

# Directed Hypercube Routing, a Generalized Lehman-Ron Theorem, and Monotonicity Testing

Deeparnab Chakrabarty   

Dartmouth College, Hanover, NH, USA

C. Seshadhri   

University of California, Santa Cruz, CA, USA

---

## Abstract

---

Motivated by applications to monotonicity testing, Lehman and Ron (JCTA, 2001) proved the existence of a collection of vertex disjoint paths between comparable sub-level sets in the directed hypercube. The main technical contribution of this paper is a new proof method that yields a generalization of their theorem: we prove the existence of *two* edge-disjoint collections of vertex disjoint paths. Our main conceptual contributions are conjectures on directed hypercube flows with simultaneous vertex and edge capacities of which our generalized Lehman-Ron theorem is a special case. We show that these conjectures imply *directed isoperimetric theorems*, and in particular, the robust directed Talagrand inequality due to Khot, Minzer, and Safra (SIAM J. on Comp, 2018). These isoperimetric inequalities, that relate the directed surface area (of a set in the hypercube) to its distance to monotonicity, have been crucial in obtaining the best monotonicity testers for Boolean functions. We believe our conjectures pave the way towards combinatorial proofs of these directed isoperimetry theorems.

**2012 ACM Subject Classification** Theory of computation → Streaming, sublinear and near linear time algorithms; Theory of computation → Randomness, geometry and discrete structures

**Keywords and phrases** Monotonicity testing, isoperimetric inequalities, hypercube routing

**Digital Object Identifier** 10.4230/LIPIcs.ITCS.2025.34

**Related Version** *arXiv Version*: <https://www.arxiv.org/abs/2409.02206>

**Funding** *Deeparnab Chakrabarty*: Supported by NSF-CAREER CCF-2041920 and CCF-2402571. *C. Seshadhri*: Supported by NSF CCF-1740850, CCF-1839317, CCF-2402572, and DMS-2023495.

## 1 Introduction

We let  $d \geq 2$  denote a natural number. The directed  $d$ -dimensional hypercube graph  $H$  has vertices  $V(H)$  which correspond to bit-vectors  $x \in \{0, 1\}^d$ , and edges  $E(H)$  corresponding to pairs of bit-vectors  $(x, y)$  that differ in exactly one coordinate. Edges point from lower Hamming weight vectors to larger ones. We use  $x_i$  to denote the  $i$ th coordinate of vertex  $x$ . There is a natural partial order on the vertices/elements of the Boolean hypercube:  $x \preceq y$  iff  $\forall i, x_i \leq y_i$ . Note that the directed hypercube is precisely the Hasse diagram of this partial order. Equivalently, one can consider the vertices as subsets of  $[d]$ , and the partial order is given by containment.

Two subsets  $S, T$  of  $V(H)$  are called a *matched pair* if there exists a *bijection*  $\phi : S \rightarrow T$  such that  $s \prec \phi(s)$  for all  $s \in S$ ; we denote a matched pair by  $(S, T; \phi)$ . An early writeup of Goldreich-Goldwasser-Ron [30] posed a routing question, inspired by questions in monotonicity testing, which was solved by Lehman and Ron [36]. (More discussion in §4.) They were interested in the following natural question: given any matched pair  $(S, T; \phi)$ , can one find (edge or vertex) *disjoint* directed paths<sup>1</sup> from  $S$  to  $T$ ? Remarkably, they proved that if all

---

<sup>1</sup> In their paper, Lehman and Ron consider these paths to be disjoint chains of subsets.



## 34:2 Directed Hypercube Routing, a Generalized Lehman-Ron Theorem

points in  $S$  (resp.  $T$ ) have the same Hamming weight, then the answer is affirmative: one can find *vertex disjoint paths* from  $S$  to  $T$ ! More precisely, for an integer  $0 \leq i \leq d$ , let  $L_i$  denote the  $i$ th layer of  $H$ , that is,  $L_i := \{x \in V(H) : \|x\|_1 = i\}$ . We refer to this beautiful statement as the “Lehman-Ron (LR) theorem”.

► **Theorem 1** (Lehman-Ron Theorem [36]). *Fix any two integers  $i < j$ . Let  $(S, T; \phi)$  be a matched pair with  $S \subseteq L_i$  and  $T \subseteq L_j$ . Then, there are  $|S| = |T|$  vertex disjoint paths between  $S$  and  $T$ . We refer to such a set of vertex disjoint paths as an LR solution.*

► **Remark.** The paths may not respect the bijection  $\phi$ . More precisely, the above doesn’t prescribe vertex disjoint paths from  $s$  to  $\phi(s)$  for all  $s \in S$ . There exist concrete counterexamples (one is given in Lehman and Ron’s paper attributed to Dan Kleitman) for such paths. A follow-up work by proves that *edge-disjoint paths* from  $s$  to  $\phi(s)$  don’t exist either [16].

We give an alternate proof of this theorem. But more importantly, we use our new proof technique to strengthen the LR theorem. If the terminals are at distance at least 2, there exist *two* edge-disjoint LR solutions.

► **Theorem 2.** *Fix any two integers  $i < j$  with  $j - i \geq 2$ . Let  $(S, T; \phi)$  be a matched pair with  $S \subseteq L_i$  and  $T \subseteq L_j$ . Then, there are 2 collections of vertex disjoint paths between  $S$  and  $T$ , such that their union is edge disjoint.*

Lehman-Ron’s proof of Theorem 1 is by induction on  $|S|$  and on the quantity  $(j - i)$ , the distance between the layers in which  $S$  and  $T$  lie. The base case of  $j - i = 1$  is obvious as the bijection gives us the matching between  $S$  and  $T$ . The heart of the proof essentially shows the existence of a set of vertices  $U$  either in layer  $L_{j-1}$  or  $L_{i+1}$  and two bijections  $\phi' : S \rightarrow U$  and  $\phi'' : U \rightarrow T$  such that  $(S, U; \phi')$  and  $(U, T; \phi'')$  are matched pairs. This last part is a neat argument which uses Menger’s theorem, which is a special case of the max-flow-min-cut theorem, on an auxiliary graph that they create. But how can one get *two* edge disjoint LR solutions? The reader may notice that even the “base case” of  $j - i = 2$ , that is, when  $S$  and  $T$  are two levels apart is itself non-trivial (indeed, we don’t really know a much simpler way to solve this than the general case). And so, a new idea is needed to prove Theorem 2.

Our proof of the Lehman-Ron theorem brings the *flow-cut duality* idea front and center. We note that Theorem 1 is actually a statement about the structure of flows and cuts in the directed hypercube. More precisely, it states the existence of  $|S|$  units of flow from vertices in  $S$  to vertices in  $T$  when all vertices have *vertex capacity* 1 unit. We exploit the duality between cuts and flows, and more precisely the notion of *complementary slackness*, to give an alternate proof of the LR Theorem. In this flow-cut language, Theorem 2 states the existence of  $2|S|$  units of flow when both edges and vertices have capacities (1 and 2 units each, respectively). The existence of the two kinds of capacities makes the argument slightly more involved, but the essence is still the same. For completeness, we show proofs of both Theorem 1 and Theorem 2 in Section 2 and Section 3, respectively.

We end our introduction with a natural conjecture that our techniques have been unable to solve. We discuss the connections between this conjecture and monotonicity testing in Section 4. As the layers  $L_i$  and  $L_j$  move further apart, there should exist more collections of edge-disjoint LR solutions between  $S$  and  $T$ .

► **Conjecture 3.** *Fix any two integers  $i < j$  with  $r := j - i$ . Let  $(S, T; \phi)$  be a matched pair with  $S \subseteq L_i$  and  $T \subseteq L_j$ . Then, there are  $r$  collections of vertex disjoint paths between  $S$  and  $T$ , such that their union is edge disjoint.*

**Other LR connections.** Recent work has generalized the LR theorem different directions in [3]. These results find vertex disjoint paths that “cover” any collection of points specifying certain properties. Consider a subset  $X$  of the hypercube that is partitioned into subsets of paths (or chains). Meaning, we partition  $X = \bigcup_i X_i$  such that, the vertices of  $X_i$  can be ordered according to  $\prec$ . (Moreover, this is the partition that minimizes the number of sets.) In the vanilla LR setting, each  $X_i$  is just  $(s, \phi(s))$  for each  $s \in S$ . The main theorem of [3] shows that  $X$  can be covered by a collection of vertex disjoint paths. A nice implication of their result is that Theorem 1 holds even if  $S$  and  $T$  were not contained in levels, but were antichains.

We also note that the routing perspective in §4 answers a question of Sachdeva from a collection of open problems on Boolean functions [27] (Pg 19, “Routing on the hypercube”). We discuss more in §4.

## 2 Alternate proof of the Lehman-Ron Theorem

As a warm-up, we set up the main idea with a proof of the Lehman-Ron theorem. We begin with an important definition.

► **Definition 4.** *Given two sets  $S$  and  $T$  of the directed hypercube, the cover graph  $G_{S,T}$  is formed by the union of all paths from  $S$  to  $T$ .*

In other words, the cover graph is the subset of the hypercube, that contains all vertices  $v$  such that  $s \prec v \prec t$  (for  $s \in S, t \in T$ ). The cover graph inherits “layers” via intersection with the original hypercube layers. In particular, layer  $L_i$  of the cover graph is only  $S$  and layer  $L_j$  is only  $T$ .

For the sake of contradiction, consider the *minimal counterexample* of Theorem 1 in terms of  $|S| + |T|$ . Consider the following flow network which contains  $V(H)$  and also supernodes  $\textcircled{S}$  and  $\textcircled{T}$ .  $\textcircled{S}$  has a directed edge to every vertex in  $s \in S$ , and every vertex  $t \in T$  has a directed edge to  $\textcircled{T}$ . We construct a flow network by setting the following *vertex* capacities to  $G_{S,T}$ : the supernodes have infinite capacity while every vertex in  $V(H)$  has capacity 1. Since  $(S, T)$  is a counterexample, by the theory of flows and cuts, the maximum  $\textcircled{S}, \textcircled{T}$  flow in this vertex-capacitated network is  $< |S|$ . And so, using flow-cut duality, we know that there exists a *cut*  $C \subseteq V(H)$  such that (a)  $|C| < |S|$ , (b) *every* path from  $\textcircled{S}$  to  $\textcircled{T}$  contains a  $C$ -vertex. Call a path *cut-free* if it doesn’t contain a vertex from  $C$ . We can partition all vertices into three sets  $\mathcal{S}, C, \mathcal{T}$ , where  $\mathcal{S}$  contains all vertices that are reached by a cut-free path from  $\textcircled{S}$ , and vertices of  $\mathcal{T}$  can reach  $\textcircled{T}$  by a cut-free path. In particular, there is no edge from a vertex in  $\mathcal{S}$  to a vertex in  $\mathcal{T}$ ; all edges leaving  $\mathcal{S}$  enter  $C$ , and all edges entering  $\mathcal{T}$  originate from  $C$ . We make a quick observation using the minimality of our counterexample.

► **Lemma 5.**  *$C$  is disjoint from  $S \cup T$ .*

**Proof.** If  $C$  contains a vertex  $S \cup T$ , then one obtains a smaller counterexample. If  $v \in C \cap S$ , then  $S' := S \setminus \{v\}$ ,  $T' := T \setminus \{\phi(v)\}$  and  $\phi' := \phi|_{S'}$ , forms a matched pair  $(S', T'; \phi')$  which is also a counterexample: the cut  $(C - \{v\})$  is a valid cut of value  $|C| < |S| - 1 = |S'|$ . So,  $C \cap S = \emptyset$ . The proof of  $C \cap T = \emptyset$  is analogous. ◀

Our setup so far is a restructuring of the original Lehman-Ron proof. The following lemma is where we start to differ. This lemma is a consequence of *complementary slackness* from the theory of linear optimization, and is the central tool for our new proof.

## 34:4 Directed Hypercube Routing, a Generalized Lehman-Ron Theorem

► **Lemma 6.** *There exists a collection of vertex disjoint paths  $\mathcal{P}$  where every path  $p \in \mathcal{P}$  begins at a vertex in  $S$  and ends at a vertex in  $T$  and*

- *Every path  $p \in \mathcal{P}$  contains exactly one vertex in  $C$ .*
- *Every vertex  $v \in C$  is in exactly one path in  $\mathcal{P}$ .*

Note that all these paths are in the cover graph  $G_{S,T}$ . Using the above collection of paths, we make a key definition.

► **Definition 7.** *A vertex  $v$  is a gateway if (i)  $v \in S$ , (ii)  $v$  doesn't lie on any path in  $\mathcal{P}$ , and (iii) there is at least one edge  $(v, w)$  in the cover-graph.*

The Lehman-Ron theorem, Theorem 1, follows directly from the following lemma.

► **Lemma 8.** *For all  $i \leq k \leq j - 1$ , the  $k$ th layer  $L_k$  of the cover-graph contains a gateway vertex.*

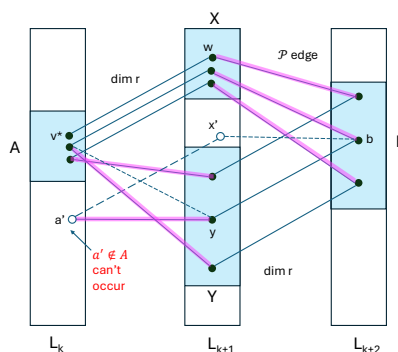
**Proof of Theorem 1.** Consider the gateway vertex  $v \in L_{j-1} \cap S$  with edge  $(v, w)$  in the cover-graph. Note  $w \in T$  and therefore in  $\mathcal{T}$ . There is an edge from  $S$  to  $\mathcal{T}$ . Contradiction. Hence, there is no (minimal) counterexample to Theorem 1. ◀

Before giving the formal proof of Lemma 8 directly, let us describe the main idea which uses the symmetry of the hypercube. First, let us observe the layer  $L_i$ , that is  $S$ , contains a gateway vertex  $s$ . Indeed, there are  $\leq |S| - 1$  paths in  $\mathcal{P}$  and so there is some  $s \in S$  not in any of these paths. Furthermore, all edges that lead  $s$  to  $\phi(s)$  lie in the cover-graph.

Now, let's see how to get a gateway in the next layer  $L_{i+1}$ . Consider any edge  $(s, x^{(1)})$  in  $G_{S,T}$ , and suppose this edge corresponds to projecting according to some dimension  $r$ . That is  $s_r = 0$  and  $x_r^{(1)} = 1$ . If  $x^{(1)}$  is not in any path in  $\mathcal{P}$ , we have discovered the desired gateway in  $L_{i+1}$ . Otherwise,  $x^{(1)}$  lies on some path, say,  $P \in \mathcal{P}$ . Follow  $P$  forwards for a single edge from  $x^{(1)}$ , to get to  $x^{(2)}$ . Note that  $x^{(2)} \in L_{i+2}$  and has  $r$ th-coordinate 1. Now, one can project “down” on the  $r$ -coordinate to get  $x^{(3)} \in L_{i+1}$ . Observe that  $(s, x^{(3)})$  is a *projection* of the edge  $(x^{(1)}, x^{(2)})$  along the  $r$ th-dimension and hence is an edge of the cover-graph. Next, observe that  $x^{(3)}$  cannot be in  $\mathcal{T}$ , so either  $x^{(3)} \in S$  or  $x^{(3)} \in C$ . If  $x^{(3)} \in S$  and not on any path in  $\mathcal{P}$ , we are done. Otherwise  $x^{(3)}$  lies on some other path  $Q \in \mathcal{P}$ . We now walk *backwards* along  $Q$ , to get  $x^{(4)} \in S$ . Noting that  $x^{(4)}$  has  $r$ -coordinate 0, we can redo the entire process above. Observe that each “step” proceeds along a *matching*. Either we project, walk from  $L_{i+1}$  to  $L_{i+2}$  using a path in  $\mathcal{P}$ , or walk backward from  $L_{i+1}$  to  $L_i$  using a path in  $\mathcal{P}$ . Each of these is using a matching edge, and no vertex is ever visited twice in the entire process. Hence, this process must terminate, at which point a gateway is discovered. And then one uses the same idea to obtain a gateway vertex in  $L_{i+2}$ , and so on. One can convert this idea into a formal proof, but it becomes notationally cumbersome. A cleaner proof method is to consider a potential “fixed point” of this process, and prove a contradiction.

**Proof of Lemma 8.** Fix a collection of paths  $\mathcal{P}$  as given by Lemma 6. As argued above,  $L_i$  has a gateway vertex. Let  $k \in [i, j - 1]$  be the largest value such that  $L_k$  contains a gateway vertex. If  $k = j - 1$ , we are done. So suppose that  $k < j - 1$ . We now engineer a contradiction. Let  $v^*$  be a gateway in  $L_k$ . So  $v^* \in S$  and has an edge  $(v^*, w)$  leaving it. Let  $r$  be the dimension of this edge implying  $v_r^* = 0$  and  $w_r = 1$ . Let  $\Pi_r$  be the projection operator which flips the  $r$ th coordinate; so  $\Pi_r(v^*) = w$  and vice versa. Define the following sets; we give an illustration for convenience where the pink highlighted edges participate in paths of  $\mathcal{P}$ .

- $A = \{a : a \in \mathcal{S} \cap L_k, a_r = 0 \text{ and } \Pi_r(a) \in G_{\mathcal{S}, \mathcal{T}}\}$ .
- $X = \Pi_r(A) = \{\Pi_r(a) : a \in A\}$ .
- $B = \{b : (x, b) \in P, \text{ for some } x \in X, P \in \mathcal{P}\}$ .
- $Y = \Pi_r(B) = \{\Pi_r(b) : b \in B\}$ .



Observe that  $X \cup Y \subseteq L_{k+1}$ , and  $B \subseteq L_{k+2}$ . We next argue these subsets lie in the cover-graph. By definition of  $A$ ,  $X \subseteq G_{\mathcal{S}, \mathcal{T}}$ , and since  $B$  is obtained via paths from  $X$ , we get  $B \subseteq G_{\mathcal{S}, \mathcal{T}}$ . Finally, for any  $y \in Y$ , note that  $b = \Pi_r(y)$  lies in  $B$ , and there's a path  $(a, x, b)$  for some  $a \in A$  and  $x = \Pi_r(a)$ . But note that  $(a, y)$  edge is the  $r$ -projection of  $(x, b)$ , and so  $(a, y, b)$  is a path implying  $y \in G_{\mathcal{S}, \mathcal{T}}$ . Thus,  $Y$  also lies in the cover-graph.

Since  $X$  is a projection of  $A$ , we get  $|X| = |A|$ . Consider any  $x \in X$ . Since  $x \in L_{k+1}$ , by our choice of  $k$ ,  $x$  isn't a gateway vertex. Since  $(a, x)$  is an edge for  $a = \Pi_r(x)$ , we have  $x \in \mathcal{S}$  or  $x \in C$ . In the former case since  $x$  isn't a gateway vertex, and in the latter case by Lemma 6, there exists  $P \in \mathcal{P}$  with  $x \in P$ . And so,  $|B| = |X|$ . By projection property,  $|Y| = |B|$ , and so following the above chain of equalities, we get  $|Y| = |A|$ .

Next consider any  $y \in Y$ . As argued above, there is a vertex  $a \in A$  such that  $(a, y)$  is an edge in the cover-graph (the projection of  $(x, b) \in P$  where  $b = \Pi_r(y)$ ). Since  $y \in L_{k+1}$ , by our choice of  $k$ ,  $y$  isn't a gateway vertex. Since  $(a, y)$  is an edge, we must have  $y \in \mathcal{S}$  or  $y \in C$ . In the former case since  $y$  isn't a gateway vertex, and in the latter case by Lemma 6, there exists  $Q \in \mathcal{P}$  with  $y \in Q$ . Let  $(a', y)$  be the edge in  $Q$  taking  $y$  "backward" along  $Q$ . We claim that  $a' \in A$ . If  $y \in \mathcal{S}$ , then  $a' \in \mathcal{S}$  since  $(a', y)$  is an edge; if  $y \in C$  then by Lemma 6,  $a' \in \mathcal{S}$  (the path  $Q$  doesn't contain two cut-vertices). Since  $(a', x' = \Pi_r(a'))$  lies in the cover graph since the  $r$ -projection of the  $(a', y)$  lies there, we get  $x'$  lies in the cover-graph, implying  $a'$  must lie in  $A$ . The above figure illustrates this. In short, there are  $|Y|$  paths of  $\mathcal{P}$  that contain vertices of  $A$ . Since  $|Y| = |A|$  and since these paths are all vertex-disjoint, we conclude *all* vertices in  $A$  are present in some path of  $\mathcal{P}$ . However, the gateway vertex  $v^* \in A$  and doesn't lie on any path of  $\mathcal{P}$ . Contradiction. ◀

### 3 Proof of the generalization Theorem 2

We now prove the generalization of the Lehman-Ron theorem using the proof strategy above. As before, we start with a minimal counter-example and use it to construct a flow network. We have the same directed graph as in the previous section with supernodes  $\textcircled{S}$  and  $\textcircled{T}$ , with every edge incident to supernodes having infinite capacity. The crucial difference is that we have both vertex and edge capacities. Each edge has capacity one, and each vertex has capacity two. (Alternately, it costs "one unit" to cut an edge, but "two units" to cut a vertex.)

Since we have a counter-example, again using the theory of flows, the maximum flow in this network is  $< 2|S|$ . And thus, by duality, we posit that there exists a *pair*  $(C, F)$  with  $C \subseteq V(H)$  and  $F \subseteq E(H)$ , such that (a)  $2|C| + |F| < 2|S|$ , (b) every path from  $s$  to  $t$  either contains a  $C$ -vertex or an  $F$ -edge or both. A path from a vertex  $u$  to  $v$  is now called *cut-free* if it contains neither a vertex from  $C$  nor an edge from  $E$ . As before, we can partition all

vertices into three sets  $\mathcal{S}, \mathcal{C}, \mathcal{T}$ , where  $\mathcal{S}$  contains all vertices that are reached by a cut-free path from  $S$ , and vertices of  $\mathcal{T}$  can reach  $T$  by a cut-free path. Note that the edges of  $F$  are from vertices in  $\mathcal{S}$  to vertices in  $\mathcal{T}$ . And, as before, by minimality of the counter-example, the following simple observation holds.

► **Lemma 9.** (i) The cut set  $C$  is disjoint from  $S \cup T$ . (ii) There exists a mincut  $(C, F)$  such that no vertex participates in more than one edge of  $F$ .

**Proof.** Proof of (i) is exactly as in Lemma 5. Suppose  $v$  participates in at least two edges of  $F$ . Observe that any  $S$ - $T$  path through any of these edges must go via  $v$ . Hence, we can remove these edges from  $F$ , add  $v$  to  $C$ , and preserve the fact that  $C \cup F$  is an  $S$ - $T$  cut. Moreover, the cut value does not increase. ◀

We can now give the analog of Lemma 6.

► **Lemma 10.** There exists a collection of paths  $\mathcal{P}$  with the following properties.

- Every path  $p \in \mathcal{P}$  begins at a vertex in  $S$  and ends at a vertex in  $T$ .
- The paths are pairwise edge-disjoint and any vertex is in at most two paths.
- Every path  $p \in \mathcal{P}$  either contains exactly one vertex in  $C$  or exactly one edge in  $F$ , but not both.
- Every vertex  $v \in C$  is in exactly two paths in  $\mathcal{P}$  and every edge  $e \in F$  is in exactly one path of  $\mathcal{P}$ .

**Proof.** By complementary slackness (Theorem A.7 of [23]), every maximum flow must saturate the min cut. Together with integrality of flow, this implies the existence of an integral flow saturating  $C \cup F$ . We give a (simple) formal explanation. By integrality of flow, there is a maximum flow that can be decomposed into paths. Let  $\mathcal{P}$  be those paths. Since these paths form a feasible flow, they satisfy the first two bullet points of the lemma. By duality,  $|\mathcal{P}| = 2|C| + |F|$ . For each path  $p \in \mathcal{P}$ , let  $c_p$  be the number of cut elements in  $C \cup F$  that the path contains. For each cut element  $e \in C \cup F$ , let  $k_e$  be the number of paths that  $e$  participates in. So  $\sum_{p \in \mathcal{P}} c_p = \sum_{e \in C \cup F} k_e$ . Note that  $\forall p, c_p \geq 1$ , since  $C \cup F$  is a valid cut. Thus,  $\sum_{p \in \mathcal{P}} c_p \geq |\mathcal{P}|$ . Now, observe that  $\forall e \in C, k_e \leq 2$  and  $\forall e \in F, k_e \leq 1$ , since  $C \cup F$  must satisfy the flow constraints. Hence,  $\sum_{e \in C \cup F} k_e \leq 2|C| + |F|$ . We get  $|\mathcal{P}| \leq \sum_{p \in \mathcal{P}} c_p = \sum_{e \in C \cup F} k_e = 2|C| + |F| = |\mathcal{P}|$ . Thus, the inequalities above are all equalities. So  $\forall p, c_p = 1$  (third bullet) and  $\forall e \in C, k_e = 2$  and  $\forall e \in F, k_e = 1$  (fourth bullet). ◀

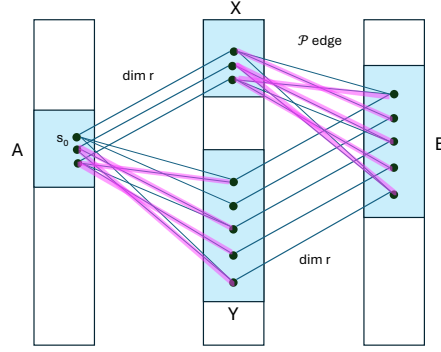
Next we provide the relevant generalization of gateway vertices earlier defined in Definition 7

► **Definition 11.** A vertex  $v$  is a gateway if (i)  $v \in \mathcal{S}$ , (ii)  $v$  lies on at most one path in  $\mathcal{P}$ , and (iii) there is at least one edge  $(v, w) \notin F$  leaving  $v$  in  $G_{\mathcal{S}, \mathcal{T}}$ .

As before, the proof of Theorem 2 follows from the following lemma, and the remainder of this section will prove it.

► **Lemma 12.** For all  $i \leq k \leq j - 1$ ,  $L_k$  contains a gateway vertex.

As in the proof of Lemma 8, we proceed via minimal counterexamples. First, we establish that there is a vertex in  $L_i$  that is a gateway vertex. There are  $< 2|S|$  paths in  $\mathcal{P}$ . Since any vertex participates in at most 2 paths, some vertex in  $s$  participates in at most 1 path. Since  $\phi(s)$  is at least distance 2 away from  $s$ ,  $s$  has degree at least 2 in  $G_{\mathcal{S}, \mathcal{T}}$ . (This is where the distance between  $S$  and  $T$  is used.) At most one of those edges is in  $F$  (Lemma 9), so there is some edge leaving  $s$  that is not in  $F$ . Thus,  $s$  is a gateway vertex.



■ **Figure 1** The setup of the proof of Lemma 12. The set  $X$  is a projection of  $A$  along dimension  $r$ , and similarly  $Y$  is a (downward) projection of  $B$ . The edges between  $A$  and  $Y$  are projections of edges between  $X$  and  $B$ .

Now, let  $k \in [i, j - 1]$  be the largest value such that  $L_k$  contains a gateway vertex. If  $k = j - 1$ , we are done. So suppose that  $k < j - 1$ , and we will engineer a contradiction. Let  $v^*$  be a gateway vertex in  $L_k$ . So  $v^* \in \mathcal{S}$ , participates in at most one path of  $\mathcal{P}$ , and there is some edge leaving  $v^*$  that is not in  $F$ . First, let us choose the projection dimension  $r$  as follows. If  $v^*$  lies on exactly one path  $P \in \mathcal{P}$ , then let  $(v^*, w)$  be the edge of  $P$  incident to  $v^*$  and let  $r$  be the coordinate that this edge flips. If  $v^*$  lies on no path  $P \in \mathcal{P}$ , then since  $k < j - 1$ , there are at least two<sup>2</sup> edges incident to  $v^*$  that are in the cover-graph; let  $(v^*, w)$  be any such edge and let  $r$  be the coordinate of this edge. As in the proof of Lemma 8, we let  $\Pi_r$  denote the projection operator, and we define the sets exactly as in the previous proof.

- $A = \{a : a \in \mathcal{S} \cap L_k, a_r = 0 \text{ and } \Pi_r(a) \in G_{\mathcal{S}, \mathcal{T}}\}$ . Note that  $v^* \in A$ .
- $X = \Pi_r(A) = \{\Pi_r(a) : a \in A\}$ .
- $B = \{b : (x, b) \in P, \text{ for some } x \in X, P \in \mathcal{P}\}$ .
- $Y = \Pi_r(B) = \{\Pi_r(b) : b \in B\}$ .

As in the proof of Lemma 8, we have  $X \cup Y \subseteq L_{k+1}$ , and  $B \subseteq L_{k+2}$  and that  $A, X, B, Y$  all lie in the cover graph  $G_{\mathcal{S}, \mathcal{T}}$ . As there, we crucially use the property that the projection of every  $(A, Y)$  edge lies in an  $(X, B)$  edge. Note that, we have  $|A| = |X|$  and  $|B| = |Y|$ . However, unlike in Lemma 8, there can be *two* paths of  $\mathcal{P}$  that may contain  $x \in X$ , and thus,  $|B|$  may be larger in size than  $|X|$ . And this makes the proof slightly more complicated.

We now restrict our attention to these four sets. As a result, we only consider the cover graph  $G_{A, B}$ , which, recall, is the union of all paths from vertices in  $A$  to vertices in  $B$  (irrespective of whether they contain cut-elements). Both  $A$  and  $B$  lie in this graph by definition. We assert  $X \subseteq G_{A, B}$ . Take a vertex  $x \in X$ . If  $x \in P$  for some path  $P \in \mathcal{P}$  then it lies in  $G_{A, B}$  by construction. We now show if  $x$  is not any path in  $\mathcal{P}$ , we already have a contradiction. Indeed, the edge  $(a, x)$  where  $a = \Pi_r(x)$  isn't in  $F$  for otherwise, by Lemma 10 there would be a path containing  $(a, x)$ . Similarly  $x \notin C$ . So,  $x \in \mathcal{S}$ . Since  $x \in L_{k+1}$  and  $k < j - 1$ , there is at least one edge leaving it, and as before, this edge isn't in  $F$ . So,  $x$  is a gateway vertex which contradicts the choice of  $k$ . Thus, the subset  $X$  lies in  $G_{A, B}$ . And since every  $(A, Y)$  edge is an  $r$ -projection of an  $(X, B)$  edge, we argue that  $Y$  also lies in  $G_{A, B}$ .

We now make another definition which will lead to our contradiction.

<sup>2</sup> There is some  $s \in \mathcal{S}$  such that  $s \prec v \prec \phi(s)$  and there are at least 2 edge disjoint paths from  $v$  to  $\phi(s)$ .

► **Definition 13.** An edge in  $G_{A,B}$  is called pink if it lies in  $\mathcal{P}$  and does not change the  $r$  coordinate. For any subset of vertices  $W$  in  $G_{A,B}$ ,  $\text{pink}(W)$  is the number of pink edges  $W$  is incident to. For singleton subsets  $\{v\}$  we abuse notation and call  $\text{pink}(\{v\})$  simply  $\text{pink}(v)$ .

Thus, every pink edge incident to  $A$  is incident to  $Y$ , and every pink edge incident to  $X$  is incident to  $B$ . The following claim is a consequence.

▷ **Claim 14.**  $\text{pink}(A) = \text{pink}(Y)$  and  $\text{pink}(X) = \text{pink}(B)$ .

Next, observe that the gateway  $v^*$  participates in at most one path of  $\mathcal{P}$ . If such a path existed, we chose the dimension  $r$  according to the incident edge of this path. Hence,  $\text{pink}(v^*) = 0$ . This is central to proving the final contradiction.

▷ **Claim 15.**  $\text{pink}(X) > \text{pink}(A)$ .

*Proof.* There is a perfect matching between  $A$  and  $X$ . We prove that  $\forall a \in A, \text{pink}(\Pi_r(a)) \geq \text{pink}(a)$ . Furthermore, for the gateway  $v^*$ , we get a strict inequality  $\text{pink}(\Pi_r(v^*)) > \text{pink}(v^*)$ . It is convenient to do a case analysis based on whether  $x = \Pi_r(a)$  is in  $\mathcal{S}, \mathcal{C}$ , or  $\mathcal{T}$ . Note that  $\text{pink}(a) \leq 2$  by Lemma 10 since any vertex participates in at most 2 paths of  $\mathcal{P}$ . Note that all edges of  $\mathcal{P}$  that leave  $X$  lead to  $B$  (by construction), so all these edges are in  $G_{A,B}$ .

$x \in \mathcal{C}$ : This is the easiest. By Lemma 10, there are two paths through  $x$ . So  $\text{pink}(x) = 2 \geq \text{pink}(a)$ .

$x \in \mathcal{T}$ : Note that  $a \in \mathcal{S}$ , and so in this case, the  $r$ -projection edge  $(a, x)$  must be in  $F$ . By Lemma 10, there is a path of  $\mathcal{P}$  through  $(a, x)$ , and the next edge leaving  $x$  goes into  $B$ . Since the  $r$ -coordinate doesn't change, this edge is pink, and so  $\text{pink}(x) \geq 1$ . Since  $a$  participates in the path through projection edge  $(a, x)$ , which is *not* pink, it can participate in at most one other path of  $\mathcal{P}$  (by Lemma 10, no vertex is in more than 2 paths). Thus, in this case  $\text{pink}(a) \leq 1$  implying  $\text{pink}(a) \leq \text{pink}(x)$ .

$x \in \mathcal{S}$ : By our assumption,  $x$  cannot be a gateway vertex. So either all edges leaving  $x$  are in  $F$  or  $x$  is in 2 paths of  $\mathcal{P}$ . If the latter happens, then  $\text{pink}(x) = 2$ , completing this case. So let us assume the former. There must be some edge leaving  $x$  in  $G_{A,B}$  because  $x$  is in the cover graph  $G_{A,B}$ ; since  $x$  is not a gateway vertex, this edge which must be in  $F$ . By Lemma 10, there is a path  $P \in \mathcal{P}$  containing this and the edge incident to  $x$  doesn't change the  $r$ -coordinate, and thus is pink. So,  $\text{pink}(x) \geq 1$ . So, if  $\text{pink}(a) \leq 1$  we are done. So, suppose  $\text{pink}(a) = 2$ . That is, there are two distinct edges  $(a, y)$  and  $(a, y')$  which are pink. Consider the  $r$ -projection of these edges,  $(x, \Pi_r(y))$  and  $(x, \Pi_r(y'))$ ; these are present in  $G_{A,B}$  and so by our assumption above, these two must be in  $F$ . But again by Lemma 10, there are two paths in  $\mathcal{P}$  containing these, and so these edges are pink, implying  $\text{pink}(x) = 2$  as well. This settles this case.

All in all, we have proven that  $\text{pink}(x) \geq \text{pink}(a)$ , and note that in all cases,  $\text{pink}(x) \geq 1$ . Since  $\text{pink}(v^*) = 0$ , we get the strict inequality  $\text{pink}(\Pi_r(v^*)) > \text{pink}(v^*)$ , completing the proof of the claim.  $\triangleleft$

The next claim which, along with Claim 14, contradicts Claim 15, completing our proof of Lemma 12.

▷ **Claim 16.**  $\text{pink}(B) \leq \text{pink}(Y)$ .

*Proof.* There is a perfect matching between  $B$  and  $Y$  by the  $\Pi_r$  projection. We will show that for every  $b \in B, \text{pink}(b) \leq \text{pink}(y)$ , where  $y = \Pi_r(b)$ . The proof is analogous to that of Claim 15, with a subtle difference. All edges of  $\mathcal{P}$  leaving  $X$  are in  $G_{A,B}$  by construction. But all edges of  $\mathcal{P}$  entering  $Y$  might not be in  $G_{A,B}$ . In particular, there could be a path  $Q \in \mathcal{P}$  which contains  $y$  and the predecessor, call it  $z$ , of  $y$  on this path may not be in  $A$ . For instance, this could occur if  $y \in \mathcal{T}$  and  $z \in \mathcal{C}$  (and thus not in  $A$ ).



With hindsight, we assert that if  $y \in S \cup C$ , then the vertex  $z$  indeed lies in  $A$ . To see this note that  $z \in S$ ; this follows from Lemma 10 since the path  $Q$  contains exactly one cut-vertex or cut-edge. Furthermore,  $\Pi_r(z)$  must lie in  $G_{S,\mathcal{T}}$  since  $(z, \Pi_r(z), b)$  is present in the hypercube. In all,  $z \in A$ . The reader may wonder why this subtlety didn't arise in the original Lehman-Ron proof that we showed in the previous section. Well, there is a vertex  $a \in A$  such that  $(a, y)$  is an edge, and since in the previous section we only had vertex cuts, we could (and did) assert  $y \in S \cup C$ . But now the edge  $(a, y)$  could be in  $F$ . As we will see, the case when  $y \in T$  is actually easy to take care of.

Now for the case analysis. Recall that  $b \in B$  and  $y = \Pi_r(w) \in Y$ .

$y \in C$ : By Lemma 10 there are two paths entering  $y$ , and since  $y \in C$  due to the discussion about the subtlety above,  $\text{pink}(y) = 2$ . Note that  $\text{pink}(b) \leq 2$  since there can be at most two paths of  $\mathcal{P}$  incident on  $b$ .

$y \in S$ : Since  $y$  is not a gateway vertex (overall assumption), either  $y$  participates in two paths of  $\mathcal{P}$  or all edges leaving  $y$  are in  $F$ . In the former case, since  $y \in S$  due to the discussion about the subtlety above,  $\text{pink}(y) = 2$ . In the latter case, the edge  $(y, b)$  must be in  $F$ . So there is at least one path through  $y$  and, again since  $y \in S$ ,  $\text{pink}(y) \geq 1$ . Now note that the  $(y, b)$  edge is *not* pink although it is on a path in  $\mathcal{P}$  because the  $r$ -coordinate changes. So,  $\text{pink}(b) \leq 1$ .

$y \in \mathcal{T}$ : Observe that all edges  $(z, y)$  with  $z \in A \subseteq S$  must be cut, that is  $(z, y) \in F$ . By Lemma 9, all such edges are on paths in  $\mathcal{P}$  and are thus pink (they are clearly in  $G_{A,B}$  and don't change  $r$ -coordinate). Suppose  $\text{pink}(b) = t$  where  $t \in \{1, 2\}$ . Then there are  $t$  distinct pink edges of the form  $(x, b)$  with these  $x$ 's in  $X$ . Consider their  $r$ -projections, that is, the  $t$  edges of the form  $(\Pi_r(x), y)$ . Since  $\Pi_r(x) \in A$ , all these edges must be pink. This shows that if  $y \in \mathcal{T}$ ,  $\text{pink}(y) \geq \text{pink}(b)$ .  $\triangleleft$

## 4 Connections to monotonicity testing

The motivation for Conjecture 3 (and indeed, Theorem 1) is a deeper understanding of the problem of monotonicity testing of functions, a problem which, especially over the hypercube and hypergrid domains, has had a rich history of more than 25 years [38, 25, 31, 24, 36, 29, 32, 1, 33, 2, 28, 40, 6, 16, 26, 13, 39, 7, 18, 19, 21, 5, 14, 20, 17, 35, 4, 22, 8, 9, 12, 34, 15, 11, 10]. A function  $f : \{0, 1\}^d \rightarrow \{0, 1\}$  is monotone if  $\forall x, y \in \{0, 1\}^d$  where  $x \prec y$ ,  $f(x) \leq f(y)$ . The distance between two functions  $f, g$  is  $|\{x : f(x) \neq g(x)\}|/2^d$ , and the distance of  $f$  to monotonicity, denote  $\varepsilon_f$ , is the minimum distance of  $f$  to a monotone  $g$ . A function  $f$  is said to be  $\varepsilon$ -far from monotone if  $\varepsilon_f \geq \varepsilon$ . The aim of a tester is to distinguish a monotone function from one that is “far” from monotone. There is a special focus on non-adaptive monotonicity testers with one-sided error. These are testers that (i) always accept monotone functions, and (ii) make all their queries in advance. After a long line of work, this has been resolved (up to  $\log d$ ,  $\text{poly}(\varepsilon^{-1})$  factors) to be  $\Theta(\sqrt{d})$  [31, 29, 19, 21, 35, 20, 22].

The study of these monotonicity testers led to the discovery of *directed isoperimetric inequalities*. Much of the study in this paper came from attempts at an alternate, more combinatorial proofs of a central isoperimetric inequality, the so-called *robust directed Talagrand theorem* due to Khot, Minzer, and Safra [35]. In this section, we give connections between monotonicity testing, directed isoperimetric inequalities (like the KMS theorem), and routing on the directed hypercube. Most importantly, we describe another routing statement, Conjecture 26, which implies the KMS theorem. We believe that Conjecture 26 and Conjecture 3 are closely related, as explained in §4.2.

## 34:10 Directed Hypercube Routing, a Generalized Lehman-Ron Theorem

► **Definition 17.** A pair of vertices of  $\{0, 1\}^d$   $x \prec y$  is called a violation of  $f(x) > f(y)$ . This pair is called a violated edge if additionally,  $(x, y)$  is an edge of the hypercube. For any  $x$ , the directed influence  $\text{Inf}_f^+(x)$  is the number of violated edges incident to  $x$ . The directed influence of  $f$ , denoted  $I_f^+$ , is  $2^{-d} \sum_x \text{Inf}_f^+(x)$ .

The most basic inequality is the directed Poincare inequality, which directly leads to  $O(d)$  query monotonicity testers (for constant  $\varepsilon$ ).

► **Theorem 18** ([31]). For any  $f : \{0, 1\}^d \rightarrow \{0, 1\}$ ,  $I_f^+ \geq \varepsilon_f$ .

The main step towards  $o(d)$  query testers (for constant  $\varepsilon$ ) is a stronger isoperimetric inequality, the directed Margulis bound. Let  $\Gamma_f^+$  denote the size of the largest matching of violated edges, which is a measure of the vertex boundary.

► **Theorem 19** ([19]). For any  $f : \{0, 1\}^d \rightarrow \{0, 1\}$ ,  $I_f^+ \cdot \Gamma_f^+ = \Omega(\varepsilon_f^2)$ .

The culmination of this line of work lead to the *robust, directed Talagrand inequality* for KMS, which yielded the (near) optimal  $\tilde{O}(\sqrt{d})$ -query non-adaptive monotonicity tester. (The original KMS result lost a log factor, which was removed by Pallavoor-Raskhodnikova-Waingarten [37].)

► **Theorem 20** ([35, 37]). Let  $\chi$  be any bicolored of the directed hypercube edges, with two colors 0 and 1. For any  $f : \{0, 1\}^d \rightarrow \{0, 1\}$  and for any  $x \in \{0, 1\}^d$ , let  $\text{Inf}_{f,\chi}^+(x)$  be the number of violated edges incident to  $x$  whose color is  $f(x)$ . Then,

$$2^{-d} \sum_x \sqrt{\text{Inf}_{f,\chi}^+(x)} = \Omega(\varepsilon_f)$$

In the next subsection, we give combinatorial interpretations to each of these statements. The reason for Conjecture 3 and a deeper study of hypercube routing was to get alternate proofs of Theorem 20. A big mystery of all these directed isoperimetric inequalities is the appearance of  $\varepsilon_f$ , the distance to monotonicity, as a “directed version” of the variance of  $f$ . It appears as if  $\varepsilon_f$  is the “right measure” of directed volume. We hope that alternate proofs of Theorem 20 may shed some light on this mystery.

### 4.1 From flows to directed isoperimetry

In what follows, all flow networks are over the directed hypercube. There is a source set  $S$ , and the aim is to route flow to the complement  $\bar{S}$ <sup>3</sup>. In the various routing theorems, we set different edge/vertex capacities and try to lower bound the maximum flow from  $S$  to  $\bar{S}$ . In all the flow settings, we have unit edge capacities.

We start with a notion of the “directed volume” of a set.

► **Definition 21.** For  $S \subseteq \{0, 1\}^d$ , the directed volume of  $S$ , denoted  $\mu^+(S)$  is

$$\max_{\substack{S' \subseteq S \\ T' \subseteq \bar{S}}} \left\{ |S'| \mid \exists \phi, (S', T'; \phi) \text{ is a matched pair} \right\}$$

Any matched pair  $(S', T'; \phi)$  that attains the maximum is called a directed volume certificate.

<sup>3</sup> Formally, one creates a supernode  $\textcircled{S}$  that connects to  $S$ , and a supernode  $\textcircled{T}$  with connections from  $T$ . All these connections have infinite capacity.

We now explain why the directed Poincare inequality of Theorem 18 essentially shows that one can send  $\mu^+(S)$  units of flow from  $S$  to  $\bar{S}$  with unit edge capacities. This is a simple application of the max-flow-min-cut theorem, and we provide the proof for completeness.

► **Theorem 22.** *Consider the directed hypercube flow network with unit edge capacities, and source set  $S$ . The maxflow is at least  $\mu^+(S)$ .*

**Proof.** Consider the indicator Boolean function  $f : \{0, 1\}^d \rightarrow \{0, 1\}$  where  $f(x) = 1$  iff  $x \in S$ . Using a standard connection between distance to monotonicity (Corollary 2 of [29]) one can argue that  $\varepsilon_f = \mu^+(S)/2^d$ . Any  $\mathbb{S}$ - $\mathbb{T}$  cut must remove all edges from  $S$  to  $\bar{S}$ . These are precisely the violated edges of  $f$ , which are at least  $\varepsilon_f 2^d = \mu^+(S)$  many (Theorem 18). The theorem follows from the duality between max-flow and min-cut. ◀

Thus, the basic directed Poincare inequality basically gives a flow bound for the directed hypercube flow network. We will now interpret the more sophisticated isoperimetric theorems as more general flow statements. A crucial notion is the separation distance of a set.

► **Definition 23.** *For any matched pair  $(S, T; \phi)$ , the separation distance is  $|S|^{-1} \sum_{s \in S} (|\phi(s)| - |s|) = |S|^{-1} \left( \sum_{t \in T} |t| - \sum_{s \in S} |s| \right)$ . Here  $|x|$  denotes the number of 1s in  $x \in \{0, 1\}^d$ . The **separation distance** of  $S$  is the smallest separation distance over directed volume certificates of  $S$ .*

A key theorem of [19] shows that larger separation distance implies more (edge disjoint) flow. This theorem is a strengthening of the directed Poincare inequality of Theorem 18, and essentially a flow rewording of Lemma 2.6 of [19]. The proof is entirely analogous to that of Theorem 22 and is omitted.

► **Theorem 24** (Lemma 2.6, [19]). *Consider the directed hypercube flow network with unit edge capacities, with source set  $S$  having separation distance  $r$ . The maxflow is at least  $r\mu^+(S)$ .*

What if we desire *vertex* disjoint paths? Lemma 2.5 of [19] answers this question, and the central tool is the Lehman-Ron theorem. Together, the two theorems above directly imply the directed Margulis statement of Theorem 19.

► **Theorem 25** (Lemma 2.5, [19]). *Consider a flow network with unit vertex capacities, with source set  $S$  having separation distance  $r$ . The maxflow is at least  $\mu^+(S)/32r$ .*

This brings us to a sort of “intellectual starting point” for this paper. The theorems above clearly show how directed isoperimetry and flows are intimately connected. Moreover, statements like Theorem 19 suggest relations between flows with edge capacities, and flow with vertex capacities. We were motivated to see if the KMS theorem (Theorem 20) could be proven from a flow perspective.

► **Conjecture 26.** *Let source set  $S$  have separation distance  $r$ . Consider a flow network with unit edge capacities and vertex capacities  $r^2$ . The maxflow is at least  $\Omega(r\mu^+(S))$ .*

Note that the above is a *simultaneous* strengthening of Theorem 24 and Theorem 25; if we remove either the edge capacity restriction or the vertex capacity restriction, then we get the above theorems. We show that Conjecture 26 implies the robust Talagrand isoperimetry theorem.

▷ **Claim 27.** Conjecture 26 implies Theorem 20.

## 34:12 Directed Hypercube Routing, a Generalized Lehman-Ron Theorem

Proof. Consider a Boolean function  $f : \{0,1\}^d \rightarrow \{0,1\}$ . Consider any bicoloring  $\chi$  of the violated edges. Our aim is to lower bound  $\sum_x \sqrt{\text{Inf}_{f,\chi}^+(x)}$ .

Let  $S$  be the set of 1-valued points. By Conjecture 26 and the maxflow-mincut theorem, the mincut of the flow network (where edges have capacity 1 and vertices have capacity  $r^2$ ) is at least  $Cr\mu^+(S)$  for some constant  $C > 0$ . Note that all edges from  $S$  to  $\bar{S}$  must be cut; moreover any separation of (the endpoints of) these edges is a valid  $(S, \bar{S})$  cut. In terms of  $f$ , these are precisely the violated edges.

Let us use  $\chi$  to construct a cut. For convenience, let  $d(x)$  denote  $\text{Inf}_{f,\chi}^+(x)$ . If  $d(x) \leq r^2$ , we cut all violated edges incident to  $x$ . Otherwise, we cut the vertex  $x$ . The total cut value is  $\sum_{x:d(x) \leq r^2} d(x) + r^2|\{x \mid d(x) > r^2\}|$ . By Conjecture 26, the cut value is at least  $Cr\mu^+(S)$ . We split into two cases.

*Case 1,*  $\sum_{x:d(x) \leq r^2} d(x) \geq Cr\mu^+(S)/2$ . Observe that  $\sum_{x:d(x) \leq r^2} \sqrt{d(x) \cdot d(x)} \leq r \sum_{x:d(x) \leq r^2} \sqrt{d(x)}$ . Thus,  $\sum_x \sqrt{\text{Inf}_{f,\chi}^+(x)} \geq C\mu^+(S)/2 = C\varepsilon_f 2^d/2$ .

*Case 2,*  $\sum_{x:d(x) \leq r^2} d(x) < Cr\mu^+(S)/2$ . So  $r^2|\{x \mid d(x) > r^2\}| \geq Cr\mu^+(S)/2$ , implying  $|\{x \mid d(x) > r^2\}| \geq C\mu^+(S)/(2r)$ . We can lower bound  $\sum_x \sqrt{d(x)} \geq r \sum_{x:d(x) > r^2} \geq C\mu^+(S)/2 = C\varepsilon_f 2^d/2$ .  $\triangleleft$

We believe that Conjecture 26 is stronger than Theorem 20, because it explicitly involves the separation distance of  $S$ .

As an aside, the connection between flows and directed isoperimetry resolves an open question in [27] (Pg 19, ‘‘Routing on the hypercube’’). It is actually a direct consequence of Theorem 18.

► **Theorem 28.** *Let  $(S, T, \phi)$  be a matched pair where  $S$  and  $T$  are disjoint. There exist  $|S|$  monotone edge disjoint paths from  $S$  to  $T$ .*

**Proof.** Consider the directed hypercube and take a mincut separating  $S$  from  $T$ . Construct a Boolean function that assigns 1 to the ‘‘ $S$ -side’’, and 0 to the ‘‘ $T$ -side’’. All remaining vertices can be assigned values such that they do not participate in any monotonicity violation. Note that these vertices cannot be on a directed path from  $S$  to  $T$ . (Process vertices according to the partial order. For  $x$ , set  $f(x)$  to be  $\max_{y \prec x: f(y) \text{ assigned}} f(y)$ . If no  $f(y)$  is assigned, set  $f(x) = 0$ . Observe that if  $f(x)$  is assigned value 1, then  $x$  must be greater than some point in  $S$ . This means that  $x$  cannot be less than any point in  $T$ , and hence does not create monotonicity violations.)

This function has distance to monotonicity at least  $|S|/2^n$ . So by Theorem 18, there are at least  $|S|$  edges which have value  $(1,0)$ . These are precisely cut edges, from the  $S$ -side to the  $T$ -side. Hence, the cut value is at least  $|S|$ . Set up a flow problem on the directed hypercube where every edge has unit capacity, vertices in  $S$  are sources, and vertices in  $T$  are sinks. By the maxflow-mincut theorem, there is a flow of value at least  $|S|$ . This flow gives edge-disjoint paths from  $S$  to  $T$ .  $\blacktriangleleft$

### 4.2 Connections between conjectures

From the perspective of monotonicity testing and directed isoperimetry, Conjecture 26 is more important. From a purely combinatorial (and maybe aesthetic) viewpoint, Conjecture 3 is more appealing. We believe that a proof of Conjecture 3 will shed light on Conjecture 26. This section is speculative, but gives some of the original motivations for studying Conjecture 3.

An uncrossing argument of [19] relates general matched pairs to matched pairs contained in level sets. (These arguments are in Section 2.4 of [19], especially Claim 2.7.2 and Claim 2.7.3.)

► **Lemma 29.** Consider a set  $S$  with separation distance  $r$ . There exist a collection of matched pairs  $(S_1, T_1, \phi_1), (S_2, T_2, \phi_2), \dots$  with the following properties.

- $\bigcup_i S_i \subseteq S, \bigcup_i T_i \subseteq \bar{S}$ .
- $\sum_i |S_i| \geq \mu^+(S)/4$ .
- Each  $S_i$  (and  $T_i$ ) is contained in a level set.
- No vertex is present in more than  $2r$  cover graphs  $G_{S_i, T_i}$ .

The main upshot of this lemma is that one can “break up” the  $(S, \bar{S})$  routing problem into a collection of  $(S_i, T_i)$  routing problems, where the  $S_i, T_i$  are level subsets. Moreover, the interaction between the various cover graphs is limited, because of the last bullet point. Given that the separation distance of  $S$  is  $r$ , we believe that for a constant fraction (by total size) of the matched pairs  $(S_i, T_i; \phi_i)$ , the distance of these pairs is  $\Omega(r)$ . If Conjecture 3 is true, we can route  $\Omega(r|S_i|)$  units of edge disjoint flow with vertex congestion  $r$ . Any vertex participates in at most  $2r$  such flow. We had hoped to overlay such flows and get an overall vertex congestion of  $O(r^2)$ . Unfortunately, a direct overlay of flows leads to an edge congestion of  $O(r)$ , which is not useful for Conjecture 26. Nonetheless, it felt that a proof of Conjecture 3 with the proof techniques of Theorem 24 might yield insight into Conjecture 26.

---

## References

- 1 Nir Ailon and Bernard Chazelle. Information theory in property testing and monotonicity testing in higher dimension. *Information and Computation*, 204(11):1704–1717, 2006. doi:10.1016/J.IC.2006.06.001.
- 2 Nir Ailon, Bernard Chazelle, Seshadhri Comandur, and Ding Liu. Estimating the distance to a monotone function. *Random Structures Algorithms*, 31(3):371–383, 2007. Prelim. version in Proc., RANDOM 2004. doi:10.1002/RSA.20167.
- 3 Paul Bastide, Carla Groenland, Hugo Jacob, and Tom Johnston. Exact antichain saturation numbers via a generalisation of a result of Lehman-Ron. *Combinatorial Theory*, 4(1), 2024. doi:10.5070/C64163848.
- 4 Aleksandrs Belovs and Eric Blais. A polynomial lower bound for testing monotonicity. *SIAM Journal on Computing (SICOMP)*, 50(3):406–433, 2021. Prelim. version in Proc., STOC 2016. doi:10.1137/16M1097006.
- 5 Piotr Berman, Sofya Raskhodnikova, and Grigory Yaroslavtsev.  $L_p$ -testing. In *Proceedings, ACM Symposium on Theory of Computing (STOC)*, 2014.
- 6 Arnab Bhattacharyya. A note on the distance to monotonicity of boolean functions. Technical Report 012, Electronic Colloquium on Computational Complexity (ECCC), 2008.
- 7 Arnab Bhattacharyya, Elena Grigorescu, Madhav Jha, Kyoming Jung, Sofya Raskhodnikova, and David Woodruff. Lower bounds for local monotonicity reconstruction from transitive-closure spanners. *SIAM Journal on Discrete Mathematics (SIDMA)*, 26(2):618–646, 2012. Prelim. version in Proc., RANDOM 2010. doi:10.1137/100808186.
- 8 Hadley Black, Deeparnab Chakrabarty, and C. Seshadhri. A  $o(d) \cdot \text{polylog}(n)$  monotonicity tester for Boolean functions over the hypergrid  $[n]^d$ . In *Proceedings, ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2018.
- 9 Hadley Black, Deeparnab Chakrabarty, and C. Seshadhri. Domain reduction: A  $o(d)$  tester for boolean functions in  $d$ -dimensions. In *Proceedings, ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2020.
- 10 Hadley Black, Deeparnab Chakrabarty, and C. Seshadhri. A  $d^{1/2+o(1)}$  monotonicity tester for boolean functions on  $d$ -dimensional hypergrids. In *Proceedings, IEEE Symposium on Foundations of Computer Science (FOCS)*, 2023.
- 11 Hadley Black, Deeparnab Chakrabarty, and C. Seshadhri. Directed isoperimetric theorems for boolean functions on the hypergrid and an  $\tilde{O}(n\sqrt{d})$  monotonicity tester. In *Proceedings, ACM Symposium on Theory of Computing (STOC)*, 2023.

- 12 Hadley Black, Iden Kalemaj, and Sofya Raskhodnikova. Isoperimetric inequalities for real-valued functions with applications to monotonicity testing. *arXiv*, abs/2011.09441, 2020. [arXiv:2011.09441](https://arxiv.org/abs/2011.09441).
- 13 Eric Blais, Joshua Brody, and Kevin Matulef. Property testing lower bounds via communication complexity. *Computational Complexity*, 21(2):311–358, 2012. Prelim. version in Proc., CCC 2011. [doi:10.1007/S00037-012-0040-X](https://doi.org/10.1007/S00037-012-0040-X).
- 14 Eric Blais, Sofya Raskhodnikova, and Grigory Yaroslavtsev. Lower bounds for testing properties of functions over hypergrid domains. In *Proceedings, IEEE Conference on Computational Complexity (CCC)*, 2014. [doi:10.1109/CCC.2014.38](https://doi.org/10.1109/CCC.2014.38).
- 15 Mark Braverman, Subhash Khot, Guy Kindler, and Dor Minzer. Improved monotonicity testers via hypercube embeddings. In *Innovations in Theoretical Computer Science (ITCS)*, pages 25:1–25:24, 2023. [doi:10.4230/LIPICs.ITCS.2023.25](https://doi.org/10.4230/LIPICs.ITCS.2023.25).
- 16 Jop Briët, Sourav Chakraborty, David García Soriano, and Ari Matsliah. Monotonicity testing and shortest-path routing on the cube. *Combinatorica*, 32(1):35–53, 2012. [doi:10.1007/S00493-012-2765-1](https://doi.org/10.1007/S00493-012-2765-1).
- 17 Deeparnab Chakrabarty, Kashyap Dixit, Madhav Jha, and C Seshadhri. Property testing on product distributions: Optimal testers for bounded derivative properties. *ACM Trans. on Algorithms (TALG)*, 13(2):1–30, 2017. Prelim. version in Proc., SODA 2015. [doi:10.1145/3039241](https://doi.org/10.1145/3039241).
- 18 Deeparnab Chakrabarty and C. Seshadhri. Optimal bounds for monotonicity and Lipschitz testing over hypercubes and hypergrids. In *Proceedings, ACM Symposium on Theory of Computing (STOC)*, 2013. [doi:10.1145/2488608.2488661](https://doi.org/10.1145/2488608.2488661).
- 19 Deeparnab Chakrabarty and C. Seshadhri. An  $o(n)$  monotonicity tester for Boolean functions over the hypercube. *SIAM Journal on Computing (SICOMP)*, 45(2):461–472, 2014. Prelim. version in Proc., STOC 2013. [doi:10.1137/13092770X](https://doi.org/10.1137/13092770X).
- 20 Xi Chen, Anindya De, Rocco A. Servedio, and Li-Yang Tan. Boolean function monotonicity testing requires (almost)  $O(n^{1/2})$  non-adaptive queries. In *Proceedings, ACM Symposium on Theory of Computing (STOC)*, 2015.
- 21 Xi Chen, Rocco A. Servedio, and Li-Yang Tan. New algorithms and lower bounds for monotonicity testing. In *Proceedings, IEEE Symposium on Foundations of Computer Science (FOCS)*, 2014.
- 22 Xi Chen, Erik Waingarten, and Jinyu Xie. Beyond Talagrand: New lower bounds for testing monotonicity and unateness. In *Proceedings, ACM Symposium on Theory of Computing (STOC)*, 2017.
- 23 William Cook, William Cunningham, William Pulleybank, and Alexander Schrijver. *Combinatorial Optimization*. Wiley Interscience Series, 1998.
- 24 Yevgeny Dodis, Oded Goldreich, Eric Lehman, Sofya Raskhodnikova, Dana Ron, and Alex Samorodnitsky. Improved testing algorithms for monotonicity. *Proceedings, International Workshop on Randomization and Computation (RANDOM)*, 1999.
- 25 Funda Ergun, Sampath Kannan, Ravi Kumar, Ronitt Rubinfeld, and Mahesh Viswanathan. Spot-checkers. *J. Comput. System Sci.*, 60(3):717–751, 2000. Prelim. version in Proc., STOC 1998. [doi:10.1006/JCSS.1999.1692](https://doi.org/10.1006/JCSS.1999.1692).
- 26 Shahar Fattal and Dana Ron. Approximating the distance to monotonicity in high dimensions. *ACM Trans. on Algorithms (TALG)*, 6(3), 2010. [doi:10.1145/1798596.1798605](https://doi.org/10.1145/1798596.1798605).
- 27 Yuval Filmus, Hamed Hatami, Steven Heilman, Elchanan Mossel, Ryan O’Donnell, Sushant Sachdeva, Andrew Wan, and Karl Wimmer. Real analysis in computer science: A collection of open problems. <https://simons.berkeley.edu/sites/default/files/openprobsmerged.pdf>, 2014.
- 28 Eldar Fischer. On the strength of comparisons in property testing. *Information and Computation*, 189(1):107–116, 2004. [doi:10.1016/J.IC.2003.09.003](https://doi.org/10.1016/J.IC.2003.09.003).

- 29 Eldar Fischer, Eric Lehman, Ilan Newman, Sofya Raskhodnikova, and Ronitt Rubinfeld. Monotonicity testing over general poset domains. *Proceedings, ACM Symposium on Theory of Computing (STOC)*, 2002. doi:10.1145/509907.509977.
- 30 O. Goldreich, S. Goldwasser, and S. Ron. A note of testing monotonicity, 1997. Technical report. URL: [https://www.wisdom.weizmann.ac.il/~oded/p\\_testMON.html](https://www.wisdom.weizmann.ac.il/~oded/p_testMON.html).
- 31 Oded Goldreich, Shafi Goldwasser, Eric Lehman, Dana Ron, and Alex Samordinsky. Testing monotonicity. *Combinatorica*, 20:301–337, 2000. Prelim. version in Proc., FOCS 1998, with authors Goldreich, Goldwasser, Lehman, and Ron. doi:10.1007/S004930070011.
- 32 Shirley Halevy and Eyal Kushilevitz. Distribution-free property testing. *Proceedings, International Workshop on Randomization and Computation (RANDOM)*, 2003.
- 33 Shirley Halevy and Eyal Kushilevitz. Testing monotonicity over graph products. *Random Structures Algorithms*, 33(1):44–67, 2008. Prelim. version in Proc., ICALP 2004. doi:10.1002/RSA.20211.
- 34 Nathaniel Harms and Yuichi Yoshida. Downsampling for testing and learning in product distributions. In *Proceedings, International Colloquium on Automata, Languages and Programming (ICALP)*, volume 229, pages 71:1–71:19, 2022. doi:10.4230/LIPICSL.ICALP.2022.71.
- 35 Subhash Khot, Dor Minzer, and Muli Safra. On monotonicity testing and boolean isoperimetric-type theorems. *SIAM Journal on Computing*, 47(6):2238–2276, 2018. Prelim. version in Proc., FOCS 2015. doi:10.1137/16M1065872.
- 36 Eric Lehman and Dana Ron. On disjoint chains of subsets. *Journal of Combinatorial Theory, Series A*, 94(2):399–404, 2001. doi:10.1006/JCTA.2000.3148.
- 37 Ramesh Krishnan S. Pallavoor, Sofya Raskhodnikova, and Erik Waingarten. Approximating the distance to monotonicity of boolean functions. *Random Structures Algorithms*, 60(2):233–260, 2022. Prelim. version in Proc., SODA 2020. doi:10.1002/RSA.21029.
- 38 Sofya Raskhodnikova. Monotonicity testing. *Masters Thesis, MIT*, 1999.
- 39 Dana Ron, Ronitt Rubinfeld, Muli Safra, Alex Samorodnitsky, and Omri Weinstein. Approximating the influence of monotone boolean functions in  $O(\sqrt{n})$  query complexity. *ACM Trans. Comput. Theory*, 4(4):11:1–11:12, 2012. Prelim. version in Proc., RANDOM 2011. doi:10.1145/2382559.2382562.
- 40 Michael E. Saks and C. Seshadhri. Parallel monotonicity reconstruction. In *Proceedings, ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2008. URL: <http://dl.acm.org/citation.cfm?id=1347082.1347187>.