Approximation Algorithms for Spanner Problems and Directed Steiner Forest*

Piotr Berman[a], Arnab Bhattacharyya[b,1], Konstantin Makarychev[c], Sofya Raskhodnikova[a,2], Grigory Yaroslavtsev[a,2]

[a] *Pennsylvania State University, University Park, PA 16802, USA.*
[b] *Massachusetts Institute of Technology, Cambridge, MA 02139, USA.*
[c] *IBM T.J. Watson Research Center, Yorktown Heights, NY 10598, USA.*

Abstract

We present an $O(\sqrt{n}\log n)$-approximation algorithm for the problem of finding the sparsest spanner of a given *directed* graph G on n vertices. A spanner of a graph is a sparse subgraph that approximately preserves distances in the original graph. More precisely, given a graph $G = (V, E)$ with nonnegative edge lengths $d : E \to \mathbb{R}^{\geq 0}$ and a *stretch* $k \geq 1$, a subgraph $H = (V, E_H)$ is a *k-spanner* of G if for every edge $(s, t) \in E$, the graph H contains a path from s to t of length at most $k \cdot d(s, t)$. The previous best approximation ratio was $\tilde{O}(n^{2/3})$, due to Dinitz and Krauthgamer (STOC '11).

We also improve the approximation ratio for the important special case of directed 3-spanners with unit edge lengths from $\tilde{O}(\sqrt{n})$ to $O(n^{1/3}\log n)$. The best previously known algorithms for this problem are due to Berman, Raskhodnikova and Ruan (FSTTCS '10) and Dinitz and Krauthgamer. The approximation ratio of our algorithm almost matches Dinitz and Krauthgamer's lower bound for the integrality gap of a natural linear programming relaxation. Our algorithm directly implies an $O(n^{1/3}\log n)$-approximation for the 3-spanner problem on undirected graphs with unit lengths. An easy $O(\sqrt{n})$-approximation algorithm for this problem has been the best known for decades.

Finally, we consider the Directed Steiner Forest problem: given a directed graph with edge costs and a collection of ordered vertex pairs, find a minimum-cost subgraph that contains a path between every prescribed pair. We obtain an approximation ratio of $O(n^{2/3+\epsilon})$ for any constant $\epsilon > 0$, which improves the $O(n^{\epsilon} \cdot \min(n^{4/5}, m^{2/3}))$ ratio due to Feldman, Kortsarz and Nutov (SODA '09).

[1] Arnab Bhattacharyya is supported by the National Science Foundation grants CCF-1065125 and CCF-0728645.

[2] Sofya Raskhodnikova and Grigory Yaroslavtsev are supported by the National Science Foundation (NSF/CCF CAREER award 0845701). Grigory Yaroslavtsev is also supported by a University Graduate Fellowship and a College of Engineering Fellowship.

* A preliminary version of this paper appeared in the proceedings of ICALP 2011 [BBM+11].

1. Introduction

A spanner of a graph is a sparse subgraph that approximately preserves distances in the original graph. This notion was first used by Awerbuch [Awe85] and explicitly introduced by Peleg and Schäffer [PS89].

Definition 1.1 (k-spanner, [Awe85, PS89])**.** *Given a graph $G = (V, E)$ with non-negative edge lengths $d : E \to \mathbb{R}^{\geq 0}$ and a real number $k \geq 1$, a subgraph $H = (V, E_H)$ is a **k-spanner** of G if for all edges $(s, t) \in E$, the graph H contains a path from s to t of length at most $k \cdot d(s, t)$. The parameter k is called the **stretch**.*

Spanners have numerous applications, such as efficient routing [Cow01, CW04, PU89b, RTZ08, TZ01], simulating synchronized protocols in unsynchronized networks [PU89a], parallel, distributed and streaming algorithms for approximating shortest paths [Coh98, Coh00, Elk01, FKM+08], algorithms for distance oracles [BS06, TZ05], property testing, property reconstruction and key management in access control hierarchies (see [BGJ+09, BGJ+12, JR11], the survey in [Ras10] and references therein).

We study the computational problem of finding the sparsest k-spanner of a given *directed* graph G, that is, a k-spanner of G with the smallest number of edges. We refer to this problem as Directed k-Spanner and distinguish between the case of unit edge lengths (i.e., $d(e) = 1$ for all $e \in E$) and arbitrary edge lengths. The Undirected k-Spanner problem refers to the task of finding the sparsest k-spanner of a given undirected graph. The natural reduction from Undirected k-Spanner to Directed k-Spanner preserves the approximation ratio.

Our main results are an algorithm with approximation ratio $O(\sqrt{n} \log n)$ for Directed k-Spanner with arbitrary edge lengths and an algorithm with approximation ratio $O(n^{1/3} \log n)$ for Directed 3-Spanner with unit edge lengths, where n is the number of nodes in the input graph G. Our approximation guarantee for Directed 3-Spanner almost matches the integrality gap of $\Omega(n^{1/3-\epsilon})$ by Dinitz and Krauthgamer [DK11] for a natural linear programming relaxation of the problem. Our result also directly implies the same approximation ratio for the Undirected 3-Spanner problem with unit edge lengths.

Our techniques also apply to the Directed Steiner Forest problem. Our result for this problem is discussed in Section 1.3.

1.1. Relation to Previous Work

Directed k-Spanner with *unit edge lengths* has been extensively studied. Note that in this case, we can assume that k is a positive integer. For $k = 2$, the problem has been completely resolved: Kortsarz and Peleg [KP94] and Elkin and Peleg [EP01] gave an $O(\log n)$-approximation, and Kortsarz [Kor01] proved that this approximation ratio cannot be improved unless P=NP. Elkin and Peleg [EP05] gave an $\tilde{O}(|E|^{1/3})$-approximation for Directed 3-Spanner, which is an $\tilde{O}(n^{2/3})$-approximation for dense graphs with $\Theta(n^2)$ edges. For general $k \geq 3$, Bhattacharyya *et al.* [BGJ+09] presented an $\tilde{O}(n^{1-1/k})$-approximation; then Berman, Raskhodnikova and Ruan [BRR10] improved it to $\tilde{O}(n^{1-1/\lceil k/2 \rceil})$, and recently Dinitz

and Krauthgamer [DK11] gave $\tilde{O}(n^{2/3})$-approximation, presenting the first algorithm with approximation ratio independent of k. For the special cases of $k = 3$ and $k = 4$, Berman, Raskhodnikova and Ruan's algorithm gives an $\tilde{O}(\sqrt{n})$-approximation. Dinitz and Krauthgamer also gave an $\tilde{O}(\sqrt{n})$-approximation for the case $k = 3$, using different techniques than in [BRR10]. Thus, our algorithms improve on [BRR10] for all $k \geq 3$, where $k \neq 4$, and on [DK11] for all $k \geq 3$.

Dinitz and Krauthgamer's algorithms also work for Directed k-Spanner with arbitrary edge lengths. For this case, one can no longer assume that k is an integer. Dinitz and Krauthgamer achieved an $\tilde{O}(n^{2/3})$-approximation for all $k > 1$ and $\tilde{O}(\sqrt{n})$ for $k = 3$ for arbitrary edge lengths. We improve this approximation ratio to $\tilde{O}(\sqrt{n})$ for all $k > 1$.

In contrast to the directed case, a simple approximation algorithm for Undirected k-Spanner was known for decades. For all integer $k \geq 3$ and for all undirected graphs G with arbitrary edge lengths, a k-spanner can be constructed in polynomial time by a greedy algorithm proposed by Althofer, Das, Dobkin, Joseph and Soares [ADD+93]. It follows from the Moore bound for irregular graphs by Alon, Hoory and Linial [AHL02] that the graph constructed by this greedy algorithm has $O(n^{1+\frac{1}{\lfloor k/2 \rfloor}})$ edges. Since a k-spanner of a connected graph must have at least $n-1$ edges, an approximation ratio $O(n^{\frac{1}{\lfloor k/2 \rfloor}})$ follows. Our result improves the ratio for Undirected 3-Spanner from $O(\sqrt{n})$ to $\tilde{O}(n^{1/3})$ in the case of unit-length edges.

Elkin and Peleg [EP00, EP07], improving on [Kor01], showed that it is quasi-NP-hard to approximate Directed k-Spanner, even when restricted to unit edge lengths, with ratio better than $2^{\log^{1-\epsilon} n}$ for $k \in (3, n^{1-\delta})$ and all $\delta, \epsilon \in (0, 1)$. For Undirected k-Spanner with unit-length edges, such a strong hardness result does not hold since the problem is $O(1)$-approximable when $k = \Omega(\log n)$.

1.2. Our Techniques

Our algorithms operate by combining two graphs: the first obtained from randomized rounding of a fractional solution to a linear programming relaxation of the problem and the second obtained by growing shortest-path trees from randomly selected vertices. The idea of combining a linear programming approach with sampling of shortest-path trees to solve Directed k-Spanner first appeared in [BGJ+09]. Dinitz and Krauthgamer [DK11] used the same approach in their main algorithm (for arbitrary stretch k), but with a novel, flow-based linear program (LP). In this paper, we propose alternative randomized LP rounding schemes that lead to better approximation ratios. Sampling and randomized rounding has been previously used by Kortsarz and Peleg [KP98] to construct undirected low-degree 2-spanners. In that work, the sampling step selects uniformly random edges, and the LP is different from ours.

We also give new LP relaxations of Directed k-Spanner, slightly simpler than that in [DK11], although they describe the same polytope. Our LP relaxation for the general case is stated in terms of *antispanners*, a graph object "dual" to spanners. An antispanner for an edge (s, t) is a set of edges whose removal from the graph destroys all paths of stretch at most k from s to t. Like in [DK11], our LP

has a polynomial number of variables and an exponential number of constraints. We use the ellipsoid algorithm with a randomized separation oracle to solve it. In the case of unit edge lengths, we present a different LP that has an extra advantage: it has a polynomial number of constraints and thus can be solved quickly without using the ellipsoid algorithm. We apply two different rounding schemes to the fractional solution of this LP: one for general stretch, another for stretch $k = 3$.

We note, however, that our method would yield the same approximation ratios with the LP of Dinitz and Krauthgamer [DK11] and, in the case of 3-spanners for graphs with unit edge lengths, with their rounding method as well. Dinitz and Krauthgamer gave a separate algorithm for Directed 3-Spanner that uses randomized rounding, but does not combine it with sampling. By combining with sampling, we obtain an algorithm with better approximation ratio for the case of unit lengths. Our rounding method allows for simpler analysis.

1.3. Directed Steiner Forest

Finally, we apply our techniques to the Directed Steiner Forest (DSF) problem, a fundamental network design problem on directed graphs. In this problem, the input is a directed graph $G = (V, E)$ with edge costs and a collection $D \subseteq V \times V$ of vertex pairs. The goal is to find a minimum-cost subgraph of G that contains a path from s to t for every pair $(s, t) \in D$. DSF is an NP-hard problem and is known [DK99] to be quasi-NP-hard to approximate with ratio better than $2^{\log^{1-\epsilon} n}$ for all $\epsilon \in (0, 1)$. DSF is also known [FKN09] to be as hard as MAX-REP, a basic problem used for hardness reductions, for which the current best approximation ratio is $O(n^{1/3})$ [CHK11].

Previous to this work, the best known approximation ratio for DSF, independent of the size of D, was $O(n^{\epsilon} \cdot \min(n^{4/5}, m^{2/3}))$ due to Feldman, Kortsarz and Nutov [FKN09]. Their algorithm has the same structure as the algorithms for Directed k-Spanner in [BGJ+09, DK11]: it combines two graphs obtained, respectively, by sampling and solving an LP. In addition, the LP relaxation they formulate is closely related to that developed by Dinitz and Krauthgamer, with edge costs replaced by edge lengths. Our technique for the spanner problem also applies to the DSF problem, yielding an improved approximation ratio of $O(n^{2/3+\epsilon})$ for any fixed $\epsilon > 0$.

1.4. Organization

In Section 2, we explain the general outline of our algorithms, introduce antispanners and show how to find an $\tilde{O}(n^{1/2})$-approximate solution to Directed k-Spanner in polynomial time. In Section 3, we present a more efficient algorithm for the special case when all the edges of the graph are of unit length. In Section 4, we show the $\tilde{O}(n^{1/3})$-approximation for Directed 3-Spanner with unit-length edges. Finally, Section 5 describes the $O(n^{2/3+\epsilon})$-approximation for Directed Steiner Forest. In Section 6 we give a conclusion and directions for future work.

2. An $\tilde{O}(\sqrt{n})$-Approximation for Directed k-Spanner

Our first result is stated in the following theorem.

Theorem 2.1. *There is a polynomial time randomized algorithm for* Directed k-Spanner *with expected approximation ratio* $O(\sqrt{n}\log n)$.

All algorithms in this paper have the same structure. They break the problem into two parts and obtain separate solutions to each part: one by random sampling and the other by randomized rounding of a solution to a linear program. We start by explaining how we break Directed k-Spanner into two parts. In Section 2.1, we describe how to obtain a solution to the first part using random sampling. Section 2.2 describes our randomized rounding scheme for Directed k-Spanner. In Section 2.3, we introduce antispanners, a graph object used to formulate and analyze our linear programming relaxations. In Section 2.4, we formulate our linear programming relaxation and separation oracle, and finish the description and analysis of the algorithm, completing the proof of Theorem 2.1.

Let $G = (V, E)$ be a directed graph with edge lengths $d : E \to \mathbb{R}^{\geq 0}$, given as an input to our algorithm, and OPT be the size of its sparsest k-spanner. We assume that G is weakly connected. Otherwise, our algorithm should be executed for each weakly connected component separately.

Definition 2.1 (Local graph $G^{s,t}$)**.** *For an edge $(s,t) \in E$, let $G^{s,t} = (V^{s,t}, E^{s,t})$ be the subgraph of G induced by the vertices that belong to paths from s to t of length at most $k \cdot d(s,t)$.*

We classify edges according to the sizes of their local graphs.

Definition 2.2 (Thick and thin edges)**.** *Let β be a parameter in $[1,n]$. If $|V^{s,t}| \geq n/\beta$, the corresponding edge (s,t) is* thick*, and otherwise, it is* thin*. The set of all thin edges is denoted by $\mathcal{E}$. In Sections 2.1–3, we set $\beta = \sqrt{n}$ and in Section 4, $\beta = n^{1/3}$.*

Definition 2.3. *A set $E' \subseteq E$* settles *an edge $(s,t) \in E$ if (V, E') satisfies the k-spanner property for this edge, i.e., it contains a path of length at most $k \cdot d(s,t)$ from s to t.*

Our algorithm must find a small subset of edges that settles all edges in E. To accomplish this, it finds two subsets of edges, E' and E'', such that E' settles all thick edges and E'' settles all thin edges. The output of the algorithm is $(V, E' \cup E'')$.

2.1. Sampling

The following procedure uses random sampling to construct an edge set E' that settles all thick edges. Recall that an *in-arborescence is a directed rooted tree where all edges are oriented towards the root; an* out-arborescence is defined similarly.

Algorithm 1 SAMPLE(β)

1: $E' \leftarrow \varnothing$, $S \leftarrow \varnothing$;
2: **for** $i = 1$ to $\beta \ln n$ **do**
3: $v \leftarrow$ a uniformly random element of V;
4: $T_v^{in} \leftarrow$ a shortest path in-arborescence rooted at v;
5: $T_v^{out} \leftarrow$ a shortest path out-arborescence rooted at v;
6: $E' \leftarrow E' \cup T_v^{in} \cup T_v^{out}$, $S \leftarrow S \cup \{v\}$; //Set S is used only in the analysis.
7: **end for**
8: Add all unsettled thick edges to E';
9: **return** E'.

Lemma 2.2. *Algorithm 1, in polynomial time, computes a set E' that settles all thick edges and has expected size at most $3\beta \ln n \cdot OPT$.*

Proof. After the execution of the **for**-loop in Algorithm 1, $|E'| \leq 2(n-1)\beta \ln n \leq 2\beta \ln n \cdot OPT$. The last inequality holds because $OPT \geq n-1$ for weakly connected graphs G.

If some vertex v from a set $V^{s,t}$ appears in the set S of vertices selected by SAMPLE, then T_v^{in} and T_v^{out} contain shortest paths from s to v and from v to t, respectively. Thus, both paths are contained in E'. Since $v \in V^{s,t}$, the sum of lengths of these two paths is at most $k \cdot d(s,t)$. Therefore, if $S \cap V^{s,t} \neq \varnothing$, then the edge (s,t) is settled. For a thick edge (s,t), the set $S \cap V^{s,t}$ is empty with probability at most $(1-1/\beta)^{\beta \ln n} \leq e^{-\ln n} = 1/n$. Thus, the expected number of unsettled thick edges added to E' in Step 8 of SAMPLE is at most $|E|/n \leq n-1 \leq OPT$.

Step 8 ensures that the set E', returned by the algorithm, settles all thick edges. Computing shortest path in- and out-arborescences and determining whether an edge is thick can be done in polynomial time. □

2.2. Randomized Rounding

To obtain a set E'' that settles all thin edges, each of our algorithms solves a linear program and rounds the resulting fractional solution. The LP is a relaxation of DIRECTED k-SPANNERfor the set of all thin edges. It has a variable x_e and a constraint $x_e \geq 0$ for each edge $e \in E$. The variable x_e in the corresponding optimal {0,1}-solution indicates whether the edge e is present in the smallest spanner for all thin edges. The following randomized rounding procedure is used in our algorithms for DIRECTED k-SPANNER, both for arbitrary and for unit lengths. As an input it gets a fractional vector $\{\hat{x}_e\}$ with nonnegative entries.

Algorithm 2 RANDOMIZEDSELECTION($\hat{x}_e$)

1: $E'' \leftarrow \varnothing$;
2: **for** each edge $e \in E$ **do**
3: Add e to E'' with probability $\min(\sqrt{n} \ln n \cdot \hat{x}_e, 1)$;
4: **end for**
5: **return** E''.

The following proposition shows that if the sum of values assigned by $\{\hat{x}_e\}$ to edges in some $A \subseteq E$ is at least 1 then E'' intersects A with high probability.

Claim 2.3. *Let $A \subseteq E$. If Algorithm 2 receives a fractional vector $\{\hat{x}_e\}$ with nonnegative entries satisfying $\sum_{e\in A} \hat{x}_e \geq 1$, the probability that it outputs a set E'' disjoint from A is at most* $\exp(-\sqrt{n}\ln n)$.

Proof. If A contains an edge e, such that $\hat{x}_e \geq (\sqrt{n}\ln n)^{-1}$, then $e \in E''$ with probability 1. That is, E'' is never disjoint from A.

Otherwise, for all edges $e \in A$, the probability that $e \in E''$ is exactly $\sqrt{n}\ln n \cdot \hat{x}_e$. The probability that no edges of A are in E'' is, therefore,

$$\prod_{e\in A}(1 - \sqrt{n}\ln n \cdot \hat{x}_e) \leq \exp\left(-\sum_{e\in A} \sqrt{n}\ln n \cdot \hat{x}_e\right) \leq \exp(-\sqrt{n}\ln n).$$

The first inequality above follows from the fact that $1 - x \leq \exp(-x)$ for $x \geq 0$. The second one holds because $\sum_{e\in A} \hat{x}_e \geq 1$. □

2.3. Antispanners

In this section, we introduce antispanners, a graph object used in the description of our LP for Directed k-Spanner and crucial in the analysis of the parts of our algorithms that settle thin edges. After giving the definition, we show how to construct minimal antispanners (in Claim 2.4) and give an upper bound on their number (in Claim 2.5.)

For a given edge (s, t), we define an antispanner to be a subset of edges of G, such that if we remove this subset of edges from G, the length of the shortest path from s to t becomes larger than $k \cdot d(s, t)$.

Definition 2.4 (Antispanner)**.** *A set $A \subseteq E$ is an antispanner for an edge $(s, t) \in E$ if $(V, E \setminus A)$ contains no path from s to t of length at most $k \cdot d(s, t)$. If no proper subset of an antispanner A for (s, t) is an antispanner for (s, t) then A is minimal. The set of all* minimal *antispanners for all* thin *edges is denoted by $\mathcal{A}$.*

The edge set of a k-spanner of G must intersect all antispanners for all edges of G. In other words, it has to be a hitting set for all minimal antispanners. Specifically, a set E'' that settles all thin edges must be a hitting set for all minimal antispanners in $\mathcal{A}$. We now prove that if a set E'' does not settle some thin edge, then we can efficiently find a minimal antispanner $A \in \mathcal{A}$ disjoint from E''.

Claim 2.4. *There exists a polynomial time algorithm that, given a set of edges $E'' \subset E$ that does not settle some thin edge, outputs a minimal antispanner $A \in \mathcal{A}$ for some thin edge, such that $A \subseteq E \setminus E''$.*

Proof. The algorithm first finds a thin edge (s, t) with no directed path from s to t of length at most $k \cdot d(s, t)$ in E''. Recall that all paths from s to t of length at most $k \cdot d(s, t)$ in G lie in the local graph $G^{s,t} = (V^{s,t}, E^{s,t})$. (See Definition 2.1.) Therefore, $E^{s,t} \setminus E''$ is an antispanner for (s, t). The algorithm sets $A = E^{s,t} \setminus E''$ and then greedily deletes edges e from A while $A \setminus \{e\}$ is an antispanner, that is, while $(V^{s,t}, E^{s,t} \setminus A)$ contains no paths of length at most $k \cdot d(s, t)$ from s to t. When no more such edges can be deleted, the algorithm returns A. □

$$\text{Minimize} \sum_{e \in E} x_e \text{ subject to:} \tag{1}$$

$$\sum_{e \in A} x_e \geq 1 \qquad \forall A \in \mathcal{A} \tag{2}$$

$$x_e \geq 0 \qquad \forall e \in E \tag{3}$$

Figure 1: Linear program for the arbitrary-length case, LP-A. $\mathcal{A}$ is the set of all minimal antispanners for thin edges.

Next, we give an upper bound on the number of minimal antispanners for thin edges.

Claim 2.5. *$|\mathcal{A}| \leq |E| \cdot (n/\beta)^{n/\beta}$. In particular, if $\beta = \sqrt{n}$, then $|\mathcal{A}| \leq \sqrt{n}^{\sqrt{n}+4}$.*

Proof. Fix a thin edge (s,t) and a minimal antispanner A for (s,t). Let T_A be an out-arborescence (shortest-path tree) rooted at s in the graph $(V^{s,t}, E^{s,t} \setminus A)$. Denote by $d_{T_A}(u)$ the distance from s to u in the tree T_A. If T_A contains no directed path from s to u, let $d_{T_A}(u) = \infty$. We show that $A = \{(u,v) \in E^{s,t} : d_{T_A}(u) + d(u,v) < d_{T_A}(v)\}$, and thus T_A uniquely determines A for a given thin edge (s,t).

Consider an edge $(u,v) \in A$, and let A^- denote $A \setminus \{(u,v)\}$. Since the antispanner A is minimal, the graph $(V, E \setminus A^-)$ contains a path from s to t of length at most $k \cdot d(s,t)$. This path must lie in $(V^{s,t}, E^{s,t} \setminus A^-)$ and must contain the edge (u,v). Thus, the distance from s to t in the graph $(V^{s,t}, E^{s,t} \setminus A^-)$ is at most $k \cdot d(s,t)$ and is strictly less than $d_{T_A}(t)$. Hence, T_A is not a shortest-path tree in the graph $(V^{s,t}, E^{s,t} \setminus A^-)$. Therefore, $d_{T_A}(u) + d(u,v) < d_{T_A}(v)$.

If $(u,v) \in E^{s,t}$ satisfies the condition $d_{T_A}(u) + d(u,v) < d_{T_A}(v)$, then $(u,v) \notin E^{s,t} \setminus A$; otherwise, T_A would not be a shortest-path tree. Hence, $(u,v) \in A$.

We now count the number of out-arborescences rooted at s in $(V^{s,t}, E^{s,t} \setminus A)$. For every vertex $u \in V^{s,t}$, we may choose the parent vertex in at most $|V^{s,t}|$ possible ways (if a vertex is a not reachable from s, we choose it as its own parent). Thus, the total number of trees is at most $|V^{s,t}|^{|V^{s,t}|} \leq (n/\beta)^{n/\beta}$.

Since there are at most $|E|$ thin edges, the claim follows. □

2.4. LP, Separation Oracle and Overall Algorithm

In this section, we describe a randomized algorithm for constructing a small subset of edges $E'' \subseteq E$ that settles all thin edges. First, we formulate an LP relaxation of this problem. Then we describe how to solve it using the ellipsoid method with a separation oracle (Section 2.4.1). Finally, in Section 2.4.2, we summarize the resulting algorithm for Directed k-Spanner and complete the proof of Theorem 2.1.

A set E'' that settles must intersect all minimal antispanners for all thin edges. This condition can be expressed using linear program LP-A (see Fig. 1). LP-A has a variable x_e for each edge $e \in E$ and a constraint (2) for each minimal antispanner A for every thin edge. Recall that $\mathcal{A}$ is the set of all minimal antispanners for thin

edges. In the integral solution $\{x_e^{int}\}$ corresponding to a k-spanner with an edge set $E'' \subseteq E$, we set $x_e^{int} = 1$ if $e \in E''$ and $x_e^{int} = 0$ otherwise. All constraints in (2) are satisfied for $\{x_e^{int}\}$ since E'' intersects every antispanner. The value of the objective function $\sum_e x_e^{int}$ is equal to the size of E''. Hence, LP-A is a relaxation of Directed k-Spanner.

For ease of presentation, we assume that we have guessed OPT, the size of the sparsest spanner. (We can try all values in $\{n-1, \ldots, n^2\}$ for OPT and output the sparsest spanner found in all iterations). We replace the objective function (1) with

$$\sum_{e \in E} x_e \leq OPT, \tag{4}$$

and call the resulting linear program LP-A′.

2.4.1. Separation Oracle

LP-A′ has a polynomial number of variables and, by Claim 2.5, an exponential in $\tilde{O}(\sqrt{n})$ number of constraints. We solve it using the ellipsoid algorithm with a separation oracle. Our separation oracle receives a fractional vector $\{\hat{x}_e\}$, satisfying (3) and (4). If $\{\hat{x}_e\}$ is a feasible solution to LP-A′, then the separation oracle outputs a set E'' of size at most $2OPT \cdot \sqrt{n} \ln n$, which settles thin edges. Otherwise, it outputs either a set E'' with the same guarantee or a violated constraint from (2) for some antispanner A. The separation oracle can also fail with small probability. If it happens during an execution of the ellipsoid algorithm, we output the input graph with all its edges as a k-spanner.

Algorithm 3 SeparationOracle($\hat{x}_e$)

```
//Sample a random set of edges E'', picking each e ∈ E
//with probability min(x̂_e √n ln n, 1) (see Algorithm 2).
E'' ← RandomizedSelection(x̂_e)
if E'' settles all thin edges then
    if |E''| ≤ 2OPT · √n ln n then return E'';
    else fail;
else
    Find an antispanner A ⊆ E\E'' from 𝒜 using the algorithm from Claim 2.4.
    if Σ_{e∈A} x_e < 1 then return violated constraint Σ_{e∈A} x_e ≥ 1;
    else fail.
end if
```

The separation oracle is described in Algorithm 3. Next we analyze the probability that the separation oracle fails.

Lemma 2.6. *The probability that the separation oracle fails during an execution of the ellipsoid algorithm is exponentially small in n.*

Proof. The separation oracle can fail for two reasons:

1. The size of the sampled set E'' is too large.
2. The minimal antispanner A found by the oracle does not correspond to a violated constraint.

To analyze the probability of the first event, note that the expected size of E'' is at most $\sqrt{n}\ln n \sum_{e\in E} x_e \leq OPT \cdot \sqrt{n}\ln n$. By the Chernoff bound,

$$\Pr[|E''| > 2OPT \cdot \sqrt{n}\ln n] \leq \exp(-c \cdot OPT \cdot \sqrt{n}\ln n) = \exp(-\Omega(n\sqrt{n}\ln n)).$$

Thus, the probability that the separation oracle fails because $|E''| > 2OPT \cdot \sqrt{n}\ln n$ is exponentially small in n.

To analyze the probability of the second event, consider one call to the separation oracle. Fix a minimal antispanner A satisfying $\sum_{e\in A} \hat{x}_e \geq 1$. Claim 2.3 shows that the probability that E'' is disjoint from A is at most $\exp(-\sqrt{n}\ln n)$. Claim 2.5 demonstrates that $|\mathcal{A}| \leq \sqrt{n}^{\sqrt{n}+4}$. Therefore, by a union bound, the probability that there is a minimal antispanner $A \in \mathcal{A}$ satisfying $\sum_{e\in A} \hat{x}_e \geq 1$ and also disjoint from E'' is at most $\sqrt{n}^{\sqrt{n}+4} \cdot \exp(-\sqrt{n}\ln n) = \exp(-\frac{1}{2}\sqrt{n}\ln n + 2\ln n)$. Thus, the probability that the separation oracle fails during one call because $\sum_{e\in A} \hat{x}_e \geq 1$ is exponentially small in n. Since the number of iterations of the ellipsoid algorithm is polynomial in n, a union bound over all iterations gives that the overall probability that the separation oracle fails during an execution of the ellipsoid algorithm is exponentially small in n. □

Lemma 2.6 implies, in particular, that when the separation oracle is given a feasible solution to LP-A′, it fails to output a set E'' with exponentially small probability. Since E'' is obtained by running Algorithm 2, we obtain the following corollary that will be used in Section 3.

Corollary 2.7. *Given a feasible solution to LP-A′, Algorithm 2 with all but exponentially small probability produces a set E'' that settles thin edges and has size at most $2OPT \cdot \sqrt{n}\ln n$.*

2.4.2. *Overall Algorithm for* Directed k-Spanner

Proof of Theorem 2.1. We settle thick edges by running Sample($\sqrt{n}$), according to Lemma 2.2. We settle thin edges by running the ellipsoid algorithm as described in the beginning of Section 2.4 and in Section 2.4.1. If the separation oracle fails, which, by Lemma 2.6, happens with exponentially small probability, we output a spanner containing all edges E. Thus, the expected size of the set E'' is at most $2OPT \cdot \sqrt{n}\ln n + o(1)$, and the resulting approximation ratio of the algorithm is $O(\sqrt{n}\ln n)$. The ellipsoid algorithm terminates in polynomial time, so the overall running time is polynomial. □

3. LP and Rounding for Graphs with Unit-Length Edges

In this section, we describe how to settle all thin edges, and thus prove Theorem 2.1, for the case of unit-length edges. Our motivation for presenting this

special case is two-fold. First, we show that for the unit-length case, one can directly formulate a polynomial-sized LP relaxation, and this makes the approximation algorithm more efficient. Second, we also use the LP from this section to present a better algorithm for 3-spanners in Section 4.

Our LP for the case of unit lengths, LP-U, is stated in terms of *local layered graphs* which we introduce next.

Definition 3.1 (Layered expansion)**.** *Given a directed graph $G = (V, E)$, its* layered expansion *is a directed graph $\bar{G} = (\bar{V}, \bar{E})$, satisfying the following:*

1. *Let $\bar{V} = \{v_i : v \in V \text{ and } i \in \mathbb{Z}^{\geq 0}\}$, where v_i denotes the i-th copy of v. The set of all the i-th copies of nodes in V is the i-th layer of $\bar{V}$.*
2. *Let $L = \{(u, u) : u \in V\}$ be the set of loops. Define the i-th copy of an edge $e = (u, v)$ to be $e_i = (u_i, v_{i+1})$, and the i-th copy of a loop $e = (u, u)$ to be $e_i = (u_i, u_{i+1})$. Let $\bar{E} = \{e_i : e \in E \cup L \text{ and } i \in \mathbb{Z}^{\geq 0}\}$.*

Layered expansion $\bar{G}$ contains a path from s_0 to t_k if and only if G contains a path from s to t of length at most k. A *local layered graph* for a thin edge (s, t) is defined next. It consists of all paths in the layered expansion $\bar{G}$ that correspond to paths from s to t of length at most k in the original graph G or, in other words, to paths in the local graph $G^{s,t}$, defined in Definition 2.1.

Definition 3.2 (Local layered graph)**.** *For a thin edge (s, t) and $k \geq 1$, the* local layered graph *is a subgraph $\bar{G}^{s,t} = (\bar{V}^{s,t}, \bar{E}^{s,t})$ of $\bar{G}$ with a source $\bar{s} = s_0$ and a sink $\bar{t} = t_k$, such that $\bar{G}^{s,t}$ contains all nodes and edges on paths from $\bar{s}$ to $\bar{t}$.*

Our algorithm solves the linear program LP-U defined in Figure 2. Recall that $\mathcal{E}$ denotes the set of thin edges. LP-U has variables of two types: x_e, where $e \in E$, and $f_{e_i}^{s,t}$, where $(s, t) \in \mathcal{E}$ and $e_i \in \bar{E}^{s,t}$. A variable x_e represents whether the edge e is included in the k-spanner. We think of a path from s to t of length at most k in G as a unit flow from $\bar{s}$ to $\bar{t}$ in $\bar{G}^{s,t}$. A variable $f_{e_i}^{s,t}$ represents flow along the edge e_i in $\bar{G}^{s,t}$. We denote the sets of incoming and outgoing edges for a vertex $v_i \in \bar{G}^{s,t}$ by $In(v_i)$ and $Out(v_i)$, respectively.

Given $\hat{x}_e$, a fractional solution of LP-U, we construct the set E'' by first running Algorithm 2 and then adding all unsettled thin edges.

Lemma 3.1. *The algorithm described above, in polynomial time, computes a set E'' that settles all thin edges and has expected size at most $2\sqrt{n}\ln n \cdot OPT + o(1)$.*

Proof. We prove, in Claim 3.2, that in a fractional optimal solution $\{\hat{x}_e\} \cup \{\hat{f}_{e_i}^{s,t}\}$ to LP-U, the vector $\{\hat{x}_e\}$ is a fractional solution to LP-A′. Then we apply Corollary 2.7 to get the desired bound on the expected size of E''. At the end, we argue that the algorithm runs in polynomial time.

Claim 3.2. *In a fractional optimal solution $\{\hat{x}_e\} \cup \{\hat{f}_{e_i}^{s,t}\}$ to LP-U, the vector $\{\hat{x}_e\}$ is a fractional solution to LP-A′.*

Minimize $\sum_{e\in E} x_e$ subject to:

$$\text{Flow requirement} \qquad \sum_{e_0\in Out(s_0)} f^{s,t}_{e_0} \geq 1 \quad \forall (s,t)\in\mathcal{E}$$

$$\text{Flow conservation} \qquad \sum_{e_{i-1}\in In(v_i)} f^{s,t}_{e_{i-1}} - \sum_{e_i\in Out(v_i)} f^{s,t}_{e_i} = 0 \quad \forall (s,t)\in\mathcal{E}, \forall v_i \in \bar{V}^{s,t}\setminus\{\bar{s},\bar{t}\}$$

$$\text{Capacity constraints} \qquad x_e - \sum_{i=0}^{k-1} f^{s,t}_{e_i} \geq 0 \quad \forall (s,t)\in\mathcal{E}, \forall e\in E$$

$$x_e \geq 0 \quad \forall e\in E$$

$$f^{s,t}_{e_i} \geq 0 \quad \forall (s,t)\in\mathcal{E}, \forall e_i \in \bar{E}^{s,t}$$

Figure 2: Linear program for the unit-length case, LP-U.

Proof. First, we argue that LP-U is a relaxation of Directed k-Spanner for the unit-length case or, in other words, that an optimal solution to this program has value at most OPT. Let H be a sparsest k-spanner of G. Assign $x_e = 1$ if e is in H and $x_e = 0$ otherwise. For each thin edge (s,t), consider a simple path from s to t in H of length ℓ, where $\ell \leq k$. Set $f^{s,t}_{e_i}$ to 1 if either e is the i-th edge on that path or $i \in \{\ell+1,\ldots,k\}$ and $e_i = (t_{i-1}, t_i)$; otherwise, set it to 0. Since the resulting assignment is a feasible solution to LP-U, the optimal solution to this program has value $\sum_{e\in E}\hat{x}_e \leq OPT$.

Next, we argue that if $\{\hat{x}_e\}\cup\{\hat{f}^{s,t}_{e_i}\}$ is a feasible solution to LP-U then $\{\hat{x}_e\}$ satisfies the antispanner constraints for LP-A′, given in (2). Consider a thin edge (s,t) and a minimal antispanner $A\in\mathcal{A}$ for (s,t). Let $\bar{A} = \{e_i :\ e\in A \text{ and } e_i\in\bar{E}^{s,t}\}$ be the set of copies of the edges in A in the local layered graph. Let $\bar{S}\subseteq\bar{V}^{s,t}$ be the set of nodes that can be reached from $\bar{s}$ in $(\bar{V}^{s,t}, \bar{E}^{s,t}\setminus\bar{A})$ and $\bar{T} = \bar{V}^{s,t}\setminus\bar{S}$ be the set of the remaining nodes. Since A is an antispanner for (s,t), node $\bar{t}$ is in $\bar{T}$, and thus $(\bar{S},\bar{T})$ is an $(\bar{s},\bar{t})$ cut in $\bar{G}^{s,t}$. Note that only edges from $\bar{A}$ can cross the cut because for an edge $(u_i, v_{i+1})\notin\bar{A}$ if u_i is reachable from $\bar{s}$ then so is v_{i+1}.

For a fractional solution $\{\hat{x}_e\}\cup\{\hat{f}^{s,t}_{e_i}\}$ to LP-U,

$$\sum_{e\in A}\hat{x}_e \geq \sum_{e\in A}\sum_{i=0}^{k-1}\hat{f}^{s,t}_{e_i} = \sum_{e_i\in\bar{A}}\hat{f}^{s,t}_{e_i} \geq \sum_{e_i\in\, cut\,(\bar{S},\bar{T})}\hat{f}^{s,t}_{e_i} = \sum_{e_0\in Out(s_0)}\hat{f}^{s,t}_{e_0} \geq 1. \tag{5}$$

The first inequality above follows from the capacity constraints in LP-U, the following equality holds by definition of $\bar{A}$, the second inequality holds because $\bar{A}$ contains the edges in the cut $(\bar{S},\bar{T})$, the last equality follows from the flow conservation, and the last inequality is the flow requirement.

We proved that in a fractional optimal solution $\{\hat{x}_e\}\cup\{\hat{f}^{s,t}_{e_i}\}$ to LP-U, the vector $\{\hat{x}_e\}$ satisfies constraints (2) and (4) of LP-A′. Since constraints (3) are also in LP-U, vector $\{\hat{x}_e\}$ is a fractional solution to LP-A′. □

By, Claim 3.2, vector $\{\hat{x}_e\}$ is a fractional solution to LP-A′. Corollary 2.7 says that, given such a solution, Algorithm 2 with all but exponentially small probability produces a set E'' that settles thin edges and has size at most $2OPT \cdot \sqrt{n} \ln n$. After we add all unsettled thin edges, the expected size of the resulting set E'' is at most $2OPT \cdot \sqrt{n} \ln n + o(1)$.

It remains to argue that the described algorithm takes polynomial time. To write down LP-U, we only need to know V, E, k and the set of thin edges, $\mathcal{E}$. The first three are inputs to the algorithm, and $\mathcal{E}$ can be computed in polynomial time. LP-U can be written down and solved in polynomial time because it has $O(|E|^2 \cdot k) = O(n^5)$ variables and constraints. □

Proof of Theorem 2.1 for the case of unit-lengths. We run Algorithm 1 to get E'. We construct E'' by running Algorithm 2 and adding all unsettled thin edges. Let the edge set of our k-spanner be $E' \cup E''$. By Lemmas 2.2 and 3.1, E' settles all thick edges, E'' settles all thin edges, the expected size of $E' \cup E''$ is $O(\sqrt{n} \ln n \cdot OPT)$, and the resulting algorithm runs in polynomial time, as required. □

4. An $\tilde{O}(n^{1/3})$-Approximation for Directed 3-Spanner with Unit-Length Edges

In this section, we show an improved approximation for the special case of Directed 3-Spanner with unit-length edges. The algorithm follows the general strategy explained in Section 2. The LP rounding scheme here is different from that presented in Section 2.2 and used in the two algorithms for Directed k-Spanner in Sections 2 and 3. We note that Algorithm 2 from [DK11] with $\rho = \tilde{\Theta}(n^{1/3})$ could also be used to prove our result. The rounding scheme we present is simpler and allows for simpler analysis.

As in [DK11], we use random variables for vertices instead of edges to guide edge selection process. Intuitively, this allows us to introduce positive correlations in selection of edges adjacent to the same vertex. Because the correlations are local, the improvement in approximation deteriorates for larger values of k. To simplify analysis, instead of threshold rounding (as in the previous sections) we use Poisson random variables.

Theorem 4.1. *There is a polynomial time randomized algorithm for* Directed 3-Spanner *for graphs with unit edge lengths with expected approximation ratio* $O(n^{1/3} \log n)$.

Proof. We define thick and thin edges as in Definition 2.2, with $\beta = n^{1/3}$, and run Sample($n^{1/3}$). By Lemma 2.2, the resulting edge set E' settles all thick edges and has expected size at most $3n^{1/3} \ln n \cdot OPT$. Then we obtain an optimal solution $\{\mathring{x}_e\} \cup \{\mathring{f}^{s,t}_{e_i}\}$ of the linear program LP-U from Fig. 2. Our rounding scheme is stated in Algorithm 4. It consists of two stages: first, we round $\{\mathring{x}_e\}$ to obtain a new solution $\{\hat{x}_e\}$, where every assignment $\hat{x}_e$ is an integer multiple of $n^{-2/3}$; second, we round $\{\hat{x}_e\}$ to obtain an edge set E'' that settles all thin edges with high probability.

In the first step we sample a random variable from Poisson distribution for every edge. Recall, that a Poisson random variable X with mean λ is supported over nonnegative integers and has a probability density function:

$$\Pr[X = k] = \frac{\lambda^k e^- \lambda}{k!}, \forall k \in \mathbb{Z}^{\geq 0}.$$

The only properties of the Poisson distribution that we use in the analysis are concentration bound stated in the Appendix, integrality of the support and the fact that the sum of Poisson random variables is again a Poisson random variable.

Algorithm 4 RANDOMIZED3SPANNERSELECTION($\mathring{x}_e$)

1: $E'' \leftarrow \varnothing$;
   //Obtain a new solution $\{\hat{x}_e\}$, where each coordinate $\hat{x}_e$ is a multiple of $n^{-2/3}$:
2: **for** each edge $e \in E$ **do**
3: $\quad P_e \leftarrow$ sample from the Poisson distribution with mean $\lambda_e = 6n^{2/3}\mathring{x}_e$;
4: $\quad \hat{x}_e \leftarrow P_e n^{-2/3}$;
5: **end for**
   //Round $\{\hat{x}_e\}$ to get E'':
6: **for** each vertex $u \in V$ **do**
7: $\quad r_u \leftarrow$ uniform sample from $(0, 1)$;
8: **end for**
9: **for** each edge $e = (u, v) \in E$ **do**
10: $\quad$ **if** $\min(r_u, r_v) \leq \hat{x}_e \alpha\, n^{1/3} \ln n$ **then** add e to E'';
11: $\quad$ //$\alpha > 1$ is an absolute constant
12: **end for**
13: **return** E''.

Lemma 4.2 below analyzes the first stage. Then Lemmas 4.3 and 4.4 analyze the set E'' produced by the second stage. Lemma 4.3 bounds the expected size of E'' by $O(OPTn^{1/3} \ln n)$. Lemma 4.4 shows that E'' settles a given thin edge with probability at least $1 - 1/n$. Consequently, the expected number of unsettled thin edges is at most $|E|/n \leq n - 1 \leq OPT$, and they can be added to the solution without affecting the approximation ratio. This completes the proof of Theorem 4.1. □

It remains to prove the lemmas that were used in the proof of Theorem 4.1.

Recall that $\bar{s}$ and $\bar{t}$ are used to denote the source and the sink the local layered graph of an edge (s, t) as in Definition 3.2.

Lemma 4.2. *Given a feasible solution $\{\mathring{x}_e\} \cup \{\mathring{f}^{s,t}_{e_i}\}$ of LP-U of cost LP, Algorithm 4 on lines 2–5 computes a vector $\{\hat{x}_e\}$ of cost at most $20LP$ (i.e., satisfying $\sum_e \hat{x}_e \leq 20LP$) such that all $\hat{x}_e$ are integer multiples of $n^{-2/3}$. Moreover, for every thin edge (s, t) and cut $(\bar{S}, \bar{T})$ in the local layered graph $\bar{G}^{s,t}$ with $\bar{s}, s_1 \in \bar{S}$ and $t_2, \bar{t} \in \bar{T}$, vector $\{\hat{x}_e\}$ satisfies*

$$\sum_{(u,v) \in E^{s,t} : (u_i, v_{i+1}) \in \bar{S} \times \bar{T}} \hat{x}_{(u,v)} \geq 1. \tag{6}$$

This stage of the algorithm succeeds with probability $1 - \exp(-cn^{2/3})$ *for some constant* $c > 0$.

Proof. For every edge e, we independently sample a Poisson random variable P_e with mean $\lambda_e = 6n^{2/3}\mathring{x}_e$, and set $\hat{x}_e = P_e n^{-2/3}$. Since the support of the Poisson distribution is on nonnegative integers, all $\hat{x}_e$ are integer multiples of $n^{-2/3}$. We need to verify that $\hat{x}_e$ satisfies (6) and that its cost is bounded by $20LP$.

Fix a thin edge (s,t) and a cut $(\bar{S},\bar{T})$ in $\bar{G}^{s,t}$ with $\bar{s}, s_1 \in S$ and $t_2, \bar{t} \in T$. Let $A = \{(u,v) \in E^{s,t} : (u_i, v_{i+1}) \in \bar{S} \times \bar{T}\}$. We will show that it is an antispanner for (s,t). For every path $p = s \to u \to v \to t$ of length 3 in $G^{s,t}$, one of the edges on the path $\bar{s} \to u_1 \to v_2 \to \bar{t}$ crosses the cut $(\bar{S},\bar{T})$ and, consequently, one of the edges of p belongs to A. Similarly, for every path $p = s \to u \to t$ of length 2 (respectively, path $p = s \to t$ of length 1) one of the edges on the path $\bar{s} \to u_1 \to t_2 \to \bar{t}$ (respectively, path $\bar{s} \to s_1 \to t_2 \to \bar{t}$) crosses the cut $(\bar{S},\bar{T})$, and one of the edges of p belongs to A. Therefore, A is an antispanner for (s,t). By Claim 3.2, if $\{\mathring{x}_e\} \cup \{\mathring{f}^{s,t}_{e_i}\}$ is a feasible solution to LP-U then $\{\mathring{x}_e\}$ satisfies the antispanner constraints for LP-A′, given in (2). That is, $\sum_{e\in A} \mathring{x}_e \geq 1$.

Next, we bound $\sum_{e\in A} \hat{x}_e = n^{-2/3} \sum_{e\in A} P_e$. The sum $\sum_{e\in A} P_e$ is distributed as a Poisson random variable with mean $\lambda_A = \sum_{e\in A} \lambda_e \geq 6n^{2/3}$. By Lemma A.1 in the Appendix,

$$\Pr[\sum_{e\in A} \hat{x}_e < 1] = \Pr[\sum_{e\in A} P_e < n^{2/3}] \leq \Pr[\sum_{e\in A} P_e \leq 6/e \cdot n^{2/3}] \leq \exp(-6n^{2/3}/4).$$

Since (s,t) is a thin edge, $|\bar{V}^{s,t} \setminus \{\bar{s},\bar{t}\}| \leq 2n^{2/3}$, and the number of cuts $(\bar{S},\bar{T})$ in $\bar{V}^{s,t}$ separating $\bar{s}$ and $\bar{t}$ is at most $2^{2n^{2/3}} = \exp(\ln 4 \cdot n^{2/3})$. Hence, by a union bound, (6) holds for all such cuts simultaneously with probability at least $1 - e^{-cn^{2/3}}$, where $c = 6/4 - \ln 4 > 0$. By a union bound, the previous sentence is true for all thin edges (s,t) simultaneously with a constant if c is set to $c/2 = \frac{1}{2}(6/4 - \ln 4)$.

Finally, observe that the cost of $\{\hat{x}_e\}$ is $n^{-2/3} \times \sum_{e\in E} P_e$. The sum $\sum_{e\in E} P_e$ is a Poisson random variable with mean $6n^{2/3} \sum \mathring{x}_e = 6n^{2/3}LP$. By Lemma A.1,

$$\Pr[\sum_{e\in E} P_e \geq 20n^{2/3}LP] \leq \Pr[\sum_{e\in E} P_e \geq 6 \cdot e \cdot n^{2/3}LP] \leq \exp(-6n^{2/3}LP) \leq \exp(-6n^{2/3}).$$

Thus, the probability that the cost of $\{\hat{x}_e\}$ exceeds $20LP$ is exponentially small. □

Lemma 4.3 (Analog of Lemma 4.1 in [DK11])**.** $\mathbb{E}[|E''|] = O(OPT\, n^{1/3} \ln n)$.

Proof. By a union bound, the probability that an edge e belongs to E'' is at most $2\hat{x}_e \alpha n^{1/3} \ln n$. Therefore, since α is a constant,

$$\mathbb{E}[|E''|] \leq \sum_{e\in E} 2\hat{x}_e \alpha\, n^{1/3} \ln n = O(OPT\, n^{1/3} \ln n).$$

□

Lemma 4.4 (Analog of Lemma 4.2 in [DK11])**.** *If* (s,t) *is a thin edge for which condition (6) holds, then* E'' *settles* (s,t) *with probability at least* $1 - 1/n$.

Proof. Fix a thin edge (s,t). Let

$$\bar{E}'' = \{e_i : \ e \in E'' \text{ and } e_i \in \bar{E}^{s,t}\} \cup \{(\bar{s}, s_1), (t_2, \bar{t})\}$$

be the set of copies of the edges in E'' in the local layered graph $\bar{G}^{s,t}$. We show that, with probability at least $1 - 1/n$, there is a path from $\bar{s}$ to $\bar{t}$ in $(\bar{V}^{s,t}, \bar{E}'')$ and, consequently, the edge (s,t) is settled.

Let $\{\hat{f}^{s,t}_{e_i}\}$ be the maximum flow from $\bar{s}$ to $\bar{t}$ in the graph $\bar{G}^{s,t}$ with capacities $\hat{x}_{e_i}$ set to $\hat{x}_{(u,v)}$ on edges $e_i = (u_i, v_{i+1})$ for $u \neq v$, infinite capacities ($\hat{x}_{e_i} = \infty$) on edges $e_i \in \{(\bar{s}, s_1), (t_2, \bar{t})\}$ and zero capacities ($\hat{x}_{e_i} = 0$) on edges $e_i = (u_i, u_{i+1})$ for $e_i \notin \{(\bar{s}, s_1), (t_2, \bar{t})\}$. Note that this flow may be different from the flow $\{\mathring{f}^{s,t}_{e_i}\}$ obtained by LP-U. By (6), the capacity of the minimum cut between $\bar{s}$ and $\bar{t}$ is at least 1. Thus, the value of the flow $\{\hat{f}^{s,t}_{e_i}\}$ is at least 1.

In the simplest case, the flow is routed along $n^{2/3}$ disjoint paths of capacity $n^{-2/3}$ each. The probability that a given path $\bar{s} \to u_1 \to v_2 \to \bar{t}$ belongs to $(\bar{V}^{s,t}, \bar{E}'')$ is at least $\Pr[r_u \leq \alpha\, n^{-1/3} \ln n \text{ and } r_v \leq \alpha\, n^{-1/3} \ln n] \geq (\alpha\, n^{-1/3} \ln n)^2$. The probability that at least one path belongs to $\bar{E}''$ is $1-(1-\alpha^2\, n^{-2/3} \ln^2 n)^{n^{2/3}} > 1-1/n$. In the general case, however, we need a more involved analysis.

To analyze the general case, we partition the set $V^{s,t}$ into two disjoint sets S and T such that at least $1/4$ units of flow $\{\hat{f}^{s,t}_{e_i}\}$ are routed along the paths $\bar{s} \to u_1 \to v_2 \to \bar{t}$, where $u \in S$ and $v \in T$. To see that such a partition exists, randomly add every vertex in $V^{s,t} \setminus \{s,t\}$ to S or T with probability $1/2$. Add s to S and t to T. Then for every path $\bar{s} \to u_1 \to v_2 \to \bar{t}$ (where $u \neq v$), $\Pr[u \in S \text{ and } v \in T] \geq 1/4$, so the expected contribution of every path to the new flow is at least $1/4$ of the original flow over the path. Because the total new flow from $\bar{s}$ to $\bar{t}$ can be represented as a sum of flows over such paths, the expected flow routed from $\bar{s}$ to $\bar{t}$ through S and T (as described above) is at least $1/4$. That is, for at least one partition (S,T) the flow is at least $1/4$. Fix this partition.

Let $\{f_{e_i}\}$ be the maximum flow in $\bar{G}^{s,t}$ (with the same capacities as above) routed from $\bar{s}$ to $\bar{t}$ through S and T, such that all f_{e_i} are multiples of $n^{-2/3}$. Such a flow exists because all capacities are multiples of $n^{-2/3}$. Observe that $(u,v) \in \bar{E}''$ if $\min(r_u, r_v) \leq \hat{x}_{(u,v)} \alpha\, n^{1/3} \ln n$ and, consequently, also if $\min(r_u, r_v) \leq f_{(u,v)} \alpha\, n^{1/3} \ln n$, since $f_{(u,v)} \leq \hat{x}_{(u,v)}$.

Consider the following two cases:

1. $f_{(\bar{s},u_1)} \geq n^{-1/3}$ for some vertex $u \in S$.
2. $f_{(\bar{s},u_1)} < n^{-1/3}$ for all vertices $u \in S$.

Case 1. Fix a vertex $u \in S$ for which $f_{(\bar{s},u_1)} \geq n^{-1/3}$. We will show that with probability at least $1 - 1/n$ there is a path from $\bar{s}$ to $\bar{t}$ via u_1 in $\bar{G}^{s,t}$. The edge $(\bar{s}, u_1)$ always belongs to $\bar{E}''$ because $\alpha \hat{x}_{(\bar{s},u_1)} n^{1/3} \ln n \geq \alpha f_{(\bar{s},u_1)} n^{1/3} \ln n > 1 \geq r_u$. Consider an arbitrary path $u_1 \to v_2 \to \bar{t}$. Note that $f_{(v_2,\bar{t})} \geq f_{(u_1,v_2)}$, since all flow from u_1 to v_2 must be routed to $\bar{t}$ along the edge $(v_2, \bar{t})$. Thus, if $r_v \leq \alpha f_{(u_1,v_2)}\, n^{1/3} \ln n$, then $(u_1, v_2) \in \bar{E}''$ and $(v_2, \bar{t}) \in \bar{E}''$. Therefore, if there is no path from u_1 to $\bar{t}$ in $\bar{E}''$,

then $r_v > \alpha f_{(u_1,v_2)}\, n^{1/3} \ln n$ for all $v \in T$. This happens with probability at most

$$\begin{aligned}
\prod_{v\in T} \min(1 - \alpha f_{(u_1,v_2)}\, n^{1/3} \ln n, 0) &\leq \prod_{v\in T} \exp\left(-\alpha f_{(u_1,v_2)}\, n^{1/3} \ln n\right) \\
&= \exp\left(-\left(\sum_{v\in T} f_{(u_1,v_2)}\right) \alpha n^{1/3} \ln n\right) \\
&\leq \exp\left(-f_{(\bar{s},u_1)}\, \alpha n^{1/3} \ln n\right) \\
&\leq \exp(-\ln n) = \frac{1}{n}.
\end{aligned}$$

Therefore, with probability at least $1 - 1/n$, there is a path from $\bar{s}$ to $\bar{t}$ in $\bar{G}^{s,t}$ and, consequently, the edge (s,t) is settled.

Case 2. For every $u \in S$, define a random variable F_{u_1}:

$$F_{u_1} = \sum_{v\in T : r_u \leq \alpha f_{(u_1,v_2)} n^{1/3} \ln n} f_{(u_1,v_2)}.$$

This random variable gives a lower bound on the amount of flow that can be routed along the edges $\bar{E}''$ from the source $\bar{s}$ to the set of copies of nodes in T through the vertex u_1. (Recall that $\bar{E}''$ is a random set.)

Claim 4.5. $\Pr_{r_u :\, u\in S}\left[\sum_{u\in S} F_{u_1} \geq \dfrac{\alpha n^{-1/3} \ln n}{8}\right] \geq 1 - \dfrac{1}{2n}.$

Proof. The value of F_{u_1} depends only on r_u, and hence all random variables F_{u_1} are independent. If $f_{(u_1,v_2)} > 0$ then $f_{(u_1,v_2)} \geq n^{-2/3}$ because $f_{(u_1,v_2)}$ is a multiple of $n^{-2/3}$. Therefore, for all nodes $u \in S$ and $v \in T$ with positive flow $f_{(u_1,v_2)}$,

$$\Pr_{r_u}\left[r_u \leq \alpha f_{(u_1,v_2)} n^{1/3} \ln n\right] = \min(\alpha f_{(u_1,v_2)} n^{1/3} \ln n, 1) \geq \alpha n^{-2/3} n^{1/3} \ln n \geq \alpha n^{-1/3} \ln n.$$

This implies that for all nodes $u \in S$ and $v \in T$,

$$f_{(u_1,v_2)} \cdot \Pr_{r_u}\left[r_u \leq \alpha f_{(u_1,v_2)} n^{1/3} \ln n\right] \geq f_{(u_1,v_2)} \cdot \alpha n^{-1/3} \ln n.$$

Therefore,

$$\begin{aligned}
\mathbb{E}[\sum_{u\in S} F_{u_1}] &= \sum_{u\in S}\left(\sum_{v\in T} f_{(u_1,v_2)} \Pr_{r_u}\left[r_u \leq \alpha f_{(u_1,v_2)} n^{1/3} \ln n\right]\right) \\
&\geq \sum_{u\in S}\Big(\sum_{v\in T} f_{(u_1,v_2)}\Big)\alpha n^{-1/3} \ln n = \Big(\sum_{u\in S} f_{(\bar{s},u_1)}\Big)\alpha n^{-1/3} \ln n \geq \frac{\alpha}{4} n^{-1/3} \ln n.
\end{aligned}$$

Now we use the assumption that $f_{(\bar{s},u_1)} \leq n^{-1/3}$ for all $u \in S$. By flow conservation, it implies that all F_{u_1} are bounded from above by $n^{-1/3}$. By the Hoeffding

inequality [3] applied with $\epsilon = 1/2$ and $c = n^{-1/3}$,

$$\begin{aligned}\Pr_{r_u}\left[\sum_{u\in S} F_{u_1} \geq \frac{\alpha n^{-1/3}\ln n}{8}\right] &= 1 - \Pr_{r_u}\left[\sum_{u\in S} F_{u_1} < \frac{1}{2}\mathbb{E}[\sum_{u\in S} F_{u_1}]\right] \\ &\geq 1 - \exp\left(-\frac{\alpha \ln n}{32}\right) \geq 1 - \frac{1}{2n}.\end{aligned}$$

□

Next, we condition on the event that $\sum_{u\in S} F_{u_1} \geq \alpha n^{-1/3}\ln n/8$, and bound the conditional probability that there exists a path from $\bar{s}$ to $\bar{t}$.

Claim 4.6. *For any fixed $\{r_u\}_{u\in S}$, such that $\sum_{u\in S} F_{u_1} \geq \alpha n^{-1/3}\ln n/8$, we have*

$$\Pr_{r_v:\, v\in T}\left[\textit{there is no path } \bar{s} \to u_1 \to v_2 \to \bar{t} \textit{ in } E''\right] \leq \frac{1}{2n}.$$

Proof. For every $v \in T$, let

$$F_{v_2} = \sum_{u_1 : r_u \leq \alpha f_{(u_1,v_2)} n^{1/3}\ln n} f_{(u_1,v_2)}.$$

If for some $\tilde{v} \in T$ we have $F_{\tilde{v}_2} > 0$, then for some $\tilde{u} \in S$, $r_{\tilde{u}} \leq \alpha f_{(\tilde{u}_1,\tilde{v}_2)} n^{1/3}\ln n$, $r_{\tilde{u}} \leq \alpha f_{(\bar{s},\tilde{u}_1)} n^{1/3}\ln n$ and, hence, the path $\bar{s} \to \tilde{u}_1 \to \tilde{v}_2$ belongs to E''. Also,

$$f_{(\tilde{v}_2,\bar{t})} = \sum_{u\in S} f_{(u_1,\tilde{v}_2)} \geq F_{v_2}.$$

Now for a fixed $\{r_u\}_{u\in S}$ and a vertex $\tilde{v} \in T$, we bound the probability that $(\tilde{v}_2, \bar{t}) \in E''$ from below by

$$\Pr_{r_{\tilde{v}}}\left[r_{\tilde{v}} \leq \alpha f_{(\tilde{v}_2,\bar{t})} n^{1/3}\ln n\right] \geq \min(\alpha f_{(\tilde{v}_2,\bar{t})} n^{1/3}\ln n, 1) \geq \min(\alpha F_{\tilde{v}_2} n^{1/3}\ln n, 1).$$

Note that we have a lower bound on the sum of F_{v_2}'s:

$$\sum_{v\in T} F_{v_2} = \sum_{u\in S} F_{u_1} \geq \frac{\alpha n^{-1/3}\ln n}{8}.$$

Thus, we can use the same argument as in Claim 2.3 to get a lower bound on the overall probability:

$$\begin{aligned}\Pr_{r_v}\left[(v_2,\bar{t}) \notin E'', \text{ for all } v \in T \text{ with } F_{v_2} > 0\right] &\leq \exp(-\sum_{v\in T} \alpha F_{v_2} n^{1/3}\ln n) \\ &\leq \exp(-\alpha^2 \ln^2 n/8) < \frac{1}{2n}.\end{aligned}$$

□

[3] Here we use the following variant of the Hoeffding's inequality. Let $X_1, \ldots, X_n$ be independent random variables taking values in $[0, c]$. Let $S_n = \sum X_i$, let $\mu = \mathbb{E}[S_n]$. Then, for every positive ε,

$$\Pr[S_n \leq (1-\varepsilon)\mu] \leq e^{-\frac{1}{2}\varepsilon^2\mu/c}.$$

For reference see, e.g., [HMRR98] Theorem 2.3(c) on page 200.

By Claims 4.5 and 4.6, the probability that there exists a path $\bar{s} \to u_1 \to v_2 \to \bar{t}$ is at least $(1 - 1/(2n))^2 > 1 - 1/n$. □

5. An $O(n^{2/3+\epsilon})$-Approximation for Directed Steiner Forest

Let us first recall the Directed Steiner Forest (DSF) problem. Given a directed graph $G = (V, E)$, a cost function $c : E \to \mathbb{R}^+$ and a set $D \subseteq V \times V$ of ordered pairs, the goal is to find a min-cost subgraph H of G that contains a path from s to t for every $(s, t) \in D$. In contrast to spanners, there is no restriction on the paths used to connect pairs, but the objective to be optimized depends on arbitrary edge costs.

Theorem 5.1. *For any fixed $\epsilon > 0$, there is a polynomial time randomized algorithm for* Directed Steiner Forest *with expected approximation ratio $O(n^{2/3+\epsilon})$.*

Our algorithm for DSF builds on the algorithm of Feldman, Kortsarz and Nutov [FKN09] for the problem. We describe their algorithm and most of their analysis, using notation compatible with previous sections of this paper, and show where we make our improvement. As mentioned in [FKN09], one can assume without loss of generality that $D \subseteq S \times T$ for two *disjoint* subsets S and T of V and that the costs are metric.

Let τ denote our guess for the optimal value of OPT. We start from $\tau = 1$ and repeatedly double our guess each time we find it is too small. Thus, it suffices to give the approximation guarantee for the iteration when $OPT \leq \tau \leq 2 \cdot OPT$. The algorithm has two parameters: β and ℓ. We set $\beta = n^{1/3}$ and $\ell = \tau/n^{2/3}$ below.

Let us adapt some terminology from the previous sections to this new setting.

Definition 5.1 (Thick and thin pairs)**.** *For a pair $(s, t) \in D$, let $G^{s,t} = (V^{s,t}, E^{s,t})$ be the subgraph of G induced by the vertices on paths from s to t of cost at most ℓ. A pair $(s, t) \in D$ is* thick *if $|V^{s,t}| \geq n/\beta$ and it is* thin *otherwise.*

Definition 5.2. *A set $E' \subseteq E$* settles *a pair $(s, t) \in D$ if the subgraph (V, E') contains a path from s to t.*

The high-level structure of the algorithm is the same as for the spanner problem. We will describe how to find in polynomial time two sets $E', E'' \subseteq E$,such that E' settles all the thick pairs and E'' settles all the thin pairs.

The thick pairs can be settled by random sampling, just as in Section 2.1. For $p = O((\log n)/(n/\beta))$, if each vertex is selected with probability p to lie in a set R, then for every $(s, t) \in D$, $R \cap V^{s,t} \neq \emptyset$ with high probability. Let the set E' be constructed by adding, for each $u \in R, s \in S, t \in T$, the edges of a path from s to u of cost at most ℓ if one exists and the edges of a path from u to t of cost at most ℓ if one exists. The expected number of thick pairs still not settled is at most $|D|/n^2 \leq 1$. Thus, we can add the edges of a minimum-cost path from s to t for any unsettled thick pair (s, t) and still have that the expected cost of E' be $O(n \cdot pn \cdot \ell + \tau) = \tilde{O}(n\ell\beta + \tau) = \tilde{O}(n^{2/3}\tau)$, where we use τ as an upper bound on the cost of a minimum-cost (s, t)-path.

$$\text{Minimize} \sum_{e \in E} c(e) \cdot x_e \text{ subject to:} \tag{7}$$

$$\sum_{(s,t) \in D} y_{s,t} \geq |D|/2$$

$$\sum_{\Pi(s,t) \ni P \ni e} f_P \leq x_e \qquad \forall (s,t) \in D, e \in E$$

$$\sum_{P \in \Pi(s,t)} f_P = y_{s,t} \qquad \forall (s,t) \in D$$

$$0 \leq y_{s,t}, f_P, x_e \leq 1 \qquad \forall (s,t) \in D, P \in \Pi, e \in E$$

Figure 3: Linear program LP-DSF for the case $|D - C| > |D|/2$

We remove the settled thick pairs from D, so that it only consists of the unsettled thin pairs. Next, we construct an edge set E'' that settles all the thin pairs. Define the *density* of a subset of E to be the ratio between the total cost of the subset and the number of pairs in D settled by it. We show how to efficiently construct a subset K with expected density $O(n^{2/3+\epsilon}) \cdot \tau/|D|$. This allows us to compute the set E'': starting from $|D|$ unsettled thin pairs and $E'' = \emptyset$, find K of expected density $O(n^{2/3+\epsilon}) \cdot \tau/|D|$, add the edges in K to E'', remove the settled pairs from D, and repeat. As shown in Theorem 2.1 of [FKN09], this greedy procedure produces a subset E'' of expected cost $O(n^{2/3+\epsilon}) \cdot \tau$ that settles all the thin pairs, completing the proof of Theorem 5.1.

The edge set K is produced by constructing two sets K_1 and K_2 and letting K be the set of smaller density. We guarantee that one of K_1 and K_2 has expected density $O(n^{2/3+\epsilon}) \cdot \tau/|D|$. Whether the guarantee is provided for K_1 or K_2 depends upon which one of the two cases below holds. Suppose H is an optimal solution with cost τ (we ignore the factor of 2 for simplicity). Let C be the set of pairs $(s,t) \in D$ for which the minimum cost of an (s,t)-path in H is at least ℓ; that is, these are the costly pairs to settle. The two cases are: $|C| \geq |D|/2$ and $|C| < |D|/2$.

Case 1: $|C| \geq |D|/2$. This case relies on a result of Chekuri, Even, Gupta and Segev [CEGS11]. Define a *junction tree* to be the union of an ingoing tree and an outgoing tree (not necessarily disjoint) rooted at the same vertex. Chekuri *et al.* [CEGS11] show an $O(n^\epsilon)$-approximation for the minimum density junction tree of a graph. Fortunately, there exists a junction tree of density at most $\tau^2/(|C|\ell)$. To see why, take the paths in H connecting the pairs in C. The sum of the costs of all such paths is at least $|C|\ell$. If we denote the maximum number of these paths that any edge belongs to as μ, then the sum of the costs of the paths is at most $\mu \cdot \tau$ and thus there exists an edge, which belongs to $\mu \geq |C|\ell/\tau$ paths. Therefore, there must be a junction tree K_1 which contains this edge and connects at least $|C|\ell/\tau$ pairs in D. K_1 has density at most $\tau/(|C|\ell/\tau) = \tau^2/(|C|\ell)$. Thus, when $|C| \geq |D|/2$, the algorithm of [CEGS11] (deterministically) returns a junction tree of density $O(n^\epsilon \cdot \tau/\ell \cdot \tau/|D|) = O(n^{2/3+\epsilon}) \cdot \tau/|D|$.

Case 2: $|D - C| > |D|/2$. In this case, we attempt to find a subgraph that connects many pairs of D using low-cost edges. Consider the problem of connecting at least $|D|/2$ pairs from D using paths of cost at most ℓ while minimizing the total cost of the edges. For $(s,t) \in D$, let $\Pi(s,t)$ be the set of (s,t)-paths of cost at most ℓ, and let $\Pi = \bigcup_{(s,t)\in D} \Pi(s,t)$. We can formulate an LP relaxation for this problem, LP-DSF, shown in Figure 3, which closely resembles the LP used by [DK11] for Directed k-Spanner. Each edge e has a capacity x_e, each path $P \in \Pi$ carries f_P units of flow, and $y_{s,t}$ is the total flow through all paths from s to t. Also, the total flow through all paths in Π should be at least $|D|/2$. It is clear that LP-DSF is a relaxation of the problem of connecting at least $|D|/2$ pairs in D while minimizing the cost of the edges. Feldman, Kortsarz and Nutov [FKN09] show that in polynomial time, we can find a solution $\{\hat{x}_e\} \cup \{\hat{y}_{s,t}\}$ such that $\sum_{e\in E} c(e) \cdot \hat{x}_e$ is within $(1+\epsilon)$ factor of OPT, the optimal solution to LP-DSF, for any fixed $\epsilon > 0$.

Our improvement comes in the analysis of the rounding algorithm for LP-DSF. Suppose $\{\hat{x}_e\} \cup \{\hat{y}_{s,t}\}$ is a feasible solution to LP-DSF. Let K_2 be the edge set obtained by selecting each edge in E with probability $\min((8n \ln n)/\beta \cdot x_e, 1)$.

Lemma 5.2. *With probability $\geq 1 - 1/n^2$, set K_2 settles every thin pair (s,t) with $\hat{y}_{s,t} \geq 1/4$.*

Proof. We reinterpret Definition 2.4 in terms of edge costs instead of lengths. More precisely, define a set $A \subseteq E$ to be an antispanner for a pair $(s,t) \in D$ if $(V, E \setminus A)$ contains no path from s to t of cost at most ℓ. By exactly the same argument as in Claim 2.5, the set of all minimal antispanners for thin pairs is of size at most $n^2(n/\beta)^{n/\beta}$.

For every thin pair $(s,t) \in D$ with $\hat{y}_{s,t} \geq 1/4$, if A is an antispanner for (s,t), then $\sum_{e\in A} \hat{x}_e \geq \sum_{P\in\Pi(s,t)} \hat{f}_P \geq 1/4$, where $\hat{f}_P$ is the value of the variable f_P in LP-DSF that corresponds to the solution $\{\hat{x}_e\} \cup \{\hat{y}_{s,t}\}$. So, the probability that K_2 is disjoint from A is at most $\exp(-(n \ln n)/\beta)$, by the same argument as in Claim 2.3. Thus, by the bound on the total number of antispanners of thin pairs from above, the union bound, and Claim 2.4, it follows that with high probability, K_2 settles every thin pair (s,t) with $\hat{y}_{s,t} \geq 1/4$. □

We add to K_2 a minimum-cost path between any pair (s,t) with $\hat{y}_{s,t} \geq 1/4$ that is still not settled. In expectation, the number of such pairs is $|D|/n^2 \leq 1$, so that the total expected cost[4] of K_2 is at most $(16n \ln n)/\beta \cdot \tau$. A simple argument shows that the number of pairs (s,t) in D for which $\hat{y}_{s,t} < 1/4$ is at most $2|D|/3$; assuming the opposite makes the total amount of flow between all pairs strictly less than $|D|/2$. So, the expected density of K_2 is at most:

$$\left(\frac{16n \ln n}{\beta} \cdot \tau\right) / \left(|D| - 2|D|/3\right) = \frac{48n \ln n}{\beta} \cdot \frac{\tau}{|D|} = \tilde{O}(n^{2/3}) \cdot \tau/|D|.$$

[4]This is where we save over [FKN09]. The cost of their comparable K_2 is $O(n^2/\beta^2 \cdot \tau)$.

As we said earlier, the set K is taken to be either K_1 or K_2, depending upon which has the smaller density. By the above, expected density of K is at most $O(n^{2/3+\epsilon}) \cdot \tau/|D|$. This concludes the proof of Theorem 5.1.

6. Conclusion

For general Directed k-Spanner, we obtained an approximation ratio of $\tilde{O}(\sqrt{n})$ and for Directed 3-Spanner with unit edge lengths we obtained an approximation ratio of $\tilde{O}(n^{1/3})$. The second bound almost matches the LP integrality gap of Dinitz and Krauthgamer [DK11]. It remains an interesting open question whether one can get an approximation ratio of $\tilde{O}(n^{1/3})$ for the general case.

All our algorithms are randomized and have an expected approximation factor. Our algorithms consist of multiple stages and the analysis of concentration of the cost of the solution can be done using standard concentration bounds for each stage separately in the same way as in the previous work (e.g. [BGJ+09, DK11, FKN09]), so we omit it to simplify presentation. It remains open whether these algorithms can be derandomized.

References

[ADD+93] Ingo Althöfer, Gautam Das, David Dobkin, Deborah Joseph, and José Soares. On sparse spanners of weighted graphs. *Discrete & Computational Geometry*, 9(1):81–100, 1993.

[AHL02] Noga Alon, Shlomo Hoory, and Nathan Linial. The moore bound for irregular graphs. *Graphs and Combinatorics*, 18:53–57, 2002.

[Awe85] Baruch Awerbuch. Communication-time trade-offs in network synchronization. In *PODC*, pages 272–276, 1985.

[BBM+11] Piotr Berman, Arnab Bhattacharyya, Konstantin Makarychev, Sofya Raskhodnikova, and Grigory Yaroslavtsev. Improved approximation for the directed spanner problem. In Luca Aceto, Monika Henzinger, and Jiri Sgall, editors, *ICALP (1)*, volume 6755 of *Lecture Notes in Computer Science*, pages 1–12. Springer, 2011.

[BGJ+09] Arnab Bhattacharyya, Elena Grigorescu, Kyomin Jung, Sofya Raskhodnikova, and David Woodruff. Transitive-closure spanners. In *SODA*, pages 932–941, 2009.

[BGJ+12] Arnab Bhattacharyya, Elena Grigorescu, Madhav Jha, Kyomin Jung, Sofya Raskhodnikova, and David Woodruff. Lower bounds for local monotonicity reconstruction from transitive-closure spanners. *SIAM J. Discrete Math.*, 26(2):618–646, 2012.

[BRR10] Piotr Berman, Sofya Raskhodnikova, and Ge Ruan. Finding sparser directed spanners. In Kamal Lodaya and Meena Mahajan, editors, *FSTTCS*, volume 8 of *LIPIcs*, pages 424–435. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2010.

[BS06] Surender Baswana and Sandeep Sen. Approximate distance oracles for unweighted graphs in expected $O(n^2)$ time. *ACM Trans. Algorithms*, 2(4):557–577, 2006.

[CEGS11] Chandra Chekuri, Guy Even, Anupam Gupta, and Danny Segev. Set connectivity problems in undirected graphs and the directed steiner network problem. *ACM Trans. Algorithms*, 7(2):18, 2011.

[CHK11] Moses Charikar, MohammadTaghi Hajiaghayi, and Howard J. Karloff. Improved approximation algorithms for label cover problems. *Algorithmica*, 61(1):190–206, 2011.

[Coh98] Edith Cohen. Fast algorithms for constructing t-spanners and paths with stretch t. *SIAM J. Comput.*, 28(1):210–236, 1998.

[Coh00] Edith Cohen. Polylog-time and near-linear work approximation scheme for undirected shortest paths. *JACM*, 47(1):132–166, 2000.

[Cow01] Lenore Cowen. Compact routing with minimum stretch. *J. Algorithms*, 38(1):170–183, 2001.

[CW04] Lenore Cowen and Christopher G. Wagner. Compact roundtrip routing in directed networks. *J. Algorithms*, 50(1):79–95, 2004.

[DK99] Yevgeniy Dodis and Sanjeev Khanna. Designing networks with bounded pairwise distance. In *STOC*, pages 750–759, 1999.

[DK11] Michael Dinitz and Robert Krauthgamer. Directed spanners via flow-based linear programs. In Lance Fortnow and Salil P. Vadhan, editors, *STOC*, pages 323–332. ACM, 2011.

[Elk01] M. Elkin. Computing almost shortest paths. In *PODC*, pages 53–62, 2001.

[EP00] Michael Elkin and David Peleg. Strong inapproximability of the basic k-spanner problem. In *ICALP*, pages 636–647, 2000.

[EP01] Michael Elkin and David Peleg. The client-server 2-spanner problem with applications to network design. In *SIROCCO*, pages 117–132, 2001.

[EP05] Michael Elkin and David Peleg. Approximating k-spanner problems for $k > 2$. *Theor. Comput. Sci.*, 337(1-3):249–277, 2005.

[EP07] Michael Elkin and David Peleg. The hardness of approximating spanner problems. *Theory Comput. Syst.*, 41(4):691–729, 2007.

[FKM+08] Joan Feigenbaum, Sampath Kannan, Andrew McGregor, Siddharth Suri, and Jian Zhang. Graph distances in the data-stream model. *SIAM J. Comput.*, 38(5):1709–1727, 2008.

[FKN09] Moran Feldman, Guy Kortsarz, and Zeev Nutov. Improved approximating algorithms for Directed Steiner Forest. In Claire Mathieu, editor, *SODA*, pages 922–931. SIAM, 2009.

[HMRR98] M. Habib, C. McDiarmid, J. Ramirez-Alfonsin, and B. Reed. *Probabilistic methods for algorithmic discrete mathematics*, volume 16. Springer Verlag, 1998.

[JR11] Madhav Jha and Sofya Raskhodnikova. Testing and reconstruction of lipschitz functions with applications to data privacy. In *FOCS*, pages 433–442. IEEE, 2011.

[Kor01] Guy Kortsarz. On the hardness of approximating spanners. *Algorithmica*, 30(3):432–450, 2001.

[KP94] G. Kortsarz and D. Peleg. Generating sparse 2-spanners. *J. Algorithms*, 17(2):222–236, 1994.

[KP98] Guy Kortsarz and David Peleg. Generating low-degree 2-spanners. *SIAM J. Comput.*, 27(5):1438–1456, 1998.

[PS89] David Peleg and Alejandro A. Schäffer. Graph spanners. *J. Graph Theory*, 13(1):99–116, 1989.

[PU89a] David Peleg and Jeffrey D. Ullman. An optimal synchronizer for the hypercube. *SIAM J. Comput.*, 18(4):740–747, 1989.

[PU89b] David Peleg and Eli Upfal. A trade-off between space and efficiency for routing tables. *JACM*, 36(3):510–530, 1989.

[Ras10] Sofya Raskhodnikova. Transitive-closure spanners: a survey. In Oded Goldreich, editor, *Property Testing*, volume LNCS 6390, pages 167–196. Springer, Heidelberg, 2010.

[RTZ08] Liam Roditty, Mikkel Thorup, and Uri Zwick. Roundtrip spanners and roundtrip routing in directed graphs. *ACM Trans. Algorithms*, 4(3):1–17, 2008.

[TZ01] Mikkel Thorup and Uri Zwick. Compact routing schemes. In *SPAA*, pages 1–10. ACM, 2001.

[TZ05] Mikkel Thorup and Uri Zwick. Approximate distance oracles. *JACM*, 52(1):1–24, 2005.

Appendix A. Concentration of Poisson Random Variables

We use the following standard concentration result on Poisson random variables, giving the proof for completeness.

Lemma Appendix A.1. *Let P be a Poisson random variable with parameter $\lambda \geq 1$. Then*

$$\Pr[P < \lfloor \lambda/e \rfloor] \leq e^{-\lambda/4};$$
$$\Pr[P > e\lambda] \leq e^{-\lambda}.$$

Proof. Let $T = \lfloor \lambda/e \rfloor$. Then

$$\Pr[P < \lambda/e] = \sum_{t=0}^{T-1} \frac{\lambda^t e^{-\lambda}}{t!}.$$

The terms in the sum increase exponentially: $\frac{\lambda^{t+1}}{(t+1)!} \Big/ \frac{\lambda^t}{(t)!} = \frac{\lambda}{t+1} \geq e$. Hence,

$$\begin{aligned} \Pr[P < \lambda/e] &\leq \frac{\lambda^T e^{-\lambda}}{T!} \sum_{t=0}^{\infty} e^{-t} \leq \frac{\lambda^T e^{-\lambda}}{\sqrt{2\pi}(T/e)^T} \times 2 \leq e^{-(\lambda-T)} \Big(\frac{\lambda}{T}\Big)^T \\ &\leq e^{-(\lambda-T)} \Big(\frac{\lambda}{\lambda/e}\Big)^{\lambda/e} \leq e^{-\lambda(1-2/e)} \leq e^{-\lambda/4}. \end{aligned}$$

To estimate $T!$ we use Stirling's approximation.

Similarly, let $T' = \lceil e\lambda \rceil$. Then

$$\Pr[P > e\lambda] = \sum_{t=T'}^{\infty} \frac{\lambda^t e^{-\lambda}}{t!}.$$

The terms in the sum decrease exponentially: $\frac{\lambda^{t+1}}{(t+1)!} \Big/ \frac{\lambda^t}{(t)!} = \frac{\lambda}{t+1} \leq 1/e$. Hence,

$$\Pr[P > e\lambda] \leq \frac{2\lambda^{T'} e^{-\lambda}}{T'!} \leq \frac{2\lambda^{T'} e^{-\lambda}}{\sqrt{2\pi}(T'/e)^{T'}} \leq e^{-\lambda} \Big(\frac{e\lambda}{T'}\Big)^{T'} \leq e^{-\lambda}.$$

□