( s \) with parity vector \((x, y, z)\), we reach a vertex with parity vector \((0, 0, z)\). Thus, we see that the parity vectors 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): \((x, y, z)\).

Observe that, while the source \(s\) is different from its sink \(t\), the two parity vectors are the same. Also notice that the parity vectors at level 1 are distinct, as are the parity vectors at level 2.

The three directions from \(s\) to \(t\), i.e., the indices of the support of \(t - s\), are determined by \(p(s) = (x, y, z) = p(t)\). By construction, \(t_{(x,y,0)} = t_{(x,0,z)} = t_{(0,y,z)} = 1\), implying \(t \notin S\) as it does not meet the requirements for being a source. Likewise, if two different sources \(s_1\) and \(s_2\) were associated with the same sink \(t \supseteq s_1, s_2\), we would get \(p(s_1) = p(t) = p(s_2)\), so \(t - s_1 = t - s_2\), implying \(s_1 = s_2\). Hence \(P\) is indeed a pairing of size \(|P| = |S| = \Omega(2^n)\).

Now we argue that \(P\) admits a small cut. Let \(Q_s\) be the set of vertices at level 1 or 2 in the subcube from a source \(s \in S\) to its sink \(t\) (that is, lying on one of the six paths from \(s\) to \(t\) and different from \(s\) and \(t\)). Let \(Q = \bigcup_{s \in S} Q_s\). All vertices in \(Q\) have parity vectors 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 (directed) edges of \(H_n\) with both endpoints in \(Q\); it is clearly a \(P\)-cut. Besides, each vertex of \(Q\) is incident with at most \(3 \cdot 2^{k/3} = O(n^{1/3})\) edges from \(C\). To see this, consider an arbitrary element of \(q \in Q\) with parity vector \(p(q) = (x, 0, 0)\), say. It 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 \(y, z, x' \in \{0, 1\}^{k/3}\). Since \(x\) is fixed for this particular vertex \(q\), there are at most \(3 \cdot 2^{k/3} = O(n^{1/3})\) edges in \(C\) going out of \(q\). Therefore \(|C| \leq |Q| \cdot O(n^{1/3}) = O(2^n/n^{1/3})\), concluding the proof. \(\Box\)

### 8.4.2 Improved upper bound on the edge sparsity of \(H_n\)

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 nonzero 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 implicitly used in the previous proof.

For our purposes, all parameters involved except \(k\) and \(n\) should be thought of as constants.

#### 8.4.3. Lemma. Suppose \(V \subseteq \{0, 1\}^m\), \(d = |V| > 0\), 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 \(|W| = [d/2]|\), \(|\bigoplus_{v \in W} v| \geq [m/2]|\).
Let \( k \geq m \log m \) be a positive 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

\[
|\mathcal{P}| \geq \frac{1}{4} 2^{n-d}
\]

that has a \( \mathcal{P} \)-cut \( C \subseteq E_n \) of size

\[
|C| \leq \frac{2^n}{\sqrt{n+1}} (n+1)^{w/m} \sqrt{d} 2^d
\]

and with the additional property that each source in \( \mathcal{P} \) is at distance exactly \( d \) from its sink.

**Proof.** Partition \([k]\) 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 \( G_1, \ldots, G_m \) 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 \( Z_v = \bigcup_{i \in v} G_i \subseteq [k] \). Given \( r \subseteq [k] \), define the *projection* of \( r \) on \( v \) to be \( \Pi_v(r) = r \cap Z_v \), (remember that \( r \) and \( \Pi_v(x) \) can be interpreted as strings in \( \{0,1\}^k \) as well). For example, in the preceding subsection, we would write \( \Pi_{110}((x,y,z)) = (x,y,0) \). Consider the set

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

This will be our 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 \( \Pi_v(p(a)) ) \neq 0 \). Moreover, \( \Pi_v(p(a)) \neq \Pi_w(p(a)) \) for \( v \neq w \).

The set of \( d \) directions between a source \( s \) and the corresponding sink \( t \) will be determined by the parity vector of \( s \) alone. This set will be \( D(p(s)) \) for a function \( D \) defined in the following way: for \( r \in \{0,1\}^k \), let \( D(r) = \bigcup_{v \in V} \{ \Pi_v(r) \} \). Condition 1 in the hypothesis of the lemma implies that for \( s \in S \), all elements of \( D(p(s)) \) have weight \( \geq 2 \); note also that \( s \) is disjoint with \( D(p(s)) \) and that \( |D(p(s))| = |V| = d \). For each source \( s \in S \), we define the sink \( t = s \cup D(p(s)) \); by construction \( s \subseteq t \), and \( |t - s| = |D(p(s))| = d \). The cut \( \mathcal{P} \) is defined as the union of all such ordered pairs \( (s,t) \): \( \mathcal{P} \triangleq \bigcup_{s \in S} \{(s,s \cup D(p(s)))\} \). Note that

\[
|\mathcal{P}| = |S| = (2^{k/m} - 1)^m 2^{n-k-d}.
\]

We can bound

\[
n + 1 = 2^k \geq (2^{k/m} - 1)^m = \left( 1 - \frac{1}{2^{k/m}} \right)^m 2^k \geq \frac{1}{4} 2^k. \tag{8.1}
\]

\(^2\) Actually, in order to do this we first impose an arbitrary ordering on the elements of each \( G_i \).
Chapter 8. Monotonicity testing and shortest-path routing

since \( k \geq m \log m \) and \( m \geq 2 \). Hence

\[
|\mathcal{P}| = |S| \geq \frac{1}{4}2^{n-d}.
\]

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

\[
p(t) = p(s) \oplus \bigoplus_{v \in V} \Pi_v(p(s)) = p(s) \oplus \Pi_{\bigoplus_{v \in V}}(p(s)) = p(s),
\]

where we used the second property of \( V \) and simple properties of the projection operator. Since for every \( i \in D(p), i \notin s \) but \( i \in t \), 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 upper-bound the size of a smallest \( \mathcal{P} \)-cut. Consider the set of vertices halfway between a source and a sink:

\[
Q \triangleq \{ x \in \{0,1\}^n \mid \exists (s, t) \in \mathcal{P} \text{ s.t. } s \subseteq x \subseteq t \text{ and } |x - s| = \lceil d/2 \rceil \}
\]

(note the slightly different definition of \( Q \), compared to that in the warmup at the start of the section).

The set \( Q \) is a \( \mathcal{P} \)-vertex-cut; we use it to construct an edge cut for \( \mathcal{P} \). Due to the third property of \( V \) and the definition of \( D(p(s)) \), it follows that for \( b \in Q \), at least half of \( x_1(b), \ldots, x_m(b) \) are zero. For any \( b \in \{0,1\}^n \), let \( r(b) \) be the \( m \)-bit string such that for all \( i \in [m] \), the equality \( x_i(b) = 0^{k/m} \) holds iff \( r(b)_i = 0 \). Then the set \( \{ r(b) \mid b \in Q \} \) has size bounded by \( \binom{d}{d/2} \): for all \( s \in S \), \( r(s) \) is the all-ones string and for any \( b \in Q \), \( r(b) \) is \( r(s) \) XOR-ed with some \( d/2 \) vectors in \( V \). So the set \( \{ p(b) \mid b \in Q \} \) has size at most \( \binom{d}{d/2}(2^{k/m} - 1)^{m/2} \), and does not contain unit vectors; therefore using \( (8.1) \) we obtain

\[
|Q| \leq 2^{n-k} \binom{d}{d/2} (2^{k/m} - 1)^{m/2} \leq \frac{2^n}{\sqrt{n} + 1} \frac{2^d}{\sqrt{d}}.
\]

An edge cut is given by

\[
C \triangleq \{(b, c) \in E_n \mid b \in Q \land c - b \in D(p(s))\},
\]

where \( D(p(S)) = \bigcup_{s \in S} \{ D(p(s)) \} \). To verify this, observe that any path from a source \( s \) to its sink \( t \) must go through some vertex \( q \in Q \) at distance \( \lfloor d/2 \rfloor \) from \( s \), and then take one of the directions in the set \( D(p(s)) \) (or else the sink would not be reachable from the next vertex in the path). Finally, observe that

\[
|D(p(S))| \leq d(2^{k/m} - 1)^w \leq d(n + 1)^{w/m},
\]
8.4. Upper bounds on sparsity

because every element of $D(p(s))$ is determined by the projection of $p(s)$ on some $v \in V$ (with weight at most $|v| \leq w$). The claim follows from our bounds on $|Q|$ and $|D(p(s))|$, since $|C| \leq |Q| \cdot |D(p(S))|$. 

8.4.4. Lemma. Let $w \in \mathbb{N}, w \geq 2$ and set $m = w^2, d = 2w$. Then there is a set $V \subseteq \{0,1\}^m$ of $d$ vectors satisfying the three conditions in Lemma 8.4.3.

Proof. Arrange the $w^2$ elements of $[m]$ into a square matrix $A \in \{0,1\}^{w \times w}$. For each row and each column of $A$ we form an element of $\{0,1\}^w$ whose support is that row or column (there are $2w$ vectors in total). The $i$-th row is then associated with the subset (or vector in $\{0,1\}^w$)

$$R_i \triangleq \{ r \in [m] | (i-1)w < r \leq iw \};$$

the $j$-th column will correspond to the subset

$$C_j \triangleq \{ r \in [m] | (r-1) \mod w = j - 1 \}.$$

Let

$$V \triangleq \bigcup_{i \in [w]} \{R_i, C_i\}.$$

Clearly, $|V| = 2w$ and for all $v \in V$, we have $|v| = w > 1$. It is also apparent that $\bigoplus_{v \in V} v = 0$, 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 $i, j \in [w]$. (This is a rephrasing of the fact that every entry of the matrix $A$ is in the intersection of precisely one row and one column.)

Finally, we show that, for any $W \subseteq V$ with $|V| = d/2 = w$,

$$\left| \bigoplus_{v \in W} v \right| \geq \frac{m}{2} = \frac{w^2}{2}.$$

Suppose that

$$W = \{R_{i_1}, R_{i_2}, \ldots, R_{i_a}, C_{j_1}, C_{j_2}, \ldots, C_{j_{(w-a)}}\};$$

then

$$\left| \bigoplus_{v \in W} v \right| = a^2 + (w-a)^2 \geq \frac{w^2}{2}$$

by the quadratic mean-arithmetic mean inequality. 

Proof of Theorem 8.4.1. Let $\delta \in (0, 1), n > (4/\delta^2)^{4/\delta^2+1}$ and set

$$w \triangleq \lceil 1/\delta \rceil, m \triangleq w^2, d \triangleq 2w.$$
Let $k$ be the largest multiple of $m$ which is at most $\log(n + 1)$. Note that $w \leq 2/\delta, n > m^{m+1}$ and

$$k > \log(n + 1) - m \geq \log(n/m) \geq m \log m.$$  

Now set $n_0 \triangleq 2^k - 1 > n/2^{m+1}$. By Lemmas 8.4.4 and 8.4.3 there is a pairing $P_{n_0} \subseteq V_{n_0} \times V_{n_0}$ of size $2^{n_0-d-2}$ which admits a cut $C_{n_0}$ of size

$$\frac{2^{n_0}}{\sqrt{n_0} + 1} (n_0 + 1)^{1/w} \sqrt{d} 2^d$$

and thus $H_{n_0}$ has sparsity at most

$$\frac{|C_{n_0}|}{|P_{n_0}|} \leq (n_0 + 1)^{\frac{1}{\sqrt{d}} - \frac{1}{2}} \cdot 2^{2(d+1)\sqrt{d}} \leq \frac{1}{\sqrt{n}} \cdot n^\delta \cdot 2^{O(\frac{1}{\delta})).}$$

Observe that $H_n$ can embed $2^{n-n_0}$ disjoint copies of $H_{n_0}$, for example, according to each of the settings of the last $n-n_0$ bits of a vertex label. We can thus embed $2^{n-n_0}$ copies of the pairing and its cut, and still obtain a pairing of size $\Omega(2^{n-d})$ and a cut of sparsity $n^{-1/2+\delta} \cdot 2^{O(1/\delta^3)}$. This may look slightly weaker than the second part of the theorem, but it implies it by choosing for example $\delta_0 = \delta/2$ and $n = 2^{O(1/\delta^3)}$.

For the first part, recall that the pairing $P$ we obtain has the additional property that all pairs in $P$ have distance exactly $d$. Knowing this, the first part of Theorem 8.4.3 can be proved in the following way. Let $C$ be a $P$-cut of sparsity $\leq n^{-1/2+\delta}$. Partition the pairs of $P$ into $d+1$ parts $P_1, \ldots, P_{d+1}$ according to the Hamming weight of their source modulo $d+1$. At least one of them, say $P_i$, has size $\geq |P|/(d+1)$, so $C$ is a $P_i$-cut with sparsity $\leq (d+1) \cdot n^{-1/2+\delta}$. If $(s^i, t^i), (s^k, t^k) \in P_i$ and $|s^i| \neq |s^k|$, then $|s^i| - |s^k|$ is a multiple of $d+1$, and since $|t^j - s^j| = |t^k - s^k| = d$, we conclude that no edge in any path from $s^j$ to $t^j$ lies in a path from $s^k$ to $t^k$ as well.

Now, for $j = 0, \ldots, [(n-i)/(d+1)] - 1$, let

$$A_j = \{ x \in \{0,1\}^n \mid j(d+1) \leq |x| - i < (j+1)(d+1) \}.$$  

It follows that we can partition $P_i$ into $[n/d]$ parts, according to which set $A_j$ their sources (and sinks) belong to. Moreover, there is a subset $C' \subseteq C$ that is a $P_i$-cut and only contains edges both of whose endpoints are inside the same $A_j$ (possibly different for different pairs), i.e., $C' = \bigcup C'_j$ where $C'_j \subseteq A_j \times A_j$. (The reason is that we can safely remove any edges that do not satisfy this condition.) From $\sum_j |C_j| = |C'|$ and $\sum_j |P_i \cap (V_j \times V_j)| = |P_i|$, it follows that there is some $j$ with $|C'_j|/|P_i \cap (V_j \times V_j)| \leq |C'|/|P| \leq |C|/|P|$, i.e., there is an aligned pairing $P_i \cap (V_j \times V_j)$ with sparsity at most $(d+1) \cdot n^{-1/2+\delta}$. Again, choosing a smaller $\delta$ proves the claim.  

\[ \square \]
8.4. Upper bounds on sparsity

8.4.3 Upper bound on the vertex sparsity of $H_n$

While it is edge sparsity that bears the strongest relationship with monotonicity testing, it is natural to study the related quantity of vertex sparsity as well. Here we present a simple result on the vertex sparsity of $H_n$.

8.4.5. Theorem (Briët et al. [BCGM12]). The vertex sparsity of $H_n$ is $O(1/n)$.

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

Consider the following set of pairs:

$P_0 = \{(0001, 1011), (1000, 1101), (0010, 0111), (0100, 1110)\}$.

Any $P_0$-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

$P_i = \{(01a, 01b) \mid (a, b) \in P_{i-1}\} \cup \{(10a, 10b) \mid (a, b) \in P_{i-1}\} \cup \{(00a, 11a) \mid a \in M_{i-1}\}$.

Let $M_i$ denote the set of all internal vertices that lie on some $P_i$-path. Notice that $M_i = \{01a \mid a \in M_{i-1}\} \cup \{10a \mid a \in M_{i-1}\}$. So we have:

- $|P_i| = 2|P_{i-1}| + |M_{i-1}|$;
- $|M_i| = 2|M_{i-1}|$.

Solving these recurrence relations we get $|M_i| = 2^i|M_0|$ and $|P_i| = 2^i|P_0| + i2^{i-1}|M_0|$. Let $P \triangleq P_{(n-4)/2} \subseteq V_n \times V_n$ and $M \triangleq M_{(n-4)/2} \subseteq V_n$. Since $M$ is a $P$-vertex-cut by definition, we only need to show that the sizes of $M$ and $P$ are as claimed. Indeed, $|M| = 2^{(n-4)/2}|M_0| = 2^{n/2}$ and

$|P| = 2^{(n-4)/2}|P_0| + 2^{(n-4)/2-1} \left(\frac{n-4}{2}\right) |M_0| = 2^{n/2}(n/4)$.

\qed
Chapter 8. Monotonicity testing and shortest-path routing

8.5 Interlude: Routing on the hypercube

The hypercube is a natural and well-studied architecture for multiprocessor 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 $H_n^{↑↓}$. A permutation $\pi$ of $V_n$ is 1-realizable if there exists a set of pairwise edge-disjoint paths in $H_n^{↑↓}$ such that for every $v$, there is a path in the set that connects $v$ with $\pi(v)$. Similarly, a permutation $\pi$ is $k$-realizable if there exist paths connecting every $v$ with $\pi(v)$ such that each edge is used in at most $k$ paths. In terms of the circuit-switching capability of interconnection networks this means the following. If a computer is located at every node of the directed hypercube and two neighboring vertices can communicate and send at most $k$ messages simultaneously, then for any permutation $\pi$ it is possible to set up connection paths allowing us to send messages from every computer $v$ to its addressee $\pi(v)$ without needing to send more than $k$ simultaneous messages between any pair of computers. Szymanski [Szy89] conjectured that any permutation $\pi$ 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, 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 of Theorem 8.4.1 can be used to prove the following.

8.5.1. Theorem (Briet et al. [BCGM12]). For any $\delta > 0$ and large enough $n$, there are permutations of $V_n$ that cannot be $n^{1/2-\delta}$-realized in $H_n^{↑↓}$ with shortest paths.

Proof. Let $P$ and $C$ be the pairing and cut constructed in the proof of Theorem 8.4.1. Let $\pi$ be any permutation of $V_n$ that maps each source in $P$ to its sink. Notice that any shortest path in $H_n^{↑↓}$ that connects a source of $P$ to its sink must also be a directed path in $H_n$, because the elements on each pair are comparable. So such a path must intersect $C$. Hence any realization of $P$ with shortest paths must use some edge in $C$ at least $|P|/|C| = n^{1/2-\delta}$ times.

8.5.2. Remark. Any upper bound $\mu(n)$ on the sparsity of $H_n$ can be used to show that $H_n^{↑↓}$ is not $1/\mu(n)$-realizable with shortest paths. But the opposite is

---

3Since the original conjecture was shown to be false, the weaker version not requiring shortest paths is now sometimes called Szymanski’s conjecture.
not true; in particular, the counterexample from [Lub90] does not imply that the sparsity of $H_5$ is less than 1.

8.6 New bounds on testing monotonicity

At the time of writing the best known query-complexity bounds for testing monotonicity (non-adaptively with one-sided error) of functions $f: \{0, 1\}^n \rightarrow \mathbb{R}$ were:

- an upper bound of $O(\frac{n}{\varepsilon} \log |\mathcal{R}|)$ for any range $\mathcal{R}$, by Dodis et al. [DGL+99];
- a lower bound of $\Omega(\sqrt{n}/\varepsilon)$ for boolean ranges (and hence for wider ranges too), by Fischer et al. [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 an edge-test. [DGL+99] prove that the probability that a single execution of an edge-test rejects is $\Omega(\frac{x_{opt}(f)}{n \log |\mathcal{R}|})$, 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 $|\mathcal{R}| \leq 2^n$, any upper bound of $o(\frac{n^2}{\varepsilon})$ queries would be an improvement. We make a small step in this direction. Call a function $f: \{0, 1\}^n \rightarrow \mathbb{R}$ dist-$k$ monotone if $f(y) \geq f(x)$ for every $y > x$ with $|y| > |x| + k$. In this terminology, “dist-0 monotone” simply means “monotone”.

In Section 8.6.1 we prove that given a dist-3 monotone function $f$, we can test if $f$ is monotone with $O(\frac{n^{3/2}}{\varepsilon})$ queries. The reasons for considering dist-3 monotonicity here are twofold. Firstly, it is the first non-trivial case: It is easy to see that the sparsity of any pairing contained in the set of violated pairs of a dist-2 monotone function is one, because to disconnect any pair of points at distance 2 from each other at least two edges must be removed, but in this case any edge can only be involved in two pairs (and a similar argument takes care of the pairs at distance one too). This implies (see Section 8.3) that 2-monotone functions can be tested for monotonicity with $O(n/\varepsilon)$ queries. Secondly, we saw in Section 8.4 that non-trivial sparsity upper bounds can already hold for pairings in which every source is at distance 3 from its sink.

In Section 8.6.2 we also extend the lower bound of $\Omega(\sqrt{n}/\varepsilon)$ of [FLN+02] to $\Omega(n/\varepsilon)$, for large enough $|\mathcal{R}|$. Using the “Range-Reduction Lemma” of [DGL+99], the new bound implies an improved lower bound of $\Omega(n/(\varepsilon \log n))$ for the boolean range, in the special case of pair testers whose query complexity can be written as $q(n)/\varepsilon$ for some function $q$. (A pair tester picks independent pairs of comparable vertices according to some distribution, and rejects if and only if one of the pairs
forms a violation.) We note that such testers 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 of this kind. Furthermore, the new lower bound almost matches the aforementioned upper bound of $O(n/\varepsilon)$ achieved by edge-tests (a special case of pair tests).

Section 8.6.3 describes the recent work of Blais et al., who used a clever reduction to well-studied problems in communication complexity to show that the $\Omega(n/\varepsilon)$ query complexity lower bound holds even for adaptive, two-sided-error testers of monotonicity.

8.6.1 Sparsity and dist-3-monotone functions

8.6.1. Theorem (Briët et al. [BCGM12]). Let $\varepsilon > 0$, $R \subseteq \mathbb{Z}$ and consider a dist-3 monotone function $f : \{0,1\}^n \to R$. If $f$ is $\varepsilon$-far from being monotone then $|\text{EdgeViol}(f)| \geq \Omega\left(\frac{2^n}{\varepsilon \sqrt{n}}\right)$.

**Proof.** Let $\varepsilon > 0$, $R \subseteq \mathbb{Z}$ and consider a dist-3 monotone function $f : \{0,1\}^n \to R$. If $f$ is $\varepsilon$-far from being monotone, then by Lemma 8.3.1 there is a set $P$ of $\varepsilon^2 n - 1$ vertex-disjoint pairs in $H_n$ that are violated by $f$. Furthermore, since $f$ is dist-3 monotone, for every $(s_i, t_i) \in P$ we have $|t_i| \leq |s_i| + 3$. To prove Theorem 8.6.1 we show that the sparsity of such $P$ must be $\Omega(1/\sqrt{n})$.

Let $C$ be a smallest $P$-cut, and let us prove that $|C|/|P| = \Omega(1/\sqrt{n})$. There is nothing to prove if $|C| \geq |P|/2$, so assume the opposite. It is possible to assume further that $C$ has no edges incident with any source $s^i$ or sink $t^j$ from $P$ (and in particular, this will mean that no pair in $P$ has distance 1 or 2): Let $p \leq |C| < |P|/2$ be the number of edges in $C$ that are incident to some source or sink of a pair in $P$. Removing these $p$ edges from $C$ and the corresponding pairs from $P$ leaves a set $C'$ of size $|C| - p \geq 0$ that cuts a subset $P' \subseteq P$ of at least $|P| - 2p > 0$ pairs. This is due to the fact that the pairs in $P$ are disjoint, and hence each edge can be incident with at most two pairs. We have

$$(|C| - p)|P| = |C||P| - |P|p \leq |C||P| - 2|C|p = (|P| - 2p)|C|,$$

so the sparsity of the $P'$-cut $C'$ is

$$\frac{|C'|}{|P'|} = \frac{|C| - p}{|P| - 2p} \leq \frac{|C|}{|P|}.$$

Therefore it is enough to prove the claim for $C \triangleq C'$ and $P \triangleq P'$.

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

Fix $h$. Each pair $(s^i, t^i) \in P^h$ defines a (directed) subcube graph $H_3^i = (V(H_3^i), E(H_3^i))$ of dimension 3. This subcube contains all vertices and edges that belong to one of the six possible paths from $s^i$ to $t^i$.

**8.6.2. Lemma.** For any two pairs $(s^i, t^i), (s^j, t^j) \in P^h$, $|E(H_3^i) \cap E(H_3^j)| \leq 1$.

**Proof.** Assume that $|E(H_3^i) \cap E(H_3^j)| \geq 2$ for some $i \neq j$, 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 connect 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 $e \neq e'$. □

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

Consider vertex $a_i \in A$. For every pair $(s^i, t^i) \in P^h$ such that $a_i \in V(H_3^i)$ there are two edges in $H_3^i$ going out of $a_i$. Since for any two pairs $(s^i, t^i), (s'^i, t'^i) \in P^h$ we have $|E(H_3^i) \cap E(H_3^j)| \leq 1$, it follows that the source-sink pair $(s^i, t^i)$ is completely determined by the vertex $a_i$ and the two possible edges to take from $a^i$ to reach the sink. In other words, for any $a_i \in A$, any pair of two distinct outgoing edges from $a_i$ “encodes” a unique ingoing edge. This implies $(\beta_i)^2 \geq \alpha_i$. So $\beta_i > \sqrt{2}\alpha_i$ for all $i$ and hence

$$|C^h| = \sum_{i \in [k]} \beta_i > \sum_{i \in [k]} \sqrt{\alpha_i} = \sum_{i \in [k]} \frac{\alpha_i}{\sqrt{\alpha_i}} \geq \frac{3|P^h|}{\sqrt{n}},$$

as $\alpha_i \leq n$. □

**8.6.2 A lower bound for general functions**

**8.6.3. Theorem (Briët et al. [BCGM12]).** Let $\mathcal{R} \subseteq \mathbb{Z}$, $|\mathcal{R}| > 2\sqrt{n}$. Testing monotonicity of functions $f : \{0, 1\}^n \rightarrow \mathcal{R}$ non-adaptively with one-sided error requires $\Omega(n/\varepsilon)$ queries.

**Proof.** We first prove a lower bound of $\Omega(n)$ for some constant $\varepsilon$ and argue at the end how we can achieve the promised lower bound of $\Omega(n/\varepsilon)$.

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 \( \varepsilon \)-far from monotone (for a fixed \( \varepsilon > 0 \) and all \( n \)) and such that, for any fixed set \( Q \subseteq \{0,1\}^n \) of size \( o(n) \), a uniformly random \( f \sim \mathcal{F}_n \) induces a violated pair in \( Q \) with probability less than \( 1/3 \).

For every \( n \), we will define a family \( \mathcal{F}_n = \{f_1, \ldots, f_n\} \) of \( n \) functions
\[
f_i: \{0,1\}^n \to \mathcal{R}
\]
with the following properties:
- every \( f_i \) is \( \varepsilon \)-far from monotone, for some absolute constant \( \varepsilon > 0 \);
- for any set \( Q \subseteq \{0,1\}^n \), \( \Pr_{i \in [n]}[(Q \times Q) \cap \text{Viol}(f_i) \neq \emptyset] \leq \frac{|Q|-1}{n} \).

This implies any tester making fewer than \( \frac{2n}{3} \) queries will fail with probability \( > 1/3 \).

As in the proof of the lower bound for boolean functions given in \[ FLN^02 \], each \( f_i \in \mathcal{F}_n \) will violate only pairs differing in the \( i \)-th coordinate. But unlike their construction, where distant vertices may cause violations, ours will take advantage of the larger range size to make sure that only the actual edges of \( H_n \) are violated, making it more difficult to catch violated pairs.

We formally define \( \mathcal{F}_n \) formally. Without loss of generality, let the range be \( \mathcal{R} = \{0,1, \ldots, 2\sqrt{n}\} \) (the labels can be chosen freely, and a lower bound for range \( \mathcal{R} \) also holds for ranges containing \( \mathcal{R} \)). 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:
\[
f_i(x) = \begin{cases} 
0, & h(x) < 0 \\
2\sqrt{n}, & h(x) > 2\sqrt{n} \\
h(x), & h(x) \in \mathcal{R} \text{ and } x_i \neq h(x) \mod 2 \\
h(x) + (-1)^{x_i}, & h(x) \in \mathcal{R} \text{ and } x_i = h(x) \mod 2.
\end{cases}
\]

Observe that for all \( i \in [n] \), \( \text{Viol}(f_i) = \text{EdgeViol}(f_i) \) (all violated pairs in \( f_i \) are neighbouring vertices of the hypercube), and the edges in \( \text{EdgeViol}(f_i) \) are vertex-disjoint. So by Lemma \[ 8.3.1 \] the functions \( f_i \in \mathcal{F}_n \) are \( \varepsilon \)-far from monotone (for some fixed \( \varepsilon > 0 \)) if \( |\text{EdgeViol}(f_i)| \geq \varepsilon 2^n \). Indeed, \( |\text{EdgeViol}(f_i)| \) equals the number of points \( x \in \{0,1\}^n \) such that: \( h(x) \in \mathcal{R}, h(x) \equiv 0 \pmod{2} \) 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 \( \varepsilon = \rho/5 \), we conclude that all functions \( f_i \in \mathcal{F}_n \) are \( \varepsilon \)-far from monotone.

Now we prove that \( \Pr_{i \in [n]}[(Q \times Q) \cap \text{Viol}(f_i) \neq \emptyset] \leq \frac{|Q|-1}{n} \). Fix \( Q \) and consider the undirected graph \( G = (V,E) \), where \( V = Q \) and
\[
E = \left\{ (x,y) \in Q \times Q \mid (x,y) \in E_n \right\}.
\]
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,

$$T = \{ j \in [n] \mid \text{there exists } \{ x, y \} \in E \text{ such that } x = y^{(j)} \}.$$  

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 $T$. Then $|E(G')| = |T|$. Since any cycle in the undirected skeleton of $H_n$ travels along any direction an even number of times, $G'$ is acyclic. So $|T| = |E(G')| \leq |V(G')| - 1 = |Q| - 1$.

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

Note 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 take. Our result then implies a query complexity lower bound of $\Omega(n \log n/\varepsilon)$ for pair testers of boolean monotonicity (see Section 8.6), by dint of the range reduction lemma of [DGL+99]. Although the lemma holds for every poset, we formulate it here for the particular case of the n-cube:

**8.6.4. Lemma (Dodis et al.: Range Reduction [DGL+99, Theorem 3]).** Let $c: \mathbb{N} \to \mathbb{N}^+$ and suppose for some distribution $D_n$ on pairs $(x, y) \in V_n \times V_n$ with $x \leq y$, and for every function $f: V_n \to \{0, 1\}$,

$$\Pr_{(x, y) \sim D_n} [f(x) > f(y)] \geq \frac{\varepsilon_M(f)}{c(n)}.$$

Then, for every $\mathcal{R}$ and every function $g: V_n \to \mathcal{R}$,

$$\Pr_{(x, y) \sim D_n} [f(x) > f(y)] \geq \frac{\varepsilon_M(f)}{c(n) \log |\mathcal{R}|}.$$  

**8.6.5. Corollary.** Suppose there is a pair tester of boolean monotonicity over $H_n$ whose query complexity, for each distance parameter $\varepsilon > 0$, is upper bounded by $q(n)/\varepsilon$. Then $q(n) = \Omega(n/ \log n)$. 


This is tight up to the log \( n \) factor in view of the existing \( O(n/\varepsilon) \) upper bound via pair testers.

**Proof.** Fix a distribution \( D_n \) and pick a sufficiently large range \( R \). For each function \( f : V_n \rightarrow \{0,1\} \), let \( p(f) \triangleq \Pr_{(x,y) \sim D_n}[f(x) > f(y)] \); define in the same way \( p(g) \) for each \( g : V_n \rightarrow \mathbb{R} \). The probability that a violation is detected between the values of \( f \) on two elements of the same pair after \( \frac{q(n)}{\varepsilon} \) samples is bounded by \( p(f) \frac{q(n)}{\varepsilon} \). Putting \( \varepsilon = \varepsilon_M(f) \), the existence of a tester implies \( p(f) \geq \frac{2\varepsilon_M(f)}{3\varepsilon_M(f)/(q(n) \log n)} \) for any \( f \). It follows that the range reduction lemma applies for \( c(n) = \frac{O(q(n) \log n)}{\varepsilon} \).

Choosing \( |R| > 2\sqrt{n} \), we obtain \( p(g) = \Omega(\varepsilon_M(g)/(q(n) \log n)) \) for every \( g : V_n \rightarrow \mathbb{R} \).

The expected number of draws of pairs from \( D_n \) before a violation of \( g \) is caught is \( 1/p(g) \), which is the expectation of a geometric random variable with parameter \( p(g) \). By Markov’s inequality, \( 3/p(g) \) samples suffice with probability at least \( 1/3 \).

This quantity is bounded by \( O(q(n) \log n/\varepsilon) \) whenever \( \varepsilon_M(g) \geq \varepsilon \), so we obtain a pair tester of monotonicity of functions with range \( R \) with query complexity \( O(q(n) \log n/\varepsilon) \). This contradicts Theorem 8.6.3 unless \( q(n) = \Omega(n/\log n) \).

### 8.6.3 Recent developments

After completion of this work, the lower bound has been greatly extended in a recent paper that exploits connections with some problems in communication complexity (c.f. Section 5.1).

**8.6.6. Theorem (Blais, Brody & Matulef [BBM11]).** Any adaptive, two-sided error \( \varepsilon \)-tester of monotonicity over the hypercube must make \( \Omega(n/\varepsilon) \) queries.

(As before, the range of \( f \) can be taken to be \( O(\sqrt{n}) \); see their paper for details.)

**Proof.** Let \( f : H_n \rightarrow \mathbb{Z} \). We apply a reduction from the \( n \)-disjointness problem. Let \( A, B \in \{0,1\}^n \) be Alice’s and Bob’s strings, respectively. As before, Alice builds the parity defined on \( A \), but here it is best to write it as a \( \pm 1 \) valued parity \( \chi_A \) that maps \( z \) to \( (-1)^{\sum_{i \in A} z_i} \). Likewise, Bob builds the character \( \chi_B \). Now they communicate to test whether the function \( h : \{0,1\}^n \rightarrow \mathbb{Z} \) defined by

\[
h(z) = 2 \cdot |x| + \chi_A(z) + \chi_B(z)
\]

is monotone.

Now we study when the function \( h \) is monotone. Suppose \( i \notin z \) and let us determine when \( h(z \cup \{i\}) - h(z) \) is negative. The term \( 2 \cdot |x| \) increases by two when going from \( z \) to \( z \cup \{i\} \). If \( i \notin A \) then \( \chi_A(z) = \chi_A(z \cup \{i\}) \) and the second summand stays the same; otherwise it changes by \( \pm 2 \). Similarly for \( \chi_B(z) \).
Consequently, the only way for $h(z \cup \{i\})$ to be less than $h(z)$ is for $i$ to belong to both $A$ and $B$ and to have $\chi_A(z) = \chi_B(z) = 1$.

Clearly this means that $h$ is monotone when $A$ and $B$ are not disjoint. It remains to be seen that $h$ is $\Omega(1)$-far from monotone when $A \cap B = \{i\}$. Consider the $2^{n-1}$ edges of $H_n$ in the $i$th direction. Exactly $1/4$ of these edges $(x, x \cup \{i\})$ satisfy the condition $\chi_A(x) = \chi_B(x) = 1$, and the set of those which do is vertex-disjoint. For each of these pairs of vertices, either $h(x)$ or $h(x \cup \{i\})$ needs to be modified to make $h$ monotone, because $h(x) > h(x \cup \{i\})$. Therefore, when $A \cap B \neq \emptyset$, the function $h$ is $1/8$-far from monotone, as we wished to show.

\[ \Box \]

### 8.7 Summary

We studied 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. We constructed a family of instances of $\Omega(2^n)$ 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 about routing on the hypercube.

We also proved a lower bound of $\Omega(n/\varepsilon)$ 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.

We suggest two open problems related to this line of research. The first one 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 8.4.1, 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.

As to the second one, recall from Section 8.6.1 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 8.4.1 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.