Graph Spanners for Group Steiner Distances

Davide Bilò
Department of Information Engineering, Computer Science and Mathematics, University of L'Aquila, Italy

Luciano Gualà
Department of Enterprise Engineering, University of Rome "Tor Vergata", Italy

Stefano Leucci
Department of Information Engineering, Computer Science and Mathematics, University of L'Aquila, Italy

Alessandro Straziota
Department of Enterprise Engineering, University of Rome "Tor Vergata", Italy

Abstract

A spanner is a sparse subgraph of a given graph G which preserves distances, measured w.r.t. some distance metric, up to a multiplicative stretch factor. This paper addresses the problem of constructing graph spanners w.r.t. the *group Steiner metric*, which generalizes the recently introduced beer distance metric. In such a metric we are given a collection of *groups* of *required* vertices, and we measure the distance between two vertices as the length of the shortest path between them that traverses at least one required vertex from each group.

We discuss the relation between *group Steiner spanners* and classic spanners and we show that they exhibit strong ties with *sourcewise* spanners w.r.t. the shortest path metric. Nevertheless, group Steiner spanners capture several interesting scenarios that are not encompassed by existing spanners. This happens, e.g., for the *singleton case*, in which each group consists of a single required vertex, thus modeling the setting in which routes need to traverse certain points of interests (in any order).

We provide several constructions of group Steiner spanners for both the *all-pairs* and *single-source* case, which exhibit various size-stretch trade-offs. Notably, we provide spanners with almost-optimal trade-offs for the *singleton* case. Moreover, some of our spanners also yield novel trade-offs for classical sourcewise spanners.

Finally, we also investigate the query times that can be achieved when our spanners are turned into group Steiner distance oracles with the same size, stretch, and building time.

2012 ACM Subject Classification Theory of computation → Sparsification and spanners; Mathematics of computing → Graph algorithms

Keywords and phrases Network sparsification, Graph spanners, Group Steiner tree, Distance oracles

Digital Object Identifier 10.4230/LIPIcs.ESA.2024.25

Related Version *Full Version*: https://arxiv.org/abs/2407.01431

1 Introduction

Given an edge-weighted graph $G = (V, E)$, the distance between two vertices s, t is typically measured as the length of a shortest path having s and t as endvertices. Such *shortest path metric* is pervasive in the study of optimization problems on graphs, yet there are natural scenarios that cannot be readily captured by such a metric. For example, consider the case in which a route from s to t needs to pass through at least one vertex from a distinguished set $R \subseteq V$ of *required* vertices. These vertices might represent, e.g., grocery stores on your commute to work, charging stations when planning a trip in an electric vehicle, or special hosts when routing packets in a communication network. The above metric was introduced

32nd Annual European Symposium on Algorithms (ESA 2024).
Editors: Timothy Chan, Johannes Fischer, John Iacono, and Grzegorz Herman; Article No. 25; pp. 25:1–25:17
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

in [5] under the name *beer-distance*.[1] In particular, several works have been devoted to the problem of building a compact data structure that is able to quickly report (exact or approximate) beer distances between pair of vertices for special classes of graphs, such as outer planar graphs, interval graphs, or bounded tree-width graphs [5, 6, 15, 20, 21]. Such data structures are the beer distance rendition of *distance oracles*, which are analogous data structures for the shortest path metric. Distance oracles and the closely related concept of *graph spanners* have received a vast amount of attention in the area of graph sparsification. Informally, an α-spanner of G is a sparse subgraph H of G that preserves the all-pairs distances in G up to a multiplicative *stretch* factor of α, and a distance oracle can be seen as data structure which allows for quick queries on the underlying spanner.[2]
The above discussion begs the following two natural questions:

- What happens if paths are required to traverse more than one kind of required vertices? For example, in the commute from home to work one needs to visit both a grocery store *and* a gas station, in some order.
- What can be said about beer-distances for *general* graph, i.e., when G does not fall into one of the special classes of graphs mentioned above?

Answering these questions is the focus of our paper, which will be devoted to designing spanners and distance oracles for general graphs and for the natural generalization of beer distance, which we name *group Steiner distance*.

Formally, given an undirected connected graph $G = (V, E)$ on n vertices and with non-negative edge weights, and a collection of $k \geq 1$ (not necessarily disjoint) *groups* $R_1, \ldots, R_k \subseteq V$ of *required* vertices, a *group Steiner path* between two vertices s and t is a (not necessarily simple) path π between s and t in G such that π includes at least one vertex from each group R_i. The *group Steiner distance* between s and t is the length (w.r.t. the edge weights, with multiplicity) of the shortest group Steiner path between s and t (see Figure 1).

Not surprisingly, as we discuss in the full version of the paper, the problem of computing the group Steiner distance between two vertices is NP-hard in general for large values of k. On the other hand, the group Steiner distance between two vertices can be computed in *Fixed Parameter Tractable (FPT)* time $2^k k n^{O(1)}$, which is polynomial when $k = O(\log n)$. Notice that it is easy to imagine scenarios in which k is a small constant.

The group Steiner distance coincides with the beer distance for $k = 1$, but it also captures practical scenarios in which one wants to route entities through multiple points of interest, as in waypoint routing and other related motion planning problems [16, 26, 4].

Using this novel metric, we can define a *group Steiner α-spanner* which is the analogous of α-spanner when distances are measured w.r.t. the group Steiner distance metric. Group Steiner spanners exhibit interesting relations with classical graph spanners (for the shortest path metric). Indeed, one can observe that any shortest group Steiner path can be seen as the concatenation of up to $k+1$ sub-paths, each of which has s, t, or a vertex in $R = \cup_i R_i$ as endvertices, and is a shortest path in G (see Section 2.1 for the details). Hence any

[1] In [5], the required nodes in R correspond to breweries, and one seeks a shortest path from s to t among those that traverse at least one brewery.

[2] Actually, a distance oracle does not immediately imply the existence of a corresponding spanner, while a sparse spanner can always be thought as a compact distance oracle albeit with a large query time. Nevertheless, it is often the case that a spanner and its corresponding oracle are provided together. The main challenge in designing distance oracles lies in organizing the implicit distance information contained in the spanner in a way that allows for quick queries.

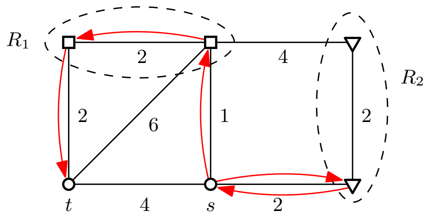

Figure 1 A shortest group Steiner path from s to t with length 9. The required vertices of the two groups R_1 and R_2 are depicted as squares and triangles, respectively.

sourcewise[3] $R \times V$ α-spanner for G is also a group Steiner spanner with the same stretch factor. As a consequence, all the upper bounds on the sizes for classical sourcewise (and all-pairs) α-spanners carry over to group Steiner spanners. It turns out that these two notions coincide when $R_1 = \cdots = R_k = R$, which implies that, in general, it is not possible to obtain better size-stretch trade-offs than the ones for sourcewise spanners. This is always the case for $k = 1$.

1.1 Our results

We mainly focus on group Steiner spanners and we show that their landscape exhibits a quite rich structure. In fact, while the problem of designing group Steiner spanners generalizes the problem of computing sourcewise spanners w.r.t. the shortest path metric, there are several interesting and natural classes of instances, like going from a source to a destination by passing through k waypoints (in any order), for which the lower bounds for the sourcewise spanners do not apply.

We start our investigation by considering exactly this scenario, that we call *singleton case* because each group contains only one vertex. First, we pinpoint the extremal size-stretch trade-offs: on the one hand it is possible to build a group Steiner spanner with $O(kn)$ edges that preserves *exact distances*; on the other hand $n - 1$ edges are already sufficient to build a group Steiner *tree spanner* with stretch 2.[4] Both spanners can be constructed in polynomial-time. Moreover, we show that both results are tight, meaning that there are instances in which any group Steiner spanner preserving exact distances must contain $\Omega(kn)$ edges and instances for which any group Steiner spanner with stretch strictly less than 2 must contain at least n edges. We then consider intermediate stretch factors and show that $O(n/\varepsilon^2)$ edges suffice to build a group Steiner spanner having stretch $\gamma + \varepsilon$ in polynomial time, where γ is the approximation factor of a polynomial-time algorithm for the minimum-cost metric Hamiltonian path problem[5] and $\varepsilon \in (0, 1)$. If one is willing to settle for an FPT building time w.r.t. k, then the above stretch can be improved to $1 + \varepsilon$. These results are summarized in Table 1 (a).

For general instances, we provide two *recipes*, one of which can be thought as a generalization of Theorem 3 in [17]. These recipes use existing α-spanners w.r.t. the shortest-path metric to construct group Steiner spanners with stretch $2\alpha + 1$ (see Table 1 (b)). One recipe

[3] A sourcewise $R \times V$ α-spanner of G is a subgraph of G that approximates all the $R \times V$ distances within a multiplicative factor of α.

[4] We observe that, in general, no sourcewise spanner w.r.t. the shortest path metric with stretch 2 and size $n - 1$ exists. As an example, consider a complete unweighted $R \times (V \setminus R)$ bipartite graph and observe that none of the edges can be discarded if we are seeking for a sourcewise $R \times V$ α-spanner, for any $\alpha < 3$.

[5] The formal definition of this problem is given in the full version of the paper.

Table 1 Summary of our results. The size-stretch trade-offs of rows marked with ☆ are tight. The rightmost column reports the query time attained by a distance oracle with the same stretch, asymptotic size, and building time class (i.e., polynomial or FPT) of the associated spanner. We use γ to denote the approximation ratio of a polynomial-time algorithm for the minimum-cost metric Hamiltonian path problem, while $R = \bigcup_i R_i$ denotes the set of all required vertices. Expanded versions of (b) and (c) are respectively shown in Table 3 and in the full version of the paper.

(a) All-pairs, singleton case ($|R_i| = 1\ \forall i = 1, \dots, k$).

	Stretch	Size	Building time	Reference	D.O. query time
☆	1	$O(kn)$	polynomial	Theorem 7	$O(2^k \cdot k^3)$
	$1+\varepsilon$	$O(n/\varepsilon^2)$	FPT	Theorem 8	$O(1/\varepsilon^2)$
	$\gamma+\varepsilon$	$O(n/\varepsilon^2)$	polynomial	Theorem 9	$O(1/\varepsilon^2)$
☆	2	$n-1$	polynomial	Theorem 11	$O(1)$

(b) All-pairs, general group sizes.

Stretch	Size	Building time	Reference	D.O. query time
$2\alpha+1$	$kn + \lvert\bigcup_i (R_i \times R_i\ \alpha\text{-spanner})\rvert$	polynomial	Theorem 13	$O(2^k k \cdot \lvert R\rvert^2 + \lvert R\rvert^3)$
$2\alpha+1$	$n + \lvert R \times R\ \alpha\text{-spanner}\rvert$	polynomial	Theorem 14	$O(2^k k \cdot \lvert R\rvert^2 + \lvert R\rvert^3)$

(c) Single-source, general group sizes.

Stretch	Size	Building time	Reference	D.O. query time
1	$O(2^k n)$	FPT	See full version	$O(1)$
3	$n-1$	FPT	See full version	$O(1)$
$\alpha+1$	$O(n) + \lvert R \times R\ \alpha\text{-spanner}\rvert$	polynomial	See full version	$O(2^k k \cdot \lvert R\rvert^2 + \lvert R\rvert^3)$

(Theorem 14) upper bounds the size of the group Steiner spanner in terms of n and $|R|$, while the other (Theorem 13) provides bounds w.r.t. the sizes of the groups R_i. Despite the simplicity of our constructions, by combining Theorem 14 with the spanners in [1, 8, 14, 18], we obtain new trade-offs for sourcewise $R \times V$ spanners w.r.t. the shortest path metric. These results are marked with ❇ in Table 3, which summarizes the current state of the art for group Steiner spanners.

We also consider the *single-source* case[6], and we show that any $R \times R$ α-spanner for the shortest path metric can be used to construct a single-source group Steiner spanner with stretch $\alpha + 1$. We also provide ad-hoc constructions achieving either the minimum stretch $\alpha = 1$ or the minimum conceivable size $n - 1$. These results are summarized in Table 1 (c)), while the corresponding current state of the art for the single-source case is given the full version of the paper. Finally, as an instrumental result to achieve our trade-offs, we also consider the *single-pair* case and show that any group Steiner path can be sparsified to have $O(n)$ edges without increasing its length (see Corollaries 5 and 6).

Turning our spanners into group Steiner distance oracles. We also investigate the problem of turning our group Steiner spanners into group Steiner distance oracles. For each of our spanners, we provide a corresponding oracle with the same stretch, the same asymptotic

[6] Roughly speaking, a single-source group Steiner spanner is only required to contain approximate group Steiner paths between a distinguished source vertex and all other vertices in V. See Section 2.1 for a formal definition.

Table 2 Known bounds for classical spanners for both weighted and unweighted graphs that yield the best trade-offs when used in our recipes of Table 1 (b) and (c). *Pairw.* denotes a pairwise spanner, i.e., a spanner which is only required to (approximately) preserve distances between pairs of vertices in $P \subseteq V^2$. Randomized constructions are marked with ⚂. A *mixed* stretch of (α, β) means that the corresponding spanner H approximates the distance from s to t in G within a *multiplicative* stretch of α plus an *additive* stretch of β times the maximum edge-weight, say $W_{s,t}$, along a shortest path from s to t in G, i.e., $d_H(s,t) \leq \alpha d_G(s,t) + \beta W_{s,t}$. Notice that a spanner with a mixed stretch of (α, β) is also a spanner with a purely multiplicative stretch of $\alpha + \beta$.

Weighted

Type	Stretch	Size	Ref.
$V \times V$	$2h-1$	$O(n^{1+1/h})$	[3]
Pairw.	1	$O(n + \|P\|n^{1/2})$	[13]
Pairw.	1	$O(n\|P\|^{1/2})$	[13]
Pairw.	⚂ $(1,2)$	$O(n\|P\|^{1/3})$	[1]
Pairw.	⚂ $(1,4)$	$O(n\|P\|^{2/7})$	[1]
$R \times V$	$4h-1$	$O(n + n^{1/2}\|R\|^{1+1/h})$	[17]
$R \times R$	$(1, 2+\varepsilon)$	$O(n\|R\|^{1/2}/\varepsilon)$	[18]

Unweighted

Type	Stretch	Size	Ref.
Pairw.	1	$O(n^{2/3}\|P\|^{2/3} + n\|P\|^{1/3})$	[8]
$R \times V$	$(1,2)$	$\widetilde{O}(n^{5/4}\|R\|^{1/4})$	[9]
$R \times V$	$(1,4)$	$\widetilde{O}(n^{11/9}\|R\|^{2/9})$	[23]
$R \times V$	$(1,6)$	$\widetilde{O}(n^{6/5}\|R\|^{1/5})$	[23]
$R \times R$	$(1,2)$	$O(n\|R\|^{1/2})$	[14]

size and the same class of building time.[7] The distance query times are reported in Table 1. Some of these query times are constant, and in this case our oracles are also able to report a corresponding group Steiner path in an additional time proportional to the number path's edges. The remaining query times are exponential in k and this is unavoidable. Indeed, consider the group Steiner spanner in Table 1 (a) with stretch 1 and polynomial building time. It can be shown (see full version of the paper for details) that any corresponding oracle with polynomial query time would be able to report the cost of a minimum-cost metric Hamiltonian path which is known to be NP-hard. For similar reasons, any oracle for general group sizes that has polynomial building time and stretch $\log^{2-\varepsilon} k$, for constant $\varepsilon > 0$, cannot have polynomial query time (regardless of its size), even for the single-pair oracle, due to the inaproximability of computing group Steiner distances.

Finally, we emphasize that the two oracles for the singleton case with stretch $1+\varepsilon$ and $\gamma+\varepsilon$ are, in some sense, tight. Indeed, since a distance oracle can be used to compute group Steiner distances, the building time of the former oracle (having stretch $1+\varepsilon$) cannot be improved to polynomial time since the minimum-cost metric Hamiltonian path problem is APX-hard [22]; while improving the stretch to a value better than γ in the latter oracle would provide a better than γ-approximation for the minimum-cost metric Hamiltonian path problem.

Due to space limitations, the results for the single-source case, along with some constructions of group Steiner oracles, and some proofs are omitted and can be found in the full version of the paper.

1.2 Related work

There is a huge body of literature on graph spanners and distance oracles w.r.t. the shortest-path metric. Since we mostly focus on the stretch-size trade-offs of our group Steiner spanners, in the following we discuss the related work providing the best size-stretch trade-offs for spanners. The reader interested in efficient computation of spanners is referred to [24] and to the references therein.

[7] We classify the building times into one of two coarse classes, namely *polynomial* and *FPT*, depending on whether the spanner/oracle can be computed in time $n^{O(1)}$ or $f(k)n^{O(1)}$.

Table 3 State of the art for all-pairs group Steiner spanners of weighted graphs (left), along with additional bounds that only apply to unweighted graphs (right). Randomized constructions are marked with ⚁. Combinations that are dominated by results with a better size-stretch trade-off are omitted. All building times are polynomial. Results marked with ❇ are also novel sourcewise $R \times V$ spanners w.r.t. the shortest path metric. The classical spanners used for our combinations are reported in Table 2.

All-pairs Weighted

	α	Size	Reference
	⚁ 3	$O(n^{4/3}\|R\|^{1/3})$	Ob. 2 + [1]
	3	$O(kn + kR_{\max}^2 n^{1/2})$	Th. 13 + [13]
	3	$O(n + \|R\|^2 n^{1/2})$	Ob. 2 + [17]
	3	$O(n^{3/2})$	Ob. 2 + [3]
	⚁ 5	$O(n^{9/7}\|R\|^{2/7})$	Ob. 2 + [1]
	5	$O(n^{4/3})$	Ob. 2 + [3]
❇	⚁ 7	$O(n\|R\|^{2/3})$	Th. 14 + [1]
	7	$O(n + \|R\|^{3/2} n^{1/2})$	Ob. 2 + [17]
	7	$O(n^{5/4})$	Ob. 2 + [3]
❇	$7+\varepsilon$	$O(n\|R\|^{1/2}/\varepsilon)$	Th. 14 + [18]
	$2h-1$	$O(n^{1+1/h})$	Ob. 2 + [3]
	$4h-1$	$O(n + n^{1/2}\|R\|^{1+1/h})$	Ob. 2 + [17]

All-pairs Unweighted

	α	Size	Reference
❇	3	$O(n^{2/3}\|R\|^{4/3} + n\|R\|^{2/3})$	Th. 14 + [8]
	3	$O(kn^{2/3}R_{\max}^{4/3} + knR_{\max}^{2/3})$	Th. 13 + [8]
	3	$\widetilde{O}(n^{5/4}\|R\|^{1/4})$	Ob. 2 + [9]
	5	$\widetilde{O}(n^{11/9}\|R\|^{2/9})$	Ob. 2 + [23]
	7	$\widetilde{O}(n^{6/5}\|R\|^{1/5})$	Ob. 2 + [23]
	7	$O(n + \|R\|^{3/2} n^{1/2})$	Ob. 2 + [17]
❇	7	$O(n\|R\|^{1/2})$	Th. 14 + [14]

A classical result shows that it is possible to build all-pairs spanners with stretch $2h-1$ and size $O(n^{1+\frac{1}{h}})$, for every integer $h \geq 1$ [3] . For $h \in \{1, 2, 3, 5\}$ these asymptotic bounds are (unconditionally) tight [29, 32], and in general, for every h, matching asymptotic lower bounds can be proved assuming the Erdős girth conjecture [19]. The reader is referred to [2] for a survey which also discusses other notions of stretch (e.g., additive and mixed distortions), as well as generalizations in which good distance approximations only need to be maintained between specific pairs of vertices of interest (as sourcewise, subsetwise, or pairwise spanners). In [28, 11, 12], the authors show how to build, in polynomial time, distance oracles achieving the same size-stretch trade-offs and having constant query time.

As mentioned above, group Steiner paths for the special case $k = 1$ are known in the literature as *beer paths*. The notion of beer paths (and the corresponding *beer distance*) has been first introduced in [5, 6], where the authors show how to construct *beer distance oracles* for outerplanar graphs that are able to report exact beer distances. Subsequent works showed how to construct beer distance oracles for interval graphs [15] and graphs with bounded treewidth [20]. Construction of beer distance oracles for graphs that admit either good tree decomposition or good triconnected component decompositions have been studied in [21]. None of the above results yields a non-trivial beer distance oracle for general graphs, hence they cannot readily be compared with group Steiner distance oracles (which also handle $k > 1$ groups).

Several distance metrics involving paths that are required to traverse groups of vertices have already been considered in the context of optimization problems. For example, the *generalized TSP* problem [26] asks to find a shortest tour in a graph that visits at least one vertex from each group, which corresponds to finding the shortest group Steiner path from a required vertex in the optimal tour to itself. Elbassoni et al., studied a geometric version of the above problem called *Euclidean group TSP* [16]. A related optimization problem involving both waypoints and capacity constraints is known as *waypoint routing* and has been studied in [4].

Finally, we point out that our metric should not be confused with a different measure also called *Steiner distance* that is defined as the weight of the lightest Steiner tree connecting as set of vertices (see, e.g., [10, 25]).

2 Preliminaries

2.1 Notation

We denote by $G = (V, E)$ a connected undirected graph with n vertices, m edges, and with a non-negative edge-weight $w(e)$ associated with each $e \in E$, We also denote by $\mathcal{R} = \{R_1, \dots, R_k\}$ a collection of k non-empty subsets of V, which we refer to as *groups* of *required* vertices. We denote by $R = \bigcup_{i=1}^{k} R_i$.

Throughout this work, we use the term path to refer to walks, i.e., our paths are not necessarily simple. A *group Steiner path* from s to t in G w.r.t. $\mathcal{R}$ is a path from s to t that contains at least one vertex from each group. The length $w(\pi)$ of a path π is the sum of all its edge weights, with multiplicity (see Figure 1). The *group Steiner distance* $\sigma_G(s, t \mid \mathcal{R})$ between s and t w.r.t. $\mathcal{R}$ in G is the length of the shortest group Steiner path from s to t in G. Whenever $\mathcal{R}$ is clear from the context, we omit it from the notation.

Given $\alpha \geq 1$, a *group Steiner α-spanner of G* w.r.t. $\mathcal{R}$ is a spanning subgraph H of G such that:

$$\sigma_H(s, t) \leq \alpha \cdot \sigma_G(s, t), \tag{1}$$

for every $s, t \in V$. We denote by $|H|$ the *size* of H which corresponds to the number of edges contained in H. When $\alpha = 1$, a group Steiner 1-spanner of G is called a *group Steiner preserver* as it preserves group Steiner distances between all-pairs of vertices.

The classical notion of graph α-spanner w.r.t. the shortest-path metric is analogous, once the group Steiner distances $\sigma_H(s, t)$ and $\sigma_G(s, t)$ in Equation (1) are replaced with the lengths $d_H(s, t)$ and $d_G(s, t)$ of a shortest path in H and G, respectively.

Regardless of the distance metric of interest, we can restrict the pairs of vertices for which the distances in H must α-approximate the corresponding distances in G to those in the set $S \times T$, for some choice of $S, T \subseteq V$. The definition of α-spanner in Equation (1) corresponds to the *all-pairs* case in which $S, T = V$. In *sourcewise* spanners we have $S \subseteq V$ and $T = V$. *Single-source* spanners are a special case of sourcewise spanners in which $S = \{s\}$, for some source vertex $s \in V$. Finally, in *subsetwise* spanners we have $S = T$, with $S \subseteq V$.

Given a path π from s to v in G and a path π' from v to t in G, we denote by $\pi \circ \pi'$ the path obtained by concatenating π with π'. Moreover, given two occurrences x, y of vertices in π, we denote by $\pi[x : y]$ the subpath of π between x and y. Notice that the same vertex can appear multiple times in π however, whenever the occurrence of interest is clear from context, we may slightly abuse the notation and use vertices in place of their specific occurrences.

We conclude this preliminary section by observing a structural property of group Steiner paths and its important immediate consequences. The property can be proved by a simple cut-and-paste argument, and intuitively shows that a shortest group Steiner path can be seen as a concatenation of up to $k + 1$ subpaths, each of which is a shortest path in G between two vertices in $R \cup \{s, t\}$. This is formalized in the following:

▶ **Lemma 1.** *Let $\pi = \langle s = v_0, v_1, \dots, v_\ell = t \rangle$ be a shortest group Steiner path between two vertices s and t in G. Let $j_1, \dots, j_h$ be h indices such that $0 \leq j_1 < j_2 < \cdots < j_h \leq \ell$ and $\{v_{j_1}, \dots, v_{j_h}\} \cap R_i \neq \emptyset$ for all $i = 1 \dots, k$. For every $i = 0, \dots, h$, $\langle v_{j_i}, v_{j_i+1}, \dots, v_{j_{i+1}} \rangle$ is a shortest path between v_{j_i} and $v_{j_{i+1}}$ in G, where $j_0 = 0$ and $j_{h+1} = \ell$.*

The above property immediately implies the following results, whose simple proofs are given in the full version of the paper:

▶ **Observation 2.** *Any sourcewise $R \times V$ (and hence also any all-pairs $V \times V$) α-spanner w.r.t. the shortest path metric is an all-pairs group Steiner α-spanner.*

▶ **Observation 3.** *Any group Steiner α-spanner with $R_1 = R_2 = \cdots = R_k = R$ is a sourcewise $R \times V$ α-spanner w.r.t. the shortest path metric.*

2.2 Single-pair group Steiner spanners

Our technical discussion begins with a result (whose proof is given in the full version of the paper) for the single-pair case that, other than being interesting in its own regard, will also be instrumental to construct our group Steiner spanners.

▶ **Lemma 4.** *Let π be a group Steiner path from s to t in G. We can process π in polynomial time to build a group Steiner path π' from s to t in G that traverses at most $2(n-1)$ edges (with multiplicity) and such that $w(\pi') \leq w(\pi)$.*

From Lemma 4, we can easily derive the following corollaries. The first corollary is a direct consequence that a shortest group Steiner path can be computed in $2^k k \cdot n^{O(1)}$ time, while the second corollary, which holds only for the singleton case, comes from the fact that we can compute a 3/2-approximation of the shortest group Steiner path in G. Both the aforementioned results are discussed in the full version of the paper.

▶ **Corollary 5.** *We can compute a single-pair shortest group Steiner path of size at most $2(n-1)$ in $2^k k \cdot n^{O(1)}$ time.*

▶ **Corollary 6.** *For the singleton case in which $|R_i| = 1$ for all $i = 1, \ldots, k$, there is a polynomial-time algorithm that computes a single-pair group Steiner path with stretch $3/2$ and size at most $2(n-1)$.*

3 Group Steiner spanners in the singleton case

In this section we consider the special case in which each group R_i in $\mathcal{R}$ contains a single vertex r_i, hence $R = \{r_1, r_2, \ldots, r_k\}$.

3.1 A group Steiner preserver

Lemma 1 implies that union of k shortest-path trees $T_1, \ldots, T_k$ of G, where T_i is rooted in r_i, is a group Steiner preserver of size $O(kn)$. The size of such a preserver is asymptotically optimal, even when G is unweighted. To see this, let $k \geq 3$ and consider a graph G consisting of a cycle $\langle r_1, r_2, \ldots, r_k, r_1 \rangle$ on the k required vertices, along with $n-k$ additional vertices $v_1, \ldots, v_{n-k}$ (see Figure 4).[8] All vertices v_i have $\lfloor k/3 \rfloor$ incident edges, where the j-th such edge is $e_{i,j} = (v_i, r_{1+3(j-1)})$. Given any v_i and $r_{1+3(j-1)}$, there exists a unique simple path from v_i to $r_{2+3(j-1)}$ spanning v_i and $\{r_1, \ldots, r_k\}$, and such a path uses the edge $e_{i,j}$. As a consequence, any group Steiner preserver H must contain all edges $e_{i,j}$, which implies that H must have size $\Omega(kn)$.

▶ **Theorem 7.** *In the singleton case, it is possible to compute a group Steiner preserver of size $O(kn)$ in polynomial time. Moreover, there are unweighted graphs G such that any group Steiner preserver of G has size $\Omega(kn)$.*

A corresponding distance oracle with query time $O(2^k k^3)$ is discussed in the full version of the paper.

[8] For $k \in 1, 2$, a trivial lower bound of $\Omega(n) = \Omega(kn)$ clearly holds.

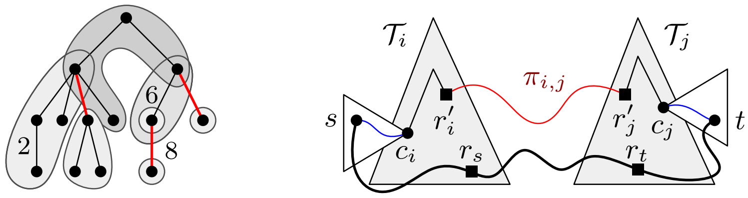

Figure 2 On the left: a tree with edge-weights, where unlabeled edges have weight 1, and a possible decomposition into micro-trees as computed by our procedure with $W = 6$. The edges (v, u_i) are highlighted in red. On the right: a qualitative depiction of the paths used in the analysis of the stretch of our group Steiner $(1 + \varepsilon)$-spanner. The shortest group Steiner path between s and t is shown in bold, while π_s and π_t are shown in blue. The white triangles are the trees in F_i and F_j rooted in c_i and c_j, respectively.

3.2 A spanner with stretch $1 + \varepsilon$ and $O(n/\varepsilon^2)$ edges

By the lower bound in Theorem 7, any group Steiner α-spanner of size $o(kn)$ must have a stretch of $\alpha > 1$. In this section we present an algorithm that builds a group Steiner spanner of linear size and stretch $1 + \varepsilon$ in time $2^k kn^{O(1)}$, for every constant $\varepsilon > 0$. We then show how to reduce the building time of our group Steiner spanner to polynomial at the cost of increasing the stretch to $\gamma + \varepsilon$ by using a γ-approximation algorithm for the minimum-cost metric Hamiltonian path problem as a black-box. By keeping the same trade-offs among size, stretch, and building time, we show how to convert both spanners to group Steiner distance oracles with query time $O(1/\varepsilon^2)$. The pseudocode for constructing our $(1 + \varepsilon)$ group Steiner spanner can be found in Algorithm 1.

We start by defining an auxiliary *clustering procedure* that will be useful for describing all our spanner constructions. Given a set of *centers* $C \subseteq V$, the procedure computes a spanning forest F of G with $|C|$ rooted trees with the following properties: (i) the root of each tree is a distinct vertex in C, (ii) the unique path in F from a vertex $v \in V$ to the root c of its tree is a shortest path between v and c in G, (iii) c is (one of) the closest center(s) to v. The procedure first constructs a graph G' which is obtained from G by adding a *dummy source vertex* s^* along a *dummy edge* (s^*, c) of weight 0 for each $c \in C$. Then, it computes a shortest-path tree $\widetilde{T}$ of G' from s^* that contains all the dummy edges. Finally, it returns the forest F obtained by deleting s^* (and all its incident dummy edges) from $\widetilde{T}$.

We show how to compute our group Steiner spanner H. Let T be a Steiner tree of G w.r.t. the required vertices R whose total weight $w(T)$ is at most twice that of the optimal Steiner tree T^*. It is known that T can be computed in polynomial time [31].

We subdivide T into $O(\frac{1}{\varepsilon})$ edge-disjoint *micro-trees* $\mathcal{T}_1, \ldots, \mathcal{T}_h$ that altogether span all vertices of T and such that each micro-tree is a subtree of T of weight at most $W = \frac{\varepsilon}{4} w(T)$.

The subdivision is computed by the following iterative procedure, which keeps track of the part T' yet to be divided. Initially, T' is obtained by rooting T in an arbitrary vertex. As long as T' has weight larger than W, we find a node v such that the weight of the subtree T'_v of T' rooted in v is larger than W and the depth of v in T' is maximized. Let $u_1, u_2, \ldots$ be the children of v in T', and let i be the smallest index such that $\sum_{j=1}^{i} \big(w(T'_{u_j}) + w(u_j, v)\big) > W$. We create two micro-trees: one consists of the subtree of T'_v induced by v and all the vertices in $T'_{u_1}, \ldots, T'_{u_{i-1}}$ (if any), and the other consists of T'_{u_i}. Notice that our choice of v and i ensures that both micro-trees have weight at most W. Finally, we delete all vertices in $T'_{u_1}, \ldots, T'_{u_i}$ from T' (along with their incident edges) and repeat.

Algorithm 1 Our algorithm for computing a group Steiner $(1+\epsilon)$-spanner of a graph G in the singleton case. R denotes the set of required vertices.

/* Returns a rooted forest F of G with one tree T_c for each $c \in C$. If vertex v is in T_c then $d_{T_c}(v,c) = d_G(v,c) \le d_G(v,c')$ for $c' \in C$. */

Function Cluster(C):
  $G' \leftarrow$ graph obtained from G by adding a new vertex s^* and all edges (s^*, c) of weight 0 for $c \in C$;
  $\widetilde{T} \leftarrow$ a shortest path tree of G' from s^* that contains all edges incident to s^*;
  $F \leftarrow$ the forest obtained from $\widetilde{T}$ by deleting s^* and all its incident edges;
  return F;

/* Returns a partition of a group Steiner tree T of G w.r.t. R into a collection $\mathcal{T}$ containing $O(\frac{1}{\epsilon})$ edge-disjoint *micro-trees*. */

Function PartitionIntoMicroTrees(T):
  $W \leftarrow \frac{\varepsilon}{4} w(T)$;
  $\mathcal{T} \leftarrow \emptyset$; // A collection of micro-trees
  $T' \leftarrow$ tree obtained by rooting T in an arbitrary vertex;
  while $w(T') > W$ **do**
    $v \leftarrow$ deepest node in T' such that $w(T_v) \ge W$;
    $u_1, \dots, u_k \leftarrow$ children of v in T', in an arbitrary order;
    $i \leftarrow$ smallest index such that $\sum_{j=1}^{i} \left(w(T'_{u_j}) + w(u_j, v) \right) > W$;
    $T'' \leftarrow$ subtree of T' induced by v and all the vertices in T_{u_j} for $j < i$;
    $\mathcal{T} \leftarrow \mathcal{T} \cup \{T'', T'_{u_i}\}$;
    $T' \leftarrow$ tree obtained from T' by deleting all the vertices in T_{u_j} for $j \le i$;
  return $\mathcal{T} \cup \{T'\}$;

$T \leftarrow$ a Steiner tree w.r.t. R such that $w(T)$ is at most twice the weight of an optimal Steiner tree;
$\mathcal{T}_1, \dots, \mathcal{T}_h \leftarrow$ PartitionIntoMicroTrees (T);
$\overline{G} \leftarrow$ complete graph with vertex set R, the weight of a generic edge (u,v) is $d_G(u,v)$;
for $i \leftarrow 1, \dots, h$ **do**
  for $j \leftarrow i, \dots, h$ **do**
    $\pi'_{i,j} \leftarrow$ minimum cost path among all Hamiltonian paths in $\overline{G}$ having an endvertex in $\mathcal{T}_i$ and the other endvertex in $\mathcal{T}_j$;
    $\pi''_{i,j} \leftarrow$ path obtained from π by replacing each edge (u,v) with a shortest path from u to v in G (w.r.t. the shortest path metric);
    $\pi_{i,j} \leftarrow$ sparsify $\pi''_{i,j}$ as shown in Lemma 4;

for $i \leftarrow 1, \dots, h$ **do**
  $F_i \leftarrow$ Cluster($V(T_i)$), where $V(T_i)$ denotes the set of vertices in T_i;

return $H = T \cup \left(\bigcup_{i=1}^{h} \bigcup_{j=i}^{h} \pi_{i,j} \right) \cup \left(\bigcup_{i=1}^{h} F_i \right)$;

We stop the above procedure as soon as $w(T') \le W$, and we choose T' as the last micro-tree of our subdivision (see Figure 2). Notice that the edges deleted in each iteration have a total weight of at least W, and that each iteration creates at most 2 micro-trees. It follows that the resulting collection contains at most $2\left(\frac{w(T)}{W} + 1\right) \le \frac{8}{\varepsilon} + 2 = O(\frac{1}{\varepsilon})$ micro-trees.

We then compute a complete graph $\overline{G}$ on the required vertices, where the weight of a generic edge (u, v) is $d_G(u, v)$. Then, for each unordered pair of (not necessarily distinct) micro-trees $\{\mathcal{T}_i, \mathcal{T}_j\}$, we consider all pairs (r, r') of required vertices such that r is in $\mathcal{T}_i$ and r' is in $\mathcal{T}_j$, compute a minimum-cost Hamiltonian path in $\overline{G}$ between r and r', and we call $\pi_{i,j}$ the shortest of such paths in which each edge of $\overline{G}$ has been replaced by the corresponding shortest path in G. Thanks to Lemma 4, we can assume that each $\pi_{i,j}$ contains at most $O(n)$ edges.

Finally, we compute h forests $F_1, \ldots, F_h$, where F_i is obtained by our clustering procedure using the vertices of $\mathcal{T}_i$ as centers.

Our group Steiner spanner H consists of the union of T, all paths $\pi_{i,j}$ for $1 \le i \le j \le h$, and all the edges in F_i for $1 \le i \le h$. The size of H is $O(n/\varepsilon^2)$ as each of (i) T, (ii) the $O(\frac{1}{\varepsilon^2})$ paths $\pi_{i,j}$, and (iii) $O(\frac{1}{\varepsilon})$ forests F_i, all contain $O(n)$ edges.

To analyze the stretch of H, fix a shortest group Steiner path π^* between any two vertices s, t in G and let r_s (resp. r_t) be the first (resp. last) required vertex encountered in a traversal of π^* from s to t. Let $\mathcal{T}_i$ and $\mathcal{T}_j$ the micro-trees containing r_s and r_t, respectively. Moreover, let r'_i and r'_j be the endvertices of $\pi_{i,j}$, where r'_i lies in $\mathcal{T}_i$ and r'_j lies in $\mathcal{T}_j$. Finally, let c_i (resp. c_j) be the root of the tree containing s in F_i (resp. t in F_j). The situation is depicted in Figure 2.

Notice that the weight of a minimum-cost group Steiner tree is a lower bound to $w(\pi^*)$, hence $\frac{w(T)}{2} \le \sigma_G(s, t)$. Moreover, since r_s is a center of the clustering procedure for F_i, we have $d_{F_i}(s, c_i) \le w(\pi^*[s : r_s])$, and symmetrically $d_{F_i}(c_j, t) \le w(\pi^*[t : r_t])$. Therefore:

$$\begin{aligned}\sigma_H(s,t) &\le d_{F_i}(s,c_i) + d_{\mathcal{T}_i}(c_i, r'_i) + w(\pi_{i,j}) + d_{\mathcal{T}_j}(r'_j, c_j) + d_{F_j}(r_j, t) \\ &\le w(\pi^*[s : r_s]) + W + w(\pi^*[r'_i : r'_j]) + W + w(\pi^*[r_t, t]) \\ &\le w(\pi^*) + 2 \cdot \frac{\varepsilon}{4} w(T) \le \sigma_G(s,t) + \varepsilon\sigma_G(s,t) = (1+\varepsilon)\sigma_G(s,t).\end{aligned} \tag{2}$$

Observe that all steps of the above construction can be carried out in polynomial time, except for the computation of the paths $\pi_{i,j}$, which requires time $2^k k n^{O(1)}$ (see the full version of the paper). Hence, we have the following:

▶ **Theorem 8.** *In the singleton case, it is possible to compute a group Steiner spanner having stretch $1 + \varepsilon$ and size $O(\frac{n}{\varepsilon^2})$ in $2^k k \cdot n^{O(1)}$ time.*

To obtain a polynomial building time, we can redefine each $\pi_{i,j}$ starting from a γ-approximation of minimum-cost Hamiltonian path between (a vertex of) $\mathcal{T}_i$ and (a vertex of) $\mathcal{T}_j$ in $\overline{G}$. We use $w(\pi_{i,j}) \le \gamma w(\pi^*[r'_i, r'_j])$ in Equation (2) to show that

$$\begin{aligned}\sigma_H(s,t) &\le w(\pi^*[s : r_s]) + W + \gamma w(\pi^*[r'_i : r'_j]) + W + w(\pi^*[r_t, t]) \\ &\le \gamma w(\pi^*) + 2 \cdot \frac{\varepsilon}{4} w(T) \le \gamma\sigma_G(s,t) + \varepsilon\sigma_G(s,t) = (\gamma+\varepsilon)\sigma_G(s,t).\end{aligned}$$

We can then state the following:

▶ **Theorem 9.** *In the singleton case, it is possible to compute a group Steiner spanner having stretch $\gamma + \varepsilon$ and size $O(\frac{n}{\varepsilon^2})$ in polynomial time, where γ is the approximation ratio for the minimum-cost metric Hamiltonian path problem.*

If G is weighted, we can choose $\gamma = \frac{3}{2}$ [33], while if G is unweighted (i.e., $\overline{G}$ is the metric closure of an unweighted graph) we can choose $\gamma = \frac{7}{5} + \delta$ [27, 30], for any constant $\delta > 0$.

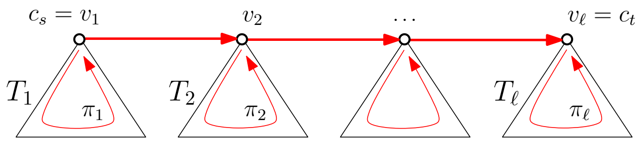

Figure 3 A qualitative depiction of the path constructed in the proof of Lemma 10. The path in bold is the unique path π from c_s to c_t in T. Each π_i is the Eulerian tour of the corresponding tree T_i, and the final path is in red.

A corresponding distance oracle. Here we show how to transform the above spanners into group Steiner distance oracles with the same size, stretch and building time. The oracles will be able to answer distance queries in $O(1/\varepsilon^2)$ time and report a corresponding path with an additional time proportional to the number of the path's edges.

We only discuss how to build the oracle with stretch $1+\varepsilon$ since the version with stretch $\gamma+\varepsilon$ is analogous. Our oracle explicitly maintains all the $O(1/\varepsilon^2)$ paths $\pi_{i,j}$ and their lengths, all the $h = O(1/\varepsilon)$ micro-trees $\mathcal{T}_1, \dots, \mathcal{T}_h$ and the forests $F_1, \dots, F_h$. Moreover, for each micro-tree $\mathcal{T}_i$, we maintain a linear-size data structure that, given any two vertices a and b of $\mathcal{T}_i$, returns the length of the (unique) path in $\mathcal{T}_i$ between a and b in constant time, and the corresponding path in time proportional to its number of edges.[9] Notice that the size of our oracle is $O(n/\varepsilon^2)$.

To answer a distance query for the vertices s and t, we look at all pairs of micro-trees $\mathcal{T}_i, \mathcal{T}_j$. For each such pair, arguments analogous to the ones used in the analysis of the stretch factor of the spanner, show that there exists a group Steiner path in H of length

$$\ell_{i,j}(s,t) = d_{F_i}(s, c_i) + d_{\mathcal{T}_i}(c_i, r'_i) + w(\pi_{i,j}) + d_{\mathcal{T}_j}(r'_j, c_j) + d_{F_j}(c_j, t),$$

where c_i (resp. c_j) is the root of the tree containing s in F_i (resp. t in F_j) and r'_i, r'_j are the endpoints of $\pi_{i,j}$. Notice that $\ell_{i,j}(s,t)$ can be evaluated in constant time. We return $\min_{1 \le i \le j \le h} \ell_{i,j}(s,t)$, which is guaranteed to be at most $(1+\varepsilon) d_G(s,t)$ by Equation (2). Once the pair i, j minimizing $\ell_{i,j}(s,t)$ is known, we can also report a group Steiner path of length $\ell_{i,j}(s,t)$ in constant time per path edge by navigating F_i (resp. F_j) from s (resp. t) to the root of its tree, and by querying our data structures for the micro-trees $\mathcal{T}_i$ and $\mathcal{T}_j$.

3.3 A tree spanner with a tight stretch of 2

We now describe how to obtain group Steiner *tree spanner* with stretch 2 (and $n-1$ edges).

We first compute a complete graph $\overline{G}$ on the required vertices, where the weight of a generic edge (u, v) is $d_G(u, v)$. We then compute an MST M of $\overline{G}$, and we construct a subgraph $\widetilde{M}$ of G by replacing each edge (u, v) in M with a shortest path between u and v in G. Finally, we select any spanning tree T of $\widetilde{M}$. Our spanner H is obtained as the union of T with the forest F computed by our clustering procedure using the vertices in T as centers.

Notice that H has $n-1$ edges since it is a spanning tree of G. We now show that the stretch factor of H is at most 2. Consider any pair of vertices s, t and let c_s and c_t be the roots of the trees of F containing s and t, respectively (notice that c_s and c_t might coincide).

▶ **Lemma 10.** *There exists a path between c_s and c_t in T that traverses all vertices of T and has length $2w(T) - d_T(c_s, c_t)$.*

[9] For instance, such a data structure can be implemented by rooting $\mathcal{T}_i$ at an arbitrary vertex and using the *least-common-ancestor* data structure in [7].

Proof. Let $\pi = \langle c_s = v_1, v_2, \dots, v_\ell = c_t \rangle$ be the unique path from c_s to c_t in T (if $c_s = c_t$ then $\pi = \langle c_s \rangle$), and call T_i the unique tree containing v_i in the forest obtained from T by deleting the edges of π. Let π_i be an Eulerian tour of T_i that starts and ends in v_i and observe that π_i traverses each edge in T_i twice. The sought path is obtained by joining all tours $\pi_1, \dots, \pi_\ell$ with the edges of π, i.e., it is the path $\pi_1 \circ (v_1, v_2) \circ \pi_2 \circ (v_2, v_3) \circ \cdots \circ (v_{\ell-1}, v_\ell) \circ \pi_\ell$ (see Figure 3). ◀

Using the above lemma, we consider the path from s to t in H consisting of the composition of (i) the unique path π_s from s to c_s in F, (ii) a path π from c_s to c_t of length at most $2w(T)$ that traverses all vertices in T (whose existence is guaranteed by Lemma 10), and (iii) the unique path π_t from c_t to t.

We now argue that the length of the above path is at most $2\sigma(s,t)$. Let π^* be the optimal group Steiner path from s to t and let r_1^* and r_k^* be the first and the last occurrences of a required vertex in π^*, respectively. Since r_1^* is a vertex of T, the length of $\pi^*[s : r_1^*]$ is at least $w(\pi_s)$. Similarly, the length of $\pi^*[r_k^* : t]$ is at least $w(\pi_t)$. Moreover, observe that the weight of M is a lower bound for the length of $\pi^*[r_1^* : r_k^*]$, and hence $w(T) \le w(M) \le w(\pi^*[r_1^* : r_k^*])$. Therefore:

$$\begin{aligned} \sigma_H(s,t) &\le w(\pi_s) + w(\pi) + w(\pi_t) = d_F(s, c_s) + 2w(T) - d_T(c_s, c_t) + d_F(c_t, t) \\ &\le w(\pi^*[s : r_1^*]) + 2w(T) + w(\pi^*[r_k^* : t]) \\ &\le w(\pi^*[s : r_1^*]) + w(\pi^*[r_1^* : r_k^*]) + w(\pi^*[r_k^* : t]) + w(T) \\ &\le w(\pi^*) + w(T) \le 2w(\pi^*) = 2\sigma_G(s,t). \end{aligned} \tag{3}$$

▶ **Theorem 11.** *In the singleton case, it is possible to compute a group Steiner tree spanner having stretch 2 and $n-1$ edges in polynomial time.*

We can prove that the stretch of the above tree spanner is tight, since its stretch cannot be improved even for the single-source case.

▶ **Theorem 12.** *In the singleton case, there are unweighted graphs G such that any single-source group Steiner spanner of G having stretch strictly smaller than $2 - \frac{2}{k}$ must contain at least n edges.*

Proof. To prove our lower bound consider a graph G consisting of a cycle C on the k required vertices $r_1, \dots, r_k$, plus $n-k$ additional vertices $v_1, \dots, v_{n-k}$ connected to an arbitrary required vertex r via the edges in $F = \{(v_1, r), \dots, (v_{n-k}, r)\}$, and let r_1 be the source vertex (see Figure 4). Clearly all single-source group Steiner spanners of G need to contain all edges in F, since otherwise they would be disconnected. Consider now any subgraph H obtained from G by deleting a generic edge e of the cycle. Observe that the shortest group Steiner path from s to itself in H requires traversing each edge $C - e$ twice, hence $\sigma_H(s,s) = 2(k-1) = 2k-2$. Since the shortest group Steiner path in G from s to itself has length k, the stretch factor of H is at least $\frac{2k-2}{k} = 2 - \frac{2}{k}$. ◀

Our group Steiner spanner can also be turned a corresponding distance oracle with constant query time, as we discuss in the full version of the paper.

4 Group Steiner spanners for general group sizes

We now describe how to obtain an all-pairs group Steiner spanner H with stretch factor $2\alpha + 1$, for each $\alpha \ge 1$. We will provide two different constructions that build different spanners of sizes $kn + |\bigcup_i R_i \times R_i\ \alpha\text{-spanner}|$ and $n + |R \times R\ \alpha\text{-spanner}|$, respectively. The

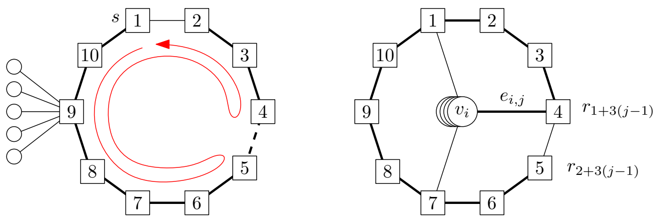

Figure 4 The lower bound constructions of Theorem 12 (left) and Theorem 7 (right) with $k = 10$ required vertices. The generic required vertex r_i is depicted as a squared labelled with i.

first construction is preferable over the second one when the k groups are somewhat disjoint and of uniform sizes, i.e., $|R_i| = O(|R|/k)$ for all i. Corresponding distance oracles having query time $O(2^k k \cdot |R|^2 + |R|^3)$ are discussed in the full version of the paper.

The first construction. The first construction is the following. For each group $R_i \subseteq V$, we build the spanner H as the union of a *subsetwise* spanner $R_i \times R_i$ with stretch α, plus k spanning forests $F_1, \dots, F_k$, where each F_i is obtained by our clustering procedure using the vertices in R_i as centers.

We now discuss the stretch factor of H. Fix any two vertices $s, t \in V$, and, without loss of generality, assume that the shortest group Steiner path π^* from s to t in G traverses the groups $R_1, \dots, R_k$ in this order (otherwise, re-index the groups accordingly), and let r_i be the first required vertex in R_i reached by π^* (see Figure 5).

Let r'_1 (resp. r'_{k+1}) be the root of the tree in F_1 (resp. F_k) containing s (resp. t) and let π_1 (resp. π_{k+1}) the unique path in F_1 (resp. F_k) between s and r'_1 (resp. between t and r'_{k+1}). Moreover, for each $i = 2, \dots, k$, let r'_i be the root of the tree in F_i containing r_{i-1}, and let π_i be the corresponding path in F_i between them. Finally, denote by π'_i the path in H between r'_i and r_i, for $i = 1, \dots, k$, and by π'_{k+1} the path in H between r_k and r'_{k+1}. Consider now the group Steiner path $\widetilde{\pi}$ in H made by the concatenation of $\pi_1 \circ \pi'_1 \circ \cdots \circ \pi_k \circ \pi'_k \circ \pi'_{k+1} \circ \pi_{k+1}$ (see Figure 5). We now show that $w(\widetilde{\pi}) \leq (2\alpha + 1)w(\pi^*)$.

For technical convenience, we let $r_0 = s$ and $r_{k+1} = t$ and we notice that, for $i = 1, \dots, k+1$, we have $w(\pi_i) \leq w(\pi^*[r_{i-1} : r_i])$.

For $i = 1, \dots, k$, $w(\pi'_i) \leq \alpha d_G(r'_i, r_i) \leq \alpha(w(\pi_i) + w(\pi^*[r_{i-1} : r_i])) \leq 2\alpha w(\pi^*[r_{i-1} : r_i])$. Finally, $w(\pi'_{k+1}) \leq \alpha d_G(r_k, r'_{k+1}) \leq \alpha(w(\pi^*[r_k : r_{k+1}]) + w(\pi_{k+1})) \leq 2\alpha w(\pi^*[r_k : r_{k+1}])$.

$$\sigma_H(s,t) \leq w(\widetilde{\pi}) = \sum_{i=1}^{k+1} (w(\pi_i) + w(\pi'_i)) \leq \sum_{i=1}^{k+1} w(\pi^*[r_{i-1} : r_i]) + 2\alpha \sum_{i=1}^{k+1} w(\pi^*[r_{i-1} : r_i])$$
$$= w(\pi^*) + 2\alpha w(\pi^*) = (1 + 2\alpha)\sigma_G(s,t).$$

▶ **Theorem 13.** *Given k subsetwise spanners $H_1, \dots, H_k$, where H_i is an $R_i \times R_i$ α-spanner of G, it is possible to compute in polynomial time a group Steiner spanner of G with stretch $2\alpha + 1$ and size $O\left(nk + \left|\bigcup_{i=1}^k H_i\right|\right)$.*

The second construction. The second construction is the following. We build the spanner H as the union of a subsetwise $R \times R$ α-spanner H' of G, plus a spanning forest F that is obtained by our clustering procedure using the vertices in R as centers. Clearly, the size of H is $O(n + |H'|)$.

We now show that the stretch factor of H is $2\alpha + 1$. Fix any two vertices $s, t \in V$, and, without loss of generality, assume that the shortest group Steiner path π^* from s to t in G traverses the groups $R_1, \dots, R_k$ in this order (otherwise, re-index the groups accordingly).

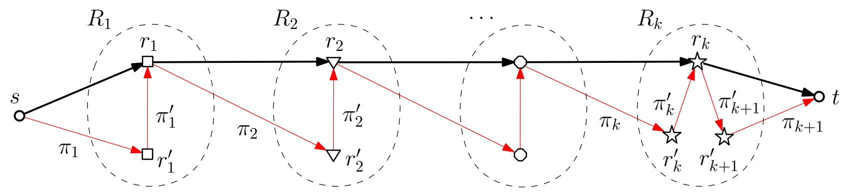

Figure 5 A qualitative depiction of analysis of the stretch $2\alpha + 1$. The shortest group Steiner path π^* from s to t is in bold, while the path $\widetilde{\pi}$ in the spanner is in red.

Let r_i be the first required vertex in R_i reached by π^*. Let r_s be the vertex of R that corresponds to the center of the cluster in F that contains s. Similarly, let r_t be the vertex of R that corresponds to the center of the cluster in F that contains t (it might happen that $r_s = r_t$). We have $d_H(s, r_s) = d_G(s, r_s) \leq d_G(s, r_1)$ and $d_H(r_t, t) = d_G(r_t, t) \leq d_G(r_k, t)$. Moreover, we can upper bound the group Steiner distance from r_s to r_t in G by the following

$$\sigma_G(r_s, r_t) \leq d_G(r_s, s) + \sigma_G(s, t) + d_G(t, r_t) \leq d_G(s, r_1) + \sigma_G(s, t) + d_G(r_k, t) \leq 2\sigma_G(s, t).$$

As a consequence, since H contains an $R \times R$ α-spanner of G, using Lemma 1, we obtain that $\sigma_H(r_s, r_t) \leq \alpha\sigma_G(r_s, r_t) \leq 2\alpha\sigma_G(s, t)$. Therefore $\sigma_H(s, t)$ is at most:

$$\leq d_H(s, r_s) + \sigma_H(r_s, r_t) + d_H(r_t, t) \leq d_G(s, r_1) + 2\alpha\sigma_G(s, t) + d_G(r_k, t) \leq (2\alpha + 1)\sigma_G(s, t).$$

▶ **Theorem 14.** *Given a subsetwise $R \times R$ α-spanner H' of G, it is possible to compute in polynomial time a group Steiner spanner of G with stretch $2\alpha + 1$ and size $O(n + |H'|)$.*

5 Conclusions

We conclude this work by mentioning some problems that we deem significant. Our construction of the $(1 + \varepsilon)$-spanner with size $O(n/\varepsilon^2)$ for the singleton case requires a building time of $2^k k \cdot n^{O(1)}$, can a spanner with the same stretch and size $O(f(\varepsilon) \cdot n \operatorname{polylog}(n))$ be built in polynomial time? Regarding the single-source case, we conjecture that the dependency on k in the $O(2^k \cdot n)$ size of our preserver is too weak. Can stronger upper bounds be proved for either the same or novel constructions? Can a lower bound that is polynomial in k and linear in n be shown?

References

1. Abu Reyan Ahmed, Greg Bodwin, Keaton Hamm, Stephen G. Kobourov, and Richard Spence. On additive spanners in weighted graphs with local error. In Lukasz Kowalik, Michal Pilipczuk, and Pawel Rzazewski, editors, *Graph-Theoretic Concepts in Computer Science - 47th International Workshop, WG 2021, Warsaw, Poland, June 23-25, 2021, Revised Selected Papers*, volume 12911 of *Lecture Notes in Computer Science*, pages 361–373. Springer, 2021. doi:10.1007/978-3-030-86838-3_28.
2. Reyan Ahmed, Greg Bodwin, Faryad Darabi Sahneh, Keaton Hamm, Mohammad Javad Latifi Jebelli, Stephen Kobourov, and Richard Spence. Graph spanners: A tutorial review. *Computer Science Review*, 37:100253, 2020. doi:10.1016/j.cosrev.2020.100253.
3. Ingo Althöfer, Gautam Das, David P. Dobkin, Deborah Joseph, and José Soares. On sparse spanners of weighted graphs. *Discret. Comput. Geom.*, 9:81–100, 1993. doi:10.1007/BF02189308.
4. Saeed Akhoondian Amiri, Klaus-Tycho Foerster, and Stefan Schmid. Walking through waypoints. *Algorithmica*, 82(7):1784–1812, 2020. doi:10.1007/S00453-020-00672-Z.

5 Joyce Bacic, Saeed Mehrabi, and Michiel Smid. Shortest beer path queries in outerplanar graphs. In Hee-Kap Ahn and Kunihiko Sadakane, editors, *32nd International Symposium on Algorithms and Computation, ISAAC 2021, December 6-8, 2021, Fukuoka, Japan*, volume 212 of *LIPIcs*, pages 62:1–62:16. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021. doi:10.4230/LIPICS.ISAAC.2021.62.

6 Joyce Bacic, Saeed Mehrabi, and Michiel Smid. Shortest beer path queries in outerplanar graphs. *Algorithmica*, 85(6):1679–1705, 2023. doi:10.1007/S00453-022-01045-4.

7 Michael A. Bender and Martin Farach-Colton. The level ancestor problem simplified. In Sergio Rajsbaum, editor, *LATIN 2002: Theoretical Informatics, 5th Latin American Symposium, Cancun, Mexico, April 3-6, 2002, Proceedings*, volume 2286 of *Lecture Notes in Computer Science*, pages 508–515. Springer, 2002. doi:10.1007/3-540-45995-2_44.

8 Greg Bodwin and Virginia Vassilevska Williams. Better distance preservers and additive spanners. In Robert Krauthgamer, editor, *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2016, Arlington, VA, USA, January 10-12, 2016*, pages 855–872. SIAM, 2016. doi:10.1137/1.9781611974331.CH61.

9 Keren Censor-Hillel, Telikepalli Kavitha, Ami Paz, and Amir Yehudayoff. Distributed construction of purely additive spanners. In Cyril Gavoille and David Ilcinkas, editors, *Distributed Computing - 30th International Symposium, DISC 2016, Paris, France, September 27-29, 2016. Proceedings*, volume 9888 of *Lecture Notes in Computer Science*, pages 129–142. Springer, 2016. doi:10.1007/978-3-662-53426-7_10.

10 Gary Chartrand, Ortrud R. Oellermann, Song Lin Tian, and Hung Bin Zou. Steiner distance in graphs. *Časopis pro pěstování matematiky*, 114(4):399–410, 1989.

11 Shiri Chechik. Approximate distance oracles with constant query time. In David B. Shmoys, editor, *Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 - June 03, 2014*, pages 654–663. ACM, 2014. doi:10.1145/2591796.2591801.

12 Shiri Chechik. Approximate distance oracles with improved bounds. In Rocco A. Servedio and Ronitt Rubinfeld, editors, *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing, STOC 2015, Portland, OR, USA, June 14-17, 2015*, pages 1–10. ACM, 2015. doi:10.1145/2746539.2746562.

13 Don Coppersmith and Michael Elkin. Sparse sourcewise and pairwise distance preservers. *SIAM J. Discret. Math.*, 20(2):463–501, 2006. doi:10.1137/050630696.

14 Marek Cygan, Fabrizio Grandoni, and Telikepalli Kavitha. On pairwise spanners. In Natacha Portier and Thomas Wilke, editors, *30th International Symposium on Theoretical Aspects of Computer Science, STACS 2013, February 27 - March 2, 2013, Kiel, Germany*, volume 20 of *LIPIcs*, pages 209–220. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2013. doi:10.4230/LIPICS.STACS.2013.209.

15 Rathish Das, Meng He, Eitan Kondratovsky, J. Ian Munro, Anurag Murty Naredla, and Kaiyu Wu. Shortest beer path queries in interval graphs. In Sang Won Bae and Heejin Park, editors, *33rd International Symposium on Algorithms and Computation, ISAAC 2022, December 19-21, 2022, Seoul, Korea*, volume 248 of *LIPIcs*, pages 59:1–59:17. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022. doi:10.4230/LIPICS.ISAAC.2022.59.

16 Khaled M. Elbassioni, Aleksei V. Fishkin, Nabil H. Mustafa, and René Sitters. Approximation algorithms for euclidean group TSP. In Luís Caires, Giuseppe F. Italiano, Luís Monteiro, Catuscia Palamidessi, and Moti Yung, editors, *Automata, Languages and Programming, 32nd International Colloquium, ICALP 2005, Lisbon, Portugal, July 11-15, 2005, Proceedings*, volume 3580 of *Lecture Notes in Computer Science*, pages 1115–1126. Springer, 2005. doi:10.1007/11523468_90.

17 Michael Elkin, Arnold Filtser, and Ofer Neiman. Terminal embeddings. *Theor. Comput. Sci.*, 697:1–36, 2017. doi:10.1016/J.TCS.2017.06.021.

18 Michael Elkin, Yuval Gitlitz, and Ofer Neiman. Improved weighted additive spanners. *Distributed Comput.*, 36(3):385–394, 2023. doi:10.1007/S00446-022-00433-X.

19 Paul Erdös. On some extremal problems in graph theory. *Israel Journal of Mathematics*, 3:113–116, 1965.

20 Joachim Gudmundsson and Yuan Sha. Shortest beer path queries in digraphs with bounded treewidth. In Satoru Iwata and Naonori Kakimura, editors, *34th International Symposium on Algorithms and Computation, ISAAC 2023, December 3-6, 2023, Kyoto, Japan*, volume 283 of *LIPIcs*, pages 35:1–35:17. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2023. doi:10.4230/LIPICS.ISAAC.2023.35.

21 Tesshu Hanaka, Hirotaka Ono, Kunihiko Sadakane, and Kosuke Sugiyama. Shortest beer path queries based on graph decomposition. In Satoru Iwata and Naonori Kakimura, editors, *34th International Symposium on Algorithms and Computation, ISAAC 2023, December 3-6, 2023, Kyoto, Japan*, volume 283 of *LIPIcs*, pages 37:1–37:20. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2023. doi:10.4230/LIPICS.ISAAC.2023.37.

22 Marek Karpinski, Michael Lampis, and Richard Schmied. New inapproximability bounds for TSP. *J. Comput. Syst. Sci.*, 81(8):1665–1677, 2015. doi:10.1016/J.JCSS.2015.06.003.

23 Telikepalli Kavitha. New pairwise spanners. *Theory Comput. Syst.*, 61(4):1011–1036, 2017. doi:10.1007/S00224-016-9736-7.

24 Hung Le and Shay Solomon. Near-optimal spanners for general graphs in (nearly) linear time. In Joseph (Seffi) Naor and Niv Buchbinder, editors, *Proceedings of the 2022 ACM-SIAM Symposium on Discrete Algorithms, SODA 2022, Virtual Conference / Alexandria, VA, USA, January 9 - 12, 2022*, pages 3332–3361. SIAM, 2022. doi:10.1137/1.9781611977073.132.

25 Yaping Mao. Steiner distance in graphs–a survey, 2017. arXiv:1708.05779.

26 Petrică C. Pop, Ovidiu Cosma, Cosmin Sabo, and Corina Pop Sitar. A comprehensive survey on the generalized traveling salesman problem. *European Journal of Operational Research*, 314(3):819–835, 2024. doi:10.1016/j.ejor.2023.07.022.

27 András Sebö and Jens Vygen. Shorter tours by nicer ears: 7/5-approximation for the graph-tsp, 3/2 for the path version, and 4/3 for two-edge-connected subgraphs. *Comb.*, 34(5):597–629, 2014. doi:10.1007/S00493-014-2960-3.

28 Mikkel Thorup and Uri Zwick. Approximate distance oracles. *J. ACM*, 52(1):1–24, January 2005. doi:10.1145/1044731.1044732.

29 Jacques Tits. Sur la trialité et certains groupes qui s'en déduisent. *Publications Mathématiques de l'Institut des Hautes Études Scientifiques*, 2:14–60, 1959.

30 Vera Traub, Jens Vygen, and Rico Zenklusen. Reducing path TSP to TSP. In Konstantin Makarychev, Yury Makarychev, Madhur Tulsiani, Gautam Kamath, and Julia Chuzhoy, editors, *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing, STOC 2020, Chicago, IL, USA, June 22-26, 2020*, pages 14–27. ACM, 2020. doi:10.1145/3357713.3384256.

31 Vijay V. Vazirani. *Approximation algorithms*. Springer, 2001. URL: http://www.springer.com/computer/theoretical+computer+science/book/978-3-540-65367-7.

32 R. Wenger. Extremal graphs with no c4,s, or c10,s. *J. Comb. Theory Ser. B*, 52(1):113–116, May 1991. doi:10.1016/0095-8956(91)90097-4.

33 Rico Zenklusen. A 1.5-approximation for path TSP. In Timothy M. Chan, editor, *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2019, San Diego, California, USA, January 6-9, 2019*, pages 1539–1549. SIAM, 2019. doi:10.1137/1.9781611975482.93.