On Pairwise Spanners*

Marek Cygan[1], Fabrizio Grandoni[1], and Telikepalli Kavitha[2]

1   IDSIA, University of Lugano, Switzerland; {marek, fabrizio}@idsia.ch
2   Tata Institute of Fundamental Research, India; kavitha@tcs.tifr.res.in

─── **Abstract** ───

Given an undirected n-node unweighted graph $G = (V, E)$, a spanner with *stretch function* $f(\cdot)$ is a subgraph $H \subseteq G$ such that, if two nodes are at distance d in G, then they are at distance at most $f(d)$ in H. Spanners are very well studied in the literature. The typical goal is to construct the sparsest possible spanner for a given stretch function.

In this paper we study *pairwise spanners*, where we require to approximate the u-v distance only for pairs (u, v) in a given set $\mathcal{P} \subseteq V \times V$. Such $\mathcal{P}$-spanners were studied before [Coppersmith,Elkin'05] only in the special case that $f(\cdot)$ is the identity function, i.e. distances between relevant pairs must be preserved exactly (a.k.a. *pairwise preservers*).

Here we present pairwise spanners which are at the same time sparser than the best known preservers (on the same $\mathcal{P}$) and of the best known spanners (with the same $f(\cdot)$). In more detail, for arbitrary $\mathcal{P}$, we show that there exists a $\mathcal{P}$-spanner of size $O(n(|\mathcal{P}| \log n)^{1/4})$ with $f(d) = d + 4 \log n$. Alternatively, for any $\varepsilon > 0$, there exists a $\mathcal{P}$-spanner of size $O(n|\mathcal{P}|^{1/4} \sqrt{\frac{\log n}{\varepsilon}})$ with $f(d) = (1 + \varepsilon)d + 4$. We also consider the relevant special case that there is a critical set of nodes $S \subseteq V$, and we wish to approximate either the distances within nodes in S or from nodes in S to any other node. We show that there exists an $(S \times S)$-spanner of size $O(n\sqrt{|S|})$ with $f(d) = d + 2$, and an $(S \times V)$-spanner of size $O(n\sqrt{|S| \log n})$ with $f(d) = d + 2 \log n$. All the mentioned pairwise spanners can be constructed in polynomial time.

1   Introduction

Let $G = (V, E)$ be an undirected unweighted graph. A subgraph H of G is a *spanner* with stretch function $f(\cdot)$ if, given any two nodes $s, t \in V$ at distance $\delta_G(s, t)$ in G, the distance $\delta_H(s, t)$ between the same two nodes in H is at most $f(\delta_G(s, t))$. An (α, β) spanner is a spanner with stretch functions $f(d) = \alpha \cdot d + \beta$. ($\alpha$ and β are the *multiplicative stretch* and *additive stretch* of the spanner, respectively). If $\beta = 0$ the spanner is called *multiplicative*, and if $\alpha = 1$ the spanner is called *purely-additive*. Spanners are very well studied in the literature (see Section 1.2). The typical goal is to achieve the sparsest possible spanner for a given stretch function $f(\cdot)$ [4, 5, 11, 12, 13, 15, 17, 19, 20, 22].

In this paper we address the natural problem of finding (even sparser) spanners in the case that we want to approximately preserve distances only among a given subset $\mathcal{P} \subseteq V \times V$ of pairs. More formally a *pairwise spanner* on pairs $\mathcal{P}$, or $\mathcal{P}$-spanner for short, with stretch function $f(\cdot)$ is a subgraph $H \subseteq G$ such that, for any $(s, t) \in \mathcal{P}$, $\delta_H(s, t) \leq f(\delta_G(s, t))$. In

30th Symposium on Theoretical Aspects of Computer Science (STACS'13).
Editors: Natacha Portier and Thomas Wilke; pp. 209–220
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

SYMPOSIUM
ON THEORETICAL
ASPECTS
OF COMPUTER
SCIENCE

particular, a classical (all-pairs) spanner is a $(V \times V)$-spanner. Pairwise spanners capture scenarios where we only (or mostly) care about some distances in the graph.

To the best of our knowledge, pairwise spanners were studied before only in the special case that $f(\cdot)$ is the identity function, i.e. distances between relevant pairs have to be preserved exactly. Coppersmith and Elkin [8] call such spanners *pairwise (distance) preservers*, and show that one can compute pairwise preservers of size (i.e., number of edges) $O(\min\left\{|\mathcal{P}|\sqrt{n},\; n\sqrt{|\mathcal{P}|}\right\})$.

The authors left it as an open problem to study the *approximate* variants of these preservers, i.e. what we call pairwise spanners here. This paper takes the first step in answering this question. We show that (for suitable $\mathcal{P}$) it is possible to achieve $\mathcal{P}$-spanners which are at the same time sparser than the preservers in [8] (on the same set $\mathcal{P}$) and than the sparsest known classical spanners (with the same stretch function).

1.1   Our Results and Techniques

In this paper we present some polynomial-time algorithms to construct (α, β) $\mathcal{P}$-spanners for unweighted graphs. Our spanners are either purely-additive (i.e. $\alpha = 1$) or *near-additive* (i.e. $\alpha = 1 + \varepsilon$ for an arbitrarily small $\varepsilon > 0$). For arbitrary $\mathcal{P}$, we achieve the following main results (see Section 5).

▶ **Theorem 1. (near-additive pairwise)** *For any $\varepsilon > 0$ and any $\mathcal{P} \subseteq V \times V$, there is a polynomial time algorithm to compute a $(1 + \varepsilon, 4)$ $\mathcal{P}$-spanner of size $O(n|\mathcal{P}|^{1/4}\sqrt{\log n/\varepsilon})$.*

▶ **Theorem 2. (purely-additive pairwise)** *For any integer $k \geq 1$ and any $\mathcal{P} \subseteq V \times V$, there is a polynomial time algorithm to compute a $(1, 4k)$ $\mathcal{P}$-spanner of size $O(n^{1+1/(2k+1)}(\sqrt{(4k+5)|\mathcal{P}|})^{k/(2k+1)})$.*

We also consider the relevant special case that all the pairs involve at least one node from a critical set $S \subseteq V$. More precisely, we distinguish two types of such pairwise spanners: in *subsetwise spanners* (see Section 3) we wish to approximate distances *between* nodes in S, i.e. $\mathcal{P} = S \times S$; in *sourcewise spanners* (see Section 4) we wish to approximate distances *from* nodes in S, i.e. $\mathcal{P} = S \times V$. We obtain the following improved results for the mentioned cases.

▶ **Theorem 3. (subsetwise)** *For any $S \subseteq V$, there is a polynomial time algorithm to compute a $(1, 2)$ $(S \times S)$-spanner of size $O(n\sqrt{|S|})$.*

▶ **Theorem 4. (sourcewise)** *For any $S \subseteq V$ and any integer $k \geq 1$, there is a polynomial time algorithm to compute a $(1, 2k)$ $(S \times V)$-spanner of size $O(n^{1+1/(2k+1)}(k|S|)^{k/(2k+1)})$.*

In particular, by choosing $k = \log n$, we obtain a $(1, 2\log n)$ sourcewise spanner of size $O(n\sqrt{|S|\log n})$, and a $(1, 4\log n)$ pairwise spanner of size $O(n(|\mathcal{P}|\log n)^{1/4})$.

All our spanners rely on a path-buying strategy which was first exploited in the $(1, 6)$ spanner by Baswana et al. [4]. The high-level idea is as follows. There is an initial clustering phase, where we compute a suitable clustering of the nodes, and an associated subset of edges which are added to the spanner. Then there is a path-buying phase, where we consider an appropriate sequence of paths, and decide whether to add or not each path in the spanner under construction[1]. In particular, each path has a *cost* which is given by the number of

---

[1] In the spanner from Theorem 1 there is also a final step where we add a multiplicative $(2\log n, 0)$-spanner.

edges of the path not already contained in the spanner, and a *value* which measures *how much* the path helps to satisfy the considered set of constraints on pairwise distances. If the value is sufficiently larger than the cost, we add the considered path to the spanner, otherwise we do not.

In more detail, all our pairwise spanners exploit the same clustering phase. We compute a partition $\mathcal{C} = \{C_1, \dots, C_q\}$ of a subset of the nodes, and call *unclustered* the remaining nodes $V - \cup_i C_i$. The initial value of the spanner is $G_{\mathcal{C}} = (V, E_{\mathcal{C}})$, where $E_{\mathcal{C}}$ contains all the edges of G but possibly a subset of the *inter-cluster* edges (with endpoints in two different clusters). The common clustering phase is described in Section 2.

During the path-buying phase we add to the spanner some extra *inter-cluster* edges. Here we need to finely tune the sequence of paths that we consider, and also the definition of value of a path. In our subsetwise and sourcewise spanners the value of a path ρ reflects the number of pairs (v, C), where v is the endpoint of some pair and C is a cluster, such that adding ρ to the current spanner decreases the distance between v and (the closest node in) C. In the remaining pairwise spanners, we use a similar notion of value, but considering the distance between pairs of clusters (C', C'').

The sequence of paths used in our subsetwise spanner and near-additive pairwise spanner is simply given by the shortest paths among the relevant pairs. This naturally generalizes the set of paths considered in [4]. However, for the sourcewise spanner and the purely-additive pairwise spanner we need to consider a carefully constructed sequence of paths, which includes slightly suboptimal paths. In more detail, we start with the set of shortest paths between the relevant pairs. Then, for each such path ρ, if the cost of ρ is sufficiently smaller than its value, we include ρ in the spanner. Otherwise, we replace ρ with a *slightly longer* path ρ' between the same endpoints which is *much cheaper*, and iterate the process on ρ'. After a small number of iterations, the considered path becomes cheap enough and hence we include it in the spanner. This (non-trivial) iterative construction of candidate paths during the path-buying phase is probably the main algorithmic contribution of this paper.

1.2   Related Work

Graph spanners were introduced by Peleg and Schäffer [17] in 1989. Spanners have been extensively studied since then, and there are numerous applications involving spanners, such as algorithms for approximate shortest paths [1, 7, 12], labeling schemes [16, 14], approximate distance oracles [21, 6, 3], routing [2, 9, 10], and network design [18].

There are several algorithms for computing multiplicative and additive spanners in weighted and unweighted graphs. In unweighted graphs, for any integer $k \geq 1$, Halperin and Zwick [15] gave a linear time algorithm to compute a multiplicative $(2k - 1, 0)$-spanner of size $O(n^{1+1/k})$, where n is the number of vertices. Note that for $k = \log n$ one obtains a spanner with multiplicative stretch $O(\log n)$ and with size $O(n)$: we will use this type of spanner in Theorem 1. Analogous results are also known for weighted graphs [5, 20, 19].

The first purely-additive spanner (for unweighted graphs) is due to Dor et al. [11]. They describe a $(1, 2)$ spanner of size $O(n^{3/2} \log n)$. This was subsequently improved to $O(n^{3/2})$ [13]. Note that our subsetwise spanner from Theorem 3 generalizes this result: in particular, it has the same stretch function and is sparser whenever $|S| = o(n)$. Baswana et al. [4] describe a $(1, 6)$-spanner of size $O(n^{4/3})$. Whenever $|\mathcal{P}| = O(n^{4/3-\delta})$ for some constant $\delta > 0$, we achieve an asymptotically sparser pairwise spanner with constant additive stretch (depending on δ). The same holds for our sourcewise spanner if $|S| = O(n^{2/3-\delta})$.

The result in [15] shows an elegant trade-off between the size of the spanner and its

multiplicative stretch. No such trade-off is known for purely-additive spanners. In particular, the spanner in [4] is the sparsest known purely-additive spanner. Theorems 2 and 4 show a non-trivial trade-off between the size and additive stretch of pairwise spanners.

There have also been several results on near-additive spanners [13, 12, 22]. For example, there is a $(1 + \epsilon, 4)$-spanner of size $O(\frac{n^{4/3}}{\epsilon})$ for any $\epsilon > 0$ [13]. Our pairwise spanner from Theorem 1 has the same stretch function, and is sparser for $|\mathcal{P}| = o(n^{4/3}/(\varepsilon \log n)^2)$.

Compared to the preservers in [8], we achieve sparser pairwise spanners with additive stretch $O(\log n)$ for $|\mathcal{P}| = \omega(n^{2/3} \log^{1/3} n)$, and a sparser subsetwise spanners for $|S| = \omega(n^{1/3})$. Interestingly, our sourcewise spanners are always sparser than the pairwise preservers from [8].

2   Clustering

A *clustering* of a graph $G = (V, E)$ is a collection $\mathcal{C} = \{C_1, \ldots, C_q\}$ of pairwise disjoint subsets of nodes $C_i \subseteq V$. Note that we do not require $\mathcal{C}$ to span all the nodes V: we call *unclustered* the nodes $V - \cup_i C_i$.

We will crucially exploit the following construction of a clustering $\mathcal{C}$ and of an associated *cluster subgraph* $G_{\mathcal{C}}$.

▶ **Lemma 5.** *There is a polynomial time algorithm which, given $\beta \in [0, 1]$ and a graph $G = (V, E)$, computes a clustering $\mathcal{C}$ with at most $n^{1-\beta}$ clusters and a subgraph $G_{\mathcal{C}}$ of size $O(n^{1+\beta})$ such that:*
1. **(missing-edge property)** *If an edge $uv \in E$ is absent in $G_{\mathcal{C}}$, then u and v belong to two different clusters.*
2. **(cluster-diameter property)** *The distance in $G_{\mathcal{C}}$ between any two vertices of the same cluster is at most 2.*

Proof. Let U be the set of nodes which are not yet clustered (initially we set $U := V$). As long as there exists a vertex $v \in V$ with at least $\lceil n^{\beta} \rceil$ neighbors in U, let C contain exactly $\lceil n^{\beta} \rceil$ arbitrary neighbors of v in U. Add C to $\mathcal{C}$, set $U := U \setminus C$ and add to $G_{\mathcal{C}}$ all the edges of G with both endpoints in $C \cup \{v\}$. When no node v satisfies the mentioned property, we stop creating new clusters and add to $G_{\mathcal{C}}$ all the edges incident to the final set of unclustered nodes U.

By construction, clusters are pairwise disjoint. Each time we create a new cluster, the size of U decreases by at least n^{β}, hence there cannot be more than $n^{1-\beta}$ clusters. Any two nodes in the same cluster C have some common neighbor v in $G_{\mathcal{C}}$, hence Property 2 is satisfied. By construction, all the edges incident to unclustered nodes plus the intra-cluster edges (with both endpoints in the same cluster) belong to $G_{\mathcal{C}}$, which implies Property 1.

It remains to bound the number of edges of $G_{\mathcal{C}}$. Each time we create a new cluster, the number of edges of $G_{\mathcal{C}}$ grows by at most $O(n^{2\beta})$: this gives $O(n^{1-\beta}n^{2\beta}) = O(n^{1+\beta})$ edges altogether. When we stop creating clusters, each (clustered or unclustered) node v has at most n^{β} neighbors in U: consequently the number of edges incident to unclustered nodes that we add at the end of the procedure is at most $O(n^{1+\beta})$.   ◀

The following technical lemma turns out to be useful in the remaining sections.

▶ **Lemma 6.** *Let $\mathcal{C}$ and $G_{\mathcal{C}}$ be constructed with the procedure from Lemma 5 w.r.t. a given graph $G = (V, E)$. If the shortest path ρ in G between any two nodes $u, v \in V$ contains t edges that are absent in $G_{\mathcal{C}}$, then there are at least $t/2$ clusters of $\mathcal{C}$ having at least one vertex on ρ.*

Proof. We prove the lemma by counting pairs (u, e), where e is an edge of ρ absent in $G_\mathcal{C}$ and u is one of the endpoints of e: let $\mathcal{S}$ be the set of such pairs. Since ρ contains t edges that are absent in $G_\mathcal{C}$ there are exactly $2t$ pairs in $\mathcal{S}$ (each edge $e = uv$ belongs to two pairs: (u, e) and (v, e)). We say that a cluster $C \in \mathcal{C}$ *owns* a pair (u, e) if $u \in C$. By the missing-edge property, each edge e of ρ absent in $G_\mathcal{C}$ has both endpoints clustered, hence each pair of $\mathcal{S}$ is owned by some cluster.

Let us assume that there are x clusters of $\mathcal{C}$ having at least one vertex on ρ. By the cluster-diameter property any cluster $C \in \mathcal{C}$ contains at most 3 vertices on ρ, since otherwise ρ would not be a shortest path between u and v. However, if a cluster $C \in \mathcal{C}$ contains exactly 3 vertices on ρ, those have to be consecutive vertices a, b, c of ρ, since ρ is a shortest path and we know by the cluster-diameter property that there is a path of length at most 2 between every pair in $\{a, b, c\}$. By the missing-edge property both edges ab and bc are present in $G_\mathcal{C}$, and consequently C owns at most two pairs of $\mathcal{S}$. Clearly if a cluster $C \in \mathcal{C}$ contains at most 2 vertices on ρ, then it owns at most 4 pairs of $\mathcal{S}$. Therefore each cluster owns at most 4 pairs of $\mathcal{S}$: since $\mathcal{S}$ has $2t$ pairs we have $x \geq t/2$. ◄

3 Subsetwise Spanners

In this section we present our algorithm to compute a subsetwise spanner, and prove Theorem 3.

Our algorithm consists of two main phases: a clustering phase and a path-buying phase. In the clustering phase we invoke Lemma 5 and obtain a cluster subgraph $G_\mathcal{C}$ of G of size $O(n^{1+\beta})$, together with a set $\mathcal{C}$ of at most $n^{1-\beta}$ clusters. The value of β will be defined later.

In the path-buying phase we proceed as follows. Initially set $G_0 := G_\mathcal{C}$ and let $\{\rho_1, \ldots, \rho_z\}$ denote the set of $z = \binom{|S|}{2}$ shortest paths between all pairs of vertices in S. We let (u_i, v_i) denote the endpoints of ρ_i. Next, we iterate over the paths ρ_i for $i = 1, \ldots, z$. To determine which paths are affordable, we define the functions value$(\cdot)$ and cost$(\cdot)$:

- let cost(ρ_i) be the number of edges of ρ_i that are absent in G_{i-1}
- let value(ρ_i) be the number of pairs (x, C), where $x \in \{u_i, v_i\}$ and $C \in \mathcal{C}$ is a cluster, such that ρ_i contains at least one vertex of C and the distance between x and C in the graph G_{i-1} is strictly greater than the distance between u and C in ρ_i, i.e., $\delta_{G_{i-1}}(x, C) > \delta_{\rho_i}(x, C)$.

Our path-buying strategy is as follows. If

$$\text{cost}(\rho_i) \leq 2\,\text{value}(\rho_i)$$

then we buy the path ρ_i, that is we set $G_i := G_{i-1} \cup \rho_i$ (in words, G_i is given by G_{i-1} plus the edges of ρ_i not in G_{i-1}). Otherwise (i.e., $2\text{value}(\rho_i) < \text{cost}(\rho_i)$), we do not buy ρ_i and set $G_i := G_{i-1}$. The subsetwise spanner is given by $G_s := G_z$.

The next two lemmas bound the stretch and the size of the constructed spanner G_s, respectively

▶ **Lemma 7.** *For any $(u_i, v_i) \in \mathcal{P}$, $\delta_{G_s}(u_i, v_i) \leq \delta_G(u_i, v_i) + 2$.*

Proof. Clearly the claim holds if our algorithm bought the path ρ_i, hence we assume $2\text{value}(\rho_i) < \text{cost}(\rho_i)$. Let cost$(\rho_i) = t$, that is there are exactly t edges of ρ_i which are not present in the graph G_{i-1}. By Lemma 6 there are at least $t/2$ clusters having at least one vertex on ρ_i. If there is no cluster C among them such that $\delta_{G_{i-1}}(u_i, C) = \delta_G(u_i, C)$ and $\delta_{G_{i-1}}(v_i, C) = \delta_G(v_i, C)$, then all these clusters would contribute to value(ρ_i) (either with

u_i or with v_i or both) which leads to a contradiction, because $t = 2 \cdot (t/2) \leq 2\text{value}(\rho_i) < \text{cost}(\rho_i) = t$.

Thus there is a cluster C having a vertex of ρ_i such that $\delta_{G_{i-1}}(u_i, C) = \delta_G(u_i, C)$ and $\delta_{G_{i-1}}(v_i, C) = \delta_G(v_i, C)$. This implies:

$$
\begin{aligned}
\delta_{G_s}(u_i, v_i) \; \leq \; \delta_{G_{i-1}}(u_i, v_i) \; &\leq \; \delta_{G_{i-1}}(u_i, C) + \delta_{G_{i-1}}(v_i, C) + 2 \\
&\leq \; \delta_G(u_i, C) + \delta_G(v_i, C) + 2 \\
&\leq \; \delta_G(u_i, v_i) + 2,
\end{aligned}
$$

where the first inequality is because G_{i-1} is a subgraph of G_s, the second inequality holds since any two vertices of C are at distance at most two in $G_{\mathcal{C}} \subseteq G_{i-1}$ (by the cluster-diameter property) and the last inequality follows from the assumption that C contains a vertex of ρ_i. ◄

▶ **Lemma 8.** *For β such that $n^\beta = \sqrt{|S|}$ the graph G_s contains $O(n\sqrt{|S|})$ edges.*

Proof. The clustering phase produces a graph with $O(n^{1+\beta}) = O(n\sqrt{|S|})$ edges. Let $\mathcal{B}$ be the set of paths bought in the path-buying phase. The total number of edges that appear in $\mathcal{B}$ and do not appear in $G_{\mathcal{C}}$ is equal to $\sum_{\rho_i \in \mathcal{B}} \text{cost}(\rho_i)$, which is upper bounded by $\sum_{\rho_i \in \mathcal{B}} 2\text{value}(\rho_i)$. Observe that after the first contribution of a pair (x, C) to the above sum, the distance between x and C is at most $\delta_G(x, C) + 2$, hence each pair (x, C) can contribute to the sum at most 3 times. Therefore the total number of edges added in the second phase of our algorithm is upper bounded by $O(n^{1-\beta}|S|) = O(n\sqrt{|S|})$. ◄

The proof of Theorem 3 follows from Lemmas 7 and 8.

4   Sourcewise spanners

In this section we present our algorithm to compute a sourcewise spanner from sources S, and prove Theorem 4.

Our algorithm again consists of two phases, where the first is a clustering phase and the second is a path-buying phase. The clustering phase is as in the algorithm from previous section, for a proper value of β to be defined later. Let $\mathcal{C}$ and $G_{\mathcal{C}}$ be the resulting clustering and cluster subgraph.

At the start of the second phase we set $G_0 := G_{\mathcal{C}}$ and define $\{\rho_1, \ldots, \rho_z\}$ as the set of shortest paths between any two vertices of V such that at least one of them belongs to S. Let us assume that the path ρ_i is a shortest path between $u_i \in S$ and $v_i \in V$. Next, we iterate over paths ρ_i for $i = 1, \ldots, z$. For a given i we are going to define paths ρ_i^j, where $0 \leq j \leq k$, maintaining the following invariants:

(i) ρ_i^j is a path between u_i and v_i in G of length at most $\delta_G(u_i, v_i) + 2j$,
(ii)   any cluster $C \in \mathcal{C}$ contains at most three vertices of ρ_i^j,
(iii) $\text{cost}(\rho_i^j) \leq 2n^{1-\beta}/\gamma^j$, where $\text{cost}(\rho_i^j)$ is the number of edges of ρ_i^j absent in G_{i-1}, and $\gamma = (3n^{1-\beta})^{1/k}$.

Our algorithm will buy exactly one path ρ_i^j for $0 \leq j \leq k$, which will ensure (by Invariant (i)) that in G_i, the distance between u_i and v_i is at most $\delta_G(u_i, v_i) + 2k$.

We set $\rho_i^0 := \rho_i$. Observe that for $j = 0$, Invariant (i) is trivially satisfied, Invariant (ii) is satisfied by the cluster-diameter property (otherwise ρ_i would not be a shortest path), and Invariant (iii) is satisfied because there are at most $n^{1-\beta}$ clusters in $\mathcal{C}$ and consequently by Lemma 6 $\text{cost}(\rho_i) \leq 2n^{1-\beta}$.
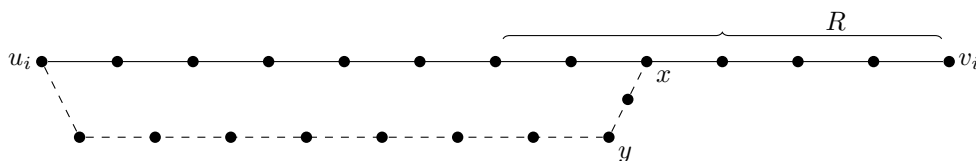
Figure 1 The solid edges represent a path ρ_i^j, while the dashed edges denote the new prefix of the path ρ_i^{j+1}.

Say we have constructed ρ_i^j, where $j \in \{0, \ldots, k\}$. Let us define the function value(ρ_i^j) as the number of clusters $C \in \mathcal{C}$ such that C contains a vertex of ρ_i^j and the distance between u_i and C in G_{i-1} is strictly greater than the distance between u_i and C in ρ_i^j, i.e., $\delta_{G_{i-1}}(u_i, C) > \delta_{\rho_i^j}(u_i, C)$. Now we check the condition

$$\text{cost}(\rho_i^j) \leq 3\gamma\text{value}(\rho_i^j).$$

If that is the case, then we buy the path ρ_i^j. That is, G_i is set to $G_{i-1} \cup \rho_i^j$. We ignore the remaining values of j and proceed with the next value of i. Else we construct ρ_i^{j+1} as follows:

Let R be the longest suffix of ρ_i^j containing exactly $\lfloor \text{cost}(\rho_i^j)/\gamma \rfloor$ edges that are absent in G_{i-1}. Observe that the first node of R is clustered: by the maximality of R, the edge e of ρ_i^j preceding R is absent in G_{i-1}, and hence both the endpoints of e (one of which is the first node of R) are clustered by the missing-edge property of $G_{\mathcal{C}}$. Consequently at least $1 + \lfloor \text{cost}(\rho_i^j)/\gamma \rfloor \geq \text{cost}(\rho_i^j)/\gamma$ vertices of R are clustered, as R contains $\lfloor \text{cost}(\rho_i^j)/\gamma \rfloor$ edges absent in G_{i-1} and the endpoints of these edges are clustered.

By Invariant (ii) there are at least $\text{cost}(\rho_i^j)/(3\gamma)$ clusters in $\mathcal{C}$ having at least one vertex of R. Since we did not buy ρ_i^j, there exists a cluster $C \in \mathcal{C}$ containing a vertex $x \in C$ of R such that the distance between u_i and C in G_{i-1} is at most the distance between u_i and x in ρ_i^j. We construct the path ρ_i^{j+1} by taking a shortest path in G_{i-1} from u_i to the closest node $y \in C$, then we add a path of length at most two between y and x (which exists in $G_{\mathcal{C}}$ hence in G_{i-1} by the cluster-diameter property), and finally add the suffix of R starting at x (see Fig. 1).

Let us show that ρ_i^{j+1} maintains the invariants. Note that by construction, Invariant (i) is satisfied, since the length of ρ_i^{j+1} is at most the length of ρ_i^j plus 2. Then, as long as there is a cluster $C \in \mathcal{C}$ containing at least four vertices on ρ_i^{j+1}, we let a, b be the vertices of ρ_i^{j+1} closest to u_i and v_i respectively. Note that there are at least three edges on ρ_i^{j+1} between a and b, hence we can replace the subpath of ρ_i^{j+1} by adding the at most two edges of $G_{\mathcal{C}}$ guaranteed by the cluster-diameter property. Consequently, Invariant (ii) is satisfied. Moreover, by the choice of R, Invariant (iii) is also satisfied. This finishes the construction of ρ_i^{j+1}.

Observe that by Invariant (iii) we have $\text{cost}(\rho_i^k) \leq 2/3$: since $\text{cost}(\cdot)$ has only integral values, it has to be that $\text{cost}(\rho_i^k) = 0$, which ensures that we buy a path ρ_i^j for some $j \leq k$.

Finally, as our spanner G_s we take $G_s := G_z$.

▶ **Lemma 9.** *For any pair $(u_i, v_i) \in \mathcal{P}$, $\delta_{G_s}(u_i, v_i) \leq \delta_G(u_i, v_i) + 2k$.*

Proof. From the above discussion, we buy at least one path ρ_i^j for some $0 \leq j \leq k$. By Invariant (i), the length of the latter path is at most the length of the shortest path ρ_i between u_i and v_i plus $2k$. ◀

▶ **Lemma 10.** *For β such that $n^\beta = (n^{1/k}(2k+3)|S|)^{k/(2k+1)}$, the subgraph G_s contains $O(n^{1+1/(2k+1)}(k|S|)^{k/(2k+1)})$ edges.*

Proof. To bound the size of G_s we recall that in the first phase we have inserted $O(n^{1+\beta})$ edges. Let $0 \le j_i \le k$ be the index of a path $\rho_i^{j_i}$ bought for a given i. We claim, that any cluster C contributes to $\text{value}(\rho_i^{j_i})$ of at most $|S|(2k+3)$ bought paths. This holds because when for $u_i \in S$ a supported path is bought the distance between u_i and C is at most $2k+2$ greater than the distance between u_i and C in G: otherwise one could shorten $\rho_i^{j_i}$ by more than $2k$, obtaining a contradiction with Invariant (i). Therefore the total number of edges added during the second phase is upper bounded by $\sum_{i=1}^z \text{cost}(\rho_i^{j_i}) \le \sum_{i=1}^z 3\gamma\text{value}(\rho_i^{j_i}) \le 3\gamma(2k+3)|S|n^{1-\beta}$, since each cluster $C \in \mathcal{C}$ supports at most $|S|(2k+3)$ bought paths. The claim follows. ◀

The proof of Theorem 4 follows from Lemmas 9 and 10.

5 Pairwise spanners

In this section we present our pairwise spanners for arbitrary $\mathcal{P}$. We start with a near-additive spanner (see Section 5.1) and then present a purely-additive spanner (see Section 5.2). In both cases we let $\mathcal{P} = \{(s_1, t_1), \ldots, (s_N, t_N)\}$ denote the set of pairs, $N = |\mathcal{P}|$.

5.1 A Near-Additive Pairwise Spanner

Our algorithm to construct the near-additive $\mathcal{P}$-spanner from Theorem 1 consists of three phases. First, we use Lemma 5 with the value of β to be determined later, obtaining a cluster subgraph $G_\mathcal{C}$ of G of size $O(n^{1+\beta})$ together with a set $\mathcal{C}$ of at most $n^{1-\beta}$ clusters.

At the start of the second phase we set $G_0 := G_\mathcal{C}$ and consider the set of paths $\{\rho_1, \ldots, \rho_N\}$, where ρ_i is a shortest path between s_i and t_i in G. Next we iterate over the paths ρ_i for $i = 1, \ldots, N$. By $\text{cost}(\rho_i)$ we denote the number of edges of ρ_i absent in G_{i-1}, and by $\text{value}(\rho_i)$ we denote the number of pairs of clusters $(C_1, C_2) \in \mathcal{C}$, such that both C_1 and C_2 contain at least one vertex of ρ_i and $\delta_{\rho_i}(C_1, C_2) < \delta_{G_{i-1}}(C_1, C_2)$. For a given i if
$$\text{cost}(\rho_i) \le \frac{12\log n}{\epsilon}\sqrt{\text{value}(\rho_i)},$$
then we buy ρ_i, that is we set $G_i := G_{i-1} \cup \rho_i$. Otherwise we set $G_i := G_{i-1}$.

In the third phase we add to G_N the multiplicative $(2\log n, 0)$ spanner of size $O(n)$ given in [15]: this way we obtain the desired spanner G_s.

In the following two lemmas we bound the stretch and size of G_s, respectively.

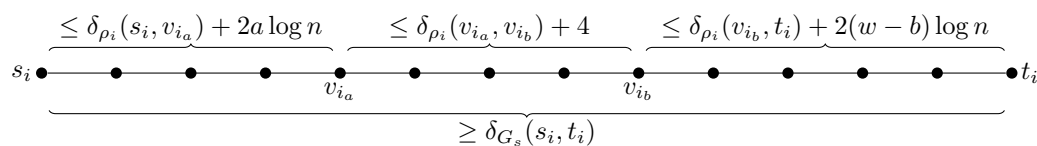▶ **Lemma 11.** *For each $(s_i, t_i) \in \mathcal{P}$, $\delta_{G_s}(s_i, t_i) \le (1+\varepsilon)\delta_G(s_i, t_i) + 4$.*

Proof. Clearly we can assume that the path ρ_i was not bought in the second phase, since otherwise the claim trivially holds. Therefore $\text{cost}(\rho_i) > \frac{12\log n}{\epsilon}\sqrt{\text{value}(\rho_i)}$.

Let $\rho_i = (v_0 = s_i, v_1, \ldots, v_{\ell-1}, v_\ell = t_i)$ and let $I \subseteq \{0, \ldots, \ell\}$ be the set of all indices j such that v_j is clustered. Observe that if $|I| \le 1$, then by the missing-edge property the whole path ρ_i is present in $G_\mathcal{C}$, and hence the claim holds. Therefore denote $I = \{i_0, \ldots, i_w\}$, where $i_0 < i_1 < \ldots < i_w$ and $w \ge 1$. Let $0 \le a \le b \le w$ be two indices, such that $v_{i_a} \in C_1$, $v_{i_b} \in C_2$ (for some $C_1, C_2 \in \mathcal{C}$), $\delta_{\rho_i}(C_1, C_2) \ge \delta_{G_{i-1}}(C_1, C_2)$ and the value of $b - a$ is maximized. Note that such a pair of indices a, b always exists, since we can take $a = b$.

Let $x = a + (w - b)$. Observe that any cluster $C \in \mathcal{C}$ contains at most 3 vertices of $V_I = \{v_{i_j} : 0 \le j \le w\}$, since otherwise by the cluster-diameter property ρ_i would not be a

$$\overbrace{\leq \delta_{\rho_i}(s_i, v_{i_a}) + 2a \log n}^{} \quad \overbrace{\leq \delta_{\rho_i}(v_{i_a}, v_{i_b}) + 4}^{} \quad \overbrace{\leq \delta_{\rho_i}(v_{i_b}, t_i) + 2(w - b) \log n}^{}$$

Figure 2 Illustration of the three paths concatenation in the proof of Lemma 11.

shortest s_i-t_i path. Therefore there are at least $x/6$ clusters $\mathcal{C}'$ having at least one vertex in the set $\{v_{i_j} : 0 \leq j < \lceil x/2 \rceil\}$, and at least $x/6$ clusters $\mathcal{C}''$ having at least one vertex in the set $\{v_{i_j} : w - \lceil x/2 \rceil < j \leq w\}$. However, each of the at least $(x/6)^2$ pairs of clusters in $\mathcal{C}' \times \mathcal{C}''$ contributes to value(ρ_i) since the difference between indices in the corresponding set is at least $w - \lceil x/2 \rceil + 1 - (\lceil x/2 \rceil - 1) > w - x$. Therefore $w \geq \text{cost}(\rho_i) > \frac{12 \log n}{\epsilon} \frac{x}{6}$ and hence $x \leq \frac{w\epsilon}{2 \log n}$.

The latter bound on x is sufficient to prove the claim. In fact, consider the path between s_i and t_i in G_s obtained by concatenating the following paths (as illustrated in Fig. 2):

- A shortest path in G_s from s_i to v_{i_a}. Note that in the prefix of ρ_i between s_i and v_{i_a} there are $a+1$ clustered nodes and hence at most a edges absent in G_{i-1} (by the missing-edge property). Since G_s contains the $(2 \log n, 0)$-spanner added in the third phase, each missing edge can be replaced by at path of length $2 \log n$. Consequently, there is a path from s_i to v_{i_a} of length at most $\delta_{\rho_i}(s_i, v_{i_a}) + 2a \log n$ in G_s.

- A shortest path in G_s from v_{i_a} to v_{i_b}. Let $C_1, C_2 \in \mathcal{C}$ be the clusters containing v_{i_a} and v_{i_b} respectively. We know that in G_{i-1} there is a path from C_1 to C_2 of length at most $\delta_{\rho_i}(v_{i_a}, v_{i_b})$, which can be extended to a path between v_{i_a} and v_{i_b} in G_{i-1} by adding at most 4 edges (by the cluster-diameter property).

- A shortest path in G_s from v_{i_b} to t_i. Observe that in the suffix of ρ_i between v_{i_b} and t_i there are at most $w - b$ edges absent in G_{i-1} by the same argument as above. Hence, thanks to the $(2 \log n, 0)$-spanner added in the third phase, there is a path from v_{i_b} to t_i of length at most $\delta_{\rho_i}(v_{i_b}, t_i) + (w - b)2 \log n$ in G_s.

The resulting path is of length at most

$$\delta_{\rho_i}(s_i, v_{i_a}) + 2a \log n + \delta_{\rho_i}(v_{i_a}, v_{i_b}) + 4 + \delta_{\rho_i}(v_{i_b}, t_i) + (w - b)2 \log n$$
$$= \delta_G(s_i, t_i) + 2x \log n + 4 \leq (1 + \epsilon)\delta_G(s_i, t_i) + 4,$$

where the last inequality follows from $x \leq \frac{\epsilon w}{2 \log n}$ together with $w \leq \delta_G(s_i, t_i)$. ◄

Due to space limitation we postpone the proof of the following lemma to the full version.

▶ **Lemma 12.** *For β such that $n^{2\beta} = \sqrt{N}\frac{\log n}{\epsilon}$ the size of G_s is $O(nN^{1/4}\sqrt{\log n/\epsilon})$.*

Having Lemmas 11 and 12, the proof of Theorem 1 follows.

5.2 A Purely-Additive Pairwise Spanner

In this section we describe an algorithm to compute the purely-additive $\mathcal{P}$-spanner from Theorem 2. To that aim we will combine ideas from the proofs of Theorems 1 and 4.

Our algorithm consists of the usual clustering phase (for an appropriate parameter β) followed by a path-buying phase that we next describe.

Let $\mathcal{C}$ and $G_\mathcal{C}$ be the clustering and the associated cluster graph. At the beginning of the path-buying phase, we set $G_0 := G_\mathcal{C}$ and consider the set $\{\rho_1, \ldots, \rho_N\}$, where ρ_i is a

shortest path between s_i and t_i in G. Next we iterate over the paths ρ_i for $i = 1, \ldots, N$. For a given i we are going to define paths ρ_i^j, where $0 \le j \le k$, maintaining the following invariants:

(i) ρ_i^j is a path between s_i and t_i in G of length at most $\delta_G(s_i, t_i) + 4j$,

(ii) any cluster $C \in \mathcal{C}$ contains at most three vertices of ρ_i^j,

(iii) $\text{cost}(\rho_i^j) \le 2n^{1-\beta}/\gamma^j$, where $\text{cost}(\rho_i^j)$ is the number of edges of ρ_i^j absent in G_{i-1}, and $\gamma = (3n^{1-\beta})^{1/k}$.

Our algorithm will buy exactly one path ρ_i^j, which will ensure by Invariant (i) that in G_i the distance between s_i and t_i is at most $\delta_G(u_i, v_i) + 4k$. By $\text{value}(\rho_i^j)$ let us denote the number of pairs of clusters $C_1, C_2 \in \mathcal{C}$, such that both C_1 and C_2 contain at least one vertex of ρ_i^j and $\delta_{\rho_i^j}(C_1, C_2) < \delta_{G_{i-1}}(C_1, C_2)$.

We set $\rho_i^0 := \rho_i$. Observe that for $j = 0$ Invariant (i) is trivially satisfied, Invariant (ii) is satisfied by the cluster-diameter property (otherwise ρ_i would not be a shortest path), and Invariant (iii) is satisfied because there are at most $n^{1-\beta}$ clusters in $\mathcal{C}$ and consequently by Lemma 6, $\text{cost}(\rho_i) \le 2n^{1-\beta}$.

Say we have constructed ρ_i^j, where $j \in \{0, \ldots, k\}$. If

$$\text{cost}(\rho_i^j) \le 6\gamma\sqrt{\text{value}(\rho_i^j)},$$

then we buy the path ρ_i^j, i.e. as G_i we take the union of G_{i-1} and ρ_i^j, ignore remaining values of j and proceed with the next value of i. Otherwise (i.e., $\text{cost}(\rho_i^j) > 6\gamma\sqrt{\text{value}(\rho_i^j)}$), we construct a path ρ_i^{j+1} as follows:

Let $\rho_i^j = (v_0 = s_i, v_1, \ldots, v_{\ell-1}, v_\ell = t_i)$ and let $I \subseteq \{0, \ldots, \ell\}$ be the set of all indices j such that v_j is clustered. Observe that if $|I| \le 1$, then by the missing-edge property the whole path ρ_i^j is present in $G_{\mathcal{C}}$, and hence it is of zero cost, which contradicts the assumption $\text{cost}(\rho_i^j) > 6\gamma\sqrt{\text{value}(\rho_i^j)}$. Therefore denote $I = \{i_0, \ldots, i_w\}$, where $i_0 < i_1 < \ldots < i_w$ and $w \ge 1$. Let $0 \le a \le b \le w$ be two indices, such that $v_{i_a} \in C_1$, $v_{i_b} \in C_2$ (for some $C_1, C_2 \in \mathcal{C}$), $\delta_{\rho_i^j}(C_1, C_2) \ge \delta_{G_{i-1}}(C_1, C_2)$ and the value of $b - a$ is maximized. Note that such a pair of indices a, b always exists, since we can take $a = b$.

Let $x = a + (w - b)$. By Invariant (ii) there are at least $x/6$ clusters $\mathcal{C}'$ having at least one vertex in the set $\{v_{i_j} : 0 \le j < \lceil x/2 \rceil\}$, and at least $x/6$ clusters $\mathcal{C}''$ having at least one vertex in the set $\{v_{i_j} : w - \lceil x/2 \rceil < j \le w\}$. However, each of the at least $(x/6)^2$ pairs of clusters in $\mathcal{C}' \times \mathcal{C}''$ contributes to $\text{value}(\rho_i^j)$ since the difference between indices in the corresponding set is at least $w - \lceil x/2 \rceil + 1 - (\lceil x/2 \rceil - 1) > w - x$. Therefore

$$\text{cost}(\rho_i^j) > 6\gamma\sqrt{\text{value}(\rho_i^j)} \ge 6\gamma\sqrt{\left(\frac{x}{6}\right)^2} = \gamma x \quad \Rightarrow \quad x \le \text{cost}(\rho_i^j)/\gamma. \tag{1}$$

We construct the path ρ_i^{j+1} by appending the following three paths A, B, and C:

- As A we take the prefix of ρ_i^j from s_i to v_{i_a}. Note that this prefix contains $a + 1$ clustered nodes and hence at most a edges absent in G_{i-1} (by the missing-edge property of $G_{\mathcal{C}}$).

- Let $C_1, C_2 \in \mathcal{C}$ be the clusters containing v_{i_a} and v_{i_b} respectively. We know that in G_{i-1} there is a path from C_1 to C_2 of length at most $\delta_{\rho_i^j}(v_{i_a}, v_{i_b})$, which can be extended to a path B between v_{i_a} and v_{i_b} in G_{i-1} by adding at most 4 edges (by the cluster-diameter property).

- As C we take the suffix of ρ_i^j from v_{i_b} to t_i, which contains at most $w - b$ edges absent in G_{i-1} by the same argument as above.

Observe that ρ_i^{j+1} contains at most $a + (w - b) = x$ edges absent in G_{i-1}, hence by (1) we ensure Invariant (iii). Moreover the length of ρ_i^{j+1} is at most the length of ρ_i^j plus 4, which ensures Invariant (i). In order to ensure Invariant (ii), as long as there exists a cluster $C \in \mathcal{C}$ containing at least 4 vertices of ρ_i^{j+1} we let u and v be two such vertices closest to s_i and t_i on ρ_i^{j+1} respectively and replace the subpath of ρ_i^{j+1} between u and v (which is of length at least three) by a path of length at most two in G_{i-1} (which exists by the cluster-diameter property).

Observe that by Invariant (iii) we have $\mathrm{cost}(\rho_i^k) \leq 2/3$, hence $\mathrm{cost}(\rho_i^k) = 0$ which ensures that we buy a path ρ_i^j for some $j \leq k$. Finally, as our spanner G_s we take $G_s := G_N$.

▶ **Lemma 13.** *For each* $(s_i, t_i) \in \mathcal{P}$, $\delta_{G_s}(s_i, t_i) \leq \delta_G(s_i, t_i) + 4k$.

Proof. From the above discussion, G_s contains at least one path ρ_i^j between s_i and t_i for some $0 \leq j \leq k$. The claim follows by Invariant (i).   ◀

▶ **Lemma 14.** *For β such that* $n^\beta = (6n^{1/k}\sqrt{(4k+5)N})^{k/(2k+1)}$ *the size of G_s is* $O(n^{1+1/(2k+1)}(\sqrt{(4k+5)N})^{k/(2k+1)})$.

Proof. The clustering phase gives $O(n^{1+\beta})$ edges, which matches the desired bound on G_s. Let $0 \leq j_i \leq k$ be the index of a path $\rho_i^{j_i}$ bought for a given i. We claim, that any pair of clusters contributes to $\mathrm{value}(\rho_i^{j_i})$ of at most $(4k + 5)$ bought paths. Observe, that if a pair of clusters C_1, C_2 contributes to $\mathrm{value}(\rho_i^{j_i})$, then when $\rho_i^{j_i}$ is bought we have $\delta_{G_i}(C_1, C_2) \leq \delta_G(C_1, C_2) + 4k + 4$, since otherwise the subpath of $\rho_i^{j_i}$ between C_1 and C_2 might be shortened by more than $4k$, contradicting Invariant (i). The total number of edges added in the second phase is upper bounded by

$$
\sum_{1 \leq i \leq N} \mathrm{cost}(\rho_i^{j_i}) \quad \leq \quad \sum_{1 \leq i \leq N} 6\gamma\sqrt{\mathrm{value}(\rho_i^{j_i})}
$$

$$
\overset{\substack{\text{Cauchy-Schwarz}\\\text{inequality}}}{\leq} \quad 6\gamma\sqrt{\sum_{1 \leq i \leq N} \mathrm{value}(\rho_i^{j_i})}\sqrt{N}
$$

$$
\leq \quad 6\gamma\sqrt{4k+5}\,n^{1-\beta}\sqrt{N}.
$$

By substituting γ and β the claim follows.   ◀

Theorem 2 follows from Lemmas 13 and 14.

6   Conclusions

We considered a natural extension to the problem of computing a sparse spanner in an undirected unweighted graph. Along with the input graph $G = (V, E)$, a subset $\mathcal{P} \subseteq V \times V$ of relevant pairs of vertices is also given here and we seek a sparse subgraph H of G such that for every pair (u, v) in $\mathcal{P}$, the u-v distance $\delta_H(u, v)$ in the subgraph is close to the u-v distance $\delta_G(u, v)$ in G. We showed sparse subgraphs H where $\delta_H(u, v)$ is a small additive or near-additive stretch away from $\delta_G(u, v)$.

The pairwise preservers in [8] are at the same time more accurate and sparser than our spanners for small enough values of $|\mathcal{P}|$. In particular, in that range of values of $|\mathcal{P}|$ the authors exploit a construction which does not seem to benefit from allowing a larger stretch. The authors also present lower bounds on the size of any preserver, however it is unclear whether those lower bounds extend to the case of pairwise spanners (where distances have to be approximated rather than preserved). Obtaining sparser pairwise spanners for very small $|\mathcal{P}|$, if possible, is an interesting open problem.

References

1 B. Awerbuch, B. Berger, L. Cowen, and D. Peleg. Near-linear time construction of sparse neighborhood covers. *SIAM Journal on Computing*, 28(1):263-277, 1998.

2 B. Awerbuch and D. Peleg. Routing with polynomial communication-space trade-off. *SIAM Journal on Discrete Math.*, 5(2):151-162, 1992.

3 S. Baswana and T. Kavitha. Faster algorithms for all-pairs approximate shortest paths in undirected graphs. *SIAM Journal on Computing*, 39(7):2865-2896, 2010.

4 S. Baswana, T. Kavitha, K. Mehlhorn, S. Pettie. Additive Spanners and (α, β)-Spanners. *ACM Transactions on Algorithms*, 7(1): 5, 2010.

5 S. Baswana and S. Sen. A simple linear time algorithm for computing a $(2k - 1)$-spanner of $O(n^{1+1/k})$ size in weighted graphs. In *Proc. 30th Int. Colloq. on Automata, Languages, and Programming (ICALP)*, pages 384-396, 2003.

6 S. Baswana and S. Sen. Approximate distance oracles for unweighted graphs in $O(n^2 \log n)$ time. In *Proc. 15th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 271-280, 2004.

7 E. Cohen. Fast algorithms for constructing t-spanners and paths of stretch t. In *Proc. 34th IEEE Symp. on Foundations of Computer Science (FOCS)*, pages 648-658, 1993.

8 D. Coppersmith and M. Elkin. Sparse source-wise and pair-wise distance preservers. In *Proc. 16th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 660-669, 2005.

9 L. J. Cowen. Compact routing with minimum stretch. *Journal of Algorithms*, 28:170-183, 2001.

10 L. J. Cowen and C. G. Wagner. Compact roundtrip routing in directed networks. *Journal of Algorithms*, 50(1):79-95, 2004.

11 D. Dor, S. Halperin, and U. Zwick. All-pairs almost shortest paths. *SIAM Journal on Computing*, 29(5):1740-1759, 2004.

12 M. Elkin. Computing almost shortest paths. In *ACM Transactions on Algorithms*, 1(2):283-323, 2005.

13 M. Elkin and D. Peleg. $(1 + \epsilon, \beta)$-spanner construction for general graphs. *SIAM Journal on Computing*, 33(3):608-631, 2004.

14 C. Gavoille, D. Peleg, S. Perennes, and R. Raz. Distance labeling in graphs. In *Proc. 12th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 210-219, 2001.

15 S. Halperin and U. Zwick. Unpublished result, 1996.

16 D. Peleg. Proximity-preserving labeling schemes. *Journal of Graph Theory*, 33(3):167-176, 2000.

17 D. Peleg and A. A. Schaffer. Graph Spanners. *Journal of Graph Theory*, 13:99-116, 1989.

18 D. Peleg and J. D. Ullman. An optimal synchronizer for the hypercube. *SIAM Journal on Computing*, 18:740-747, 1989.

19 L. Roditty and U. Zwick. On dynamic shortest paths problems. In *Proc. 12th Annual European Symposium on Algorithms (ESA)*, pages 580-591, 2004.

20 L. Roditty, M. Thorup, and U. Zwick. Deterministic constructions of approximate distance oracles and spanners. In *Proc. 32nd Int. Colloq. on Automata, Languages, and Programming (ICALP)*, pages 261-272, 2005.

21 M. Thorup and U. Zwick. Approximate Distance Oracles. *Journal of the ACM*, 52(1):1-24, 2005.

22 M. Thorup and U. Zwick. Spanners and emulators with sublinear distance errors. In *Proc. 17th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 802-809, 2006.