

Greedy Completion for Weighted (α, β) -Spanners

Elad Tzalik 

Weizmann Institute of Science, Rehovot, Israel

Abstract

We study (α, β) -spanners for weighted graphs. We propose a simple greedy completion procedure which starts from a sparse initial graph, and repeatedly fixes pairs of vertices with a bad stretch, generalizing Knudsen’s additive completion [SWAT 2014]. As an application, we construct $(k, k - 1)$ -spanners for weighted graphs of size $\tilde{O}(n^{1+1/k})$, which were previously unknown.

2012 ACM Subject Classification Theory of computation → Sparsification and spanners

Keywords and phrases Graph spanners, Greedy algorithms, Weighted graphs

Digital Object Identifier 10.4230/LIPIcs.ESA.2026.46

Related Version *Full Version*: <https://arxiv.org/abs/2603.17047>

Funding *Elad Tzalik*: Research supported by the Adams Fellowship Program of the Israel Academy of Sciences and Humanities.

Acknowledgements The author is grateful to Itai Benjamini for his encouragement over the course of this (and many other) projects. The author thanks Merav Parter for useful discussions on weighted extensions of (α, β) -spanners, Elizaveta Popova for many helpful discussions, Gur Lifshiz for useful discussions, and Tomer Even for his suggestions to improve the paper.

1 Introduction

Let G be an n -vertex weighted graph. A *spanner* of G is a subgraph that approximately preserves distances. Formally, a subgraph H is a t -spanner of G if $\text{dist}_H(u, v) \leq t \cdot \text{dist}_G(u, v)$ for all $u, v \in V$ (where dist_X denotes the shortest path distance in a graph X). Since their introduction by Peleg and Ullman [26] and Peleg and Schäffer [24], spanners have been found to be extremely useful for a wide variety of applications, including network tasks like routing and synchronization [25, 29, 13, 14, 26], preconditioning linear systems [17], distance estimation [30, 7], and many others.

The first tight construction, and one of the most influential spanner constructions, was given by [5], which described a simple greedy algorithm that produces a $(2k - 1)$ -spanner of a graph G , of size $O(n^{1+\frac{1}{k}})$ edges, which is best possible¹.

To bypass this size-stretch tradeoff, much work has gone into finding better ways to parameterize how shortest paths in G may stretch in H . The most well-studied such generalization is (α, β) -spanners which allow a multiplicative stretch α (which may be viewed as the effective stretch for pairs that are sufficiently far apart in the graph), plus an additive stretch β which is used to mask the local barrier of the stretch of the endpoints of an edge in a high girth graph. Let $W_{\max}(G)$ denote the maximum weight of an edge in G .

► **Definition 1** ((α, β) -spanner). *A subgraph $H \subseteq G$ is an (α, β) -spanner of G if for every $u, v \in V$,*

$$\text{dist}_H(u, v) \leq \alpha \cdot \text{dist}_G(u, v) + \beta \cdot W_{\max}(G).$$

¹ The $O(n^{1+\frac{1}{k}})$ upper bound is tight *assuming the girth conjecture of Erdős*. Nevertheless, the algorithm of [5] is optimal *unconditionally*.



© Elad Tzalik;

licensed under Creative Commons License CC-BY 4.0

34th Annual European Symposium on Algorithms (ESA 2026).

Editors: Philip Bille, Seth Pettie, and Sabine Storandt; Article No. 46; pp. 46:1–46:15

Leibniz International Proceedings in Informatics



Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

The quest to handle weights

Spanners are often used to compress metric spaces that correspond to weighted input graphs, see e.g. [10, 15, 22, 28]. While standard multiplicative spanners can handle weights easily, algorithms that construct (α, β) -spanners for unweighted graphs typically take non-trivial effort to be adapted to handle weights. A notable example is the +6 spanner [6, 20], where extending the construction to weighted graphs required three separate works [4, 19, 21]. Initially, the work of Cohen [12] constructed weighted $(1 + \varepsilon, \beta)$ spanners, which was then improved by [16]. In recent years, there has been a renewed and growing interest in understanding spanners for weighted graphs. The work of [18] achieved constructions of weighted (α, β) spanners with *local* stretch, meaning that $W_{\max}(G)$ is replaced by $W_{\max}(P)$, which stands for the smallest maximum edge weight along a shortest path P between nodes. A sequence of recent works [4, 19, 2, 21] gave tight bounds for weighted *additive* spanners. A key ingredient in these works is the work of Knudsen [20], which describes a simple algorithm to obtain additive spanners greedily.

Handling weights: beyond additive stretch

The main downside of additive spanners is that they may be large. The seminal work of Abboud and Bodwin [1] proved that such spanners may require $\Omega(n^{\frac{4}{3}-\varepsilon})$ edges for any $\varepsilon > 0$. This motivates the study of weighted (α, β) -spanners for $\alpha > 1$. Unfortunately, this case is far less understood with only *one prior work*² providing such constructions, by Elkin, Gitlitz and Neiman [18], constructing a weighted $(O(1), k^{O(1)})$ -spanner with $O(n^{1+\frac{1}{k}})$ edges, as well as weighted $(1 + \varepsilon, \beta)$ spanners with $\beta = O\left(\frac{\log(k)}{\varepsilon}\right)^{\log(k)}$.

The “local” recipe for (α, β) spanners, and how it fails to handle weights

We now explain why existing constructions of (α, β) -spanners fail to handle weighted graphs, e.g. those of [6, 9, 11, 27]. When the graph G is unweighted, obtaining an (α, β) spanner can be done “locally” by guaranteeing good stretch for paths of fixed length. For example, to obtain a $(k, k - 1)$ -spanner for an *unweighted* graph G , one can take a standard spanner H_1 that guarantees a stretch of $2k - 1$ for a single edge, and a subgraph H_2 which guarantees a stretch of k for vertices of distance 2. $H_1 \cup H_2$ is the desired spanner - indeed given x, y with a shortest path $P := P_{x,y}$ with t edges in G , break P into $\lfloor t/2 \rfloor$ paths of length 2 with an extra edge remaining if t is odd, and use H_2 to replace the length 2 paths, and H_1 to replace the remaining single edge. As it turns out, such a “recipe” cannot yield a $(k, k - 1)$ -spanner for weighted graphs. In a recent work, Popova and Tzalik [27] proved that in any weighted graph G the best approximation one can provide for a path P of length 2 is $W_{\min}(P) + (2k - 1)W_{\max}(P)$ ³, assuming the desired size bound on the spanner is $O(n^{1+1/k})$, hence in particular, such an approach may incur an additive $+(2k - 2)W_{\max}(G)$ error on every second edge along the path, whereas a $(k, k - 1)$ -spanner allows a $(k - 1)W_{\max}(G)$ additive error only once along the path.

Putting it all together, recent works have investigated how to construct weighted (α, β) -spanners, and while the important case of additive spanners ($\alpha = 1$) has been resolved in [4, 19, 2, 21], the *general* case of $\alpha > 1$ has resisted progress, with only a single known

² Putting aside the case of $\beta = 0$ for which the classical multiplicative spanners of [5] handle weights.

³ $W_{\max}(P)$ denotes the maximum edge weight along P , $W_{\min}$ the minimum one.

construction by [18]. Due to the lack of a desired “canonical” construction of (α, β) -spanners, in a survey on graph spanners from 2020, the following question (open question 4.7.2 [3]) was asked:

► **Question 2.** *Is there a greedy algorithm to produce additive spanners, (α, β) -spanners, subsetwise spanners, or more generally pairwise spanners?*

In this paper we answer this question in the affirmative for (α, β) -spanners by introducing a greedy completion algorithm that extends Knudsen’s approach to (α, β) -spanners.

Using the new completion method, we obtain as an application a $(k, k - 1)$ -spanner for any graph G , with essentially tight size $\tilde{O}(n^{1+1/k})^4$. To our knowledge, such spanners were not previously known.

► **Theorem 3.** *For $k \geq 2$ and any weighted graph G , there exists a $(k, k - 1)$ spanner H of G of size $\tilde{O}(n^{1+1/k})$. Moreover, H can be computed in polynomial time.*

The $(k, k - 1)$ -spanners we construct can be viewed as a spanner whose multiplicative stretch is roughly half the multiplicative stretch of the classical greedy spanner, while still allowing for a single bottleneck, reflected in the additive $(k - 1)$ term. To our knowledge, prior to this work there was no known construction of an (α, β) spanner that improves upon the stretch of the greedy algorithm for weighted graphs, and with similar size.⁵

Conceptual contribution

Beyond the $(k, k - 1)$ construction, the main contribution of this work is a new *global greedy completion paradigm* for constructing weighted (α, β) -spanners. The algorithm extends Knudsen’s additive completion by introducing a notion of *minimal segmentations* of shortest paths, and repairing only the bottlenecks that violate the target stretch. This avoids the over-correction inherent in naive greedy approaches, and crucially enables handling weighted graphs, where local path-based constructions fail. Interestingly, the analysis gives insight towards the actual structure of the replacement shortest paths, beyond just the (α, β) stretch guarantee, and leads to interesting new direction, described in Section 5.

Comparison to [27]

The two works study different aspects of greedy spanner constructions. The focus of [27] is on greedy constructions of $d \rightarrow r$ spanners in unweighted graphs, namely subgraphs in which every pair of vertices at distance exactly d in G has distance at most r in the spanner. Such constructions naturally serve as *local* ingredients in the design of unweighted (α, β) -spanners, and their analysis relies on clustering, ball-growth, and path-buying arguments around short paths. In contrast, the present work studies weighted (α, β) -spanners directly. As explained previously, the more challenging weighted setting, is inherently *global* and requires new methods. Consequently, both the algorithm and the analysis in the present paper are completely different.

⁴ The size bounds are tight assuming the girth conjecture, as a graph with girth $> 2k$ has only itself as a $(k, k - 1)$ spanner, even in the unweighted setup.

⁵ The work of [27] does provide spanners which improve upon the multiplicative $(2k - 1)$ -spanner of [5] for non-adjacent pairs of vertices. The key distinction is that in this work, the stretch is roughly $\frac{2k-1}{k}$, aka *twice* better than a multiplicative spanner, yet their methods cannot give such effective bounds.

2 Technical Overview

Throughout the technical overview we assume for simplicity the graph G is unweighted. The fact that the proposed algorithm is global, in the sense that it is independent of the actual number of edges along a path, makes it possible to handle weighted graphs.

2.1 Knudsen's algorithm

Knudsen's algorithm takes as input a graph G , and an integer β , and produces a $+\beta$ spanner of G . It consists of 2 main phases:

- (Initialize) Compute an initial graph H_0 , obtained by going over all $v \in V$ and adding n^γ arbitrary neighboring edges to v . Set $H \leftarrow H_0$.
- (Additive Completion) While there exist x, y such that $\text{dist}_H(x, y) > \text{dist}_G(x, y) + \beta$: Find an x -to- y shortest path $P_{x,y}$. Add to H all edges of $P_{x,y}$ missing from H .

We now give a high-level overview of the analysis. The analysis hinges on showing that in each completion step, if in the i^{th} step t_i new edges along $P_{x,y}$ are added, then $\Omega(t_i n^{1-\gamma})$ pairs⁶ of vertices have their distance substantially improved by the completion step, meaning that if H is the graph before the completion step, and H' is the graph after the completion step, then for $\Omega(t_i n^{1-\gamma})$ pairs (u, v) we have $\text{dist}_H(u, v) - \text{dist}_{H'}(u, v) \geq 1$. Moreover, in the first H for which (u, v) improves, we have $\text{dist}_{H'}(u, v) \leq \text{dist}_G(u, v) + O(1)$. Combining the fact that a pair (u, v) is within $O(1)$ of its distance in G , after the first improvement, there can be $O(1)$ additional steps for which $\text{dist}_H(u, v) - \text{dist}_{H'}(u, v) \geq 1$. Therefore one gets that $\sum_i \Omega(t_i n^{1-\gamma}) = O(n^2)$, hence $\sum_i t_i = O(n^{1+\gamma})$. In conclusion the final size bound of the output spanner H is:

$$|E(H)| = O(n^{1+\gamma}) + \sum_i t_i = O(n^{1+\gamma}).$$

2.2 Defining the (α, β) -completion

As usual with greedy algorithms, correctness is straightforward; the main challenge is to show that the final output is sparse. We now describe the main conceptual contribution of the paper, which lies in finding the right way to define the (α, β) -completion in a way that preserves correctness and, crucially, allows one to prove a good bound on the size of the output graph. To do so, we incorporate two delicate adaptations to Knudsen's algorithm. We describe the algorithm progressively, through a sequence of increasingly refined attempts. Assume we are given a graph G , parameters (α, β) , and some initial graph H_0 to be completed to an (α, β) -spanner of G . A natural first attempt is to apply Knudsen's original idea directly.

Attempt 1 (unsuccessful): naive (α, β) -completion

While there exist x, y such that:

$$\text{dist}_H(x, y) > \alpha \cdot \text{dist}_G(x, y) + \beta.$$

Find an x -to- y shortest path $P_{x,y}$. Add to H all edges of $P_{x,y}$ missing from H .

⁶ This doesn't hold for all $\gamma > 0$, but for $\beta = 2, 6$ one may set $\gamma = 1/2, 1/3$ respectively, and get this way a spanner of optimal size.

The naive (α, β) -completion successfully fixes pairs x, y with bad stretch – but at what cost? The key problem with the naive completion is its *over-correction*. If initially x, y satisfied $\text{dist}_H(x, y) = \alpha \cdot \text{dist}_G(x, y) + \beta + 1$, it seems plausible that a small fix could have done the job. On the other hand, by adding all missing edges along $P_{x,y}$ we make the distance between x, y in the new graph equal to their actual distance in G - this indeed fixes the bad stretch of x, y , but it seems to take it too far.

Attempt 2 (unsuccessful): (α, β) -completion by fixing the bottlenecks

The key observation is that we can find a *mild* fix to the distance of x, y by only fixing the bottlenecks along a shortest path. Let $P_{x,y}$ be an x -to- y shortest path, and let e be an edge on $P_{x,y}$. Say $e = \{u, v\}$ has α -distant endpoints in H if $\text{dist}_H(u, v) > \alpha \cdot \text{dist}_G(u, v)$. The key observation is that if *all* α -distant edges are added along the path, then in the new graph H' $\text{dist}_{H'}(x, y) \leq \alpha \cdot \text{dist}_G(x, y)$, hence this still fixes the pair x, y , but now only the “bottlenecks” along the path are added to H , which helps ensure that only a few edges are added.

Formally the new (α, β) -completion is:
While there exist x, y such that:

$$\text{dist}_H(x, y) > \alpha \cdot \text{dist}_G(x, y) + \beta.$$

Find an x -to- y shortest path $P_{x,y}$. Add to H all the α -distant edges of $P_{x,y}$ missing from H .

This idea is close to the final algorithm, but one additional ingredient is needed to control the total number of added edges.

Attempt 3 (successful): incorporating shortcuts

The last optimization is to allow bypassing bottlenecks along $P_{x,y}$ if possible. The key observation is that it may be the case that for example $e_1 = (u, v), e_2 = (v, w), e_3 = (w, x)$ are three consecutive edges along $P_{x,y}$ which are all α -distant, yet $\text{dist}_H(u, x) \leq \alpha \cdot \text{dist}_G(u, x)$. Hence if we can find in H a path of stretch α between endpoints of α -distant edges which we can use to “jump over” the bottlenecks - we do so.

In other words, we partition $P_{x,y}$ into segments that are either a single α -distant edge, or pairs of vertices with stretch $\leq \alpha$ in H , and choose one whose segments cannot be merged further; we call such segmentations *minimal*. Finally the (α, β) -completion is:
While there exist x, y such that:

$$\text{dist}_H(x, y) > \alpha \cdot \text{dist}_G(x, y) + \beta.$$

Find an x -to- y shortest path $P_{x,y}$. Compute a minimal segmentation of $P_{x,y}$ w.r.t H . Add to H all the α -distant edges in the chosen segmentation.

2.3 The Baswana-Sen algorithm, and its properties

The Baswana-Sen algorithm [8] is a useful tool for computing *multiplicative* spanners, by clustering vertices to centers. The set of centers is defined inductively by $S_0 = V$, and $S_i = \text{Sample}(S_{i-1}, p)$ where $p = \tilde{O}\left(\frac{1}{n^{1/k}}\right)$. For each vertex $u \in V$ and level i , we denote by $s_u^i \in S_i$ the center of u at level i . We do not review the algorithm here, and refer the reader to [8]; for a local description see [23]. We now record a key invariant preserved by the Baswana-Sen algorithm.

► **Theorem 4** ([8]). *For any graph G , there exists a subgraph $H' \subseteq G$, with $\tilde{O}(n^{1+1/k})$ edges such that for every $i \in \{1, \dots, k\}$, and every $\{u, v\} = e \in E(G)$ one of the following holds:*

1. $\text{dist}_H(u, v) \leq (2i - 1)w(e)$; or
2. $\exists s_u^i, s_v^i \in S_i$ such that $\text{dist}(u, s_u^i) \leq i \cdot w(e)$ and $\text{dist}(v, s_v^i) \leq i \cdot w(e)$.

We also record the following observation which follows from standard concentration bounds and a union bound over k :

► **Observation 5.** *With high probability for all $i \in \{0, \dots, k - 1\}$, we have $|S_i| = \tilde{O}(n^{1-i/k})$.*

2.4 The $(k, k - 1)$ -spanner algorithm, and its analysis

To construct the $(k, k - 1)$ -spanner we take the $(k, k - 1)$ -completion of the initial graph H_0 defined as follows: Run the Baswana-Sen algorithm [8] that produces a $(2k - 1)$ -spanner H , and set $H_0 \leftarrow H$ as the initial graph for the greedy completion. Using clustering as an initialization step is natural, and common in spanner constructions, see e.g. [20, 4, 19].

The analysis follows the high-level structure of Knudsen's original analysis, but requires several new adaptations. For simplicity we assume k is odd, and set $R := \frac{k-1}{2}$. Let t_i be the number of edges added in the i^{th} completion step. The analysis tracks distances only for pairs of *centers*, denoted by (s, s') , with $s, s' \in S_R$. We now specify how we track the progress of the pair (s, s') :

- (Set-off) The first iteration for which s, s' are centers of vertices along $P_{x,y}$ where x, y are the vertices chosen in the completion step. In Claim 16, we show that in subsequent iterations after set-off $\text{dist}_H(s, s') \leq k \text{dist}_G(s, s') + O(k^2)W_{\max}(G)$.
- (Distance improvement) These are iterations after the set-off iteration, and for which $\text{dist}_H(s, s') - \text{dist}_{H'}(s, s') = \Omega(W_{\max}(G))^7$. To complement this definition, in Lemma 20 and Lemma 24 we show that if t_i edges were added in the i^{th} steps, then $\Omega(t_i)$ *distinct* pairs of centers have their distance improved by $\Omega(W_{\max}(G))$.
- (Finalized pair) A pair (s, s') is finalized in the i^{th} iteration, if $\text{dist}_H(s, s') \leq k \text{dist}_G(s, s') - k^2 W_{\max}(G)$, where H is the state of the spanner in that iteration. The key property we use is Lemma 23, which says that once a pair is finalized, it can't be part of the $\Omega(t_i)$ improved pairs of Lemma 20.

We now explain the interplay between the above analysis and the designed completion. The bound we get on $\text{dist}_H(s, s')$ after the set-off iteration follows since in each greedy step the bottlenecks are added, hence every contiguous union of segments has stretch $\leq k$. Since s, s' are not on $P_{x,y}$ but are still somewhat close, they inherit the small stretch from the attaching points, up to the additive slack mentioned.

The distance improvement of many pairs (s, s') is quite natural, and follows since the pair x, y chosen does not satisfy the stretch guarantee. A tricky part of the proof is to show $\Omega(t_i)$ of the pairs are distinct, as it may be the case that the number of centers that attach to $P_{x,y}$ is significantly smaller than t_i . We show that this cannot happen as otherwise we could have found a shortcut (in other words, the chosen segmentation is not minimal, violating the definition of the completion step). This part is based on adapting a lemma of [4] from the additive case to the (α, β) set-up.

Finally, showing that finalized pairs are not part of the $\Omega(t_i)$ improving pairs again follows from the minimal segmentation property. Lemma 23 shows that whenever a finalized pair is one of the improving pairs considered in that iteration, one can make the segmentation smaller,

⁷ As before, H' is the state of H after the completion step.

contradicting minimality. We also highlight that here lies a technical difference between the additive spanners and (α, β) spanners. When $\alpha = 1$ knowing $\text{dist}_H(u, v) \leq \alpha \text{dist}_G(u, v)$ implies $\text{dist}_H(u, v) = \text{dist}_G(u, v)$ hence the distance in H of (u, v) cannot be improved due to a *tautology*. On the other hand, when $\alpha > 1$, pairs (u, v) satisfying $\text{dist}_H(u, v) \leq \alpha \text{dist}_G(u, v)$ for H corresponding in some iteration can still be improved.

Finally, combining all the above we have that there are $\tilde{O}(|S_R|^2) = \tilde{O}(n^{1+1/k})$ pairs of centers, and each pair improves $O(k^2)$ times. Since one may assume $k = O(\log(n))$ as larger stretch does not improve sparsity, we have that $\tilde{O}(k^2 n^{1+1/k}) = \tilde{O}(n^{1+1/k})$ pairs of centers of S_R improve in total. Concluding, we get that the number of edges added in the completion step is $\sum_i t_i = \tilde{O}(n^{1+1/k})$, as needed.

► **Remark 6.** We highlight that one may replace the clustering of [8] with the recent greedy clustering of [27], and obtain a deterministic algorithm, as well as a polylogarithmic improvement in the spanner size, though we do not take this route as it complicates notations. By optimizing the methods of this paper, it seems that one can obtain a $(k, k-1)$ -spanner of size $O(kn^{1+1/k})$, but it is unclear how to get an optimal $O(n^{1+1/k})$.

3 The Algorithm

Throughout, let $G = (V, E)$ be a weighted graph, and let $H \subseteq G$ be a subgraph. We assume that every edge of G is itself a shortest path between its endpoints; otherwise, such an edge may be safely removed without affecting the result.

3.1 Shortest paths and segmentations

For vertices $x, y \in V$, let $P_{x,y} = (x = x_0, x_1, \dots, x_t = y)$ denote a shortest x - y path in G .

► **Definition 7** (α -segmentation). *An α -segmentation of the path $P_{x,y}$ is a sequence*

$$\mathcal{S} = ([i_0, i_1], [i_1, i_2], \dots, [i_{s-1}, i_s]), \quad 0 = i_0 < i_1 < \dots < i_s = t,$$

such that for every segment satisfying $i_k - i_{k-1} \geq 2$,

$$\text{dist}_H(x_{i_{k-1}}, x_{i_k}) \leq \alpha \cdot \text{dist}_G(x_{i_{k-1}}, x_{i_k}).$$

► **Definition 8** (minimal segmentation). *An α -segmentation $\mathcal{S}$ is minimal if no block of one or more consecutive segments of $\mathcal{S}$ can be merged into a single segment so as to produce another α -segmentation of the same path.*

Given a segmentation $\mathcal{S}$ of $P_{x,y}$, let $E(\mathcal{S})$ denote the set of all edges corresponding to length 1 segments in $\mathcal{S}$.

► **Definition 9** (Distant edges of a segmentation). *Given an α -segmentation $\mathcal{S}$ of $P_{x,y}$, denote by $E_\alpha(\mathcal{S}, H) = \{\{u, v\} = e \in E(\mathcal{S}) \mid \text{dist}_H(u, v) > \alpha w(e)\}$.*

3.2 The (α, β) -completion

We now formally describe the (α, β) -completion algorithm. Initially, $H \leftarrow H_0$:

1. While H is not an (α, β) -spanner of G :
2. Pick any x, y such that $\text{dist}_H(x, y) > \alpha \text{dist}_G(x, y) + \beta W_{\max}(G)$.
3. Find a *minimal segmentation* $\mathcal{S}$ of $P_{x,y}$ (w.r.t. the current H, G).
4. Update $H \leftarrow H \cup E_\alpha(\mathcal{S}, H)$.

► **Lemma 10.** *Fix an iteration in the (α, β) completion for which (x, y) is the chosen pair violating the desired stretch. Let $\mathcal{S} = ([i_0, i_1], \dots, [i_{s-1}, i_s])$ be the chosen minimal segmentation.*

Then in the updated subgraph $H' = H \cup E_\alpha(\mathcal{S}, H)$, for any $0 \leq \ell < r \leq s$ we have $\text{dist}_{H'}(x_{i_\ell}, x_{i_r}) \leq \alpha \cdot \text{dist}_G(x_{i_\ell}, x_{i_r})$.

Proof. We first prove that for any single segment $[i_j, i_{j+1}]$ we have $\text{dist}_{H'}(x_{i_j}, x_{i_{j+1}}) \leq \alpha \cdot \text{dist}_G(x_{i_j}, x_{i_{j+1}})$. This implies that

$$\text{dist}_{H'}(x_{i_\ell}, x_{i_r}) \leq \sum_{j=\ell}^{r-1} \text{dist}_{H'}(x_{i_j}, x_{i_{j+1}}) \leq \sum_{j=\ell}^{r-1} \alpha \cdot \text{dist}_G(x_{i_j}, x_{i_{j+1}}) = \alpha \cdot \text{dist}_G(x_{i_\ell}, x_{i_r}),$$

where we use the equality $\sum_{j=\ell}^{r-1} \text{dist}_G(x_{i_j}, x_{i_{j+1}}) = \text{dist}_G(x_{i_\ell}, x_{i_r})$ which follows since the vertices $(x_{i_\ell}, x_{i_1}, \dots, x_{i_r})$ appear in order along the *shortest* x - y path in G .

To prove that $\text{dist}_{H'}(x_{i_j}, x_{i_{j+1}}) \leq \alpha \cdot \text{dist}_G(x_{i_j}, x_{i_{j+1}})$ for all j , it suffices to consider segments $[i_j, i_{j+1}]$ for which $\text{dist}_H(x_{i_j}, x_{i_{j+1}}) > \alpha \text{dist}_G(x_{i_j}, x_{i_{j+1}})$. By definition of α -segmentation, such segments must correspond to single edges on $P_{x,y}$, which by definition are included in $E_\alpha(\mathcal{S}, H)$. Hence if $\text{dist}_H(x_{i_j}, x_{i_{j+1}}) > \alpha \text{dist}_G(x_{i_j}, x_{i_{j+1}})$ then $\text{dist}_{H'}(x_{i_j}, x_{i_{j+1}}) \leq \text{dist}_G(x_{i_j}, x_{i_{j+1}})$, and the claim follows as $\alpha \geq 1$. ◀

► **Corollary 11.** *The (α, β) -completion algorithm terminates.*

Proof. By the above lemma with $\ell = 0, r = s$, we have $\text{dist}_{H'}(x, y) \leq \alpha \text{dist}_G(x, y)$, meaning that in all subsequent iterations (x, y) satisfy the stretch requirement. ◀

► **Observation 12.** *The output of the (α, β) -completion is an (α, β) -spanner of G .*

Proof. The algorithm terminates only when, for all pairs $x, y \in V$, we have

$$\text{dist}_H(x, y) \leq \alpha \text{dist}_G(x, y) + \beta W_{\max}(G). \quad \blacktriangleleft$$

Computing the (α, β) -completion

The (α, β) -completion can be computed in polynomial time. By Lemma 10 the loop runs $O(|V|^2)$ iterations. For a fixed iteration it remains to show that a minimal segmentation can be found efficiently. This can be done as follows: Start with the trivial segmentation of $P_{x,y}$, and repeatedly merge a block of consecutive segments whenever their union has stretch at most α in H . When no further merge is possible, the resulting segmentation is minimal.

3.3 The $(k, k - 1)$ spanner

Run the Baswana-Sen algorithm on G , and let H_0 be the resulting spanner. Return the $(k, k - 1)$ completion of H_0 .

By Lemma 10, the output graph is indeed a $(k, k - 1)$ -spanner, hence it remains to upper bound $|E(H)|$. The size of the initial H_0 is $\tilde{O}(n^{1+1/k})$, thus in order to prove Theorem 3 it remains to prove:

► **Theorem 13.** *During the $(k, k - 1)$ -completion, at most $\tilde{O}(n^{1+1/k})$ edges are added.*

4 The Analysis

4.1 Setup and notation

We consider a single greedy iteration in which the pair (x, y) violates the stretch condition, and let $\mathcal{S}$ be the minimal segmentation of $P_{x,y}$ used in the completion step.

Let $e_1, e_2, \dots, e_t$ denote the edges in $E_k(\mathcal{S}, H)$, ordered along $P_{x,y}$ from x to y . Each edge $e_i = (u_i, v_i)$ has endpoints whose stretch in H is greater than k , i.e., $\text{dist}_H(u_i, v_i) > k \cdot w(e_i)$.

Without loss of generality, we assume e_1 is incident to x (so $u_1 = x$) and e_t is incident to y (so $v_t = y$), otherwise set $x_{\text{new}} := u_1$, and $y_{\text{new}} := v_t$. By the property of the segmentation the path from x to x_{new} , and y to y_{new} has stretch $\leq k$ so if x, y was chosen by the greedy step, the pair $(x_{\text{new}}, y_{\text{new}})$ would also be chosen by the greedy step, using the corresponding subsegmentation $\mathcal{S}'$ of $\mathcal{S}$ which has $E_k(\mathcal{S}', H) = E_k(\mathcal{S}, H)$, hence we may analyze the addition of $E_k(\mathcal{S}, H)$ w.r.t to the pair $(x_{\text{new}}, y_{\text{new}})$ as the greedy step. Hence we assume without loss of generality, that $x \in e_1$, and $y \in e_t$.

We focus on the case of k odd, which is cleaner to describe. In the full version of the paper we include an appendix of the minor adaptations in the analysis required for k even. Throughout we set $R := \frac{k-1}{2}$, and $S := S_R$ the set of centers. We write $W := W_{\max}(G)$ throughout this section. We assume $k > 1$ as the case $k = 1$ is trivial. We use the following notations for centers:

- s_x, s_y : the R -centers of x, y respectively.
- For $1 < i < t$: for each edge $e_i = (u_i, v_i)$, let s_{u_i}, s_{v_i} denote the R -centers of u_i, v_i , respectively.

All mentioned centers exist by using Theorem 4, with $i = R + 1$, and observing that a distant edge has stretch strictly bigger than k , hence the second alternative of Theorem 4 holds for all edges e_i .

We also remark the following observation, which is relevant to the discussion in Section 5.

► **Observation 14.** *Let x, y be vertices, and $\mathcal{S}$ be a minimal segmentation of $P_{x,y}$. Assume $|E_k(\mathcal{S}, H)| = 1$, then x, y have the desired stretch in H .*

Proof. Let $e = \{u, v\}$ be the edge in $E_k(\mathcal{S}, H)$, and assume without loss of generality, u is closer to x and v is closer to y along $P_{x,y}$. Then:

$$\begin{aligned}
 \text{dist}_H(x, y) &\leq \text{dist}_H(x, u) + \text{dist}_H(u, v) + \text{dist}_H(v, y) \\
 &\leq k \cdot \text{dist}_G(x, u) + (2k - 1)\text{dist}_G(u, v) + k \cdot \text{dist}_G(v, y) \\
 &= k(\text{dist}_G(x, u) + \text{dist}_G(u, v) + \text{dist}_G(v, y)) + (k - 1)\text{dist}_G(u, v) \\
 &= k \cdot \text{dist}_G(x, y) + (k - 1)w(e).
 \end{aligned}$$

◀

4.2 Set-off and post set-off Bounds

► **Definition 15 (Set-off).** *A pair of centers (s, s') is set off at the first greedy iteration in which both s and s' serve as centers of some vertices in $\bigcup_{i=1}^t V(e_i)$.*

▷ **Claim 16 (Post set-off bounds).** When a pair of centers (s, s') is set off in an iteration updating H to H' , we have:

$$\text{dist}_{H'}(s, s') \leq k \cdot \text{dist}_G(s, s') + (k^2 - 1)W.$$

Proof. By definition of set-off, there exist $a, b \in \bigcup_{i=1}^t V(e_i)$ such that s, s' are their centers respectively. By the triangle inequality in H' :

$$\text{dist}_{H'}(s, s') \leq \text{dist}_{H'}(s, a) + \text{dist}_{H'}(a, b) + \text{dist}_{H'}(b, s').$$

46:10 Greedy Completion for Weighted (α, β) -Spanners

By the Baswana-Sen invariant, $\text{dist}_{H'}(s, a) \leq R \cdot W$ and $\text{dist}_{H'}(b, s') \leq R \cdot W$, giving $\text{dist}_{H'}(s, a) + \text{dist}_{H'}(b, s') \leq (k-1)W$. Since a, b are endpoints of edges in $E_k(\mathcal{S}, H)$, they are segment endpoints in $\mathcal{S}$, so by Lemma 10, $\text{dist}_{H'}(a, b) \leq k \cdot \text{dist}_G(a, b)$. Hence:

$$\text{dist}_{H'}(s, s') \leq k \cdot \text{dist}_G(a, b) + (k-1)W.$$

Since $H' \subseteq G$, we have $\text{dist}_G(a, s), \text{dist}_G(b, s') \leq R \cdot W$, so by triangle inequality in G : $\text{dist}_G(a, b) \leq \text{dist}_G(a, s) + \text{dist}_G(s, s') + \text{dist}_G(s', b) \leq \text{dist}_G(s, s') + (k-1)W$. Substituting:

$$\text{dist}_{H'}(s, s') \leq k(\text{dist}_G(s, s') + (k-1)W) + (k-1)W = k \cdot \text{dist}_G(s, s') + (k^2 - 1)W. \quad \triangleleft$$

We also have:

► **Definition 17** (Improved pair). *A pair of centers (s, s') is improved in a greedy iteration that updates H to H' if it is either set-off in that iteration, or:*

$$\text{dist}_H(s, s') - \text{dist}_{H'}(s, s') \geq \frac{1}{2}W.$$

► **Remark 18.** The choice of $\frac{1}{2}$ in the definition above is arbitrary, and one may choose any smaller constant and the result continues to hold, with a slight increase to the spanner size.

▷ **Claim 19.** Consider a greedy iteration with chosen pair (x, y) , a minimal segmentation $\mathcal{S}$, and edges $e_1, \dots, e_t$ added, updating H to H' . For each $1 < i < t$ and $s_i \in \{s_{u_i}, s_{v_i}\}$ we have:

$$\text{dist}_{H'}(s_x, s_i) + \text{dist}_{H'}(s_i, s_y) \leq k \text{dist}_G(x, y) - R(w(e_1) + w(e_t)).$$

Proof. Assume without loss of generality s_i is the center of u_i , the symmetric case is handled similarly. Since e_1 is added to H' during the completion phase, and $P_{x,y}[v_1, u_i]$ is a contiguous union of segments of $\mathcal{S}$, we can apply Lemma 10 to get $\text{dist}_{H'}(v_1, u_i) \leq k \text{dist}_G(v_1, u_i)$, hence by considering the path $s_x \rightarrow x \rightarrow u_i \rightarrow s_i$ and get:

$$\text{dist}_{H'}(s_x, s_i) \leq R w(e_1) + w(e_1) + k \text{dist}_G(v_1, u_i) + R w(e_i)$$

We can apply the same analysis to s_i and s_y via the path $s_i \rightarrow u_i \rightarrow v_i \rightarrow u_t \rightarrow v_t \rightarrow s_y$ (since we assumed s_i is the center of u_i), and obtain:

$$\text{dist}_{H'}(s_i, s_y) \leq R w(e_i) + w(e_i) + k \text{dist}_G(v_i, u_t) + (R+1)w(e_t)$$

In total, we have in H' :

$$\text{dist}_{H'}(s_x, s_i) + \text{dist}_{H'}(s_i, s_y) \leq (R+1)(w(e_1) + w(e_t)) + k \text{dist}_G(v_1, u_t). \quad (1)$$

Since $P_{x,y}$ is a shortest x to y path we have $w(e_1) + \text{dist}_G(v_1, u_t) + w(e_t) = \text{dist}_G(x, y)$. Combining this with Equation (1) and the fact that $R+1 = k-R$ implies the claim. $\triangleleft$

► **Lemma 20** (Distance improvement). *Consider a greedy iteration with chosen pair (x, y) , a minimal segmentation $\mathcal{S}$, and edges $e_1, \dots, e_t$ added, updating H to H' . For each $1 < i < t$ we have for $s_i \in \{s_{u_i}, s_{v_i}\}$:*

1. $\text{dist}_H(s_x, s_i) - \text{dist}_{H'}(s_x, s_i) \geq \frac{1}{2}W$, or
2. $\text{dist}_H(s_{v_i}, s_y) - \text{dist}_{H'}(s_{v_i}, s_y) \geq \frac{1}{2}W$.

Proof. Assume towards contradiction both items do not hold. Then by Claim 19 we have:

$$\text{dist}_H(s_x, s_y) \leq k \text{dist}_G(x, y) - R(w(e_1) + w(e_t)) + W.$$

By the Baswana-Sen invariant we have $\text{dist}_H(x, s_x) \leq R w(e_1)$, and $\text{dist}_H(s_y, y) \leq R w(e_t)$, hence by the triangle inequality:

$$\text{dist}_H(x, y) \leq k \text{dist}_G(x, y) + W.$$

Which contradicts the assumption that (x, y) were chosen during the completion phase. $\blacktriangleleft$

The following observation is proved in exactly the same way as Claim 19 and Lemma 20, hence omitted.

► **Observation 21.** *The pair of centers (s_x, s_y) , satisfies:*

- $\text{dist}_{H'}(s_x, s_y) \leq k \text{dist}_G(x, y) - R(w(e_1) + w(e_t)).$
- $\text{dist}_H(s_x, s_y) - \text{dist}_{H'}(s_x, s_y) \geq \frac{1}{2}W.$

We now define the finalized pairs.

► **Definition 22** (Finalized pair). *A pair of centers (s, s') is finalized in H if*

$$\text{dist}_H(s, s') \leq k \cdot \text{dist}_G(s, s') - k^2 W.$$

► **Lemma 23** (Finalized pairs do not participate in completion). *Consider a greedy completion iteration with chosen pair (x, y) and edges $e_1, \dots, e_t$. Let a, b be vertices in $\bigcup_{i=1}^t V(e_i)$, and let s, s' be their centers respectively. Then if (s, s') is finalized then a, b are the two endpoints of a segment of $\mathcal{S}$ that has stretch $\leq k$.*

Proof. Let a, b, s, s' be as stated. We show that if (s, s') is finalized in H , then $\text{dist}_H(a, b) \leq k \cdot \text{dist}_G(a, b)$. This implies the desired result as $\mathcal{S}$ is a *minimal* segmentation, hence the only possibility is that a, b are the endpoints of a segment in $\mathcal{S}$ whose stretch is at most k .

By triangle inequality for the $a \rightarrow s \rightarrow s' \rightarrow b$ in H , and the Baswana-Sen invariant we have $\text{dist}_H(a, b) \leq (k-1)W + \text{dist}_H(s, s')$. Also, since (s, s') is assumed to be finalized $\text{dist}_H(s, s') \leq k \cdot \text{dist}_G(s, s') - k^2 W$.

Now by the triangle inequality for $s \rightarrow a \rightarrow b \rightarrow s'$ in G we have $\text{dist}_G(s, s') \leq \text{dist}_G(a, b) + (k-1)W$. Hence combining it all together we get:

$$\text{dist}_H(a, b) \leq (k-1)W + k \text{dist}_G(s, s') - k^2 W \leq k \cdot \text{dist}_G(a, b) - W < k \cdot \text{dist}_G(a, b). \quad \blacktriangleleft$$

We proceed by noting the following lemma, a small variation of Lemma 1 in [4].

► **Lemma 24.** *Let $e_1, \dots, e_t$ be the edges added during the greedy completion phase. Then there exist a sub collection of $t^* \geq \min\{\frac{t}{4} - 2, 1\}$ edges $e_1^* = (a_1, b_1), \dots, e_{t^*}^* = (a_{t^*}, b_{t^*})$, such that either the centers of all the vertices $\{a_i\}_{i=1}^{t^*}$ are distinct, or the centers of $\{b_i\}_{i=1}^{t^*}$ are distinct.*

Proof. If $t < 12$ there is nothing to prove (any collection of a single edge will do), so assume $t \geq 10$. For $1 < i < t$, say that the i^{th} edge, e_i is pre-heavy if $w(e_{i-1}) \leq w(e_i)$, i.e. it is heavier than the edge preceding it (among the added edges). Otherwise, it is pre-light (ties are broken arbitrarily). As every internal edge is either pre-heavy or pre-light, $\geq \frac{t-2}{2}$ edges are pre-heavy or pre-light, assume without loss of generality that they are pre-light.

46:12 Greedy Completion for Weighted (α, β) -Spanners

We now claim that if $e'_1 = (a'_1, b'_1), \dots, e'_{t'} = (a'_{t'}, b'_{t'})$ ⁸ are pre-light, then every center s can be the center of at most 2 vertices in $\{a'_i\}_i$.

To prove the claim, assume we have $i < j < m$ such that s is the center of a'_i, a'_j, a'_m . From the initialization step we have: $\text{dist}_H(a'_i, a'_m) \leq \text{dist}_H(a'_i, s) + \text{dist}_H(s, a'_m) = R(w(e'_i) + w(e'_m))$.

Let e' be the edge preceding e'_m on $P_{x,y}$. Since e'_m is pre-light we have $w(e') \geq w(e'_m)$. Also since $i < j < m$, it must be that e' and e'_i are different edges, which both lie on $P_{x,y}[a'_i, a'_m]$. We obtain $\text{dist}_G(a'_i, a'_m) \geq (w(e'_i) + w(e'))$, and we get:

$$\text{dist}_H(a'_i, a'_m) \leq R(w(e'_i) + w(e'_m)) \leq R(w(e'_i) + w(e')) \leq R\text{dist}_G(a'_i, a'_m).$$

Since $R \leq k$ we get a contradiction since all the segments between a'_i, a'_m could be merged to one segment with stretch $\leq k$, contradicting the minimality of the segmentation.

Finally, we got a collection of $\geq \frac{t-2}{2}$ edges such that every center s is the center left endpoint of at most two of them, hence we can find a subcollection of at least $\frac{t-2}{4}$ edges whose left centers are all distinct⁹.

When most internal edges are pre-heavy, we consider them as post-light. In such a case we find a collection of $\Omega(t)$ edges with distinct right centers. ◀

► **Corollary 25.** *In a round with t edges added, $\Omega(t)$ pairs of centers that are not finalized are improving.*

Proof. If $t < 100$ then by Observation 21 we have a single pair of centers improving. Otherwise by Lemma 24 we have $\geq t/4 - 2$ edges with distinct (without loss of generality) left centers. By removing at most four edges from that collection, those whose left centers are s_x or s_y , we have $\Omega(t)$ edges with distinct left centers, that are not s_x or s_y . By Lemma 20, all these centers give an improved pair with either s_x or s_y , and the resulting center pairs are all distinct. The considered pairs of centers in Lemma 20 all attach to either x or y and another vertex on $P_{x,y}$, hence they attach to $P_{x,y}$ to two points that don't form a good stretch segment, hence by Lemma 23 all these pairs of centers are not finalized. ◀

We now complete the proof of Theorem 13:

Proof. Let t_i be the number of edges added in the i^{th} round of the completion process, and let m_i be the number of center pairs whose distance improved in the i^{th} round according to Lemma 20, which are not finalized. By Corollary 25 we have $m_i = \Omega(t_i)$. As each pair of centers can improve $O(k^2)$ times before it becomes finalized and there are $\tilde{O}(n^{1+1/k})$ center pairs we have

$$\Omega\left(\sum t_i\right) = \sum m_i = \tilde{O}(k^2 n^{1+1/k})$$

Since $k = O(\log(n))$ we get that the number of edges added during the completion phase, $\sum t_i$, is $\tilde{O}(n^{1+1/k})$, as required. ◀

⁸ We assume that the edges are ordered according to their order on the path $P_{x,y}$. That is e'_i is closer to x than e'_j for $i < j$, and that a'_i is closer to x than b'_i .

⁹ Take any maximal subcollection E' of edges whose left centers are all different, since for every edge $e \notin E'$ there must be another edge $e^* \in E'$ with the left center of e as the left center of e^* , and this mapping is injective (as each center is the left center of ≤ 2 edges), there are more edges inside the maximal subcollection than outside it - the claim follows.

5 Final Remarks and Open Problems

A main open question left from this work is whether unweighted (α, β) -spanners can be extended to the weighted setup? There are many (α, β) for which (α, β) -spanners exist for unweighted graphs, though almost all constructions do not generalize; e.g., it is open whether the construction of [9], as well as the recent constructions of [11] which have $o(k) = \alpha = \Omega(1)$ can be extended to the weighted setting.

► **Question 26.** *Is it the case that if there is an unweighted (α, β) -spanner for some (α, β) , then there is a weighted (α, β) -spanner with a similar size guarantee.*

Recall that the lightness of a spanner is the ratio of its total weight divided by the weight of a minimum spanning tree. We ask:

► **Question 27.** *For which (α, β) can we provide an (α, β) -spanner with relatively small lightness?*

Finally we propose the following more structural direction:

Where did the $k - 1$ go?

The keen reader may have noticed that in Section 3.3 the fact that $\text{dist}_H(x, y) > k \cdot \text{dist}_G(x, y) + (k - 1)W$ was not used. It actually appears only in Observation 14, hence under the assumption that (x, y) have a minimal segmentation with *more than* one missing edge we can run the completion algorithm under the same size bound and give the guarantee $\text{dist}_H(x, y) \leq k \text{dist}_G(x, y) + W_{\max}(G)$. As explained in Remark 18 the choice of the constant 1 is arbitrary. In fact, one can achieve the following: For any $\varepsilon > 0$, and any weighted graph G there exist a subgraph H of size $\tilde{O}(n^{1+1/k}/\varepsilon)$ such that for any x, y , and any shortest path $P_{x,y}$ between them in G one has:

- $((k, k - 1)$ -local stretch) $\text{dist}_H(x, y) \leq k \cdot \text{dist}_G(x, y) + (k - 1)W_{\max}(P_{x,y})$, where $P_{x,y}$ is an x to y shortest path. Or,
- (stretch k and additive $+\varepsilon W_{\max}(G)$) Namely, $\text{dist}_H(x, y) \leq k \cdot \text{dist}_G(x, y) + \varepsilon W_{\max}(G)$.

The proof follows from the analysis in Section 4. Pick a minimal segmentation $\mathcal{S}$ of $P_{x,y}$, and distinguish paths according to $|E_k(\mathcal{S}, H)|$. If $|E_k(\mathcal{S}, H)| = 0$ the first alternative above clearly holds. If $|E_k(\mathcal{S}, H)| = 1$, then by Observation 14 we have $\text{dist}_H(x, y) \leq k \cdot \text{dist}_G(x, y) + (k - 1)w(e)$ where e is the missing edge, hence the first guarantee holds. Finally if $|E_k(\mathcal{S}, H)| \geq 2$, then we check if the second guarantee holds, and if so we do not do anything, if not, we add $E_k(\mathcal{S}, H)$. By changing the definition of improved pairs, Definition 17 to have distance difference between H, H' bigger than $\frac{\varepsilon}{2}W$. All parts of the proof stay exactly the same, except for the final proof of Theorem 13, which needs to be augmented as every pair of centers can improve $O(k^2/\varepsilon)$ times, yielding an $1/\varepsilon$ larger spanner.

► **Question 28.** *Can one provide a better structural guarantee than being just an (α, β) spanner for weighted graph? E.g. is it possible to ensure that for every x, y and any shortest path $P_{x,y}$, can be segmented into $O(\beta)$ edges from G and paths of stretch α plus a relatively negligible error?*

It is plausible that resolving the question above can provide a path to allow comparison for the guarantees of (α, β) -spanners, for different α 's and β 's, similarly to the unweighted setup.

References

- 1 Amir Abboud and Greg Bodwin. The $4/3$ additive spanner exponent is tight. *J. ACM*, 64(4):28:1–28:20, 2017. doi:10.1145/3088511.
- 2 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.
- 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.
- 4 Abu Reyan Ahmed, Greg Bodwin, Faryad Darabi Sahneh, Stephen G. Kobourov, and Richard Spence. Weighted additive spanners. In Isolde Adler and Haiko Müller, editors, *Graph-Theoretic Concepts in Computer Science – 46th International Workshop, WG 2020, Leeds, UK, June 24-26, 2020, Revised Selected Papers*, volume 12301 of *Lecture Notes in Computer Science*, pages 401–413. Springer, 2020. doi:10.1007/978-3-030-60440-0_32.
- 5 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.
- 6 Surender Baswana, Telikepalli Kavitha, Kurt Mehlhorn, and Seth Pettie. Additive spanners and (α, β) -spanners. *ACM Trans. Algorithms*, 7(1):5:1–5:26, 2010. doi:10.1145/1868237.1868242.
- 7 Surender Baswana and Sandeep Sen. Approximate distance oracles for unweighted graphs in $\tilde{O}(n^2)$ time. In *Proceedings of the Fifteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 271–280, 2004. URL: <http://dl.acm.org/citation.cfm?id=982792.982830>.
- 8 Surender Baswana and Sandeep Sen. A simple and linear time randomized algorithm for computing sparse spanners in weighted graphs. *Random Structures & Algorithms*, 30(4):532–563, 2007. doi:10.1002/rsa.20130.
- 9 Uri Ben-Levy and Merav Parter. New (α, β) spanners and hopsets. In Shuchi Chawla, editor, *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms, SODA 2020, Salt Lake City, UT, USA, January 5-8, 2020*, pages 1695–1714. SIAM, 2020. doi:10.1137/1.9781611975994.104.
- 10 Leizhen Cai and J. Mark Keil. Computing visibility information in an inaccurate simple polygon. *Int. J. Comput. Geom. Appl.*, 7(6):515–538, 1997. doi:10.1142/S0218195997000326.
- 11 Shiri Chechik and Gur Lifshitz. (α, β) -spanners and hybrid spanners with nearly tight bounds. In Kasper Green Larsen and Barna Saha, editors, *Proceedings of the 2026 Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2026, Vancouver, BC, Canada, January 11-14, 2026*, pages 3511–3535. SIAM, 2026. doi:10.1137/1.9781611978971.128.
- 12 Edith Cohen. Polylog-time and near-linear work approximation scheme for undirected shortest paths. *J. ACM*, 47(1):132–166, 2000. doi:10.1145/331605.331610.
- 13 Lenore Cowen. Compact routing with minimum stretch. *J. Algorithms*, 38(1):170–183, 2001. doi:10.1006/jagm.2000.1134.
- 14 Lenore Cowen and Christopher G. Wagner. Compact roundtrip routing in directed networks. *J. Algorithms*, 50(1):79–95, 2004. doi:10.1016/j.jalgor.2003.08.001.
- 15 Andrew Dobson and Kostas E. Bekris. Sparse roadmap spanners for asymptotically near-optimal motion planning. *Int. J. Robotics Res.*, 33(1):18–47, 2014. doi:10.1177/0278364913498292.
- 16 Michael Elkin. Computing almost shortest paths. In Ajay D. Kshemkalyani and Nir Shavit, editors, *Proceedings of the Twentieth Annual ACM Symposium on Principles of Distributed Computing, PODC 2001, Newport, Rhode Island, USA, August 26-29, 2001*, pages 53–62. ACM, 2001. doi:10.1145/383962.383983.

- 17 Michael Elkin, Yuval Emek, Daniel A. Spielman, and Shang-Hua Teng. Lower-stretch spanning trees. *SIAM J. Comput.*, 38(2):608–628, 2008. doi:10.1137/050641661.
- 18 Michael Elkin, Yuval Gitlitz, and Ofer Neiman. Almost shortest paths with near-additive error in weighted graphs. In Artur Czumaj and Qin Xin, editors, *18th Scandinavian Symposium and Workshops on Algorithm Theory, SWAT 2022, June 27-29, 2022, Tórshavn, Faroe Islands*, volume 227 of *LIPIcs*, pages 23:1–23:22. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022. doi:10.4230/LIPIcs.SWAT.2022.23.
- 19 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.
- 20 Mathias Bæk Tejs Knudsen. Additive spanners: A simple construction. In *Algorithm Theory – SWAT 2014*, volume 8503 of *Lecture Notes in Computer Science*, pages 277–281. Springer, 2014. doi:10.1007/978-3-319-08404-6_24.
- 21 An La and Hung Le. New weighted additive spanners. *CoRR*, abs/2408.14638, 2024. doi:10.48550/arXiv.2408.14638.
- 22 James D. Marble and Kostas E. Bekris. Asymptotically near-optimal planning with probabilistic roadmap spanners. *IEEE Trans. Robotics*, 29(2):432–444, 2013. doi:10.1109/TR0.2012.2234312.
- 23 Merav Parter, Asaf Petruschka, Shay Sapir, and Elad Tzalik. Parks and recreation: Color fault-tolerant spanners made local. In *Proceedings of the 2025 ACM-SIAM Symposium on Discrete Algorithms (SODA)*. SIAM, 2025. doi:10.1137/1.9781611978322.139.
- 24 David Peleg and Alejandro A. Schäffer. Graph spanners. *J. Graph Theory*, 13(1):99–116, 1989. doi:10.1002/jgt.3190130114.
- 25 David Peleg and Jeffrey D. Ullman. An optimal synchronizer for the hypercube. *SIAM J. Comput.*, 18(4):740–747, 1989. doi:10.1137/0218050.
- 26 David Peleg and Eli Upfal. A tradeoff between space and efficiency for routing tables. In *Proceedings of the 20th Annual ACM Symposium on Theory of Computing, STOC*, pages 43–52, 1988. doi:10.1145/62212.62217.
- 27 Elizaveta Popova and Elad Tzalik. New Greedy Spanners and Applications. In Shubhangi Saraf, editor, *17th Innovations in Theoretical Computer Science Conference (ITCS 2026)*, volume 362 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 107:1–107:25, Dagstuhl, Germany, 2026. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.ITCS.2026.107.
- 28 Oren Salzman, Doron Shaharabani, Pankaj K. Agarwal, and Dan Halperin. Sparsification of motion-planning roadmaps by edge contraction. *Int. J. Robotics Res.*, 33(14):1711–1725, 2014. doi:10.1177/0278364914556517.
- 29 Mikkel Thorup and Uri Zwick. Compact routing schemes. In *Proceedings of the Thirteenth Annual ACM Symposium on Parallel Algorithms and Architectures, SPAA*, pages 1–10, 2001. doi:10.1145/378580.378581.
- 30 Mikkel Thorup and Uri Zwick. Approximate distance oracles. *J. ACM*, 52(1):1–24, 2005. doi:10.1145/1044731.1044732.