Designing Networks with Bounded Pairwise Distance

Yevgeniy Dodis*
Massachusetts Institute of Technology
yevgen@theory.lcs.mit.edu

Sanjeev Khanna
Bell Labs
sanjeev@research.bell-labs.com

Abstract

We study the following network design problem: Given a communication network, find a minimum cost subset of missing links such that adding these links to the network makes every pair of points within distance at most d from each other. The problem has been studied earlier [17] under the assumption that all link costs as well as link lengths are identical, and was shown to be $\Omega(\log n)$-hard for every $d \geq 4$.

We present a novel linear programming based approach to obtain an $O(\log n \log d)$ approximation algorithm for the case of uniform link lengths and costs. We also extend the $\Omega(\log n)$ hardness to $d \in \{2, 3\}$. On the other hand, if link costs can vary, we show that the problem is $\Omega(2^{\log^{1-\epsilon} n})$-hard for $d \geq 3$. This version of our problem can be viewed as a special case of the *minimum cost d-spanner* problem and thus our hardness result applies there as well. For $d = 2$, however, we show that the problem continues to be $O(\log n)$ approximable by giving an $O(\log n)$-approximation to the more general minimum cost 2-spanner problem. An $\Omega(2^{\log^{1-\epsilon} n})$-hardness result also holds when all link costs are identical but link lengths may vary (applies even when all lengths are 1 or 2). Our reduction from the *label cover* problem [3] also applies to another well-studied network design problem. We show that the *directed generalized steiner network* problem [6] is $\Omega(2^{\log^{1-\epsilon} n})$-hard, significantly improving upon the $\Omega(\log n)$ hardness known prior to our work. We also present $O(n \log d)$ approximation algorithm for our problem under arbitrary link costs and polynomially bounded link lengths. Same result holds for the minimum cost d-spanner problem.

Finally, all our positive results extend to the case where each pair (u, v) of nodes has a distinct distance requirement, say $d(u, v)$. The approximation guarantees above hold provided d is replaced by $\max_{u,v} d(u, v)$. All our algorithmic as well as hardness results hold for both undirected and directed versions of the problem.

*Supported by NSF Graduate Fellowship

1 Introduction

This paper studies the following basic network design problem. We are given a communication network and a set of additional communication links that can be added to the network. The goal is to find a minimum cost subset of links to be added such that every pair of points in the network is connected by a path of length at most d. A natural special case is when the initial network is empty; the goal then is to design a minimum cost network with bounded pairwise distance. Specifically,

PROBLEM: Mincost Distance-d

INSTANCE: A graph $G = (V, E)$ with cost function $c : \overline{E} \mapsto \mathbb{R}^+$, and length function $l : (E \cup \overline{E}) \mapsto \mathbb{Z}^+$, where $\overline{E} = \{(u, v) \mid (u, v) \notin E\}$.

GOAL: Find a minimum cost set $E' \subseteq \overline{E}$ of edges such that the distance between every pair of vertices in the graph $G' = (V, E \cup E')$ is at most d.

The graph G here represents the initial network and the set $\overline{E}$ represents the set of additional links that can be installed. Typically, the cost of an edge represents the installation cost of the corresponding link while the length represents the delay across the link. We will study the problem for both undirected and directed networks. Placing restrictions on the cost and length functions gives rise to three natural variations of the problem[1]:

- Diameter-d: unit costs, unit lengths.
- Mincost Diameter-d: arbitrary costs, unit lengths.
- Distance-d: unit costs, arbitrary lengths.

The above variations not only model network design problems with varied structure, but also capture optimization problems in other domains. For instance, the most basic version, namely the Diameter-d problem arises in the context of airline scheduling [17]. Even this basic variant is known to be NP-hard [17]. However, not much seems to be known about the complexity of approximately solving these problems. The goal of this paper is to study the approximability of these basic network design problems.

Related Work: Substantial work has been done when the design of the communication network is restricted

[1] In this paper, we use the word diameter to emphasize that the underlying graph has unit lengths on all edges.

Problem	Approximation Ratio	Hardness Factor
Diameter-d (unit cost, unit length)	$O(\log n \log d)$	$\Omega(\log n)$, $d \in \{2,3\}$ (from Set Cover)
Mincost Diameter-2 (arbitrary costs, unit length)	$O(\log n)$	$\Omega(\log n)$ (from Set Cover)
Mincost Diameter-d, $d \geq 3$ (arbitrary costs, unit length)	$O(n \log d)$	$\Omega(2^{\log^{1-\epsilon} n})$ (from Label Cover)
Distance-d (unit costs, polybounded lengths)	$O(n \log d)$	$\Omega(2^{\log^{1-\epsilon} n})$, lengths $\in \{1,2\}$ (from Label Cover)
Mincost Distance-d (arbitrary costs, polybounded lengths)	$O(n \log d)$	$\Omega(2^{\log^{1-\epsilon} n})$ (from Label Cover)

Figure 1: *Summary of Our Results*

to be a tree structure. A rather well-studied problem is the *shallow-light tree* problem where the goal is to find a spanning tree such that it has a bounded pairwise distance (shallow) and small cost (light). An (α, β)-approximation for shallow-light trees relaxes the distance bound by a factor of α and the cost bound by a factor of β. Awerbuch *et al.* [5] gave an $(O(1), O(1))$-approximation algorithm when the cost and the length functions are identical. Marathe *et al.* [18] extended this to an $(O(\log n), O(\log n))$-approximation when the cost and the distance functions are unrelated. Recently, Kortsarz and Peleg [15] studied the case when all distances are unit but costs are arbitrary and the goal is to find a minimum cost steiner tree of diameter d. They obtained an $O(d \log n)$-approximation when d is constant and an $O(n^\epsilon)$-approximation (for any $\epsilon > 0$) for general d. Another representative problem is the so-called *light, approximate shortest path trees* where the goal is to find a tree of small cost that closely approximates the shortest distances from a given single source [20, 11]. A more closely related line of research is the extensively studied area of graph spanners [1, 7, 12, 13, 14]:

PROBLEM: Mincost d-Spanner

INSTANCE: A graph $G = (V, E)$ with cost function $c : E \mapsto \mathbb{R}^+$.

GOAL: Find a minimum cost set $E' \subseteq E$ of edges such that every pair of vertices is at most a factor d further apart in $G' = (V, E')$ than it was in G.

Any feasible solution to this problem is called a *d-spanner* of G. We will show that there is a close relation between this problem and the Mincost Diameter-d problem. When all costs are the same, we refer to the problem as d-Spanner. Kortsarz and Peleg gave an $O(\log n)$-approximation for the 2-Spanner problem, and Kortsarz [12] recently showed a matching $\Omega(\log n)$-hardness result. Not much seems to be known about the approximability of the Mincost d-Spanner problem thus far.

Our Results: All our results hold for both undirected and directed networks. Our algorithmic results also extend to the case where each pair (u, v) of nodes has a distinct distance requirement, say $d(u, v)$. The approximation guarantees below hold provided d is replaced by $\max_{u,v} d(u, v)$. In this abstract, however, we restrict our attention to undirected graphs with uniform distance requirement d, leaving the details of above extensions to the full version.

Our first result is an $O(\log n \log d)$-approximation algorithm for the Diameter-d problem. Our algorithm is based on a novel linear programming formulation whose solution implicitly specifies an instance of the Hitting Set problem and a fractional solution for the instance. Randomized rounding as well as a greedy approach can then be used to make the solution integral. We also show that the problem is $\Omega(\log n)$-hard to approximate even for $d \in \{2, 3\}$, complementing the $\Omega(\log n)$-hardness result of Li *et al.* [17] for any $d \geq 4$.

Our next result is that when either non-uniform costs or non-uniform lengths are allowed on edges, the problem at once becomes very hard to approximate. Specifically, we show that for any $d \geq 3$, Mincost Diameter-d is hard to approximate within $O(2^{\log^{1-\epsilon} n})$ unless NP $\subseteq$ DTIME($n^{\text{poly}\log(n)}$). Since one can easily reduce the Mincost Diameter-d problem to the Mincost d-Spanner problem (see Lemma 2.1), this implies an identical hardness result for the latter problem. We also show that the condition $d \geq 3$ is essential by presenting an optimal $O(\log n)$-approximation algorithm for Mincost 2-Spanner (and thus Mincost Diameter-2) problem. Our algorithm generalizes the algorithm of Kortsarz and Peleg [13].

Perhaps more surprisingly, the same hardness result also holds for the Distance-d problem even when all edge lengths are *only* 1 *or* 2. These hardness results are based on reductions from a variant of the Label Cover problem [3]. We believe that our construction here is of independent interest since it seems to be adaptable to capture the hardness of some other network design problems. In particular, we can show that the *generalized steiner tree* problem in directed graphs [6] (given k pairs of vertices in a directed graph, find a minimum cost subgraph that connects each pair) is $\Omega(2^{\log^{1-\epsilon} n})$-hard, substantially improving upon the previously known $\Omega(\log n)$-hardness.

Finally, we show that the general Mincost Distance-d problem is $O(n \log d)$-approximable if all lengths are polynomially bounded. Our result is based on a suitable multicommodity flow formulation of our problem. An $O(n \log d)$ ratio can also be obtained via this approach for the Mincost d-Spanner problem. We remark that since in the presence of non-unit costs or lengths, any feasible solution to our problem may have $\Omega(n^2)$ edges, the approximation factor achieved above is indeed non-trivial. Our results are summarized in Figure 1.

2 Preliminaries

The input graph is denoted by $G = (V, E)$ and the number of vertices $|V| = n$. Given two vertices $u, v \in V$, we denote by $\texttt{dist}(u, v, G)$ the length of the shortest path from u to v in the graph G. The set of vertices adjacent to a vertex $s \in V$ is denoted by $N(s)$. We will use the following two simple lemmas.

Lemma 2.1 *An α-approximation algorithm for* Mincost d-Spanner *problem implies an α-approximation algorithm for the* Mincost Diameter-d *problem.*

Proof. For any input $G = (V, E)$ to Mincost Diameter-d with cost function c_e for $e \notin E$, create a complete graph $K = (V, F)$ where $F = V \times V$ with cost function c'_e defined as follows: if $e \in E$ then $c'_e = 0$, else $c'_e = c_e$. Then any solution E' for Mincost d-Spanner on K yields a solution $E'' = E' \backslash E$ for Mincost Diameter-d on G of identical cost. Conversely, any solution E'' for Mincost Diameter-d on G yields a solution $E' = E \cup E''$ to Mincost d-Spanner on K of identical cost. □

Lemma 2.2 *Let $X_1, X_2, \ldots, X_l$ be l independent 0/1 random variables s.t. $\Pr(X_i) = \min\{1, p_i\}$ and $\sum_i p_i \geq S$. Then $\Pr[\wedge \overline{X}_i] \leq e^{-S}$.*

Proof. If $p_i \geq 1$ for some $i \in [1 \ldots l]$, then $\Pr(\wedge \overline{X}_i) = 0 \leq e^{-S}$. Otherwise, each $p_i < 1$ and $\Pr(\wedge \overline{X}_i) = \prod_i (1 - p_i) \leq e^{-\sum_i p_i} \leq e^{-S}$. □

3 The Diameter-d Problem

3.1 An $O(\log n \log d)$ Approximation Algorithm

Theorem 3.1 *There is an $O(\log n \log d)$ approximation algorithm for* Diameter-d.

Our approach here is based on working with a certain restricted problem which we call Restricted Diameter-d, where we put special restrictions on the type of paths of length at most d that we allow. We first show that this restricted problem is closely related to our problem and thus it suffices to approximate the former. We then formulate the Restricted Diameter-d as an integer linear program and present a technique for efficiently rounding an optimal fractional solution to this program. Finally, we show that the fractional solution can be used to create an instance of the Hitting Set problem and can be derandomized by solving this hitting set instance by a greedy approach.

3.1.1 The Restricted Diameter-d Problem

In what follows, we let $U_d(G) = \{(u, v) \mid \texttt{dist}(u, v, G) > d\}$ to denote the set of "unsatisfied" pairs. We now define the restricted version of interest:

PROBLEM: Restricted Diameter-d

INSTANCE: A graph $G = (V, E)$ and a vertex $s \in V$.

GOAL: Find a minimum cardinality set of edges E' such that for each pair $(u, v) \in U_d(G)$, there is a path $\Gamma_{u,v}$ of length at most d in $G' = (V, E \cup E')$. Moreover, $\Gamma_{u,v}$ satisfies one of the following two properties:

(A) $\Gamma_{u,v}$ has exactly one edge in E' (we say that (u, v) is *covered* by a TYPE-A path), or

(B) $\Gamma_{u,v}$ has exactly 2 *consecutive* edges in E', both of which are incident on s (we say that (u, v) is *covered* by a TYPE-B path).

Lemma 3.2 *For any graph $G = (V, E)$ and any $s \in V$,*

$$1 \leq \frac{\mathsf{OPT}_{\textsf{Restricted Diameter-}d}(G)}{\mathsf{OPT}_{\textsf{Diameter-}d}(G)} \leq 3$$

Proof. The first inequality follows from the fact that any solution to Restricted Diameter-d is a solution to Diameter-d. To see the second inequality, let E' be an optimal solution to Diameter-d problem. Denote by $V' \subseteq V$ the set of vertices touched by E' and let $G' = (V, E \cup E')$. Clearly $|V'| \leq 2|E'|$. Define E'' to be E' augmented with all possible edges from s to V'; thus $|E''| \leq 3|E'|$. We claim that E'' is a solution to Restricted Diameter-d. Consider any pair $(u, v) \in U_d(G)$. Since G' has diameter at most d, there is a path $\Gamma_{u,v}$ between u and v in G' of length at most d. If $\Gamma_{u,v}$ has only one edge in E', (u, v) is covered by a TYPE-A path.

Otherwise, $\Gamma_{u,v}$ uses at least two edges in E'; let (a, b) and (x, y) denote the first and the last such edges respectively. Then the path which starts at u, follows $\Gamma_{u,v}$ to vertex a, then goes to vertex y via the vertex s, and finally follows $\Gamma_{u,v}$ to arrive at vertex v, gives either a TYPE-B path (a and y are different from s) or a TYPE-A path of length at most d. □

Thus it suffices to give an $O(\log n \log d)$-approximation to the Restricted Diameter-d problem. In what follows, we describe an ILP formulation for the restricted problem and show how its LP relaxation can be rounded to obtain the desired approximation guarantee.

3.1.2 An ILP Formulation for Restricted Diameter-d

Assume $G(V, E)$ is our input graph and let s denotes the special vertex. In order to describe the ILP, we need some further notation:

- We denote by $\overline{N}(s)$ the set of vertices not connected to s in G, i.e. the set $\{z \mid z \neq s,\ (s,z) \notin E\}$.
- Given $u \in V$, let $\text{LAYER}_i(u) = \{w \mid \texttt{dist}(u,w,G) = i\} \cap \overline{N}(s)$. Let $\text{LAYER}_{\leq j}(u) = \{w \mid \texttt{dist}(u,w,G) \leq j\} \cap \overline{N}(s) = \bigcup_{i \leq j} \text{LAYER}_i(u)$.
- Given $(u,v) \in U_d(G)$, we denote by $S_{u,v} = \{e \in \overline{E} \mid \texttt{dist}(u,v,G+e) \leq d\}$.
- Given two subsets $A, B \subseteq \{0 \ldots (d-2)\}$ such that $\max\{i \in A\} \leq \min\{j \in B\}$, we denote it by $A \prec B$. Such a pair (A,B) is called a *matched* pair.
- We denote by T the set $\{(i,j) | 0 \leq i \leq j \leq (d-2)\}$.

We use the following $0/1$ variables in our ILP:

- f_e indicates whether an edge $e \in \overline{E}$ is chosen.
- For ease of exposition we use additional variables x_w to indicate whether an edge (s,w) is chosen; we enforce that $x_w = f_{(s,w)}$.
- For each $(u,v) \in U_d(G)$, we use $\gamma_{u,v}$ to denote whether the pair (u,v) is covered by a TYPE-A or TYPE-B path. If the pair (u,v) is covered by a TYPE-A path then $\gamma_{u,v} = 1$, and it is 0 otherwise.

We now formulate the ILP constraints.

TYPE-A **Path Constraints:** A pair $(u,v) \in U_d(G)$ is covered by a TYPE-A path if and only if $\sum_{e \in S_{u,v}} f_e \geq 1$. We add the constraint $\sum_{e \in S_{u,v}} f_e \geq \gamma_{u,v}$ to determine whether u and v are covered by a TYPE-A path.

TYPE-B **Path Constraints:** A TYPE-B path for a pair $(u,v) \in U_d(G)$ has the form: $u \rightsquigarrow y \to s \to z \rightsquigarrow v$, where $y \in \text{LAYER}_i(u)$, $z \in \text{LAYER}_l(v)$ and $i + l \leq (d-2)$. Call such a path an $[i, d-2-l]$-path between u and v with *connecting points* y and z.[2] In other words, an $[i,j]$-path means that y is at distance i from u, z is at distance $d-2-j$ from v and if we add edges (s,y) and (s,z) to E, then u and v are at distance $i + 2 + (d-2-j) = d-(j-i)$. Thus, the valid settings for i and j are precisely all pairs $(i,j) \in T$. A particular $[i,j]$-path with connecting points y and z is selected if and only if $\min\{x_y, x_z\} \geq 1$. In order to capture all possible $[i,j]$-paths, we define the following two summations:

$$L_i(u) = \sum_{y \in \text{LAYER}_i(u)} x_y; \quad R_j(v) = \sum_{y \in \text{LAYER}_{d-2-j}(v)} x_z.$$

Then for any $(i,j) \in T$:

$$\text{Some } [i,j]\text{-path selected} \iff \min\{L_i(u), R_j(v)\} \geq 1.$$

Indeed, the minimum (over the integer x_w's) is at least 1 iff both sums are at least 1, i.e. some $x_y = x_z = 1$ for $y \in \text{LAYER}_i(u)$, $z \in \text{LAYER}_{d-2-j}(v)$, i.e. we included some $[i,j]$-path in our solution. The naive approach of including one such constraint for any $(i,j) \in T$ would lead to a poor approximation guarantee. So we need to cover $(i,j) \in T$ in a more effective manner. Abbreviate by $L_A(u) = \sum_{i \in A} L_i(u)$, $R_B(v) = \sum_{j \in B} R_j(v)$. Then:

$$\text{Some } [i,j]\text{-path selected for } i \in A, j \in B \iff \min\{L_A(u), R_B(v)\} \geq 1.$$

Notice, since $A \prec B$, all such paths are indeed paths of TYPE-B, i.e. we did not include paths of length more than d. This will be important later in the rounding process to ensure that we get a feasible solution, and explains why we insist that $A \prec B$. We now need to create a sequence of matched pairs $(A_1, B_1), \ldots, (A_t, B_t)$ that "cover" our set T.

Definition 3.3 (Covering Family)
A family of matched pairs $\mathcal{F} = \{(A_1, B_1), \ldots, (A_t, B_t)\}$ is a *covering family* for T if $\bigcup_{\alpha=1}^{t} A_\alpha \times B_\alpha = T$, i.e. for all $0 \leq i \leq j \leq (d-2)$ there exists an α such that $i \in A_\alpha$, $j \in B_\alpha$. Family $\mathcal{F}$ is called *C-covering* if for any $0 \leq i \leq (d-2)$,

$$|\{\alpha \mid i \in A_\alpha\}| \leq C \quad \text{and} \quad |\{\alpha \mid i \in B_\alpha\}| \leq C.$$

Now assume that we have a covering family $\mathcal{F}$ for T. Then the fact that at least one path of TYPE-B is selected between u and v is captured by the condition that $\sum_{\alpha=1}^{t} \min\{L_{A_\alpha}(u), R_{B_\alpha}(v)\} \geq 1$. Indeed, over the 0/1 domain the sum is at least 1 iff at least one of the minimums is at least 1, i.e. some path of TYPE-B is selected. Conversely, every path of TYPE-B is covered by some pair (A_α, B_α) since $\mathcal{F}$ is a covering family for T, and then the corresponding minimum will be at least 1. Let the variable $\delta_{\alpha,u,v}$ be defined as $\min\{L_{A_\alpha}(u), R_{B_\alpha}(v)\}$. Then (u,v) is covered by a TYPE-B path if and only if $\sum_{\alpha=1}^{t} \delta_{\alpha,u,v} \geq 1$. Finally, we add the constraint $\sum_{\alpha=1}^{t} \delta_{\alpha,u,v} \geq 1 - \gamma_{u,v}$ to indicate that (u,v) must be covered either by a TYPE-A path or a TYPE-B path. Putting together all the pieces, we get the following ILP:

$$\text{Minimize } \sum_{e \notin E} f_e$$

Subject To: $\forall (u,v) \in U_d(G)$

$$\sum_{e \in S_{u,v}} f_e \geq \gamma_{u,v} \tag{1}$$

$$\forall \alpha = 1 \ldots t, \quad \delta_{\alpha,u,v} \leq \min(L_{A_\alpha}(u), R_{B_\alpha}(v)) \tag{2}$$

$$\sum_{\alpha=1}^{t} \delta_{\alpha,u,v} \geq 1 - \gamma_{u,v} \tag{3}$$

$$\forall\ w \in \overline{N}(s), \quad f_{(s,w)} = x_w \tag{4}$$

$$\forall\ w \in \overline{N}(s), \forall\ e \notin E, \quad f_e, x_w \in \{0,1\} \tag{5}$$

[2] It seems more natural to call it an $[i,l]$-path, but our notation will turn out to be convenient later.

3.1.3 Rounding of an Optimal Fractional Solution

Let us now consider a relaxation of the above integer program where we replace the constraint $f_e, x_w \in \{0,1\}$ with $0 \leq f_e, x_w \leq 1$. We show how an optimal solution for this relaxation can be converted to an integral solution whose value is not much larger. Assume that we are given a C-covering family $\mathcal{F}$. The parameter C will play a direct role in determining our approximation guarantee, and we will specify later the value of C for the covering family $\mathcal{F}$ that we use.

Let O^* denote the value of the optimal fractional solution, and let f_e^*, x_w^*, $\gamma_{u,v}^*$ and $\delta_{\alpha,u,v}^*$ denote the values taken by the various variables in the optimal fractional solution. Our rounding procedure is the following:

include $e \notin E$ *in* E' *w/pr.* $\min\{1, 9Cf_e^* \log n\}$.

The set E' of chosen edges will be our solution to the Restricted Diameter-d (and Diameter-d) problem.

Theorem 3.4 *For any C-covering family $\mathcal{F}$, E' is w.h.p. a feasible solution to the* Restricted Diameter-d *problem of size at most an $O(C \log n)$ factor more than O^*.*

Proof. We will use w.h.p. to mean with probability at least $1 - 1/n$. It is easily seen that $\mathbf{E}[|E'|] \leq 9C \log n (\sum_{e \notin E} f_e^*) = (9C \log n) O^*$. By Chernoff bound, we get that w.h.p. the size of E' is at most $O(C \log n)$ times O^*. To show that w.h.p. E' is a feasible solution to Restricted Diameter-d, it suffices to show that for any $(u,v) \in U_d(G)$, $\Pr[(u,v) \text{ is not covered}] < 2/n^3$. Since $|U_d(G)| \leq \binom{n}{2}$, the union bound would give the desired result. We look at the following two cases:

$\underline{\gamma_{u,v}^* \geq 1/3}$: By condition (1), $\sum_{e \in S_{u,v}} f_e^* \geq 1/3$. Now if a pair (u,v) is not covered by a Type-A path, then no edge $e \in S_{u,v}$ is selected. Since the probability that an edge e is selected is $\min\{1, 9Cf_e^* \log n\}$ and $\sum_{e \in S_{u,v}} 9Cf_e^* \log n \geq 3 \log n$, we get by Lemma 2.2 that the probability (u,v) is not covered by a Type-A path is at most $1/n^3$.

$\underline{\gamma_{u,v}^* < 1/3}$: By condition (3), $\sum_{\alpha=1}^{t} \delta_{\alpha,u,v}^* \geq 2/3$. Denote $A \prec i_0$ to mean that $\max\{i \in A\} \leq i_0$, and similarly, $i_0 \prec B$ to mean that $i_0 \leq \min\{i \in B\}$. Choose the *unique* i_0 such that

$$\sum_{\{\alpha | A_\alpha \prec (i_0 - 1)\}} \delta_{\alpha,u,v}^* < \frac{1}{3} \leq \sum_{\{\alpha | A_\alpha \prec i_0\}} \delta_{\alpha,u,v}^* \qquad (6)$$

Such an i_0 exists as the sum goes from 0 to at least 2/3. Using inequalities (6) and (2) together with C-coverability of $\mathcal{F}$, we get

$$\begin{aligned} 1/3 &\leq \sum_{\{\alpha | A_\alpha \prec i_0\}} \delta_{\alpha,u,v}^* \leq \sum_{\{\alpha | A_\alpha \prec i_0\}} L_{A_\alpha}(u) \\ &= \sum_{i=0}^{i_0} |\{\alpha \mid i \in A_\alpha\}| \cdot L_i(u) \leq C \cdot \sum_{i=0}^{i_0} L_i(u) \\ &= C \cdot \sum_{y \in \mathsf{LAYER}_{\leq i_0}(u)} x_y^* \end{aligned}$$

Also, (6) together with our assumption implies that $\sum_{\{\alpha | A_\alpha \not\prec (i_0 - 1)\}} \delta_{\alpha,u,v}^* \geq 1/3$. But whenever it is not the case that $A_\alpha \prec (i_0 - 1)$, there is some $i \in A_\alpha$ s.t. $i \geq i_0$, and as $A_\alpha \prec B_\alpha$ (our pairs are matched), we must have that $i_0 \prec B_\alpha$. Thus, analogously to the previous case,

$$\begin{aligned} 1/3 &\leq \sum_{\{\alpha | A_\alpha \not\prec (i_0 - 1)\}} \delta_{\alpha,u,v}^* \leq \sum_{\{\alpha | i_0 \prec B_\alpha\}} \delta_{\alpha,u,v}^* \\ &\leq \sum_{\{\alpha | i_0 \prec B_\alpha\}} R_{B_\alpha}(v) = \sum_{i=i_0}^{d-2} |\{\alpha \mid i \in B_\alpha\}| \cdot R_i(v) \\ &\leq C \cdot \sum_{i=i_0}^{d-2} R_i(v) = C \cdot \sum_{z \in \mathsf{LAYER}_{\leq d-2-i_0}(v)} x_z^* \end{aligned}$$

To summarize, there is $0 \leq i_0 \leq (d-2)$ s.t.

$$\min\Big(\sum_{y \in \mathsf{LAYER}_{\leq i_0}(u)} x_y^*, \sum_{z \in \mathsf{LAYER}_{\leq d-2-i_0}(v)} x_z^*\Big) \geq \frac{1}{3C} \qquad (7)$$

Now we claim that w.h.p. we select some edge (y,s) in E' where $y \in \mathsf{LAYER}_{\leq i_0}(u)$. Indeed, such edge is selected w/pr. $\min(1, 9Cx_y^* \log n)$. As we have by (7) that $\sum_{y \in \mathsf{LAYER}_{\leq i_0}} 9Cx_y^* \log n \geq 3 \log n$, Lemma 2.2 implies that none of the y's is selected with probability at most $1/n^3$. Similar argument shows that with probability at most $1/n^3$ none of the edges (s,z) where $z \in \mathsf{LAYER}_{\leq d-2-i_0}(v)$ is selected. By union bound, we select an appropriate (y,s) and (s,z) with probability at least $1 - 2/n^3$. Such y and z create a path of Type-B of length at most d, as needed. $\Box$

3.1.4 An $O(\log d)$-covering Family

The last step is to create a C-covering family for $T = \{(i,j) \mid 0 \leq i \leq j \leq (d-2)\}$ for a small value C. We are able to achieve $C = O(\log d)$, giving us the $O(\log n \log d)$ approximation claimed by Theorem 3.1.

Let $C[l]$ denote the covering number of the family we construct for $T_l = \{(i,j) \mid 0 \leq i \leq j \leq l\}$ and let $f = \lfloor l/2 \rfloor$. We include the rectangle $R = (\{0 \ldots f\}, \{f \ldots l\})$, after which the only uncovered pairs are the ones of the sets $T_l^1 = \{(i,j) \mid 0 \leq i \leq j \leq f-1\}$, $T_l^2 = \{(i,j) \mid f+1 \leq i \leq j \leq l\}$. We cover them recursively and since coverage $C[l]$ of each point is at most $1 + C[f] \leq 1 + C[l/2]$, we get $C[l] = \Theta(\log l)$. Setting $l = d-2$ yields the desired result. It is also easy to show that for any covering family for T_l, $C[l] \geq \Omega(\log l / \operatorname{loglog} l)$; so our analysis can be improved by at most an $O(\operatorname{loglog} d)$ factor.

3.1.5 Derandomization

We observe that an optimal solution to our linear program in fact gives us an instance I of the Hitting Set problem such that any *integral* solution to this instance I is also a solution to the Diameter-d problem. Specifically, the universe U of our hitting set instance is the set $\overline{E}$ of all edges not in G. For each pair $(u, v) \in U_d(G)$ we add the following sets to the collection of sets to be hit. If $\gamma^*_{u,v} \geq 1/3$, we simply add the set $A = S_{u,v}$. And if $\gamma^*_{u,v} < 1/3$, we know that there exists a value $i_0 \in [0, d-2]$ satisfying Equation (7). For this pair, we add two sets $A' = \{e \mid e = (s, y) \text{ where } y \in \text{LAYER}_{\leq i_0}(u)\}$ and $B' = \{e \mid e = (s, z) \text{ where } z \in \text{LAYER}_{\leq d-2-i_0}(v)\}$. It is clear that any set of edges in $\overline{E}$ that hits (intersects) all these sets forms a feasible solution to our problem.

We now simply use a greedy algorithm to solve this hitting set problem, i.e. we iteratively keep picking edges that hit a largest number of sets among those yet untouched. We claim that this greedy solution is indeed an $O(\log n \log d)$ approximation. To see this, suppose we scale the value of all the variables in our LP solution by a factor of $3C$. Then by our construction, we know that the scaled variables form a fractional solution for this instance I. Moreover, the value of this fractional solution is $3C \cdot O^*$. The final fact needed to complete the analysis is the well-known result showing that the greedy algorithm always yields a solution that is within an $O(\log n)$ factor of the optimal fractional solution (see [9], for instance). Substituting $C = O(\log d)$, we get the claimed result.

Remark 3.5 We already pointed out that our algorithm extends to the directed case as well as to the case of non-uniform distance requirements $d(u, v)$ (giving the ratio of $O(\log n \log d_{\max})$ where $d_{\max} = \max_{u,v} d(u, v)$). It also extends to the case when we place the distance requirements not for all $\binom{n}{2}$ pairs (u, v), but only for a subset of pairs, say $\{(u_i, v_i)\}_{i=1}^k$ (in particular, the resulting graph need not be connected). The approximation ratio becomes $O(\log k \log d_{\max})$. Finally, we can easily obtain a "bicriteria" $(2, O(\log n))$-approximation where the number of edges we add is a factor of $O(\log n)$ away from the optimum, but the vertices are only guaranteed to be at distance at most $2d$ (rather than d) from each other.

3.2 Hardness of Diameter-d Problem

Li *et al.* [17] showed that Diameter-d is $\Omega(\log n)$-hard for $d \geq 4$ using a reduction from Dominating Set. We show an $\Omega(\log n)$ hardness for $d \in \{2, 3\}$ as well.

Theorem 3.6 *The Diameter-d problem is $\Omega(\log n)$-hard to approximate for $d \in \{2, 3\}$ unless* P = NP.

Proof. We use a reduction from Set Cover which is known to be $\Omega(\log n)$-hard unless P = NP [8, 4, 21]. Consider an instance I of Set Cover problem specified by a collections of sets $S_1, \ldots, S_m$ over the universe $U = \{u_1, \ldots, u_n\}$. The goal is to find the smallest collection of sets whose union is U. We assume that for every $u_i, u_j \in U$, there exists a set S_r containing both u_i and u_j. This assumption is w.l.o.g. since we can always add $\binom{n}{2}$ additional sets $S_{i,j} = \{u_i, u_j\}$ to our collection of sets changing the optimal set-cover value by at most a factor of 2.

Consider the following graph $G = (V, E)$. V has a vertex s_i for each set S_i and M vertices $u_{i,1}, u_{i,2}, \ldots, u_{i,M}$ for each element $u_i \in U$, where $M = 2(m + 1)$. Moreover, V contains a special additional vertex r that is not connected to any other vertex. If an element $u_i \in S_l$, there is an edge in E from s_l to each $u_{i,j}$, $1 \leq j \leq M$. Finally, E contains all edges of the form (s_i, s_j), i.e. the "set vertices" induce a clique. It is easy to verify that all vertices in G have a pairwise distance of at most 2 except for the pairs that involve the special vertex r (here we use our assumption about a common set for any pair of vertices). Also observe that there is always a solution of cost at most m on this instance of Diameter-2 problem: just connect r to all the "set vertices" in G.

We first argue that any set-cover consisting of p sets yields a solution to Diameter-2 on G of cost p. Simply connect r to p "set vertices" corresponding to the chosen sets. Conversely, consider any solution E' of size at most m for Diameter-2 on G. Let $s_{l_1}, \ldots, s_{l_p}$ be the "set vertices" connected to r in E'. We claim that the sets $S_{l_1}, \ldots, S_{l_p}$ must form a set-cover for U of size $p \leq |E'|$. Indeed, if there is some $u_i \notin \cup_{q=1}^p S_{l_q}$, then for each $j \in \{1 \ldots M\}$, $u_{i,j}$ must have an adjacent edge in E' in order to have a path of length at most two to r. But then $|E'| \geq M/2 > m$, a contradiction.

The construction and proof for $d = 3$ is almost identical, we only replace the single isolated vertex r by an M-clique $r_1, \ldots, r_M$. Again, any solution of cost p for set-cover yields a solution of cost p for Diameter-3 by connecting r_1 to the corresponding sets. Conversely, take any solution E' of cost at most m for Diameter-3 on G and look at the sets corresponding to the "set vertices" adjacent to some r_i in E'. They must form a set-cover or else we can show similar to the previous case that $|E'| > m$, a contradiction. □

4 Hardness of the Mincost Diameter-d and Distance-d Problem

We now show that our problem becomes much harder once non-uniform costs or lengths are introduced. We use the following version of the Label Cover problem that we refer to as the Symmetric Label Cover problem, to show the hardness for both Mincost Diameter-d and Distance-d problems.

Definition 4.1 (Symmetric Label Cover) We are given a complete bipartite graph $H = (U, W, E_n)$ (where $|U| = |W| = n$), two sets A and B (called the *label sets*), and a non-empty relation $R_{u,w} \subseteq A \times B$ for each edge $(u, w) \in E_n$. A feasible solution is a pair of label assignments $M_U : U \rightarrow 2^A$ and $M_W : W \rightarrow 2^B$ such that each edge (u, w) is *consistent*, i.e. there exist $a \in M_U(u)$ and $b \in M_W(w)$ such that $(a, b) \in R_{u,w}$. The objective is to find a pair of label assignments such that $\sum_{u \in U} |M_U(u)| + \sum_{w \in W} |M_W(w)|$ is minimized.

This problem is known to be $\Omega(2^{\log^{1-\epsilon} n})$-hard for any $\epsilon > 0$ (provided NP $\not\subset$ DTIME($\mathrm{n}^{\mathrm{poly}\log(\mathrm{n})}$)) via a reduction from Label Cover [2, 10]. We use it to show the following result.

Theorem 4.2 *For any $\epsilon > 0$, the following problems are $\Omega(2^{\log^{1-\epsilon} n})$-hard:*

(a) Mincost Diameter-d*, for any $d \geq 3$.*
(b) Mincost d-Spanner*, for any $d \geq 3$.*
(c) Distance-d*, even when all lengths are 1 and 2.*
(d) Directed generalized steiner network problem.

Proof. (a) We start by presenting a reduction for $d = 3$ and then sketch an extension to the case $d > 3$. Let I be an instance of Symmetric Label Cover specified by a complete bipartite graph $H = (U, W, E_n)$, relations $R_{u,w}$ for all $u \in U$, $w \in W$, and label sets A, B. We create an instance $G = (V, E)$ as the input to Mincost Diameter-3 problem, where V, $E = E_1 \cup E_2 \cup E_3$ and costs on the missing edges $\overline{E}$ are as follows:

- $V = U \cup W \cup \{U \times A\} \cup \{W \times B\} \cup \{r\}$; we denote the elements of $\{U \times A\}$ as pairs $\langle u, a\rangle$, similarly for $W \times B$.
- $E_1 = \{(\langle u, a\rangle, \langle w, b\rangle) \mid u \in U, w \in W, (a, b) \in R_{u,w}\}$; these edges capture the consistent label assignments for any pair (u, w).
- $E_2 = \{(u, u') \mid u, u' \in U\} \cup \{(w, w') \mid w, w' \in W\}$; these edges induce a clique on U and W.
- $E_3 = \{(r, \langle u, a\rangle) \mid u \in U, a \in A\} \cup \{(r, \langle w, b\rangle) \mid w \in W, b \in B\}$; these edges create a length two path via vertex r between any pair of label nodes.
- Let $E' = \{(u, \langle u, a\rangle) \mid u \in U, a \in A\} \cup \{(w, \langle w, b\rangle) \mid w \in W, b \in B\}$. Edges in E' have unit cost while every other missing edge is assigned a large cost $C = |E'| + 1$. Intuitively, edges in E' correspond to assigning labels to vertices in U, W.

Observe that the only pairs of vertices in G that are not already within a distance of 3 correspond to (u, w), $u \in U, w \in W$. Moreover, the graph $G' = (V, E \cup E')$ always has diameter 3. Since every missing edge outside of set E' has cost $|E'| + 1$, w.l.o.g. assume a feasible solution is always a subset of E'.

Let $S = (M_U, M_W)$ be any solution to the label cover instance I, and define $E_S = \{(u, \langle u, a\rangle \mid u \in U, a \in M_U(u)\} \cup \{(w, \langle w, b\rangle) \mid w \in W, b \in M_W(u)\}$. Clearly, $|E_S| = \sum_{u \in U} |M_U(u)| + \sum_{w \in W} |M_W(w)|$. Since S is consistent for each pair (u, w), we get that $G_S = (V, E \cup E_S)$ indeed has diameter 3. Conversely, consider any set of edges E' such that $G' = (V, E \cup E')$ has diameter 3. Then any pair (u, w) must be connected to each other via a path of length 3 of the form $u \rightarrow \langle u, a\rangle \rightarrow \langle w, b\rangle \rightarrow w$ such that $(a, b) \in R_{u,w}$. Thus, defining $M_U(u) = \{a \mid (u, \langle u, a\rangle) \in E'\}$ and $M_W(w) = \{b \mid (w, \langle w, b\rangle) \in E'\}$, gives a solution to I of cost $|E'|$, completing the proof.

To extend this reduction to $d > 3$, we simply augment the graph G constructed above. Let $t_U = \lfloor \frac{d-3}{2} \rfloor$ and $t_W = \lceil \frac{d-3}{2} \rceil$. For each $u \in U$, attach a path P_u of length t_U to u, and for each $w \in W$, attach a path Q_w of length t_W to w. As before, only the edges in E' have unit cost and all other missing edges have cost $|E'| + 1$. Now, an analogous argument can be used to show that for any pair (u, w), the last vertex on P_u and the last vertex on Q_w are within a distance of d if and only if u and w are within a distance of 3, establishing the desired hardness of Mincost Diameter-d for $d \geq 3$.

(b) Follows from part (a) and Lemma 2.1. We remark that the best known hardness result for d-Spanner is $\Omega(\log(n/d))$ [12], so we are able to obtain a much stronger hardness result once general costs are allowed.

(c) We use the same construction as in (a). All the edges of the original graph are assigned length 1 as well as all the (missing) edges in E'. All other edges (the ones that had large cost in the previous construction) are assigned length 2, and all the edge costs are 1. Now if we set $d = 3$, a similar argument completes the proof.

(d) We use again a modification of the construction in part (a); the details are defered to the final version. □

5 Approximating Mincost Diameter-d, Distance-d and Mincost Distance-d

5.1 Approximating Mincost Diameter-d Problem

Our main results here are as follows:

Theorem 5.1 *The* Mincost d-Spanner *problem is:*

- *$O(\log n)$-approximable for $d = 2$.*
- *$O(n \log d)$-approximable for $d \geq 3$.*

Combining this with Lemma 2.1 gives us identical results for the Mincost Diameter-d problem. This also shows why the hardness result of Theorem 4.2 holds only for $d \geq 3$.

5.1.1 $O(\log n)$-approximation for Mincost 2-Spanner

Kortsarz and Peleg [13] gave an $O(\log n)$-approximation algorithm for the (unit cost) 2-Spanner problem. We show how their algorithm may be extended to handle arbitrary edge costs. In fact, we present a much simpler analysis for this more general algorithm.

Observe that a 2-spanner $G' = (V, E')$ for $G = (V, E)$ implies that for every $e \in E$, either $e \in E'$ or there are $e_1, e_2 \in E'$ forming a triangle with e. In the latter case, we say that e is *covered* by e_1 and e_2. Given a collection F of edges, they *cover* the set of edges $\{e \in E \mid \exists\, e_1, e_2 \in F \text{ covering } e\}$. Given F, a vertex v covers all the edges in the set $C(v, F) = \{(a, b) \in E \mid (a, v), (v, b) \in F\}$.

Definition 5.2 (Density) Given a graph $G = (V, E)$ with cost $c_e \geq 0$ on every edge $e \in E$ and $w_v > 0$ on every vertex $v \in V$, the *density* of G is given by $\rho(G) = \sum_{e \in E} c_e / \sum_{v \in V} w_v$.

Using a reduction to the minimum cut problem (see [16]), we can find in polynomial time the densest subgraph G', i.e. the induced subgraph of G of maximum density. Using this result, we make the following modification to the algorithm of [13] to deal with the weighted case.

Algorithm: Let $G = (V, E)$ be the input graph with a non-negative cost c_e associated with each edge $e \in E$. We will iteratively maintain the following sets partitioning the edge set E:

- E^s – edges included in the 2-spanner. Initially consists of all zero-cost edges of E.
- E^c – edges currently covered by E^s.
- E^u – edges yet to be covered.

We repeat the following procedure until we are done in step (3). For each $v \in V$, let $G_v = (N(v), E(v))$ be the subgraph of G induced by E^u in v's neighborhood $N(v)$.

(1) Assign a vertex cost of $w_{v'} = c_{(v,v')}$ to each $v' \in N(v)$ and an edge cost c_e for every $e \in E(v)$.
(2) Find the densest subgraph $H_v = (N_v, F_v)$ in G_v. Let $\rho(v)$ be its density and let $\rho = \max_{v \in V} \rho(v)$ be achieved by $v_0 \in V$.
(3) If $\rho \leq 1$ or $E^u = \emptyset$, output $E^s \cup E^u$ and stop.
(4) Otherwise, add the star from v_0 to N_{v_0} to E^s, now covered edges F_{v_0} to E^c and remove all these edges from E^u.

Clearly, we output a 2-spanner upon termination and the algorithm terminates in polynomial time as each iteration decreases the cardinality of E^u.

Analysis: We show the following.

Theorem 5.3 *The above algorithm achieves the approximation ratio $O(1 + \log \frac{M}{M^*})$, where $M = \sum_{e \in E} c_e$ is the sum of all edge costs of G and M^* is the cost of the optimum 2-spanner for G.*

Proof. We need the following claim concerning effective "coverability".

Claim 5.4 *Let M' denote the total cost of edges in E^u at the beginning of some iteration of the algorithm. Then $\rho \geq \frac{1}{2}(\frac{M'}{M^*} - 1)$ during this iteration.*

Proof. Consider a minimum cost 2-spanner F of cost M^s which covers the edges of $E^u \subseteq E$. Since F can use any edge of the original graph, $M^s \leq M^*$. Let $W^c(v) = \sum_{e \in C(v,F) \cap E^u} c_e$ be the cost of all the edges of E^u that are covered by v in F. Then $\sum_{v \in V} W^c(v) \geq M' - M^s$, as all edges of E^u except (maybe) those of the spanner F are covered. Let $W^s(v)$ be the sum of costs of all the edges of F adjacent to v. Then $\sum_{v \in V} W^s(v) = 2 \cdot M^s$, as each edge of F is counted twice. Hence, there must be a vertex v s.t.

$$\frac{W^c(v)}{W^s(v)} \geq \frac{M' - M^s}{2 \cdot M^s} = \frac{1}{2}\left(\frac{M'}{M^s} - 1\right) \geq \frac{1}{2}\left(\frac{M'}{M^*} - 1\right)$$

By our procedure of assigning costs to vertices in the neighborhood of v, we get that the neighborhood of v in F has the claimed density. $\square$

Let W_i be the sum of the costs of new edges we added to E^s during iteration i and M_i be the cost of E^u after this iteration. By Claim 5.4,

$$M_i \leq M_{i-1} - W_i - \frac{W_i}{2} \cdot \left(\frac{M_{i-1}}{M^*} - 1\right) \quad (8)$$

$$\leq M_{i-1}\left(1 - \frac{W_i}{2M^*}\right) \quad (9)$$

This also implicitly shows that $W_i \leq 2M^*$ for each i. Let us look at the last round k s.t. $M_k \geq M^*$. Let W be the cost of edges added to our spanner after round k. We claim that $W \leq 3 \cdot M^*$. Indeed, if k was the last round, since no subgraph has density more than 1 we have by Claim 5.4 that $\frac{1}{2}(\frac{M_k}{M^*} - 1) \leq 1$, so $W = M_k \leq 3 \cdot M^*$. If k was not the last round, we have that $W_{k+1} \leq 2 \cdot M^*$ and $M_{k+1} \leq M^*$, so again $W \leq W_{k+1} + M_{k+1} \leq 3 \cdot M^*$. Also, by Equation (9) and our choice of k,

$$M^* \leq M_k \leq M \prod_{i=1}^{k} (1 - \frac{W_i}{2M^*}) \leq M e^{-\sum W_i / 2M^*}$$

$$\Longrightarrow \sum_{i=1}^{k} W_i \leq 2M^* \log \frac{M}{M^*}$$

Thus, the total cost of our spanner $W + \sum_{i=1}^{k} W_i \leq 3M^* + 2M^* \log \frac{M}{M^*} = O(1 + \log \frac{M}{M^*}) \cdot M^*$. $\square$

Eliminating Large Costs: The performance ratio of $O(1 + \log \frac{M}{M^*})$ can be very large if G has some "useless" edges of very high cost. This is overcome as follows. Let C be the smallest edge cost such that removing all edges of cost at least C leaves no 2-spanner for G (i.e. the remaining graph by itself is not a 2-spanner for G).

Clearly, $C \le M^*$. On the other hand, if we leave only the edges of cost at most C, they form a valid 2-spanner for G of cost at most Cn^2. Thus, $M^* \le Cn^2$. Hence, replacing the cost of all edges of cost more than Cn^2 by $2Cn^2$ leaves the optimal 2-spanner as well as its cost M^* unchanged. However, now the sum of all the edge costs is at most $2Cn^2 \cdot n^2 \le 2n^4 M^*$, so our algorithm on this modified graph yields approximation ratio of $O(1 + \log \frac{2n^4 M^*}{M^*}) = O(\log n)$.

5.1.2 Approximating Mincost d-Spanner ($d \ge 3$)

We now present a randomized $O(n \log d)$-approximation algorithm for the Mincost d-Spanner when $d \ge 3$. We start with the following two definitions:

Definition 5.5 (d-extension) Given a graph $G = (V, E)$, the *d-extension* of G is a $(d+1)$-layered directed graph $G[d]$ as described below:

- $G[d]$ has $d+1$ layers of vertices $V^0, \dots, V^d$ where each V^i is a copy of V. For $u \in V$, we denote by u_i the copy of u in V^i and given any $U \subseteq V$, let $U^i = \bigcup_{u \in U} u_i$.
- For each $(u, v) \in E$, there is a directed edge from u_i to v_{i+1}, where $0 \le i < d$. In addition, we add "self" edges (u_i, u_{i+1}) for $0 \le i < d$.

Definition 5.6 (d-Ascending u-v Cut) A *d-ascending u-v cut* in a graph $G = (V, E)$ is a cut in its d-extension $G[d]$ and is specified by a sequence $C = \{U_0 \subseteq U_1 \subseteq \ldots \subseteq U_d \mid U_i \subseteq V, u \in U_0, v \notin U_d\}$. The two sides of the cut induced by C are given by $L(C) = \bigcup_{i=0}^{d} U_i^i$ and $R(C) = \bigcup_{i=0}^{d} V^i \backslash U_i^i$. We say that C is *satisfied* if at least one edge in $G[d]$ goes from $L(C)$ to $R(C)$.

Observe that since $G[d]$ contains self edges, d-ascending cuts are the only cuts not satisfied when $E = \emptyset$. It is easy to see that there are exactly $(d+1)^{n-2}$ d-ascending cuts for a n-vertex graph. This follows from the fact that each $w \in V \backslash \{u, v\}$ has $(d+1)$ disjoint sets to choose from: $U_0 \backslash \{u\}, U_1 \backslash U_0, \dots, U_d \backslash U_{d-1}, V \backslash (U_d \cup \{v\})$.

Lemma 5.7 *Given a graph $G = (V, E)$, for any $u, v \in V$ and $E' \subseteq E$, the following are equivalent:*

(1) $\mathtt{dist}(u, v, G') \le d$*, where* $G' = (V, E')$.

(2) u_0 *and* v_d *are connected in* $G'[d]$.

(3) *Every d-ascending u-v cut of G' is satisfied.*

Proof. Let $\mathtt{dist}(u, v, G') = l \le d$ and $u^0 = u, u^1, \dots, u^l = v$ be a $u \rightsquigarrow v$ path in G'. Let $u^i = v$ for $l < i \le d$. Then the sequence $u_0^0, u_1^1, \dots, u_d^d$ is a $u_0 \rightsquigarrow v_d$ path in $G'[d]$. Conversely, any $u_0 \rightsquigarrow v_d$ path in $G'[d]$ naturally defines a path of length at most d in G' by removing self edges. Finally, u_0 and v_d are disconnected in $G'[d]$ if and only if there exists a regular cut between u_0 and v_d that is not satisfied. But we already observed that due to self edges, the only cuts that may not be satisfied are the d-ascending cuts. □

A Multicommodity Flow Formulation: The preceding lemma tells us that the Mincost d-Spanner problem on input $G = (V, E)$ can be equivalently stated as follows: Choose a minimum cost subset of edges $E' \subseteq E$ s.t. for any $(u, v) \in E$, the vertices u_0 and v_d are connected in $G'[d]$ where $G' = (V, E')$. This can be formulated as a "non-aggregating" multicommodity flow problem in the graph $G[d]$. There is a commodity $q_{(u,v)}$ for each edge $(u, v) \in E$, and let $Q = \{q_{(u,v)} \mid (u, v) \in E\}$ be the set of all commodities. For each $(u, v) \in E$, we require that one unit of $q_{(u,v)}$-flow be sent from u_0 to v_d. We use variable $h_q(x_i, y_{i+1})$ to denote the amount of flow of commodity q across the edge (x_i, y_{i+1}) in $G[d]$. For any $(x, y) \in E$ and $q \in Q$, we define variables $g_q(x, y) = \sum_{i=0}^{d-1} h_q(x_i, y_{i+1})$ and $f(x, y) = \max_{q \in Q} g_q(x, y)$. Our objective function is simply to minimize the sum $\sum_{(x,y) \in E} c_{(x,y)} f(x, y)$. In other words, we charge each "non-self" edge in proportion to the *maximum flow of any commodity* routed over all of its d copies (x_i, y_{i+1}) in the graph $G[d]$. We omit here the description of standard flow conservation constraints. Since in the optimal solution the path of length at most d from u to v uses each edge at most once, the LP is the relaxation of our problem.

Rounding and Analysis: Let the superscript * denote the value of the variables in an optimal fractional solution to the LP. For each $(x, y) \in E$, we include (x, y) in our solution E' with probability $\min\{1, \beta f^*(x, y)\}$ where β will be chosen later. Clearly, the cost of E' is w.h.p. at most a factor $O(\beta)$ away from the cost of optimal d-spanner. Let $G' = (V, E')$. It remains to choose β s.t. for all $(u, v) \in E$, w.h.p. there is a path of length at most d connecting u and v in G', i.e. (Lemma 5.7) all d-ascending u-v cuts are satisfied in $G'[d]$.

Lemma 5.8 *For any (u, v) and any d-ascending u-v cut C, the probability that C is not satisfied in $G'[d]$ is at most $e^{-\beta}$.*

Proof. Let $q = q_{(u,v)}$, $E_C[d]$ be the set of edges crossing C in $G[d]$ and $E_C \subseteq E$ be the set of edges (x, y) such that $(x_i, y_{i+1}) \in E_C[d]$ for some $0 \le i < d$. Since one unit of commodity q must flow across C, we get

$$1 \le \sum_{(x_i, y_{i+1}) \in E_C[d]} h_q^*(x_i, y_{i+1}) \le \sum_{(x,y) \in E_C} f^*(x, y)$$

Using the fact that each $e \in E_C$ is chosen with probability $\min\{1, \beta f^*(x, y)\}$ and that $\sum_{(x,y) \in E_C} \beta f^*(x, y) \ge \beta$, we get by Lemma 2.2 that $\Pr[C \text{ is not satisfied}] = \Pr[E_C \cap E' = \emptyset] \le e^{-\beta}$. □

Choosing $\beta = O(n \log d)$ bounds the above probability by $\frac{1}{n^3 (d+1)^{n-2}}$. Since there are at most $O(n^2)$ u-v pairs to be considered and each pair has exactly $(d+1)^{n-2}$

d-ascending u-v cuts, E' forms a feasible solution with probability at least $1 - 1/n$.

5.2 Approximating Distance-d and Mincost Distance-d

We describe how to extend our algorithm to the more general version of the Mincost d-Spanner problem where the edges have arbitrary (polynomially bounded) lengths and we want to find the subgraph of the smallest cost that is a d-spanner w.r.t. this length function. Since the analog of Lemma 2.1 still holds w.r.t. general lengths (i.e. we now consider spanners for graphs with arbitrary lengths on their edges), this implies the same result for the Mincost Distance-d problem. We remark that the previous algorithm works with no changes for directed graphs, and it is actually simpler to describe our extension for the case of directed graphs as well.

The basic idea is to transform the input graph $G = (V, E)$ into a (directed) unit-length graph $H = (V_H, E_H)$ such that each edge $e = (u, v)$ in G is replaced by a path of length l_e (recall, l_e is the length of e), say $u, x_{e,1}, x_{e,2}, \ldots, x_{e,l_e-1}, v$. We put the original cost c_e on the last edge (x_{e,l_e-1}, v) of the path (referred to as the *last edge* corresponding to e) and a cost of 0 on all the other edges along the path. Since the lengths are polynomially bounded, the size of H is polynomial.

Now we simply run the algorithm of the preceding section on H (except we only ship flow between pairs of "original" vertices) and use the *same rounding*, i.e. the same $\beta = O(n \log d)$. We then take all the "last" edges in our solution and output the edges of G corresponding to them as our solution (the cost is identical). The analysis essentially remains unchanged. But the critical point is that even though the size of H is polynomially larger than that of G, the number of "relevant" u-v cuts is the same as before, $d^{O(n)}$ (yielding the $O(n \log d)$ performance guarantee). Indeed, since we can always add zero-cost edges to our solution and because of the special structure of H, the only u-v cuts in $H[d]$ that need to be non-trivially satisfied are the ones that "come from G". Specifically, we take any d-ascending u-v cut C in G, take its right side $R(C)$ in $G[d]$ and view it as the right side of a cut in $H[d]$. If all such cuts are satisfied in $H[d]$, then adding all the zero cost edges makes u_0 and v_d connected in $H[d]$. We defer the complete details to the final version.

Acknowledgments

We would like to thank Chandra Chekuri and Madhu Sudan for many valuable discussions.

References

[1] I. Althofer, G. Das, D. Dobkin, D. Joseph and J. Soares. On sparse spanners of weighted graphs. In *Discrete and Computational Geometry*, 9:81–100, 1993.

[2] E. Amaldi and V. Kann. The complexity and approximability of finding maximum feasible subsystems of linear relations. *Theoretical Computer Science*, 147(1-2):181–210, 1995.

[3] S. Arora, L. Babai, J. Stern and Z. Sweedyk. The hardness of approximate optima in lattices, codes, and systems of linear equations. *Journal of Computer and System Sciences*, 54(2):317–331, 1997.

[4] S. Arora and M. Sudan. Improved low degree testing and its applications. In *Proc. of the 29th ACM Symposium on Theory of Computing*, pp. 485–495, 1997.

[5] B. Awerbuch, A. Baratz, and D. Peleg. Cost-sensitive analysis of communication protocols. In *Proc. 9th Symp. on Principles of Dist. Comput.*, pp. 177–187, 1990.

[6] M. Charikar, C. Chekuri, T. Cheung, Z. Dai, A. Goel, S. Guha and M. Li. Approximation algorithms for directed steiner problems. In *Proc. 9th ACM-SIAM Symposium on Discrete Algorithms*, pp. 192–200, 1998.

[7] E. Cohen. Fast Algorithms for constructing t-spanners and paths with stretch t (extended abstract). In *Proc. 34th Symp. on Foundation of Computer Science*, pp. 648–658, 1993.

[8] U. Feige. A theshold of $\ln n$ for approximating set-cover. In *Proc. 26th ACM Symp. on the Theory of Computing*, pp. 314–318, 1996.

[9] D. Hochbaum (Ed.). Approximation Algorithms for NP-Hard Problems. *PWS Publishing Company*, 1997.

[10] S. Khanna, M. Sudan and L. Trevisan. Constraint satisfaction: The approximability of minimization problems. In *Proc. 12th Computational Complexity Conf.*, pp. 282–296, 1997.

[11] S. Khuller, B. Raghavachari, and N. Young. Balancing minimum spanning and shortest path trees. In *Proc. 4th Symp. on Discrete Algorithms*, pp. 243–250, 1993.

[12] G. Kortsarz. On the hardness of approximating spanners. In the *First International Workshop on Approximation Algorithms for Combinatorial Optimization*, 1998.

[13] G. Kortsarz and D. Peleg. Generating sparse 2-spanners. In *LNCS*, 621:73–83, 1992.

[14] G. Kortsarz and D. Peleg. Generating low-degree 2-spanners. In *Proc. 5th ACM-SIAM Symp. on Discrete Algorithms*, pp. 556–563, 1994.

[15] G. Kortsarz and D. Peleg. Approximating shallow-light trees. In *Proc. 8th ACM-SIAM Symp. on Discrete Algorithms*, pp. 103-110, 1997.

[16] E. Lawler. Combinatorial Optimization: Networks and Matroids. *Holt, Rinehart and Winston*, 1976.

[17] C. Li, S. McCormick, and D. Simchi-Levi. On the Minimum-cardinality-bounded-diameter and Bounded-cardinality-minimum-diameter Edge Addition Problems. In *Operation Research Letters*, 11:303–308, 1992.

[18] M. Marathe, R. Ravi, R. Sundaram, S. Ravi, D. Rosenkrantz, and H. Hunt III. Bicriteria network design problems. *J. of Algorithms*, 28(1):142–171, 1998.

[19] R. Ravi, M. Marathe, S. Ravi, D. Rosenkrantz, and H. Hunt III. Many birds with one stone: Multi-objective approximation algorithms. In *Proc. of the 25th ACM Symp. on the Theory of Computing*, pp. 438–447, 1993.

[20] R. Ravi, R. Sundaram, M. Marathe, D. Rosenkrantz, and S. Ravi. Spanning trees short or small. In *Proc. 5th ACM-SIAM Symp. on Discrete Algorithms*, pp. 546–555, 1994.

[21] R. Raz and S. Safra. A sub-constant error-probability low-degree test, and a sub-constant error-probability PCP characterization of NP. In *Proceedings of the 29th ACM Symp. on Theory of Computing*, pp. 475–484, 1997.