Parameterized Complexity of Directed Spanner Problems

Fedor V. Fomin
Department of Informatics, University of Bergen, Norway.
Fedor.Fomin@uib.no

Petr A. Golovach
Department of Informatics, University of Bergen, Norway.
Petr.Golovach@uib.no

William Lochet
Department of Informatics, University of Bergen, Norway.
William.Lochet@uib.no

Pranabendu Misra
MPI Informatik, Saarbrücken, Germany
pmisra@mpi-inf.mpg.de

Saket Saurabh
Institute of Mathematical Sciences, HBNI, Chennai, India, and Department of Informatics, University of Bergen, Norway.
saket@imsc.res.in

Roohani Sharma
Institute of Mathematical Sciences, HBNI, Chennai, India
roohani@imsc.res.in

Abstract

We initiate the parameterized complexity study of minimum t-spanner problems on directed graphs. For a positive integer t, a multiplicative t-spanner of a graph G is a spanning subgraph H such that the distance between any two vertices in H is at most t times the distance between these vertices in G, that is, H keeps the distances in G up to the distortion (or stretch) factor t. An additive t-spanner is defined as a spanning subgraph that keeps the distances up to the additive distortion parameter t, that is, the distances in H and G differ by at most t. The task of Directed Multiplicative Spanner is, given a directed graph G with m arcs and positive integers t and k, decide whether G has a multiplicative t-spanner with at most $m - k$ arcs. Similarly, Directed Additive Spanner asks whether G has an additive t-spanner with at most $m - k$ arcs. We show that

- Directed Multiplicative Spanner admits a polynomial kernel of size $\mathcal{O}(k^4t^5)$ and can be solved in randomized $(4t)^k \cdot n^{\mathcal{O}(1)}$ time,
- Directed Additive Spanner is W[1]-hard when parameterized by k even if $t = 1$ and the input graphs are restricted to be directed acyclic graphs.

The latter claim contrasts with the recent result of Kobayashi from STACS 2020 that the problem for undirected graphs is FPT when parameterized by t and k.

2012 ACM Subject Classification Mathematics of computing → Graph algorithms; Theory of computation → Parameterized complexity and exact algorithms

Keywords and phrases Graph spanners, directed graphs, parameterized complexity, kernelization

Digital Object Identifier 10.4230/LIPIcs.CVIT.2016.23

Funding *Fedor V. Fomin*: Supported by the Research Council of Norway via the project "MULTIVAL" (grant no. 263317).
Petr A. Golovach: Supported by the Research Council of Norway via the project "MULTIVAL" (grant no. 263317).

42nd Conference on Very Important Topics (CVIT 2016).
Editors: John Q. Open and Joan R. Access; Article No. 23; pp. 23:1–23:11
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Saket Saurabh: Supported by the European Research Council (ERC) via grant LOPPRE, reference 819416.

1 Introduction

Given a (directed) graph G, a *spanner* is a spanning subgraph of G that approximately preserves distances between the vertices of G. Graph spanners were formally introduced by Peleg and Schäffer in [14] (see also [15]). Originally, the concept was introduced for constructing network synchronizers [15]. However, graph spanners have a plethora of theoretical and practical applications in various areas like efficient routing and fast computing of shortest paths in networks, distributed computing, robotics, computational geometry and biology. We refer to the recent survey of Ahmed et al. [1] for the introduction to graph spanners and their applications.

We are interested in the classical *multiplicative* and *additive* graphs spanners in unweighted graphs. Let G be a (directed) graph. For two vertices $u, v \in V(G)$, $\text{dist}_G(u, v)$ denotes the *distance* between u and v in G, that is, the number of edges (arcs, respectively, for the directed case) of a shortest (u, v)-path. Let t be a positive integer. It is said that a spanning subgraph H of G is a *multiplicative t-spanner* if $\text{dist}_H(u, v) \leq t \cdot \text{dist}_G(u, v)$, i.e., H approximates distances in G within factor t. A spanning subgraph H of G is called an *additive t-spanner* if $\text{dist}_H(u, v) \leq \text{dist}_G(u, v) + t$, that is, H approximates the distances in G within the additive parameter t. The standard task in the graph spanner problems is, given an allowed distortion parameter t, find a sparsest t-spanner, i.e., a spanner with the minimum number of edges. We consider the parameterized versions of this task:

MULTIPLICATIVE SPANNER parameterized by $k + t$

Input: A (directed) graph G and integers $t \geq 1$ and $k \geq 0$.
Task: Decide whether there is a multiplicative t-spanner H with at most $|E(G)| - k$ edges (arcs, respectively).

and

ADDITIVE SPANNER parameterized by $k + t$

Input: A (directed) graph G and nonnegative integers t and k.
Task: Decide whether there is an additive t-spanner H with at most $|E(G)| - k$ edges (arcs, respectively).

Informally, the task of these problems is to decide whether we can delete at least k edges (arcs, respectively, for the directed case) in such a way that all the distances in the obtained graph are t-close to the original ones.

Previous work. We refer to [1] for the comprehensive survey of the known results and mention here only these that directly concern our work. First, we point that the considered graph spanner problems are computationally hard. It was already shown by Peleg and Schäffer in [14] that deciding whether a undirected graph G has a multiplicative t-spanner with at most m edges is NP-complete even for fixed $t = 2$. In fact, the problem is NP-complete for every fixed $t \geq 2$ [2]. Moreover, for every $t \geq 2$, it is NP-hard to approximate the minimum number of edges of a multiplicative t-spanner within the factor $c \log n$ for some $c > 1$ [10]. The same complexity lower bounds for directed graphs were also shown by Cai [2] and Kortsarz [10]. Additive t-spanners for undirected graphs were introduced by Liestman and Shermer in [11, 12]. In particular, they proved in [12], that for every fixed $t \geq 1$, it is NP-complete to decide whether a graph G admits an additive t-spanner with at most m edges. It was shown by Chlamtác et al. [4] that for every integer $t \geq 1$ and any constant $\varepsilon > 0$, there is no polynomial-time $2^{\log^{1-\varepsilon}/t^3}$-approximation for the minimum number of edges of an additive t-spanner unless $\mathsf{NP} \subseteq \mathsf{DTIME}(2^{\text{polylog(n)}})$.

The aforementioned hardness results make it natural to consider these spanner problems in the parameterized complexity framework. The investigation of Multiplicative Spanner and Additive Spanner on undirected graphs was initiated by Kobayashi in [8] and [9]. In [8], it was proved that Multiplicative Spanner admits a polynomial kernel of size $\mathcal{O}(k^2t^2)$. For Additive Spanner, it was shown in [9] that the problem can be solved in time $2^{\mathcal{O}((k^2+kt)\log t)} \cdot n^{\mathcal{O}(1)}$, that is, the problem is FPT when parameterized by k and t.

Our results. We initiate the study of Multiplicative Spanner and Additive Spanner on directed graphs and further refer to them as Directed Multiplicative Spanner and Directed Additive Spanner, respectively. We show that Directed Multiplicative Spanner admits a kernel of size $\mathcal{O}(k^4t^5)$. We complement this result by observing that the problem can be solved in $(4t)^k \cdot n^{\mathcal{O}(1)}$ time by a Monte Carlo algorithm with false negatives. Then we prove that Directed Additive Spanner becomes much harder on directed graphs by showing that the problem is W[1]-hard even when $t = 1$ and the input graphs are restricted to be directed acyclic graphs (DAGs).

Organization of the paper. In Section 2, we introduce basic notions used in the paper. In Section 3, we prove that Directed Multiplicative Spanner admits a polynomial kernel and sketch an FPT algorithm. In Section 4, we show hardness for Directed Additive Spanner. We conclude in Section 5 by stating some open problems.

2 Preliminaries

Parameterized Complexity and Kernelization. We refer to the recent books [5, 6, 7] for the detailed introduction. In the Parameterized Complexity theorey, the computational complexity is measured as a function of the input size n of a problem and an integer *parameter* k associated with the input. A parameterized problem is said to be *fixed-parameter tractable* (or FPT) if it can be solved in time $f(k) \cdot n^{\mathcal{O}(1)}$ for some function f. A *kernelization* algorithm for a parameterized problem Π is a polynomial algorithm that maps each instance (I, k) of Π to an instance (I', k') of Π such that

(i) (I, k) is a yes-instance of Π if and only if (I', k') is a yes-instance of Π, and
(ii) $|I'| + k'$ is bounded by $f(k)$ for a computable function f.

Respectively, (I', k') is a *kernel* and f is its *size*. A kernel is *polynomial* if f is polynomial. It is common to present a kernelization algorithm as a series of *reduction rules*. A reduction rule for a parameterized problem is an algorithm that takes an instance of the problem and computes in polynomial time another instance that is more "simple" in a certain way. A reduction rule is *safe* if the computed instance is equivalent to the input instance.

Graphs. Recall that an undirected graph is a pair $G = (V, E)$, where V is a set of vertices and E is a set of unordered pairs $\{u, v\}$ of distinct vertices called *edges*. A directed graph $G = (V, A)$ is a pair, where V is a set of vertices and A is a set of ordered pairs (u, v) of vertices called *arcs*; note that we allow $u = v$, i.e., D can have *loops*. We use $V(G)$ and $E(G)$ ($A(G)$, respectively) to denote the set of vertices and the set of edges (set of arcs, respectively). For a (directed) graph G and a subset $X \subseteq V(G)$ of vertices, we write $G[X]$ to denote the subgraph of G induced by X. For a set of vertices S, $G - S$ denotes the (directed) graph obtained by deleting the vertices of S, that is, $G - S = G[V(G) \setminus S]$; for a vertex v, we write $G - v$ instead of $G - \{v\}$. Similarly, for a set of edges (arcs, respectively) S (an edge or arc e, respectively), $G - S$ ($G - e$, respectively) denotes the graph obtained by the deletion of the elements of S (the deletion of e, respectively). A (directed) graph H is a spanning subgraph of G if $V(G) = V(H)$. We write $P = v_1 \cdots v_k$ to denote a *path* with the vertices $v_1, \ldots, v_k$ and the edges (arcs, respectively) $\{v_1, v_2\}, \ldots, \{v_{i-k}, v_k\}$; v_1 and v_k are the *end-vertices* of P and we say that P is an *(v_1, v_k)-path*. The *length* of the path is the number of edges (arcs, respectively). For a (u, v)-path P_1 and a (v, w)-path P_2, we denote by $P_1 \circ P_2$ the *concatenation* of P_1 and P_2. We use similar notation for walks. For two vertices $u, v \in V(G)$, $\text{dist}_G(u, v)$ denotes the *distance* between u and v in G, that is, the length of a shortest (u, v)-path; we assume that $\text{dist}_G(u, v) = +\infty$ if there is no (u, v)-path in G. Let t be a positive integer. It is said that a spanning subgraph H of G is a *multiplicative t-spanner* if $\text{dist}_H(u, v) \leq t \cdot \text{dist}_G(u, v)$. A spanning subgraph H of G is called an *additive t-spanner* if $\text{dist}_H(u, v) \leq \text{dist}_G(u, v) + t$.

3 Directed multiplicative t-spanners

In this section, we consider Directed Multiplicative Spanner. We show that the problem admits a polynomial kernel and then complement this result by obtaining an FPT algorithm. These results are based on *locality* of multiplicative spanners in the sense of the following folklore observation.

▶ **Observation 1.** *Let t be a positive integer. A spanning subgraph H of a directed graph G is a multiplicative t-spanner if and only if for every arc $(u, v) \in A(G)$, there is a (u, v)-path in H of length at most t.*

Let t be a positive integer and let G be a directed graph. For an arc $a = (u, v)$ of G, we say that a (u, v)-path P is an *t-detour* for a if the length of P is at most t and P does not contain a. By Observation 1, to solve Directed Multiplicative Spanner for (G, t, k), it is necessary and sufficient to identify k arcs that have t-detours that do not contain selected arcs. Then H can be constructed by deleting these arcs.

3.1 Polinomial kernel for Directed Multiplicative Spanner

In this subsection, we show that Directed Multiplicative Spanner admits a polynomial kernel.

▶ **Theorem 1.** Directed Multiplicative Spanner *has a kernel of size $\mathcal{O}(k^4 t^5)$.*

Proof. Let (G, t, k) be an instance of Directed Multiplicative Spanner.

Notice that loops do not contribute to the distances between vertices and, therefore, can be deleted without changing the distances. This gives the following straightforward reduction rule.

▶ **Reduction Rule 1.** *If G has a loop a, then set $G := G - a$ and $k := k - 1$.*

We apply the rule exhaustively and stop if $k = 0$ by the following rule.

▶ **Reduction Rule 2.** *If $k = 0$, then return a trivial yes-instance of* Directed Multiplicative Spanner *and stop.*

From now we assume that this is not the case, that is, from now G is a graph without loops and $k > 0$.

We say that $a \in A(G)$ is a t-*good* if G has a t-detour for a. Let S be the set of t-good arcs. Clearly, S can be constructed in polynomial time by making use of Dijkstra's algorithm. We follow the idea of Kobayashi [8] for constructing a polynomial kernel for undirected case and show that if S is sufficiently big, then (G, t, k) is a yes-instance of Directed Multiplicative Spanner.

▷ Claim 2. If $|S| \geq \frac{1}{2}k(t+1)((k-1)t+2)$, then (G, t, k) is a yes-instance of Directed Multiplicative Spanner.

Proof of Claim 2. Let $|S| \geq \frac{1}{2}k(t+1)((k-1)t+2)$. For every $a \in S$, let P_a be a t-detour for a.

Let $S_0 = \emptyset$. For $i = 1, \ldots, k$, we iteratively construct sets of arcs $S_1, \ldots, S_k$ such that

$$S_0 \subset S_1 \subset \cdots \subset S_k \subseteq S$$

and sets of arcs R_i such that $R_i \subseteq S_i \setminus S_{i-1}$ and $|R_i| = (k-i)t + 1$ for $i \in \{1, \ldots, k\}$ using the following procedure. For $i = 1, \ldots, k$,

- select an arbitrary set R_i of size $(k-i)t+1$ in $S \setminus S_{i-1}$,
- set $S_i = S_{i-1} \cup \{(A(P_a) \cap S) \cup \{a\} \mid a \in R_i\}$.

We show by induction, that the sets $S_1, \ldots, S_k$ and $R_1, \ldots, R_k$ exist. Since $|S \setminus S_0| = |S| \geq (k-1)t+1$, we conclude that R_1 of size $(k-1)t+1$ can be selected. Assume that the sets S_j and R_j have been constructed for $0 \leq j < i \leq k$. Observe that because $|\{(A(P_a) \cap S) \cup \{a\} \mid a \in R_j\}| \leq (t+1)|R_j|$,

$$|S_j \setminus S_{j-1}| \leq |R_j|(t+1) = ((k-j)t+1)(t+1)$$

for $1 \leq j < i$. Therefore,

$$|S_{i-1}| \leq \sum_{j=1}^{i-1}(((k-j)t+1)(t+1)). \tag{1}$$

Notice that

$$\frac{1}{2}k(t+1)((k-1)t+2) = \sum_{j=1}^{k}(((k-j)t+1)(t+1)). \tag{2}$$

Then by (1) and (2),

$$|S \setminus S_{i-1}| \geq \sum_{j=i}^{k}(((k-j)t+1)(t+1)) \geq (k-i)t+1.$$

This means that R_i can be selected and we can construct S_i.

Now we select arcs $a_i \in R_i$ for $i = k, k-1, \ldots, 1$. Since $|R_k| = 1$, the choice of a_k is unique. Assume that $a_k, \ldots, a_{i+1}$ have been selected for $1 < i+1 \leq k$. Then we select an arbitrary

$$a_i \in R_i \setminus \{A(P_{a_j}) \mid i+1 \leq j \leq k\}.$$

Because $|\{A(P_{a_j}) \mid i+1 \leq j \leq k\}| \leq (k-i)t$ and $|R_i| = (k-i)t+1$, a_i exists.

Let $i \in \{1, \ldots, k\}$. By the choice of a_i, we have that $a_i \notin A(P_{a_j})$ for $i < j \leq k$. From the other side, $a_i \notin A(P_j)$ for $1 \leq j < i$, because $a_i \in R_i$ and R_i does not contain the arcs of P_a for $a \in R_j$ for $1 \leq j < i$ by the construction of the sets $R_1, \ldots, R_k$. We obtain that the t-detours P_{a_i} for $i \in \{1, \ldots, k\}$ do not contain any a_j for $j \in \{1, \ldots, k\}$. By Observation 1, $H = G - \{a_1, \ldots, a_k\}$ is a multiplicative t-spanner. Therefore, (G, t, k) is a yes-instance of Directed Multiplicative Spanner. ◀

By Claim 2, we can apply the next rule:

▶ **Reduction Rule 3.** *If $|S| \geq \frac{1}{2}k(t+1)((k-1)t+2)$, then return a trivial yes-instance of* Directed Multiplicative Spanner *and stop.*

From now, we assume that $|S| < \frac{1}{2}k(t+1)((k-1)t+2)$.

The analog of Reduction Rule 3 is a main step of the kernelization algorithm of Kobayashi [8] for the undirected case, because it almost immediately allows to upper bound the total number of edges of the graph. However, the directed case is more complicated, since the arcs of t-detours for $a \in S$ may be outside S contrary to the undirected case, where all the edges of t-detours are in cycles of length at most $t+1$ and, therefore, have t-detours themselves. We use the following procedure to mark the crucial arcs of potential detours.

Marking Procedure. Let $G' = G - S$.

(i) For every $(u, v) \in S$, find a shorted (u, v)-path P in G' and if the length of P is at most t, then *mark* the arcs of P.

(ii) For every ordered pair of two distinct arcs $(u_1, v_1), (u_2, v_2) \in S$,

  (a) find a shortest (u_1, u_2)-path P_1 in G' and if the length of P_1 is at most t, then *mark* the arcs of P_1,

  (b) find a shortest (v_2, v_1)-path P_2 in G' and if the length of P_2 is at most t, then *mark* the arcs of P_2,

  (c) find a shortest (v_1, u_2)-path P_3 in G' and if the length of P_3 is at most t, then *mark* the arcs of P_3.

Observe that marking can be done in polynomial time by Dijkstra's algorithm. Denote by L the set of marked arcs. Our final rule constructs the output instance.

▶ **Reduction Rule 4.** *Consider the graph $H = (V(G), S \cup L)$. Delete the isolated vertices of H, and for the obtained G^*, output (G^*, t, k).*

We argue that the rule is safe.

▷ Claim 3. (G, t, k) is a yes-instance of Directed Multiplicative Spanner if and only if (G^*, t, k) is a yes-instance.

Proof of Claim 3. Suppose that (G, t, k) is a yes-instance of Directed Multiplicative Spanner. Then, by Observation 1, there are k distinct arcs $a_1, \ldots, a_k \in S$ with their t-detours $P_1, \ldots, P_k$, respectively, such that $a_i \notin \bigcup_{j=1}^{k} A(P_j)$. Notice that $a_1, \ldots, a_k \in A(G^*)$. Consider $i \in \{1, \ldots, k\}$ and let $a_i = (u, v)$.

Suppose that P_i does not contain arcs from S. Then P_i is a (u, v)-path in $G' = G - S$. By the first step of Marking Procedure, there is a t-detour P'_i for a_i whose arc are in G' and are marked. Then P'_i is a t-detour for a_i in G^* and $a_j \notin A(P'_i)$ for $j \in \{1, \ldots, k\}$.

Assume that P_i contains some arcs from S. Let $e_1, \ldots, e_s$ be these arcs (in the path order with respect to P_i starting from u). Note that $e_1, \ldots, e_s \in A(G^*)$ and they are distinct from $a_1, \ldots, a_k$. Let $e_j = (x_j, y_j)$ for $j \in \{1, \ldots, s\}$. Then P_i can be written as the concatenation of the paths $P_i = Q_1 \circ x_1 y_1 \circ Q_2 \circ \cdots \circ x_s y_s \circ Q_{s+1}$, where Q_1 is the (u, x_1)-subpath of P_i,

Q_j is the (y_{j-1}, x_j)-subpath of P_i for $j \in \{2, \dots, s\}$, and Q_{s+1} is the (y_s, v)-subpath of P_i; note that some of the paths $Q_1, \dots, Q_{s+1}$ may be trivial, i.e., contain a single vertex. Let $j \in \{1, \dots, s+1\}$. If Q_j is trivial, then $Q'_j = Q_j$ is a path in G^*, because the vertices incident to the arcs of S are vertices of G^*. Suppose that Q_j is not trivial. If $j = 1$, then by step (ii)(a) of Marking Procedure, there is a (u, x_1)-path Q'_1, whose arcs are in G' and are marked, and the length of Q'_1 at at most the length of Q_1. For $j = s+1$, we have that by step (ii)(b), there is a (y_s, v)-path Q'_{s+1}, whose arcs are in G' and are marked, and the length of Q'_{s+1} is at most the length of Q_{s+11}. Suppose that $2 \le j \le s$. Then by step (ii)(c), there is a (y_{j-1}, x_j)-path Q'_j, whose arcs are in G' and are marked, and the length of Q'_j is at most the length of Q_j. Consider the (u, v)-walk $W_i = Q'_1 \circ x_1 y_1 \circ Q'_2 \circ \dots \circ x_s y_s \circ Q'_{s+1}$. We have that W'_i is a (u, v)-walk of length at most t in G^* such that $a_j \notin A(W_i)$ for $j \in \{1, \dots, k\}$. This implies that G^* has a t-detour P'_i in G^* such that $a_j \notin A(P'_i)$ for $j \in \{1, \dots, k\}$.

We obtain that for every $i \in \{1, \dots, k\}$, $a_i \in A(G^*)$ has a t-detour P'_i such that $a_1, \dots, a_k \notin A(P'_i)$. By Observation 1, we conclude that $G^* - \{a_1, \dots, a_k\}$ is a multiplicative spanner for G^*, that is, (G^*, t, k) is a yes-instance of Directed Multiplicative Spanner.

For the opposite direction, assume that (G^*, t, k) is a yes-instance of Directed Multiplicative Spanner. By Observation 1, there are k distinct arcs $a_1, \dots, a_k \in A(G^*)$ with their t-detours $P_1, \dots, P_k$, respectively, such that $a_i \notin \bigcup_{j=1}^{k} A(P_j)$. Since G^* is a subgraph of G, $a_1, \dots, a_k$ have the same t-detours in G. By Observation 1, (G, t, k) is a yes-instance. ◀

To upper bound the size of G^*, observe that Marking Procedure marks at most t arcs for each $a \in S$ in step (i), that is, at most $|S|t$ arcs are marked in this step. In step (ii), we mark at most $3t$ arcs for each ordered pair of arcs of S. Hence, at most $3|S|(|S|-1)t$ arc are marked in total in the second step. Since $|S| < \frac{1}{2}k(t+1)((k-1)t+2)$, we have that G^* has $\mathcal{O}(k^4 t^5)$ arcs. Because G^* has no isolated vertices, the number of vertices is $\mathcal{O}(k^4 t^5)$.

Since each of the reduction rules and Marking Procedure can be done in polynomial time, we conclude that the total running time of our kernelization algorithm is polynomial. ◀

3.2 FPT algorithm for Directed Multiplicative Spanner

Combining Theorem 1 with the brute-force procedure that guesses k arcs of G and verifies whether the deletion of these arcs gives a multiplicative t-spanner, we obtain the straightforward $2^{\mathcal{O}(k \log(kt))} + n^{\mathcal{O}(1)}$ algorithm for Directed Multiplicative Spanner. If we use the intermediate steps of the kernelization algorithm, then the running time may be improved to $(kt)^{2k} \cdot n^{\mathcal{O}(1)}$. Namely, we can execute Reduction Rules 1–3 of the kernelization algorithm. Then we either solve the problem or obtain an instance, where the set S of t-good arcs has size at most $\frac{1}{2}k(t+1)((k-1)t+2) - 1 \le k^2 t^2$. Then for every $R \subseteq S$ of size k, we check whether $G - R$ is a multiplicative t-spanner by computing the distances between every pair of vertices. However, we can slightly improve the parameter dependence by making use of the *random separation* technique proposed by Cai, Chan, and Chan in [3] (we refer to [5, Chapter 5] for the detailed introduction to the technique). In this subsection, we briefly sketch a Monte Carlo algorithm with false negatives for Directed Multiplicative Spanner.

▶ **Theorem 4.** Directed Multiplicative Spanner *can be solved in time* $(4t)^k \cdot n^{\mathcal{O}(1)}$ *by a Monte Carlo algorithm with false negatives.*

Proof. Let (G, t, k) be an instance of Directed Multiplicative Spanner. In the same way as in the proof of Theorem 1, we can assume that G has no loops. Otherwise, we iteratively delete loops and decrease the parameter k. If $k = 0$ or $t = 1$, then the problem is trivial: if $k = 0$, then (G, t, k) is a yes-instance, and if $k > 0$ and $t = 1$, then (G, t, k) is a no-instance, because G has no loops. From now we assume that $k \geq 1$ and $t \geq 2$.

By Observation 1, to solve Directed Multiplicative Spanner for (G, t, k), it is necessary and sufficient to identify k arcs that have t-detours that do not contain selected arcs. We use random separation to distinguish the arcs that have t-detours and the arcs of the detours. We randomly color the arcs of G by two colors *red* and *blue*. An arc is colored red with probability $\frac{1}{t}$ and is colored blue with probability $\frac{t-1}{t}$. Then we try to find k red arcs that have t-detours composed by blue arcs. Let R be the set of arcs colred red and let B the set of blue arcs. For $(u, v) \in R$, it can be checked in polynomial time whether (u, v) has a t-detour with blue arcs by finding the distance between u and v in $G_B = (V(G), B)$. Then we greedily construct the set S of all red arcs with blue t-detours. If $|S| \geq k$, then we conclude that (G, t, k) is a yes-instance by Observation 1.

Suppose that (G, t, k) is a yes-instance of Directed Multiplicative Spanner. Then by Observation 1, there are k distinct arcs $a_1, \ldots, a_k$ and their t-detours $P_1, \ldots, P_k$, respectively, such that $a_1, \ldots, a_k \notin L = \bigcup_{i=1}^{k} A(P_i)$. Notice that $|L| \leq tk$. Then the probability that the considered random coloring colors the arcs $a_1, \ldots, a_k$ red is at least t^{-k} and the probability that the arcs of L are colored blue is at least $(\frac{t-1}{t})^{tk}$. We have that

$$\left(\frac{t-1}{t}\right)^t = \left(1 - \frac{1}{t}\right)^t \geq \frac{1}{4}.$$

Therefore, the probability that the arcs $a_1, \ldots, a_k$ are red and their t-detours are blue is at least $(4t)^{-k}$. Respectively, the probability that the random coloring fails to color the arcs $a_1, \ldots, a_k$ red and their t-detours blue is at most $1 - \frac{1}{(4t)^k}$. This implies that if we iterate our algorithm for $(4t)^k$ colorings, then we either find a solution and stop or we conclude that (G, t, k) is a no-instance with the mistake probability at most $\left(1 - \frac{1}{(4t)^k}\right)^{(4t)^k} \leq e^{-1}$. This gives us a Monte Carlo algorithm with running time $(4t)^k \cdot n^{\mathcal{O}(1)}$. ◀

The same approach can be used for undirected graphs and it can be shown that Multiplicative Spanner can be solved in $(4t)^k \cdot n^{\mathcal{O}(1)}$ time improving the running time given in [8].

The algorithm from Theorem 4 can be derandomized by using *universal sets* [13] instead of random colorings. Since this part is standard (see [5, Chapter 5]), we leave it to the interested readers.

4 Directed additive t-spanners

In this section, we consider Directed Additive Spanner and show that the problem is hard on DAGs even if $t = 1$.

▶ **Theorem 5.** Directed Additive Spanner *is* W[1]*-hard on DAGs when parameterized by k only even if $t = 1$.*

Proof. We reduce from the Independent Set problem. Given a graph G and a positive integer k, the problem asks whether G has an independent set of size at least k. Independent Set parameterized k is well-known to be one of the basic W[1]-complete problems (see [5, 6]).

Let (G, k) be an instance of Independent Set. Denote by $v_1, \ldots, v_n$ the vertices of G.

- For every $i \in \{1, \ldots, n\}$, construct three vertices x_i, y_i, z_i and arcs $(x_i, y_i), (y_i, z_i), (x_i, z_i)$.
- For every $i, j \in \{1, \ldots, n\}$ such that $i < j$, do the following:

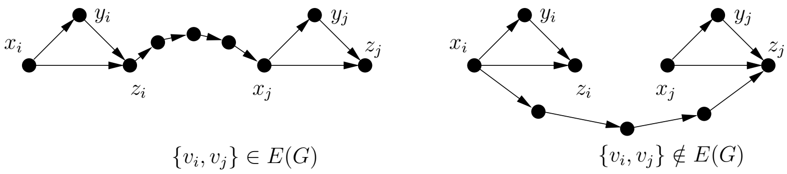

Figure 1 Construction of D.

- if $\{v_i, v_j\} \in E(G)$, then construct a directed (z_i, x_j)-path P_{ij} of length 4,
- if $\{v_i, v_j\} \notin E(G)$, then construct a directed (x_i, z_j)-path Q_{ij} of length 4.

Denote the obtained directed graph by D (see Figure 1). It is straightforward to verify that D is a DAG. We show that (G, k) is a yes-instance of Independent Set if and only if $(D, 1, k)$ is a yes-instance of Directed Additive Spanner.

Suppose that $I = \{v_{i_1}, \ldots, v_{i_k}\}$ is an independent set of size k in G. Let $R = \{(x_{i_1}, z_{i_1}), \ldots, (x_{i_k}, z_{i_k})\}$. We show that $D' = D - R$ is an additive t-spanner for D.

We claim that for every two vertices u and w of D, each shortest (u, w)-path in D contains at most one arc of R. The proof is by contradiction. Assume that there are $u, w \in V(D)$ and a shortest (u, w)-path P such that P contains at least two arcs of R. Let (x_i, z_i) and (x_j, z_j) be such arcs and let $i < j$. By the construction, (x_i, z_i) occurs before (x_j, z_j) in P. Since the arcs of R correspond to vertices of the independent set I, v_i and v_j are not adjacent in G. Therefore, D contains the (x_i, z_j)-path Q_{ij} of length 4. Since P is a shortest path containing (x_i, z_i) and (x_j, z_j), the (z_i, x_j)-subpath of P should have length at most 2. However, by the construction, the distance between z_i and x_j is at least 4; a contradiction proving the claim.

Now let u and w be two vertices of D. Let P be a shortest (u, w)-path in D. If P is a path in D', then $\text{dist}_{D'}(u, w) = \text{dist}_D(u, w)$. Suppose that P is not a path in D'. Then P contains a unique arc $(x_i, z_i) \in R$ by the proved claim. Let P_1 be the (u, x_i)-subpath of P and let P_2 be the (z_i, w)-subpath. Let $P' = P_1 \circ x_i y_i z_u \circ P_2$. Observe that P' is a path in D'. Since the length of P' is the length of P plus 1, $\text{dist}_{D'}(u, w) \leq \text{dist}_D(u, w) + 1$. This implies that D' is an additive 1-spanner of D.

Now we assume that $(D, 1, s)$ is a yes-instance of Directed Additive Spanner. Then there is a set of k arcs $R \subseteq A(D)$ such that $D' = D - R$ is an additive 1-spanner. Observe that if $(u, v) \in R$, then D has an (u, v)-path P. Otherwise, $\text{dist}_{D'}(u, v) = +\infty$ and $\text{dist}_{D'}(u, v) > \text{dist}_D(u, v) + 1$. Therefore, $R \subseteq \{(x_1, z_1), \ldots, (x_n, z_n)\}$. Let $R = \{(x_{i_1}, z_{i_1}), \ldots, (x_{i_k}, z_{i_k})\}$. We claim that $I = \{v_{i_1}, \ldots, v_{i_k}\}$ is an independent set of G. Assume that this is not the case and there are $v_i, v_j \in I$ such that v_i and v_j are adjacent in G. Let $i < j$. Consider the vertices x_j and z_j of D. Since $\{v_i, v_j\} \in E(G)$, $P = x_i z_i \circ P_{ij} \circ x_j z_j$ is an (x_i, z_j)-path of length 6, that is, $\text{dist}_D(x_i, z_j) \leq 6$. The path $P' = x_i y_i z_i \circ P_{ij} \circ x_j y_j z_j$ has length 8. Any other (x_i, z_j)-path in D' uses at least two paths of length 4: one of the paths $P_{ii'}$ and $Q_{ii'}$ for some $i' \in \{1, \ldots, n\}$ such that $i' \neq j$, and one of the paths $P_{j'j}$ and $Q_{j'j}$ for some $j' \in \{1, \ldots, n\}$ such that $j' \neq i$. This means that $\text{dist}_{D'}(x_i, z_j) - \text{dist}_D(x_i, x_j) \geq 2$ contradicting that D' is an additive 1-spanner. We conclude that I is an independent set of G and, therefore, (G, k) is a yes-instance of Independent Set. ◀

5 Conclusion

We proved that Directed Multiplicative Spanner admits a kernel of size $\mathcal{O}(k^4 t^5)$ can be solved in $(4t)^k \cdot n^{\mathcal{O}(1)}$ randomized time. We also demonstrated that Directed Additive

SPANNER is W[1]-hard even when $t = 1$ and the input graphs are restricted to DAGs. The latter result leads to the question whether DIRECTED ADDITIVE SPANNER is tractable on some special classes of directed graphs, like planar directed graphs. We believe that this problem may be interesting even if the distortion parameter t is assumed to be a constant.

Another possible direction of research is considering different types of directed graph spanners. For example, what can be said about the roundtrips spanners introduced by Roditty, Thorup, and Zwick [16]? A spanning subgraph H of a directed graph G is a multiplicative *t-roundtrip-spanner* if for every two vertices u and v, $\mathrm{dist}_H(u,v)+\mathrm{dist}_H(v,u) \leq t(\mathrm{dist}_G(u,v) + \mathrm{dist}_G(v,u))$, that is, H approximates the sum of the distances between any two vertices in both directions. Is the analog of DIRECTED MULTIPLICATIVE SPANNER for roundtrip spanners FPT? Notice that we cannot use Observation 1 that is crucial for our results for the new problem. Consider, for example, the directed graph G constructed as follows: construct two vertices u and v and an arc (u, v), and then add a (u, v)-path P_1 and a (v, u)-path P_2 of arbitrary length $\ell \geq 2$ that are internally vertex disjoint. Then it is easy to see that $H = G - (u, v)$ is a 2-roundtrip spanner for G. However, H has no short detour for (u, v). It also possible to define additive t-roundtrip-spanners and consider the analog of DIRECTED ADDITIVE SPANNER. We conjecture that this problem is at least as hard as DIRECTED ADDITIVE SPANNER.

References

1 Abu Reyan Ahmed, Greg Bodwin, Faryad Darabi Sahneh, Keaton Hamm, Mohammad Javad Latifi Jebelli, Stephen G. Kobourov, and Richard Spence. Graph spanners: A tutorial review. *CoRR*, abs/1909.03152, 2019. URL: http://arxiv.org/abs/1909.03152, arXiv:1909.03152.

2 Leizhen Cai. NP-completeness of minimum spanner problems. *Discret. Appl. Math.*, 48(2):187–194, 1994. URL: https://doi.org/10.1016/0166-218X(94)90073-6, doi:10.1016/0166-218X(94)90073-6.

3 Leizhen Cai, Siu Man Chan, and Siu On Chan. Random separation: A new method for solving fixed-cardinality optimization problems. In *Parameterized and Exact Computation, Second International Workshop, IWPEC 2006, Zürich, Switzerland, September 13-15, 2006, Proceedings*, volume 4169 of *Lecture Notes in Computer Science*, pages 239–250. Springer, 2006. URL: https://doi.org/10.1007/11847250_22, doi:10.1007/11847250_22.

4 Eden Chlamtác, Michael Dinitz, Guy Kortsarz, and Bundit Laekhanukit. Approximating spanners and directed steiner forest: Upper and lower bounds. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2017, Barcelona, Spain, Hotel Porta Fira, January 16-19*, pages 534–553. SIAM, 2017. URL: https://doi.org/10.1137/1.9781611974782.34, doi:10.1137/1.9781611974782.34.

5 Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer, 2015. URL: https://doi.org/10.1007/978-3-319-21275-3, doi:10.1007/978-3-319-21275-3.

6 Rodney G. Downey and Michael R. Fellows. *Fundamentals of Parameterized Complexity*. Texts in Computer Science. Springer, 2013. URL: https://doi.org/10.1007/978-1-4471-5559-1, doi:10.1007/978-1-4471-5559-1.

7 Fedor V. Fomin, Daniel Lokshtanov, Saket Saurabh, and Meirav Zehavi. *Kernelization*. Cambridge University Press, Cambridge, 2019. Theory of parameterized preprocessing.

8 Yusuke Kobayashi. NP-hardness and fixed-parameter tractability of the minimum spanner problem. *Theor. Comput. Sci.*, 746:88–97, 2018. URL: https://doi.org/10.1016/j.tcs.2018.06.031, doi:10.1016/j.tcs.2018.06.031.

9 Yusuke Kobayashi. An FPT algorithm for minimum additive spanner problem. In Christophe Paul and Markus Bläser, editors, *37th International Symposium on Theoretical Aspects of*

Computer Science, STACS 2020, March 10-13, 2020, Montpellier, France, volume 154 of *LIPIcs*, pages 11:1–11:16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020. URL: https://doi.org/10.4230/LIPIcs.STACS.2020.11, doi:10.4230/LIPIcs.STACS.2020.11.

10 Guy Kortsarz. On the hardness of approximating spanners. *Algorithmica*, 30(3):432–450, 2001. URL: https://doi.org/10.1007/s00453-001-0021-y, doi:10.1007/s00453-001-0021-y.

11 Arthur L. Liestman and Thomas C. Shermer. Additive spanners for hypercubes. *Parallel Process. Lett.*, 1:35–42, 1991. URL: https://doi.org/10.1142/S0129626491000197, doi:10.1142/S0129626491000197.

12 Arthur L. Liestman and Thomas C. Shermer. Additive graph spanners. *Networks*, 23(4):343–363, 1993. URL: https://doi.org/10.1002/net.3230230417, doi:10.1002/net.3230230417.

13 Moni Naor, Leonard J. Schulman, and Aravind Srinivasan. Splitters and near-optimal derandomization. In *36th Annual Symposium on Foundations of Computer Science, Milwaukee, Wisconsin, USA, 23-25 October 1995*, pages 182–191. IEEE Computer Society, 1995. URL: https://doi.org/10.1109/SFCS.1995.492475, doi:10.1109/SFCS.1995.492475.

14 David Peleg and Alejandro A. Schäffer. Graph spanners. *Journal of Graph Theory*, 13(1):99–116, 1989. URL: https://doi.org/10.1002/jgt.3190130114, doi:10.1002/jgt.3190130114.

15 David Peleg and Jeffrey D. Ullman. An optimal synchronizer for the hypercube. *SIAM J. Comput.*, 18(4):740–747, 1989. URL: https://doi.org/10.1137/0218050, doi:10.1137/0218050.

16 Liam Roditty, Mikkel Thorup, and Uri Zwick. Roundtrip spanners and roundtrip routing in directed graphs. *ACM Trans. Algorithms*, 4(3):29:1–29:17, 2008. URL: https://doi.org/10.1145/1367064.1367069, doi:10.1145/1367064.1367069.