DAG Covers for Structured Graphs: The Steiner Point Effect

Sujoy Bhore
Indian Institute of Technology Bombay, India

Hsien-Chih Chang
Dartmouth College, Hanover, NH, USA

Jonathan Conroy
Dartmouth College, Hanover, NH, USA

Arnold Filtser
Bar-Ilan University, Ramat Gan, Israel

Eunjin Oh
POSTECH, Pohang, South Korea

Nicole Wein
University of Michigan, Ann Arbor, MI, USA

Da Wei Zheng
Institute of Science and Technology Austria, Klosterneuburg, Austria

Abstract

Given a weighted digraph G, a (t, g, μ)-DAG cover is a collection of g dominating DAGs $D_1, \ldots, D_g$ such that all distances are approximately preserved: for every pair (u, v) of vertices, $\min_i d_{D_i}(u, v) \leq t \cdot d_G(u, v)$, and the total number of non-G edges is bounded by $|(\cup_i D_i) \backslash G| \leq \mu$. Assadi, Hoppenworth, and Wein [STOC 25] and Filtser [SODA 26] studied DAG covers for general digraphs. This paper initiates the study of *Steiner* DAG cover, where the DAGs are allowed to contain Steiner points.

We obtain Steiner DAG covers on the important classes of planar digraphs and low-treewidth digraphs. Specifically, we show that any digraph with treewidth tw admits a $(1, 2, \tilde{O}(n \cdot tw))$-Steiner DAG cover. For planar digraphs we provide a $(1 + \varepsilon, 2, \tilde{O}_\varepsilon(n))$-Steiner DAG cover.

We also demonstrate a stark difference between Steiner and non-Steiner DAG covers. As a lower bound, we show that any non-Steiner DAG cover for graphs with treewidth 1 with stretch $t < 2$ and sub-quadratic number of extra edges requires $\Omega(\log n)$ DAGs.

2012 ACM Subject Classification Theory of computation → Computational geometry; Theory of computation → Graph algorithms analysis

Keywords and phrases Directed graphs, DAG (directed acyclic graphs), distortion, metric embeddings, planar graph, treewidth

Digital Object Identifier 10.4230/LIPIcs.ESA.2026.94

Related Version *Full Version*: `https://arxiv.org/abs/2604.04186`

Funding *Sujoy Bhore*: Work supported in part by ANRF ARG-MATRICS, Grant 002465.
Hsien-Chih Chang: Supported by the U.S. National Science Foundation Grant No. CCF-2443017.
Jonathan Conroy: Supported by the U.S. National Science Foundation Grant No. CCF-2443017.
Arnold Filtser: This research was supported by the ISRAEL SCIENCE FOUNDATION (grant No. 1042/22).
Eunjin Oh: Supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. RS-2024-00440239, Sublinear Scalable Algorithms for Large-Scale Data Analysis) and the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. RS-2024-00358505).
Nicole Wein: Supported by NSF CAREER award 2541910.
Da Wei Zheng: This project has received funding from the Austrian Science Fund (FWF) grant DOI 10.55776/I5982. For open access purposes, the author has applied a CC BY public copyright license to any author-accepted manuscript version arising from this submission.

Acknowledgements This work was initiated at Dagstuhl Seminar 25212: Metric Sketching and Dynamic Algorithms for Geometric and Topological Graphs. We thank the organizers and other participants for a productive environment.

1 Introduction

Tree covers are fundamental graph primitives with wide-ranging applications in network routing, approximate distance oracles, spanners, and related problems. They offer sparse and structurally simple representations of large graphs while retaining key distance relationships. Formally, a *stretch-t tree cover* of an edge-weighted undirected graph G is a collection of weighted trees $\mathcal{T}$ on vertex set V such that for every pair $u, v \in V$, (1) for every tree $T \in \mathcal{T}$, we have $d_G(u, v) \leq d_T(u, v)$, and (2) there exists some $T \in \mathcal{T}$ such that $d_T(u, v) \leq t \cdot d_G(u, v)$.

Tree covers were introduced 30 years ago by Arya, Das, Mount, Salowe, and Smid [5] in the context of Euclidean spaces and have been extensively studied ever since. In $\mathbb{R}^d$, they established stretch-$(1 + \varepsilon)$ tree cover with $O(\varepsilon^{-d} \log(1/\varepsilon))$ trees, which was later improved to $O(\varepsilon^{-d+1} \log(1/\varepsilon))$ [29]. For general graphs, tree covers were studied implicitly in [7, 8], and then in their seminal work Thorup and Zwick [55] showed a tree cover with stretch $2k - 1$ and $O\big(n^{1/k} \log n\big)$ trees, for any integer $k \geq 1$. Tree covers have found many of their applications in graphs with geometric or topological structure. In particular, tree covers have been extensively studied and applied in the context of Euclidean metrics [5, 23, 29], doubling metrics [12, 21, 34, 36], planar, bounded treewidth, and minor-free graphs [17, 22, 24, 28, 38].

In sharp contrast, directed graphs admit a far weaker repertoire of usable distance-preserving structures. Many sparsification primitives that play a central role in undirected graphs are either provably impossible to obtain in directed settings (e.g., sparse spanners [3] or efficient distance oracles [55]) or remain substantially less understood (e.g., distance preservers and hopsets). For directed hopsets, recent work has made substantial progress, yet polynomial gaps between upper and lower bounds persist, in contrast to the near-tight landscape for undirected counterparts (see, e.g., [16, 19, 27, 41, 42, 45, 56]).

Directed acyclic graphs (DAGs) are natural analogs of trees in directed graphs and often distance problems become more tractable on DAGs than general directed graphs. For instance, computing single-source shortest paths with negative weights is somewhat trivial in DAGs and achievable in linear time, yet for general directed graphs the problem remains extremely difficult. Distance preservers also exhibit a gap: the best bounds known for DAGs are strictly stronger than those for general directed graphs [18]. Even for hopsets, where upper bounds for the simpler shortcut-set problem extend readily in DAGs, analogous results are more elusive for general directed graphs [16, 45].

Motivated by this phenomenon, in a recent work [6], Assadi, Hoppenworth, and Wein introduced the notion of *DAG cover* analogous to a tree cover: informally, given a weighted directed graph G, a *DAG cover* is a collection of DAGs such that no DAG underestimates any distance, and every distance $d_G(u, v)$ is approximated by at least one of the DAGs (see Definition 1). They observe that a directed cycle constitutes a trivial lower bound if each DAG is required to be a subgraph of G. In this case n DAGs are needed even to preserve reachability. (From the upper bounds side, n DAGs can always trivially preserve distances exactly by taking a shortest paths tree from each vertex.) Thus, it is necessary to allow the addition of edges not from G. However, allowing quadratically many additional edges trivializes the problem: only two dense DAGs with opposite topological orders yield exact distances by adding all $\Theta(n^2)$ edges consistent with the ordering (with appropriate edge weights). Thus, the goal becomes to trade-off between three parameters: the number of DAGs, the stretch, and the number of additional edges. Building upon [6], Filtser [33] proved the best-known upper bound for general directed graphs: with $O(\log n)$ DAGs, it is possible to achieve stretch $\tilde{O}(\log n)$, with the addition of $\tilde{O}(m)$ edges. Building off this line of work, very recently Haeupler, Jiang, and Saranurak [40] defined the related notion of a *DAG projection*, which they used to reduce a number of problems from general graphs to DAGs. See Section 2 for additional related work.

1.1 Our Setting: Steiner DAG Covers for Structured Graphs

Given that DAG covers necessarily require additional *edges* not from G, it is only natural to consider allowing additional *vertices* not from G. Indeed, this question of *Steiner DAG cover* was explicitly asked in both [6] and [33]. We initiate the study of Steiner DAG covers.

We focus, in particular, on the important graph classes of *planar digraphs* and bounded *treewidth* digraphs (i.e., the underlying undirected graph has bounded treewidth). Our focus is motivated by the fact that some of the biggest success stories of tree covers are their applications to metrics with geometric or topological structure such as planar graphs and bounded treewidth graphs (in addition to e.g. minor-free graphs, Euclidean metrics, and doubling metrics). Applications in these settings include, for example, distance oracle, spanners/emulators, low-hop emulator, distance labeling schemes, routing schemes, additive metric embeddings (e.g. [5, 22, 24, 28, 36, 38, 43]).

For these special classes of graphs, one can achieve much better tree covers than in general graphs. This raises the question:

Can we obtain DAG covers for important special classes of graphs that are much better than those for general graphs (possibly with the use of Steiner vertices)?

Our main results are Steiner DAG covers for planar digraphs and bounded-treewidth digraphs that indeed have *far better* parameters than the best-known results for general digraphs: we obtain DAG covers with only *two* DAGs, and either 1 or $1 + \varepsilon$ stretch. (Note that it is impossible to achieve only a single DAG if G itself is not a DAG, even for preserving reachability.) In contrast, in general digraphs, there is a known *lower bound* that achieving stretch 1 with $\tilde{O}(m)$ additional edges requires a *polynomial* number of DAGs, and this is conjectured to extend to larger stretch [6]. Additionally, stronger lower bounds are known for general digraphs when the number of added edges is $\tilde{O}(n^{2-\varepsilon})$: even for preserving reachability, $\Omega(n^{1-\varepsilon})$ DAGs are required [6]. Next, we will state our results formally.

1.2 Our Results

First we state the formal definition of a DAG cover.

▶ **Definition 1** (DAG Cover). *Given a weighted digraph $G = (V, E, w)$, a (t, g, μ)-DAG cover is a collection $\mathbb{D}$ of g DAGs $D_1, \ldots, D_g$ over V such that:*

- [Dominating:] *for every $u, v \in V$ and DAG $D_i \in \mathbb{D}$, $d_G(u, v) \leq d_{D_i}(u, v)$.*
- [Stretch:] *for every reachable pair (u, v) it holds that $\min_{D_i \in \mathbb{D}} d_{D_i}(u, v) \leq t \cdot d_G(u, v)$.*
- [Sparsity:] *the total number of extra edges not in G ($|\cup_i E(D_i) \setminus E(G)|$) is at most μ.*[1]

If the DAGs in $\mathbb{D}$ contain only vertices in V, we call it a non-Steiner DAG cover. Otherwise if the DAGs contain vertices that are not in V, we call it a Steiner DAG cover.

Our results are shown in Table 1. We show a stark separation between Steiner to non-Steiner DAG covers. Specifically, in Theorem 2 we show an example of a planar digraph with treewidth 1 (the bidirected star) such that every non-Steiner DAG cover with stretch $t < 2$, and μ additional edges must use $\Omega\big(\log(\frac{n^2}{\mu+1})\big)$ DAGs. In particular, for sub-quadratic $\mu \leq n^{2-\Omega(1)}$, $\Omega(\log n)$ DAGs are required. We then show that this lower bound (Theorem 2) is tight (up to second order terms). Specifically, in Theorem 3 we show that any digraph with treewidth tw admits a non-Steiner $\big(1, O(\log n), \tilde{O}(n \cdot tw)\big)$-DAG cover. That is, for bounded treewidth digraphs, $O(\log n)$ DAGs without Steiner points are sufficient to preserve all distances exactly, while using near-linear number of additional edges.

[1] Note that if all the DAGs in the cover are subgraphs of G, then $\mu = 0$.

Table 1 Summary of our results on DAG cover, in comparison to the best-known results for general digraphs. Rows with Ω in the # DAGs column are lower bounds. Stretch "finite" means that the lower bound result holds even for preserving only reachability. The number of Steiner vertices in our constructions are asymptotically the same as the number of added edges.

Family	Steiner?	Stretch t	# DAGs g	# Added edges μ	Ref.
General digraphs	✗	$\tilde{O}(\log n)$	$O(\log n)$	$\tilde{O}(m)$	[6, 33]
	✗	finite	$\Omega(n^{1-\varepsilon})$	$O(n^{2-\varepsilon})$	[6]
	✗	1	$\Omega(n^{1/6})$	$O(m^{1+\varepsilon})$	[6]
Treewidth tw digraphs	✗	< 2	$\Omega(\log \frac{n^2}{\mu+1})$	μ	Theorem 2
	✗	1	$O(\log n)$	$\tilde{O}(n \cdot tw)$	Theorem 3
	✓	1	2	$\tilde{O}(n \cdot tw)$	Theorem 5
Planar digraphs	✓	$1+\varepsilon$	2	$\tilde{O}(n \cdot \varepsilon^{-1} \cdot \log \Phi)$	Theorem 7

Treewidth. In Theorem 5 we show that any digraph with treewidth tw admits a Steiner $(1, 2, \tilde{O}(n \cdot tw))$-DAG cover. That is, we preserve all distances exactly, while using only 2 DAGs, and near linear number of additional edges. Compare this to non-Steiner DAG covers that require $\Omega(\log n)$ DAGs for such stretch and sparsity (Theorem 2).

Planar. Let Φ be the ratio between the maximum and minimum (finite) distances in G (called aspect ratio), and let $\varepsilon \in (0, 1)$. For planar digraphs, we show (Theorem 7) that with $\tilde{O}(n \cdot \varepsilon^{-1} \cdot \log \Phi)$ added edges, nearly the best possible result is achievable: 2 DAGs and stretch $1 + \varepsilon$.

In all our results, the number of Steiner vertices is asymptotically the same as the number of added edges. See Section 7 for further discussion and open problems.

2 Related Work

Trading off parameters in tree covers. While [55] seeked tree covers with small stretch, recently Bartal, Fandina, and Neiman [12] studied the other side of the parameter regime for general (undirected) graphs, seeking a small number of trees. They obtained a tree cover with stretch $O(n^{1/k} \log^{1-1/k} n)$ and k trees, for any integer $k \geq 1$. There is also a very recent lower bound in this parameter regime [26].

Ramsey tree covers. In a *Ramsey tree cover*, instead of having a collection of trees with a low-stretch estimate of each $d(u, v)$ in some tree, the goal is to have a collection of trees so that for each u there is a low-stretch estimate of all $d(u, v)$ in a single tree. That is, a collection of trees such that for every vertex, one of the trees is an approximate shortest path tree. See [13, 50, 52, 1, 35, 31, 43, 26, 28].

Probabilistic tree and DAG embeddings. Probabilistic tree embedding is an extensively studied and widely applicable primitive related to tree covers: the goal is to obtain a distribution over trees so that the expected value of each $d(u, v)$ has low stretch [44, 4, 9, 10, 30, 11, 46]. See [6] for a more detailed list of variants and applications of probabilistic tree embeddings. As an analog for directed graphs, probabilistic DAG embeddings have been recently studied [6, 33].

3 Preliminaries

The $\tilde{O}$ notation hides polylogarithmic factors, that is $\tilde{O}(g) = O(g) \cdot \text{polylog}(g)$. A *digraph* $G = (V, E, w)$ consists of a set of vertices V, a directed set of edges $E \subseteq V \times V$ (with no self-loops), and a positive weight function $w : E \to \mathbb{R}_{>0}$. By scaling, we will assume w.l.o.g. that the minimum edge weight is 1. We will also use $V(G), E(G)$ to denote the set of vertices and edges of a digraph G. For a subset $A \subseteq V$ of vertices, $G[A] = (A, E_A = E \cap A \times A, w_{\restriction E_A})$ denotes the subgraph *induced* by A, i.e., the graph with vertex set A where we keep all edges between vertices of A. Given a digraph G, $s \rightsquigarrow_G t$ denotes that there is a path from s to t in G. Similarly, we will use $s \not\rightsquigarrow_G t$ to denote that there is no path from s to t in G.

Given a path $\pi = (v_0, v_1, \dots, v_k)$, $\pi[v_i, v_j] = (v_i, v_{i+1}, \dots, v_j)$ denotes the subpath from v_i to v_j. Similarly, we use $\pi(v_i, v_j) = (v_{i+1}, v_{i+2}, \dots, v_{j-1})$ to denote the subpath without endpoints (we might also use $\pi[v_i, v_j)$ and $\pi(v_i, v_j]$ to include only one endpoint). The notation d_G denotes the shortest path (quasi)metric in G, i.e., $d_G(u, v)$ is the minimal weight of a path from u to v. If there is no such path ($u \not\rightsquigarrow_G v$) then we set $d_G(u, v) = \infty$. The *aspect ratio* is $\Phi = \frac{\max\{d(u,v) | u \rightsquigarrow_G v\}}{\min\{d(u,v) | u \rightsquigarrow_G v\}}$, the ratio between the maximum and minimum (finite) distances in G.

A *directed acyclic graph (DAG)* is a digraph not containing any directed cycle. In particular, the set of strongly connected components (SCCs) is the set of singleton vertices. Given a DAG $D = (V, E)$, a *topological order* is a total order $<_{\text{TO}}$ over V such that for every edge $(u, v) \in E$, $u <_{\text{TO}} v$. Topological order is not necessarily unique. Consider two DAGs $D = (V, E)$ and $D' = (V', E')$ with respective topological orders $<_D$ and $<_{D'}$ respectively. We say that the two orders *agree* if for every $u, v \in V \cap V'$, $u <_D v \iff u <_{D'} v$. Note that if the orders of two DAGs D, D' agree, then $D \cup D' = (V \cup V', E \cup E')$ is also a DAG, and the topological order of $D \cup D'$ restricted to D (resp. D') is identical to $<_D$ (resp. $<_{D'}$).

Treewidth. A *tree decomposition* of an undirected graph $G = (V, E)$ is a tree $\mathcal{T}$ where each node $x \in \mathcal{T}$ is associated with a subset S_x of V, called a *bag*, such that: (i) $\cup_{x \in V(\mathcal{T})} S_x = V$, (ii) for every edge $(u, v) \in E$, there exists a bag S_x for some $x \in V(\mathcal{T})$ such that $\{u, v\} \subseteq S$, and (iii) for every $u \in V$, the bags containing u induces a connected subtree of $\mathcal{T}$. The *width* of $\mathcal{T}$ is $\max_{x \in V(\mathcal{T})}\{|S_x|\}$-1. The *treewidth* of G is the minimum width among all possible tree decompositions of G. We say that a digraph G has treewidth *tw* if it's undirected counterpart (where we keep all edges and ignore directions) has treewidth *tw*.
A *path decomposition* of G is a special kind of tree decomposition where the underlying tree is a path. The *pathwidth* of G is the minimal width of a path decomposition of G.

4 Non-Steiner DAG cover for Digraphs with Bounded Treewidth

This section is devoted to non-Steiner DAG cover for digraphs with bounded treewidth. We begin with a lower bound (Theorem 2), which provides an example of a planar digraph with treewidth 1 (bidirected star), showing that any DAG cover with a subquadratic number of additional edges $\mu = O(n^{2-\delta})$, requires $\Omega(\log n)$ DAGs. Afterwards, in Theorem 3 we show that Theorem 2 is tight (up to second order terms) by proving that every digraph with bounded treewidth *tw* admits a DAG cover with stretch 1, 2 DAGs, and near-linear number of additional edges.

▶ **Theorem 2** (Non-Steiner LB). *There is a digraph G with treewidth 1 such that for every $t < 2$, and $\mu \in [0, n^2]$, if G admits a (t, g, μ)-DAG cover without Steiner points, then* $g \geq \log\left(\frac{(n-1)^2}{2\mu+n-1} + 1\right) = \Omega\left(\log(\frac{n^2}{\mu+1})\right)$.

Proof. The bidirected star $\mathcal{S}_n$ is the unweighted digraph consisting of a root vertex rt and $n-1$ "leaf" vertices $v_2, \ldots, v_n$, where there are bidirected edges between any leaf vertex v_i and the root rt (formally the edges are $\{(\mathrm{rt}, v_i)\}_{i=2}^n \cup \{(v_i, \mathrm{rt})\}_{i=2}^n$). See Figure 1.

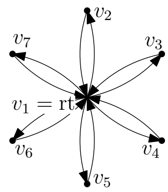

Figure 1 The bidirected star S_7.

Consider a (t, g, μ)-DAG cover $\mathbb{D} = \{D_1, D_2, \ldots, D_g\}$ of $\mathcal{S}_n$. We begin by making an observation about the shortest path P in $\mathbb{D}$ between two leaf vertices v_i and v_j with $i \neq j$. Consider a DAG D, where the shortest v_i-v_j path P goes through some other leaf vertex v_h. Then, by the dominating property of the DAG cover,

$$d_D(v_i, v_j) = d_D(v_i, v_h) + d_D(v_h, v_j) \geq d_G(v_i, v_h) + d_G(v_h, v_j) = 2 + 2 > t \cdot d_G(v_i, v_j) \ ,$$

thus the stretch guarantee is not fulfilled in D. We conclude that one of the following must hold: either there is a DAG in the cover with a direct edge between v_i and v_j, or there is a DAG in the cover containing the path (v_i, rt, v_j).

To proceed, we will define a binary string of length g that we will call a *codeword* for every vertex, and show that there must be a large number of different codewords. These codewords will be derived from the above observation. Formally, for each vertex v_i, we will construct a codeword $B_i = b_i^1 b_i^2 \ldots b_i^g$ of length g where each bit b_i^k is 1 if the DAG D_k has an edge from v_i to rt, and 0 otherwise. Consider the set of unordered pairs:

$$Q := \left\{ \{v_i, v_j\} : \begin{array}{l} \text{there is no edge in any DAG of } \mathbb{D} \text{ between} \\ v_i \text{ and } v_j \text{ (in either direction) for } i \neq j \end{array} \right\} .$$

Q contains at least $\binom{n-1}{2} - \mu$ pairs, as any edge in the DAG cover $\mathbb{D}$ (not in $\mathcal{S}_n$) can remove only a single edge from Q. Notice that for every pair $\{v_i, v_j\} \in Q$, there must be some DAG $D_q \in \mathbb{D}$ containing the path (v_i, rt, v_j). As D_q is a DAG, it will not contain the edge (v_j, rt). In particular $b_i^q = 1$, $b_j^q = 0$. We conclude that for every pair $\{v_i, v_j\} \in Q$, the codewords B_i and B_j must differ in some bit.

Consider the undirected graph G' with vertices $\{v_2, \ldots, v_n\}$, and the edge set Q. By Turán's theorem, the graph G' contains a clique of size r if

$$|Q| \geq \binom{n-1}{2} - \mu > \left(1 - \frac{1}{r-1}\right) \frac{(n-1)^2}{2} \ .$$

It follows that G' contains a clique of size at least $r = \frac{(n-1)^2}{2\mu+n-1} + 1$. Each of the vertices in this clique must have a different codeword. Since there are at most 2^g different possible codewords of length g, we conclude that $2^g \geq r$, and thus $g \geq \log\left(\frac{(n-1)^2}{2\mu+n-1} + 1\right) = \Omega\left(\log(\frac{n^2}{\mu+1})\right)$. ◀

From Theorem 5 it follows that the bidirected star (the lower bound example of Theorem 2) admits a Steiner $(1, 2, \tilde{O}(n))$-DAG cover, a huge difference! See Figure 2 for an illustration of this DAG cover.

Theorem 3 below shows that our lower bound from Theorem 2 is tight. That is, there is a DAG cover preserving all distances exactly with $O(\log n)$ DAGs, while only using near-linear number of edges.

▶ **Theorem 3.** *Every n-vertex digraph $G = (V, E, w)$ with treewidth tw admits a $\big(1, O(\log n), O(n \cdot tw \cdot \log^2 n)\big)$-DAG cover.*

Proof. The proof is by induction on the number of vertices n. Specifically, we will assume that for every digraph with treewidth tw and $n' < n$ vertices admits an $\big(1, 2\lceil\log n'\rceil, n' \cdot 2(tw+1) \cdot \lceil\log n'\rceil^2\big)$ DAG cover. The base case is trivial. Consider a digraph with treewidth tw and n vertices. Let $B = \{x_1, \dots, x_q\} \subseteq V$ be a separator bag. That is, B is a set of $|B| \le tw+1$ vertices, such that every connected component of $G \setminus B$ (in the undirected sense), contains at most $\frac{n}{2}$ vertices. Consider $G \setminus B$, and let $C_1, \dots, C_\ell$ be the connected components in the undirected sense. For every j, let $\mathbb{D}_j = \{D^j_1, D^j_2, \dots, D^j_{2\lceil\log|C_j|\rceil}\}$ be the DAG cover of $G[C_j]$ promised by induction. We also let $D^j_{2\lceil\log|C_j|\rceil+1}, \dots, D^j_{2\lceil\log n\rceil}$ to be empty digraphs over C_j.

Let $S^1, \dots, S^g \in \{0,1\}^{2\lceil\log n\rceil}$ be codewords such that for every two codewords S^i, S^j there are two bits α, β where $S^i_\alpha = 0, S^i_\beta = 1, S^j_\alpha = 1, S^j_\beta = 0$. Such codewords can be created e.g. by concatenating the binary representations of the numbers $i-1$ and $n-(i-1)$ for each i. We create the DAG cover $\mathbb{D} = \{D_1, D_2, \dots, D_{2\lceil\log n\rceil}\}$. The DAG D_i is constructed as follows:

1. D_i will contain all the edges in $D^1_i, D^2_i, \dots, D^\ell_i$.
2. For every $j \in [1, \ell]$, if $S^j_i = 1$ add the edges $C_j \times B$, that is all possible outgoing edges from C_j to B. Otherwise ($S^j_i = 0$), add the edges $B \times C_j$, that is all possible outgoing edges from B to C_j.
3. If $i = 1$, add the edges $\{(x_\alpha, x_\beta) \mid 1 \le \alpha < \beta \le q\}$, that is all possible internal edges in the bag B that respect the order $(x_1, \dots, x_q)$.
4. If $i = 2$, add the edges $\{(x_\beta, x_\alpha) \mid 1 \le \alpha < \beta \le q\}$, that is all possible internal edges in the bag B that respect the order $(x_q, \dots, x_1)$.

The weight of every added edge (x, y) will be $d_G(x, y)$.

First, we observe that each D_i is a DAG. Indeed, we can define a topological order $<_i$ over the vertices V such that all the edges respecting the order. The order inside each connected component C_j is defined with respect to a topological order of D^j_i. All the components C_j with $S^j_i = 1$ will appear before B vertices in the order, while all other components (with $S^j_i = 0$) appear after B. The internal order on B in D_1 is $(x_1, \dots, x_q)$, in D_2 is $(x_q, \dots, x_1)$, and for $i \ge 3$ is arbitrary. We conclude that we obtained a DAG cover.

Next we bound the total number of edges. In steps 2-4 we added $(n - |B|) \cdot |B|$ edges for each DAG. We also added $2 \cdot \binom{|B|}{2} \le |B|^2$ edges (in total) between the vertices of B. Thus in total $|B|^2 + (n - |B|) \cdot |B| \cdot 2\lceil\log n\rceil \le n \cdot |B| \cdot 2\lceil\log n\rceil$ edges. Using the induction hypothesis, the total number of edges added due to previously created DAG covers is $\sum_{i=1}^{\ell} |C_i| \cdot (tw+1) \cdot 2\lceil\log|C_i|\rceil^2 \le n \cdot (tw+1) \cdot 2\lceil\log\frac{n}{2}\rceil^2$. We conclude that the total number of edges in the DAG cover is

$$n \cdot |B| \cdot 2\lceil\log n\rceil + n \cdot (tw+1) \cdot 2\lceil\log\frac{n}{2}\rceil^2 \le n \cdot (tw+1) \cdot 2\lceil\log n\rceil^2 ,$$

as required.

Finally we argue that all the distances are exactly preserved. Consider a pair of vertices $x, y \in V$. Let $x = x_\alpha$ and $y = x_\beta$, where w.l.o.g. $\alpha < \beta$. We continue by case analysis.

- $x, y \in B$. The edge (x, y) was added to D_1, and thus $d_{D_1}(x, y) = d_G(x, y)$.
- $x \in B, y \notin B$, then there is some j such that $y \in C_j$, and some index α such that $S^j_\alpha = 0$. In D_α we added the edge (x, y).
- $y \in B, x \notin B$, then there is some j such that $x \in C_j$, and some index α such that $S^j_\alpha = 1$. In D_α we added the edge (x, y).

- $x, y \notin B$, and the shortest x-y path does not intersect B. By the definition of treewidth, there is some j such that the shortest x-y path lies in C_j (and in particular, $x, y \in C_j$). Using the induction hypothesis there is some DAG D_α^j such that $d_{D_\alpha}(x,y) \le d_{D_\alpha^j}(x,y) = d_G(x,y)$.
- $x, y \notin B$, and the shortest x-y path intersects B. Let $z \in B$ be a vertex such that the shortest x-y path goes through z. Let i be an index such that $S_i^j = 1$ and $S_i^{j'} = 0$. In D_i we added the edges $(x,z),(z,y)$. It thus holds that $d_{D_i}(x,y) \le d_G(x,z) + d_G(z,y) = d_G(x,y)$. ◀

▶ Remark 4. Note that the treewidth of each of the DAGs in the cover created in Theorem 3 is $O(tw \cdot \log n)$.

5 Steiner DAG cover for Digraphs with Bounded Treewidth

In this section we provide a DAG cover for bounded treewidth graphs. Furthermore, we show that the structure of the DAGs we construct preserves some structure of the original graph as it has bounded pathwidth.

▶ **Theorem 5.** *Every n-vertex digraph $G = (V, E, w)$ with treewidth tw admits a $(1, 2, O(n \cdot tw \cdot \log n))$-Steiner DAG cover. Furthermore, both of the DAGs in the cover have pathwidth $O(tw \cdot \log n)$.*

Proof. We will begin by disregarding the pathwidth requirement. Later we will adapt the construction accordingly. Fix an arbitrary permutation $\sigma = (v_1, \ldots, v_n)$ of the vertices in V. Let $\sigma^{-1} = (v_n, \ldots, v_1)$ be the reversed permutation. The work horse of our construction is a gadget that allows us to preserve all the shortest paths respecting σ that go through a certain separator vertex. We will then use this gadget on all separator vertices, and finally use it recursively to obtain our first DAG D_σ. The second DAG $D_{\sigma^{-1}}$ will be constructed in the same way with respect to the reversed permutation.

▶ **Lemma 6.** *Given an n vertex digraph $G = (V, E)$, order σ over V, and a vertex $x \in V$, there is a DAG D_x containing V such that*
- *The induced topological order of D_x on V is σ.*
- *For every $u, v \in V$ such that $u <_\sigma v$, $d_G(u,v) \le d_{D_x}(u,v) \le d_G(u,x) + d_G(x,v)$.*

D_x contains n Steiner points, and at most $3n$ edges.

Proof. Let $\sigma = (v_1, \ldots, v_n)$ be the designated order over V. For every vertex v_i, add an auxiliary Steiner point u_i. For every $i \in [1, n]$ we add to D_x the following edges:
- If $i < n$, the edge (u_i, u_{i+1}) of weight 0.
- If $x \rightsquigarrow_G v_i$, the edge (u_i, v_i) of weight $d_G(x, v_i)$.
- If $i < n$ and $v_i \rightsquigarrow_G x$, the edge (v_i, u_{i+1}) of weight $d_G(v_i, x)$.

See Figure 2 for an illustration. Note that in total we added at most $3n$ edges. D_x is indeed a DAG, where the topological order is $u_1, v_1, u_2, v_2, \ldots, u_n, v_n$ (note that all the edges respect this order). Accordingly, the induced topological order over V is σ. Finally, for every $i < j$ such that $v_i \rightsquigarrow_G x$ and $x \rightsquigarrow_G v_j$ it holds that the only outgoing edge from v_i is towards u_{i+1}, while the only ingoing edge to v_j is from u_j. It follows that

$$d_{D_x}(v_i, v_j) = d_{D_x}(v_i, u_{i+1}) + d_{D_x}(u_{i+1}, u_j) + d_{D_x}(u_j, v_j) = d_G(v_i, x) + 0 + d_G(x, v_j) \ ,$$

and in particular $d_{D_x}(v_i, v_j) \ge d_G(v_i, v_j)$. Note that if $v_i \not\rightsquigarrow_G x$ or $x \not\rightsquigarrow_G v_j$, then $d_G(v_i, x) + d_G(x, v_j) = \infty$, and the lemma holds trivially. ◀

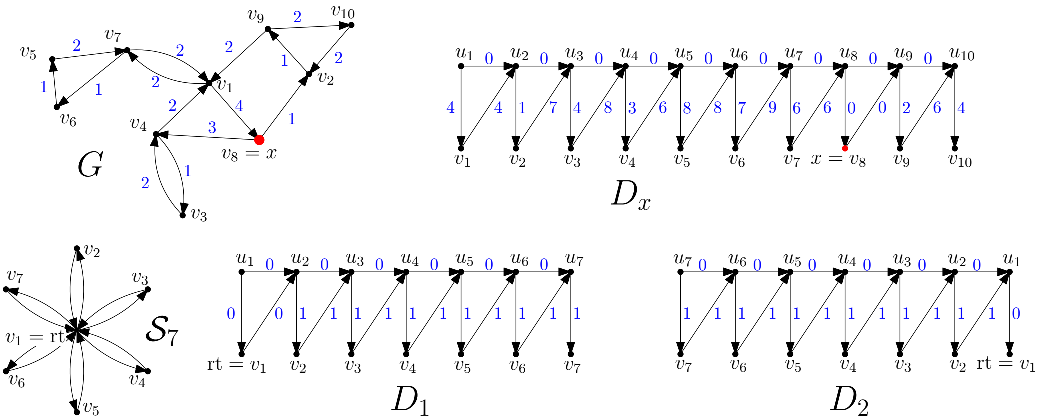

Figure 2 On top: Illustration of the DAG D_x created using Lemma 6 for the digraph G (on the left) with respect to the designated vertex x (v_8), and the order $(v_1, v_2, \ldots, v_{10})$. The DAG D_x contains an auxiliary vertex u_i for each v_i. We add a 0 weight edge between any two consecutive u_i, u_{i+1}. In addition for every i there is an edge from u_i to v_i of weight $d_G(x, v_i)$, and from v_i to u_{i+1} of weight $d_G(v_i, x)$. For every $i < j$, $d_{D_x}(v_i, v_j) \le d_{D_x}(v_i, u_{i+1}) + d_{D_x}(u_{i+1}, u_j) + d_{D_x}(u_j, v_j) = d_G(v_i, x) + 0 + d_G(x, v_j)$. That is, the shortest v_i-v_j path that goes through x.
On bottom: Illustration of the bidirected star $\mathcal{S}_7$ (used in the proof of Theorem 2), and its DAG cover. Here we take the order $\sigma = (v_1, \ldots, v_7)$ where the root vertex $v_1 = \mathrm{rt}$ is first. $D_1 = D_{\mathrm{rt}}$ is created using Lemma 6 with respect to vertex x and order σ. D_2 is created in the same way, but with respect to order σ^{-1}. As all shortest paths in $\mathcal{S}_7$ go through rt, D_1 and D_2 preserve all the distances exactly.

We will prove Theorem 5 by induction on the number of vertices n. Suppose that every digraph $G = (V, E)$ with $k < n$ vertices, treewidth tw, and for every permutation σ of G vertices, there is a dominating DAG D_σ with $3 \cdot k \cdot (tw + 1) \cdot \log k$ edges such that the induced topological order on V is σ, and for every $u <_\sigma v$, $d_{D_\sigma}(u, v) = d_G(u, v)$.

Next, consider a digraph $G = (V, E)$ with n vertices, treewidth tw, and an arbitrary permutation σ of G vertices. Let $B = \{x_1, \ldots, x_q\} \subseteq V$ be a separator bag. That is, B is a set of $|B| \le tw + 1$ vertices, such that every connected component of $G \setminus B$ (in the undirected sense), contains at most $\frac{n}{2}$ vertices. For every separator vertex $x_i \in B$, let D_{x_i} be the DAG returned by Lemma 6, with respect to the permutation σ and the vertex x_i. Consider $G \setminus B$, and let $C_1, \ldots, C_g$ be the connected components in the undirected sense. Let σ_j be the order σ induced on the vertices C_j. Using the induction hypothesis, let D_{σ_j} be the DAG created for $G[C_j]$ with respect to the order σ_j. We set D_σ to be the digraph created by taking the union of all the DAGs $D_{x_1}, \ldots, D_{x_q}, D_{\sigma_1}, \ldots, D_{\sigma_g}$. The Steiner points used in each DAG will be disjoint. Observe that D_σ is a DAG with topological order σ. This follows as all the DAGs in the union respect the order σ. Note also that all the distance in D_σ are dominating G because the Steiner points are disjoint (and a union of dominating digraphs with disjoint Steiner points is dominating).

Next we bound the size of D_σ. Using the induction hypothesis, it holds that D_{σ_j} contains at most $|C_j| \cdot 3 \cdot (tw + 1) \cdot \log |C_j| \le |C_j| \cdot 3 \cdot (tw + 1) \cdot \log \frac{n}{2}$ edges. By Lemma 6 and the fact $\sum_{j=1}^{g} |C_j| < n$, the number of vertices in D_σ is bounded by

$$
\begin{aligned}
|D_\sigma| &\le \sum_{i=1}^{q} |D_{x_i}| + \sum_{j=1}^{g} |D_{\sigma_j}| \\
&\le (tw+1)\cdot 3n + 3\cdot (tw+1)\cdot \log\frac{n}{2}\cdot \sum_{j=1}^{g}|C_j| \\
&\le (tw+1)\cdot 3n\cdot \left(1+\log\frac{n}{2}\right) = (tw+1)\cdot 3n\cdot \log n.
\end{aligned}
$$

Next, we argue that distances are preserved. Consider u, v such that $u <_\sigma v$. If $u \not\rightsquigarrow_G v$, then there is nothing to prove. We will thus suppose that $u \rightsquigarrow_G v$. Let P be the shortest u-v path in G. Suppose first that there is some vertex $x_i \in P \cap B$. In this case,

$$d_{D_\sigma}(u,v) \le d_{D_{x_i}}(u,v) \le d_G(u,x) + d_G(x,v) = d_G(u,v).$$

We will thus assume that P is disjoint from the separator B. It follows that P vertices are fully contained in some (undirected) connected component C_j of $G \setminus B$, and in particular $d_{G[C_j]}(u,v) = d_G(u,v)$. Using the inductive hypothesis, it follows that

$$d_{D_\sigma}(u,v) \le d_{D_{\sigma_j}}(u,v) = d_{G[C_j]}(u,v) = d_G(u,v)\ .$$

Finally, we construct a similar DAG $D_{\sigma^{-1}}$ with respect to the order σ^{-1}. As for every $u, v \in V$ either $u <_\sigma v$ or $u <_{\sigma^{-1}} v$, it follows that $\min\left\{d_{D_\sigma}(u,v), d_{D_{\sigma^{-1}}}(u,v)\right\} = d_G(u,v)$.

The proof that it is possible to create a DAG cover so that both DAGs have pathwidth $O(tw \cdot \log n)$ is deferred to Appendix A. ◀

6 Planar Digraphs

This section is dedicated to proving the following theorem:

▶ **Theorem 7.** *Given an n-vertex planar digraph $G = (V, E, w)$, and $\varepsilon \in (0,1)$, G admits a $\left(1+\varepsilon, 2, O(n\cdot\varepsilon^{-1}\cdot \log^2 n\cdot \log\Phi)\right)$-Steiner DAG cover.*

Shortest path separators. Thorup [54] introduced shortest path separators for planar digraphs and used them to construct a compact labeling scheme to approximate distances. The main tool in Thorup's construction is a collection of paths such that each vertex is associated with small number of paths, and every pairwise distance can be well approximated by rerouting through the associated paths. We summarize the exact details below.

▶ **Lemma 8** (Path cover for planar digraphs [54])**.** *Let $G = (V, E, w)$ be a planar digraph with aspect ratio Φ, and let $\varepsilon > 0$. There is a set of dipaths $\mathcal{P}$ in G, such that*

- *every vertex v in V is associated with a subset $\mathcal{P}[v]$ of $O(\log n \cdot \log \Phi)$ paths of $\mathcal{P}$;*
- *every pair (v, P) of a vertex $v \in V$ and a path $P \in \mathcal{P}[v]$ is associated with a set of $O(\varepsilon^{-1})$ vertices $C(v, P) \subseteq P$, called ε-covering set, with the property that for any pair of vertices u and v in V with $u \rightsquigarrow_G v$, there exists some path $P \in \mathcal{P}[u] \cap \mathcal{P}[v]$ and there exist vertices $u' \in C(u, P)$ and $v' \in C(v, P)$ such that*

$$d_G(u,u') + d_P(u',v') + d_G(v',v) \le (1+\varepsilon)d_G(u,v).$$

Lemma 8 follows from Thorup [54] implicitly. Specifically, it follows by combining [54] Lemmas 3.2, 3.4, and 3.5. Similar statements have also been made in [51, 47].

Steiner DAG cover construction. As in the bounded treewidth case (Theorem 5), we fix an arbitrary permutation $\sigma = (v_1, \ldots, v_n)$ of V. We construct one DAG D_σ that preserves all distances for pairs u, v such that $u <_\sigma v$ within a multiplicative error of $1+\varepsilon$ (the error arises from Lemma 8). We construct another DAG $D_{\sigma^{-1}}$, defined symmetrically, that preserves all remaining distances. To obtain D_σ (and $D_{\sigma^{-1}}$), similarly to the bounded treewidth case, we can take the union of DAGs constructed using Lemma 6. Suppose first that for each vertex $x \in V$, we use Lemma 6 to construct a DAG D_x with topological order σ, such that for all pairs $u <_\sigma v$ it holds that $d_{D_x}(u,v) = d_G(u,x) + d_G(x,v)$. We will then define our DAG as a union of all these DAGs $D_{\text{all}} = \cup_{x\in V} D_x$. Following our proof and construction of bounded treewidth digraphs (Theorem 5), D_{all} is a DAG (as it is a union of DAGs, with intersection only over V, where the topological order with respect to V is σ in all the DAGs), and D_{all} preserve exactly the distance between all (u,v) pairs such that $u <_\sigma v$. Unfortunately, D_{all} has quadratic size.

Instead, for each vertex $x \in V$, we will define a set of associated centers $A_x \subseteq V$. We then apply Lemma 6 to construct D_x only over A_x (and not over V). We then define $D_\sigma = \cup_{x\in V} D_x$. Clearly D_σ is a dominating DAG. Let $X_v = \{x \in V \mid v \in A_x\}$ be all the centers of the DAGs to which v joins. Following Lemma 6, the number of edges in D_σ will be $\sum_{x\in V} O(|A_x|) = \sum_{v\in V} O(|X_v|)$. Our goal is thus to choose the sets $\{X_v\}_{v\in V}$ so that their total size is bounded, while all pairwise distances are preserved. That is for all $u, v \in V$ $\min_{x\in X_u \cap X_v}\{d_G(u,x) + d_G(x,v)\} \le (1+\varepsilon)\cdot d_G(u,v)$.

Recall that in the treewidth case (Theorem 5), X_v consist of the vertices in the separator bag B, and then recursively of all the vertices in associated separator bags in the weakly connected component of v in $G \setminus B$. That ensured that $|X_v| = O(tw \cdot \log n)$, and that all the distances are preserved exactly. For the planar case, X_v will be chosen with respect to the associated paths $\mathcal{P}[v]$, and the ε-covering sets $\{C(v,P)\}_{P\in\mathcal{P}[v]}$ (given by Lemma 8).

Consider a separator path $P \in \mathcal{P}$. We construct a hierarchical decomposition of P into $O(\log|V(P)|)$ levels. The first level consists of P itself. Let $x(P)$ be the *centroid* of P. That is, a vertex whose number of predecessors and successors along P differ by at most one. P will have two child paths: the prefix of P up to (and not including) $x(P)$, and the suffix of P from (and not including) $x(P)$. We continue recursively on each of these two subpaths. See Figure 3 for an illustration. Overall, we obtain a binary tree of paths, where for each path P' in this tree with children P'_1, P'_2 it holds that $P' = P'_1 \circ x(P') \circ P'_2$. The leaves in this tree will be singleton subpaths (with a single vertex). Each vertex $y \in P$ will be associated with a set $\mathcal{Q}(y,P)$ of subpaths containing it, where $|\mathcal{Q}(y,P)| = O(\log|P|)$. For a vertex $v \in V$, the set X_v of DAG centers to which v joins is defined as follows:

- For every path $P \in \mathcal{P}[v]$,
  - for every vertex $y \in C(v,P)$ from the ε-covering set,
    - for every subpath $P' \in \mathcal{Q}(y,P)$,
      - add the centroid $x(P')$ to X_v.

Clearly, as $|\mathcal{P}[v]| = O(\log\Phi\cdot\log n)$, $C(v,P) = O(\varepsilon^{-1})$ (for every $P \in C(v,P)$), and $\mathcal{Q}(y,P) = O(\log n)$ (for every $y \in C(v,P)$), it holds that $|X_v| = O(\varepsilon^{-1}\cdot\log\Phi\cdot\log^2 n)$. The definition of the sets $\{A_x\}_{x\in V}$ follows. For every vertex $x \in V$, we apply Lemma 6 with respect to the vertex $x \in V$, the subset A_x, and the topological order σ, to obtain the DAG D_x. We then define $D_\sigma = \cup_{x\in V} D_x$. Following the lines of Theorem 5, D_σ is a DAG. The number of edges in D_σ is bounded by $\sum_{x\in V} O(|A_x|) = \sum_{v\in V} O(|X_v|) = O(n\cdot\varepsilon^{-1}\cdot\log\Phi\cdot\log^2 n)$. It remains to prove the stretch guarantee.

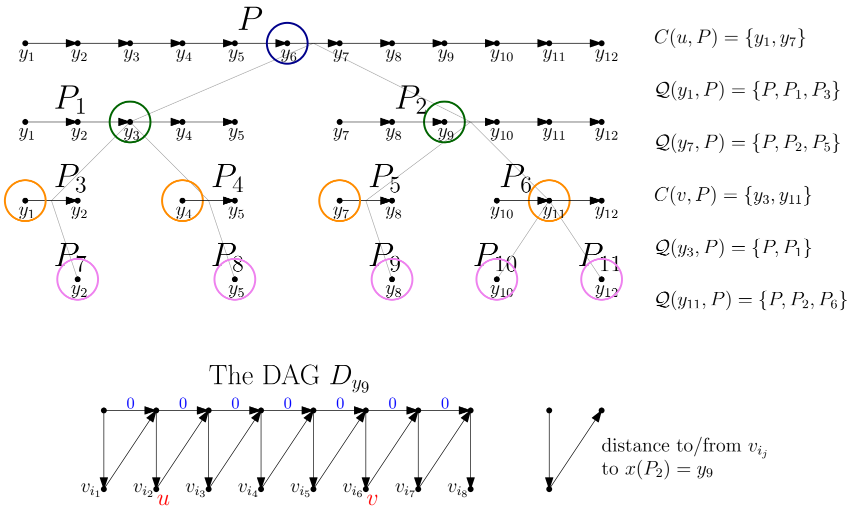

Figure 3 On top: Illustration of the path $P \in \mathcal{P}$ and how it is broken to subpaths by deleting centroid vertices. The hierarchical tree of the subpaths is drawn using a thin gray line. For each vertex $y_i \in P$, $\mathcal{Q}(y_i, P)$ contains all the subpaths containing y_i. Consider two vertices $u <_\sigma v$ such that $P \in \mathcal{P}[u] \cap \mathcal{P}[v]$, and there are vertices $u' \in C(u, p)$, $v' \in C(v, p)$ where $d_G(u, u') + d_P(u', v') + d_G(v', v) \le (1 + \varepsilon) d_G(u, v)$. In the illustration, $u' = y_7$ and $v' = y_{11}$. The subpath P_2 contains both y_7, y_{11}, and thus both X_u and X_v will contain $x(P_2) = y_9$, the centroid of P_2.
On bottom: Illustrated the DAG D_{y_9} over A_{y_9} that contains both u, v (constructed using Lemma 6). As $u <_\sigma v$, $d_{D_{y_9}} \le d_G(u, y_9) + d_G(y_9, v) \le (1 + \varepsilon) \cdot d_G(u, v)$.

Consider a pair of vertices $u, v \in V$ such that $u <_\sigma v$ and $u \rightsquigarrow_G v$. By Lemma 8, there exists some path $P \in \mathcal{P}[u] \cap \mathcal{P}[v]$ and vertices $u' \in C(u, P)$ and $v' \in C(v, P)$ such that $d_G(u, u') + d_P(u', v') + d_G(v', v) \le (1 + \varepsilon) \cdot d_G(u, v)$. Consider the hierarchical decomposition of P to subpaths, and let $P' \subseteq P$ be the maximum subpath containing both u', v'. That is, P' is a consecutive subpath of P, containing both u, v, such that once we delete the centroid $x(P')$, the (possibly) two remaining subpaths does not contain both u', v'. It follows that the path P' goes from u' to $x(P')$ to v'. Note that $x(P') \in X_u \cap X_v$. See Figure 3 for an illustration. Using Lemma 6 we conclude

$$\begin{aligned} d_{D_\sigma}(u, v) \le d_{D_{x(P')}}(u, v) = d_G(u, x(P')) + d_G(x(P'), v) \\ &\le d_G(u, u') + d_G(u', x(P')) + d_G(x(P'), v') + d_G(v', v) \\ &\le d_G(u, u') + d_P(u', v') + d_G(v', v) \le (1 + \varepsilon) \cdot d_G(u, v) \ . \end{aligned}$$

The DAG $D_{\sigma^{-1}}$ constructed in the exact same way, but with respect to the order σ^{-1}. As for every u, v, either $u <_\sigma v$ or $u <_{\sigma^{-1}} v$ we conclude that $\min\{d_{D_\sigma}(u, v), d_{D_{\sigma^{-1}}}(u, v)\} \le (1 + \varepsilon) d_G(u, v)$, as required.

7 Discussion and Open Problems

In this paper we studied Steiner DAG covers, and demonstrated a stark separation between Steiner and non-Steiner DAG covers (see Table 1). However, many fascinating problems remain unresolved. We discuss several of them:

1. **Bounds for general graphs.** Filtser [33] showed that any digraph admits a non-Steiner $(\tilde{O}(\log n), O(\log n), \tilde{O}(m))$-DAG cover. However, there is no lower bound showing that $(t, g, \tilde{O}(m))$-DAG cover is impossible even for constant t and g. Because the approach

from [6, 33] is based on directed low diameter decompositions (LDDs) [14, 15, 39, 49], and because the Lipschitz parameter of directed LDDs for general digraphs has a provable lower bound of $\Omega(\log n)$ [9], we would need a drastically different approach from [6, 33]. Understanding the best possible DAG covers for general digraphs (in both the Steiner and non-Steiner settings) is a fascinating open question.

2. **Preserving the topological structure**. In both Theorems 5 and 7 our focus was on obtaining sparse DAGs in the cover. Since small treewidth or planarity are highly desirable graph properties, it is natural to ask that all the DAGs in the DAG cover of a bounded treewidth / planar graph to remain so. For treewidth *tw* digraphs, our Theorem 5 constructs two DAGs with pathwidth $O(tw \cdot \log n)$. While pathwidth is a much more structured (and useful) property[2], we would like a smaller blow-up in the treewidth. The DAGs we construct for planar digraphs (Theorem 7) are not planar at all. Preserving planarity, or bounded treewidth, is an interesting question.
3. **Aspect ratio dependence.** Our planar result (Theorem 7), has a logarithmic dependence on the *aspect ratio*. The aspect ratio can be unbounded, so it is desirable to remove this dependency. For the much better studied $1+\varepsilon$ approximate distance oracles for planar digraphs there is also a dependency on the aspect ratio (both in Thorup [54] and in the state-of-the-art oracle of Le and Wulff-Nilsen [48]). Removing this dependency is a nice open question.
4. **Exact planar DAG covers**. Our DAG cover for planar digraph has stretch $1+\varepsilon$, while in the treewidth case distances are preserved exactly. It is interesting to understand what is the minimum number of additional edges μ such that every planar digraph admits a Steiner $(1, 2, \mu)$-DAG cover. This problem may look challenging: for undirected planar graphs, exact tree cover requires $\Omega(n^{\frac{1}{3}-\varepsilon})$ trees[3], due to an information-theoretic lower bound on distance labeling [23, 37]. However, the tree cover lower bound does not imply a DAG cover lower bound, as DAGs are more expressive than trees. Some hope comes from a recent result of Charalampopoulos *et al.* [25], who constructed an exact distance oracle for planar digraphs with space $n^{1+o(1)}$ and polylog(n) query time. Whether there is a Steiner $(1, 2, n^{1+o(1)})$-DAG cover for planar digraphs is also an interesting question.

References

1 Ittai Abraham, Shiri Chechik, Michael Elkin, Arnold Filtser, and Ofer Neiman. Ramsey spanning trees and their applications. *ACM Trans. Algorithms*, 16(2):19:1–19:21, 2020. preliminary version published in SODA 2018. doi:10.1145/3371039.

2 Ittai Abraham, Arnold Filtser, Anupam Gupta, and Ofer Neiman. Metric embedding via shortest path decompositions. *SIAM J. Comput.*, 51(2):290–314, 2022. a preliminary version apperared in the proceedings of STOC 18. doi:10.1137/19m1296021.

3 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. *Comput. Sci. Rev.*, 37:100253, 2020. doi:10.1016/j.cosrev.2020.100253.

[2] E.g., (undirected) graphs with constant pathwidth embed into ℓ_2 with constant stretch [2], while there is a graph with treewidth 2 that requires stretch $\Omega(\sqrt{\log n})$ [53]. Another example is universal Steiner tree, where bounded pathwidth graphs admit a solution with stretch $O(\log n)$ [20, 32], while nothing better than the stretch $O(\log^7 n)$ of general graph is known for graphs of treewidth 2.

[3] If one demands a tree cover that doesn't use Steiner vertices or edges, then the lower bound can be improved to $\Omega(n^{1/2})$, which is tight [38].

4 Noga Alon, Richard M. Karp, David Peleg, and Douglas B. West. A graph-theoretic game and its application to the k-server problem. *SIAM J. Comput.*, 24(1):78–100, 1995. preliminary version published in On-Line Algorithms 1991. doi:10.1137/S0097539792224474.

5 Sunil Arya, Gautam Das, David M. Mount, Jeffrey S. Salowe, and Michiel H. M. Smid. Euclidean spanners: short, thin, and lanky. In Frank Thomson Leighton and Allan Borodin, editors, *Proceedings of the Twenty-Seventh Annual ACM Symposium on Theory of Computing, 29 May-1 June 1995, Las Vegas, Nevada, USA*, pages 489–498. ACM, 1995. doi:10.1145/225058.225191.

6 Sepehr Assadi, Gary Hoppenworth, and Nicole Wein. Covering approximate shortest paths with dags. In *Proceedings of the 57th Annual ACM Symposium on Theory of Computing*, STOC '25, pages 2269–2280, New York, NY, USA, 2025. Association for Computing Machinery. doi:10.1145/3717823.3718298.

7 Baruch Awerbuch, Shay Kutten, and David Peleg. On buffer-economical store-and-forward deadlock prevention. *IEEE transactions on communications*, 42(11):2934–2937, 1994. doi:10.1109/26.328973.

8 Baruch Awerbuch and David Peleg. Routing with polynomial communication-space trade-off. *SIAM J. Discrete Math.*, 5(2):151–162, 1992. doi:10.1137/0405013.

9 Yair Bartal. Probabilistic approximations of metric spaces and its algorithmic applications. In *37th Annual Symposium on Foundations of Computer Science, FOCS '96, Burlington, Vermont, USA, 14-16 October, 1996*, pages 184–193, 1996. doi:10.1109/SFCS.1996.548477.

10 Yair Bartal. On approximating arbitrary metrics by tree metrics. In *Proceedings of the 30th Annual ACM Symposium on Theory of Computing*, pages 183–193, 1998.

11 Yair Bartal. Graph decomposition lemmas and their role in metric embedding methods. In *Algorithms - ESA 2004, 12th Annual European Symposium, Bergen, Norway, September 14-17, 2004, Proceedings*, pages 89–97, 2004. doi:10.1007/978-3-540-30140-0_10.

12 Yair Bartal, Ora Nova Fandina, and Ofer Neiman. Covering metric spaces by few trees. *J. Comput. Syst. Sci.*, 130:26–42, 2022. doi:10.1016/J.JCSS.2022.06.001.

13 Yair Bartal, Nathan Linial, Manor Mendel, and Assaf Naor. On metric ramsey-type phenomena. In *Proceedings of the 35th Annual ACM Symposium on Theory of Computing (STOC)*, pages 463–472. ACM, 2003. doi:10.1145/780542.780610.

14 Aaron Bernstein, Maximilian Probst Gutenberg, and Christian Wulff-Nilsen. Near-optimal decremental SSSP in dense weighted digraphs. In Sandy Irani, editor, *61st IEEE Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1112–1122. IEEE, 2020. doi:10.1109/FOCS46700.2020.00107.

15 Aaron Bernstein, Danupon Nanongkai, and Christian Wulff-Nilsen. Negative-weight single-source shortest paths in near-linear time. *Commun. ACM*, 68(2):87–94, 2025. preliminary version published in FOCS 2022. doi:10.1145/3631536.

16 Aaron Bernstein and Nicole Wein. Closing the gap between directed hopsets and shortcut sets. In *Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 163–182. SIAM, 2023. doi:10.1137/1.9781611977554.CH7.

17 Sujoy Bhore, Balázs Keszegh, Andrey Kupavskii, Hung Le, Alexandre Louvet, Dömötör Pálvölgyi, and Csaba D Tóth. Spanners in planar domains via steiner spanners and non-steiner tree covers. In *Proceedings of the 2025 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 4292–4326. SIAM, 2025. doi:10.1137/1.9781611978322.145.

18 Greg Bodwin. Linear size distance preservers. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 600–615. SIAM, 2017. doi:10.1137/1.9781611974782.39.

19 Greg Bodwin and Gary Hoppenworth. Folklore sampling is optimal for exact hopsets: Confirming the $\sqrt{n}$ barrier. In *2023 IEEE 64th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 701–720. IEEE, 2023. doi:10.1109/FOCS57990.2023.00046.

20 Ostas Busch, Da Qi Chen, Arnold Filtser, Daniel Hathcock, D Ellis Hershkowitz, and Rajmohan Rajaraman. One tree to rule them all: Poly-logarithmic universal steiner tree. In *2023 IEEE 64th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 60–76, 2023. doi:10.1109/FOCS57990.2023.00012.

21 T.-H. Hubert Chan, Anupam Gupta, Bruce M. Maggs, and Shuheng Zhou. On hierarchical routing in doubling metrics. *ACM Trans. Algorithms*, 12(4):55:1–55:22, 2016. doi:10.1145/2915183.

22 Hsien-Chih Chang, Jonathan Conroy, Hung Le, Lazar Milenkovic, Shay Solomon, and Cuong Than. Covering planar metrics (and beyond): O(1) trees suffice. In *64th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2023, Santa Cruz, CA, USA, November 6-9, 2023*, pages 2231–2261. IEEE, 2023. doi:10.1109/FOCS57990.2023.00139.

23 Hsien-Chih Chang, Jonathan Conroy, Hung Le, Lazar Milenkovic, Shay Solomon, and Cuong Than. Optimal euclidean tree covers. In Wolfgang Mulzer and Jeff M. Phillips, editors, *40th International Symposium on Computational Geometry, SoCG 2024, June 11-14, 2024, Athens, Greece*, volume 293 of *LIPIcs*, pages 37:1–37:15. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024. doi:10.4230/LIPIcs.SOCG.2024.37.

24 Hsien-Chih Chang, Jonathan Conroy, Hung Le, Lazar Milenković, Shay Solomon, and Cuong Than. Shortcut partitions in minor-free graphs: Steiner point removal, distance oracles, tree covers, and more. In *Proceedings of the 2024 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 5300–5331. SIAM, 2024. doi:10.1137/1.9781611977912.191.

25 Panagiotis Charalampopoulos, Pawel Gawrychowski, Yaowei Long, Shay Mozes, Seth Pettie, Oren Weimann, and Christian Wulff-Nilsen. Almost optimal exact distance oracles for planar graphs. *J. ACM*, 70(2):12:1–12:50, 2023. doi:10.1145/3580474.

26 Yu Chen, Zihan Tan, and Hangyu Xu. Lower bounds on tree covers. *arXiv preprint arXiv:2508.10376*, 2025. doi:10.48550/arXiv.2508.10376.

27 Michael Elkin and Ofer Neiman. Linear-size hopsets with small hopbound, and constant-hopbound hopsets in rnc. In *The 31st ACM Symposium on Parallelism in Algorithms and Architectures*, pages 333–341, 2019. doi:10.1145/3323165.3323177.

28 Michael Elkin and Idan Shabat. Spanning and metric tree covers parameterized by treewidth. *arXiv preprint arXiv:2511.06263*, 2025. doi:10.48550/arXiv.2511.06263.

29 Michael Elkin and Shay Solomon. Optimal euclidean spanners: Really short, thin, and lanky. *J. ACM*, 62(5):35:1–35:45, 2015. doi:10.1145/2819008.

30 Jittat Fakcharoenphol, Satish Rao, and Kunal Talwar. A tight bound on approximating arbitrary metrics by tree metrics. *J. Comput. Syst. Sci.*, 69(3):485–497, November 2004. doi:10.1016/j.jcss.2004.04.011.

31 Arnold Filtser. Hop-constrained metric embeddings and their applications. In *62nd IEEE Annual Symposium on Foundations of Computer Science, FOCS 2021, Denver, CO, USA, February 7-10, 2022*, pages 492–503. IEEE, 2021. doi:10.1109/FOCS52979.2021.00056.

32 Arnold Filtser. Scattering and sparse partitions, and their applications. *ACM Trans. Algorithms*, 20(4):30:1–30:42, 2024. doi:10.1145/3672562.

33 Arnold Filtser. Stochastic embedding of digraphs into dags. In *Proceedings of the 2026 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 105–119, 2026. doi:10.1137/1.9781611978971.5.

34 Arnold Filtser, Yuval Gitlitz, and Ofer Neiman. Light, reliable spanners. In Wolfgang Mulzer and Jeff M. Phillips, editors, *40th International Symposium on Computational Geometry, SoCG 2024, June 11-14, 2024, Athens, Greece*, volume 293 of *LIPIcs*, pages 56:1–56:15. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024. doi:10.4230/LIPIcs.SOCG.2024.56.

35 Arnold Filtser and Hung Le. Clan embeddings into trees, and low treewidth graphs. In Samir Khuller and Virginia Vassilevska Williams, editors, *STOC '21: 53rd Annual ACM SIGACT Symposium on Theory of Computing, Virtual Event, Italy, June 21-25, 2021*, pages 342–355. ACM, 2021. doi:10.1145/3406325.3451043.

36 Arnold Filtser and Hung Le. Locality-sensitive orderings and applications to reliable spanners. In Stefano Leonardi and Anupam Gupta, editors, *STOC '22: 54th Annual ACM SIGACT Symposium on Theory of Computing, Rome, Italy, June 20 - 24, 2022*, pages 1066–1079. ACM, 2022. doi:10.1145/3519935.3520042.

37 Cyril Gavoille, David Peleg, Stéphane Pérennes, and Ran Raz. Distance labeling in graphs. *J. Algorithms*, 53(1):85–112, 2004. preliminary version published in SODA 2001. doi:10.1016/j.jalgor.2004.05.002.

38 Anupam Gupta, Amit Kumar, and Rajeev Rastogi. Traveling with a pez dispenser (or, routing issues in MPLS). *SIAM J. Comput.*, 34(2):453–474, 2004. preliminary version published in STOC 2001. doi:10.1137/S0097539702409927.

39 Bernhard Haeupler, Richard Hladík, Shengzhe Wang, and Zhijun Zhang. Stronger directed low-diameter decompositions with sub-logarithmic diameter and separation. *CoRR*, abs/2509.24565, 2025. doi:10.48550/arXiv.2509.24565.

40 Bernhard Haeupler, Yonggang Jiang, and Thatchaphol Saranurak. DAG projections: Reducing distance and flow problems to DAGs. In *Proccedings of the 58nd Annual ACM SIGACT Symposium on Theory of Computing, STOC 2026*, pages 2266–2277, 2026. doi:10.1145/3798129.3800928.

41 Gary Hoppenworth, Yinzhan Xu, and Zixuan Xu. New separations and reductions for directed hopsets and preservers. In *Proceedings of the 2025 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 4405–4443. SIAM, 2025. doi:10.1137/1.9781611978322.150.

42 Shang-En Huang and Seth Pettie. Thorup–zwick emulators are universally optimal hopsets. *Information Processing Letters*, 142:9–13, 2019. doi:10.1016/J.IPL.2018.10.001.

43 Omri Kahalon, Hung Le, Lazar Milenkovic, and Shay Solomon. Can't see the forest for the trees: Navigating metric spaces by bounded hop-diameter spanners. In *PODC '22: ACM Symposium on Principles of Distributed Computing*, pages 151–162. ACM, 2022. doi:10.1145/3519270.3538414.

44 Richard M Karp. A 2k-competitive algorithm for the circle. *Manuscript, August*, 5, 1989.

45 Shimon Kogan and Merav Parter. New diameter-reducing shortcuts and directed hopsets: Breaking the $o(\sqrt{n})$ barrier. In *Proceedings of the 2022 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1326–1341. SIAM, 2022. doi:10.1137/1.9781611977073.55.

46 Rasmus Kyng, Maximilian Probst Gutenberg, and Tim Rieder. Random-shift revisited: Tight approximations for tree embeddings and l1-oblivious routings. *CoRR*, abs/2510.09124, 2025. To apear in FOCS 2025. doi:10.48550/arXiv.2510.09124.

47 Itay Laish and Shay Mozes. Efficient dynamic approximate distance oracles for vertex-labeled planar graphs. *Theory Comput. Syst.*, 63(8):1849–1874, 2019. doi:10.1007/S00224-019-09949-5.

48 Hung Le and Christian Wulff-Nilsen. Optimal approximate distance oracle for planar graphs. In *62nd IEEE Annual Symposium on Foundations of Computer Science, FOCS 2021, Denver, CO, USA, February 7-10, 2022*, pages 363–374. IEEE, 2021. doi:10.1109/FOCS52979.2021.00044.

49 Jason Li. Simpler and faster directed low-diameter decompositions, 2025. doi:10.48550/arXiv.2505.10244.

50 Manor Mendel and Assaf Naor. Ramsey partitions and proximity data structures. *Journal of the European Mathematical Society*, 9(2):253–275, 2007. preliminary version published in FOCS 2006. doi:10.4171/JEMS/79.

51 Shay Mozes and Eyal E. Skop. Efficient vertex-label distance oracles for planar graphs. *Theory Comput. Syst.*, 62(2):419–440, 2018. doi:10.1007/s00224-017-9827-0.

52 Assaf Naor and Terence Tao. Scale-oblivious metric fragmentation and the nonlinear dvoretzky theorem. *Israel Journal of Mathematics*, 192(1):489–504, 2012. doi:10.1007/s11856-012-0039-7.

53 Ilan Newman and Yuri Rabinovich. A lower bound on the distortion of embedding planar metrics into euclidean space. In *SCG '02: Proceedings of the eighteenth annual symposium on Computational geometry*, pages 94–96. ACM Press, 2002. doi:10.1145/513400.513412.

54 Mikkel Thorup. Compact oracles for reachability and approximate distances in planar digraphs. *Journal of the ACM*, 51(6):993–1024, November 2004. doi:10.1145/1039488.1039493.

55 Mikkel Thorup and Uri Zwick. Approximate distance oracles. *J. ACM*, 52(1):1–24, 2005. preliminary version published in STOC 2001. doi:10.1145/1044731.1044732.

56 Virginia Vassilevska Williams, Yinzhan Xu, and Zixuan Xu. Simpler and higher lower bounds for shortcut sets. In *Proceedings of the 2024 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2643–2656. SIAM, 2024. doi:10.1137/1.9781611977912.94.

A Pathwidth property in Theorem 5

This section is devoted to showing that there is a permutation σ such that the DAGs $D_\sigma, D_{\sigma^{-1}}$ created during the proof of Theorem 5 have pathwidth $O(tw \cdot \log n)$. Our proof of Theorem 5 holds with respect to an arbitrary permutation σ. To get DAGs with small pathwidth we will use a specific permutation. Let B be the separator bag (as above), and let $C_1, \ldots, C_g$ be the connected components of $G \setminus B$ (in the undirected sense). We will construct σ so that the vertices of each connected component will appear consecutively along σ. That is, the vertices of B will be the $|B|$ first vertices in σ. Then the vertices of C_1, then C_2, and so on, where the vertices of C_g will appear last. The internal order σ_i among the vertices of C_i is determined recursively in the same manner. That is, we find a separator bag B_i in $G[C_i]$. The vertices of B_i will appear first in σ_i, and afterwards, consecutively, the vertices of each connected component of $G[C_i] \setminus B_i$. We will construct two DAGs $D_\sigma, D_{\sigma^{-1}}$ with respect to the permutations σ, σ^{-1}.

Denote $\sigma = (v_1, \ldots, v_n)$. Let $x \in B$ be some vertex in the separator bag and consider the DAG D_x constructed in Lemma 6. Observe that D_x has pathwidth 2. For $i \in [1, n-1]$, let $X_{x,i} = \{v_i, u_i, u_{i+1}\}$, and $X_{x,n} = \{v_n, u_n\}$. Clearly, by the definition of D_x, $(X_{x,1}, X_{x,2}, \ldots, X_{x,n})$ is a path decomposition of D_x. In particular, there is a one-to-one correspondence between vertices and bags, and the order of the bags along the path decomposition is σ, the order of the vertices in the permutation. See Figure 4 of the path decomposition of the bidirected star from Figure 2.

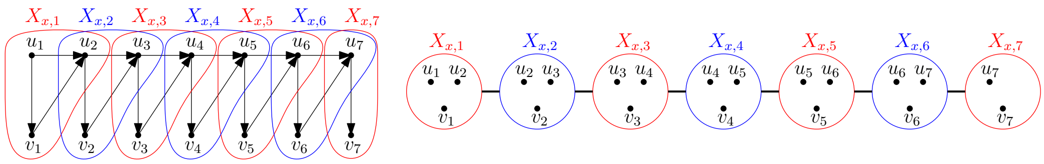

Figure 4 Bag decomposition of path construction for the biderected star.

The path decomposition for the entire DAG D_σ is constructed recursively in the natural manner. Suppose by induction that for each connected component $C_j = \{v_\alpha, v_{\alpha+1}, \ldots, v_\beta\}$ of $G \setminus B$ there is a path decomposition $\mathcal{P}_j = (X'_\alpha, \ldots, X'_\beta)$ of D_{σ_j}, where each vertex $v_i \in C_j$ belongs only to the bag X'_i. Further, we assume that the width of $\mathcal{P}_j$ is at most $2 \cdot (tw+1) \cdot \log |C_j|$. For each vertex $x \in B$ we create the DAG D_x with the path decomposition $(X_{x,1}, X_{x,2}, \ldots, X_{x,n})$. The path decomposition of D_σ will be $\mathcal{P} = (X_1, \ldots, X_n)$, where the bag X_i of $x_i \in C_j$ will consist of the union of X'_i with $\{X_{x,j}\}_{x \in B}$ (the bag of each $x_j \in B$ is only the union of X'_i with $\{X_{x,j}\}_{x \in B}$). Note that each vertex of D_σ is contained in a consecutive subset of bags in $\mathcal{P}$. This is as each original vertex $v \in V$ is contained in a single bag, and each other vertex was contained in a consecutive subset, and is contained now in respective consecutive subset. Using the fact that each bag $X_{x,j}$ contains only two vertices beyond v_j, and the inductive hypothesis, the width of the resulting path decomposition is

$$2|B| + 2(tw+1) \cdot \max_j \log |C_j| \le 2(tw+1) + 2(tw+1) \cdot \log \frac{n}{2} = 2(tw+1) \cdot \log n \ .$$

The bound on the pathwidth of $D_{\sigma^{-1}}$ follows by the exact same lines. The theorem now follows.