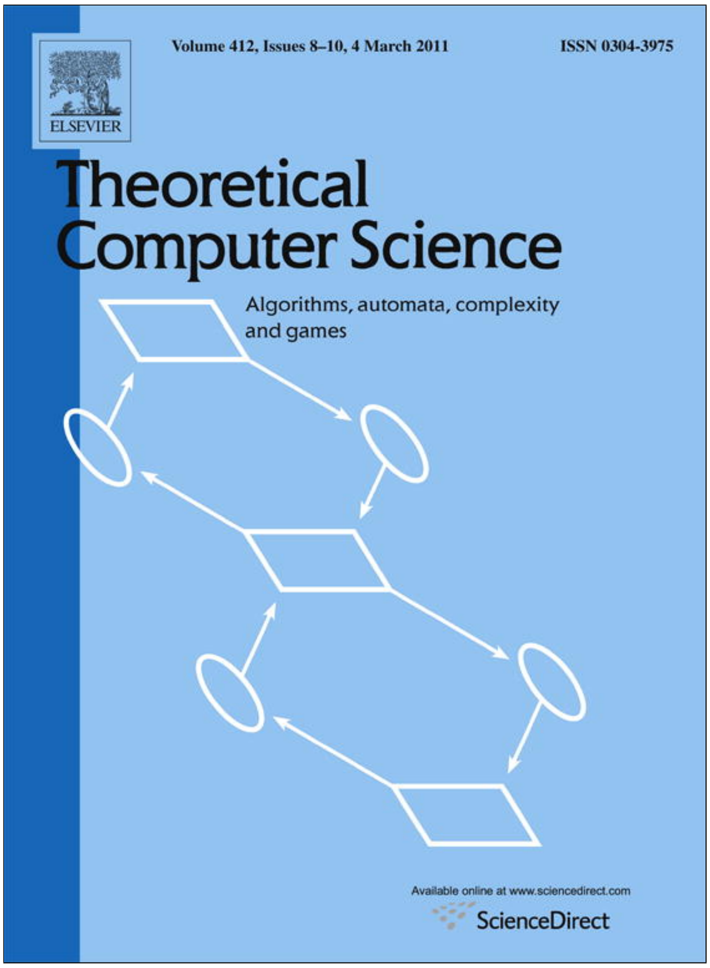
Volume 412, Issues 8–10, 4 March 2011
ISSN 0304-3975
ELSEVIER
Theoretical
Computer Science
Algorithms, automata, complexity
and games
Available online at www.sciencedirect.com
ScienceDirect

Theoretical Computer Science 412 (2011) 846–852

Contents lists available at ScienceDirect

Theoretical Computer Science

journal homepage: www.elsevier.com/locate/tcs

Approximation of minimum weight spanners for sparse graphs ☆

Feodor F. Dragan [a], Fedor V. Fomin [b], Petr A. Golovach [c,*]

[a] Department of Computer Science, Kent State University, Kent, OH 44242, USA
[b] Department of Informatics, University of Bergen, PB 7803, N-5020 Bergen, Norway
[c] School of Engineering and Computing Sciences, Durham University, South Road, DH1 3LE Durham, UK

ARTICLE INFO

Article history:
Received 21 January 2010
Received in revised form 27 September 2010
Accepted 16 November 2010
Communicated by D. Peleg

Keywords:
Graph algorithms
Approximation
Graph spanners

ABSTRACT

A t-*spanner* of a graph G is its spanning subgraph S such that the distance between every pair of vertices in S is at most t times their distance in G. The SPARSEST t-SPANNER problem asks to find, for a given graph G and an integer t, a t-spanner of G with the minimum number of edges. The problem is known to be NP-hard for all $t \geq 2$, and, even more, it is NP-hard to approximate it with ratio $O(\log n)$ for every $t \geq 2$. For $t \geq 5$, the problem remains NP-hard for planar graphs and the approximability status of the problem on planar graphs was open. We resolve this open issue by showing that the SPARSEST t-SPANNER problem admits the *efficient polynomial time approximation scheme (EPTAS)* for every $t \geq 1$. Our result holds for a much wider class of graphs, namely, the class of *apex-minor-free graphs*, which contains the classes of planar and bounded genus graphs. Moreover, it is possible to extend our results to weighted apex-minor free graphs, when the maximum edge weight is bounded by some constant.

1. Introduction

The concept of *sparse graph spanners* was introduced in [32–34] and has been studied since then in the context of wired or wireless communication networks, distributed computing, robotics, computational geometry and biology [1,2,11,12,14,15,17,32–34]. A t-*spanner* of a graph G is a spanning subgraph S in which the distance between every pair of vertices is at most t times their distance in G. One is interested in finding a sparsest t-spanner for a graph G, i.e., a t-spanner with the minimum number of edges.

The original application of spanners was in the design of low-stretch routing schemes for communication networks using small routing tables [33] (see also [5,38] and the references therein) and the efficient simulation of synchronized protocols in unsynchronized networks [4,34]. Thereafter spanners were used in computing almost shortest paths in graphs [21] and in approximation algorithms for geometric spaces [31]. A recent application of spanners is in the design of approximate distance oracles and labeling schemes for arbitrary metrics; see [38,39] for further references. In all the applications cited above the quality of the solution is directly related to the quality of the underlying spanners. For example, in [34], close relationships were established between the quality of spanners (in terms of *stretch factor* t and the number of spanner edges), and the time and communication complexities of any synchronizer for the network based on this spanner.

There is a rich literature on approximation algorithms for the sparsest t-spanner problem [1,22,23,35,37,39]. In particular, for every $t \geq 2$ there is a constant $c < 1$ such that it is NP-hard to approximate the sparsest t-spanner with the ratio $c \cdot \log n$,

☆ A preliminary version of these results (for unweighted graphs) appeared in proceedings of the 33rd International Symposium on Mathematical Foundations of Computer Science (MFCS 2008) (Dragan et al. 2008 [19]).

* Corresponding author. Tel.: +44 7826054387.
E-mail addresses: dragan@cs.kent.edu (F.F. Dragan), fedor.fomin@ii.uib.no (F.V. Fomin), petr.golovach@durham.ac.uk (P.A. Golovach).

0304-3975/$ – see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.tcs.2010.11.034

where n is the number of vertices in the graph [28]. On the other hand, the problem admits a $O(\log n)$-ratio approximation for $t = 2$ [28,29] and a $O(n^{2/(t+1)})$-ratio approximation for $t > 2$ [23].

The problem was studied on planar graphs as well. The problem of determining, for a given planar graph G and integers m and t, if G has a t-spanner with at most m edges is NP-complete for every fixed $t \geq 5$ (the case $2 \leq t \leq 4$ is still open) [10]. On the other hand, for all fixed integers t and r, it is possible to decide in a polynomial time whether a planar (and more generally, apex-minor-free) graph G has a t-spanner with at most $n - 1 + r$ edges [18]. Duckworth et al. [20] obtained a polynomial time approximation scheme (PTAS) for the sparsest 2-spanner problem on 4-connected planar triangulations, which is a special subclass of planar graphs. The question, whether there is a PTAS for the sparsest t-spanner problem for $t > 2$ on 4-connected planar triangulations, or even more generally, for all planar graphs, was left open in [20].

In this paper, we resolve positively the open problem of Duckworth, Wormald, and Zito [20] by showing that the sparsest t-spanner problem admits the *efficient polynomial time approximation scheme* (EPTAS) on the class of planar graphs for every $t \geq 1$. In other words, for every $t \geq 1$ and $\epsilon > 0$, we prove that it is possible to construct a spanner with at most $(1 + \epsilon) \cdot \mathrm{OPT}(G)$ edges in $\frac{1}{\epsilon} \cdot t^{O((t/\epsilon)^2)} \cdot n$ time for an n-vertex planar graph G, where $\mathrm{OPT}(G)$ is the number of edges of a sparsest spanner. Actually, we prove much stronger result.

- We give EPTAS for much more general class of graphs, namely apex-minor-free graphs, which include planar graphs and graphs of bounded genus.
- We consider the more general Minimum Weight t-spanner problem, which asks to find, for a given weighted graph G and an integer t, a t-spanner of G of minimum weight. In particular, we show that for all positive integers t and W, the Minimum Weight t-spanner problem admits a EPTAS on apex-minor-free graphs with positive integer weights of edges at most W with running time $\frac{1}{\epsilon} \cdot (W \cdot t)^{O((W \cdot t/\epsilon)^2)} \cdot n$.

Our proof is based on the technique for solving NP-hard problems on planar graphs proposed by Baker [6] and generalized by Eppstein [24,25] (see also [16,26]) to minor closed graph classes with bounded local treewidth (alias, apex-minor-free graphs). An extended abstract of these results appeared in [19].

2. Preliminaries

In this section we present necessary definitions, notations and some auxiliary results.

A *polynomial time approximation scheme* (PTAS) is an algorithm which takes an instance of an optimization problem and a parameter $\epsilon > 0$ and, in polynomial time, produces a solution that is within a factor $1 + \epsilon$ of being optimal. A PTAS is called an *efficient polynomial-time approximation scheme* or EPTAS if its running time is $O(f(\epsilon) \cdot n^c)$ for some function f and a constant c independent of ϵ where n is the input size (see. e.g., [13]).

Let $G = (V, E)$ be an undirected weighted graph with the vertex set V, edge set E, and weight function $w : E(G) \to \mathbb{N}$. We often use notations $V(G) = V$ and $E(G) = E$. For $U \subseteq V$, by $G[U]$ is denoted the subgraph of G induced by U. For an edge $e \in E(G)$, $w(e)$ is called the *weight* of e. We consider only graphs with positive integer weights of edges. For a set $A \subset E(G)$, $w(A) = \sum_{e \in A} w(e)$. If H is a subgraph of G, the *weight* of H is defined as $w(E(H))$ and is denoted by $w(H)$. For a path P, we also call $w(P)$ the *length* of P. The *distance* $\mathrm{dist}_G(u, v)$ between vertices u and v of a connected weighted graph G is the length of a shortest u, v-path in G. The *intersection* of graphs G and H is the graph $G \cap H = (V(G) \cap V(H), E(G) \cap E(H))$.

Let t be a positive integer. A subgraph S of a weighted connected graph G, such that $V(S) = V(G)$, is called a *(multiplicative) t-spanner* of G, if $\mathrm{dist}_S(u, v) \leq t \cdot \mathrm{dist}_G(u, v)$ for every pair of vertices u and v. The parameter t is called the *stretch factor* of S. It is easy to see that the t-spanners can equivalently be defined as follows.

Proposition 1. *Let G be a connected weighted graph, and t be a positive integer. A spanning subgraph S of G is a t-spanner of G if and only if for every edge xy of G, $\mathrm{dist}_S(x, y) \leq t \cdot w(xy)$.*

Let $A \subseteq E(G)$. We call a subgraph S of G, such that for every edge $xy \in A$, $\mathrm{dist}_S(x, y) \leq t \cdot w(xy)$, a *partial t-spanner for A*. Clearly, if $A = E(G)$ then a partial t-spanner for this set is a t-spanner for G.

The Minimum Weight t-Spanner problem asks to find, for a given weighted graph G and an integer t, a t-spanner of G of minimum weight. Correspondingly, the Minimum Weight Partial t-Spanner problem asks to find a partial t-spanner of minimum weight for a given weighted graph G, an integer t and a set $A \subseteq E(G)$. It is easy to see that for unweighted graphs the Sparsest t-Spanner problem, which asks for a t-spanner with the minimum number of edges, is a special case of the Minimum Weight t-Spanner problem.

A *tree decomposition* of a graph G defined by Robertson and Seymour [36] is a pair (X, T) where T is a tree whose vertices we call *nodes* and $X = (\{X_i \mid i \in V(T)\})$ is a collection of subsets of $V(G)$ such that

1. $\bigcup_{i \in V(T)} X_i = V(G)$,
2. for each edge $vw \in E(G)$, there is an $i \in V(T)$ such that $v, w \in X_i$, and
3. for each $v \in V(G)$ the set of nodes $\{i \mid v \in X_i\}$ forms a subtree of T.

The *width* of a tree decomposition $(\{X_i \mid i \in V(T)\}, T)$ equals $\max_{i\in V(T)}\{|X_i| - 1\}$. The *treewidth* of a graph G is the minimum width over all tree decompositions of G. We use notation $\mathbf{tw}(G)$ to denote the treewidth of a graph G. It is known that a graph G has treewidth k if and only if G is a partial k-tree (see, e.g., [3,7,27] and papers cited therein).

A graph class $\mathcal{G}$ has bounded local treewidth if there is a function $f(r)$ such that for any graph G in $\mathcal{G}$, the treewidth of every induced subgraph of G of diameter at most r is at most $f(r)$. A graph class $\mathcal{G}$ has a *linear local treewidth* if $f(r) = O(r)$. For example, planar graphs are graphs of linear local treewidth because for every planar graph G of radius at most r, the treewidth of G is at most $3r - 1$, see e.g. [8].

Given an edge $e = xy$ of a graph G, the graph G/e is obtained from G by contracting the edge e; that is, to get G/e we identify the vertices x and y and remove all loops and replace all multiple edges by simple edges. A graph H obtained by a sequence of edge-contractions is said to be a *contraction* of G. H is a *minor* of G if H is a subgraph of a contraction of G. A graph class $\mathcal{G}$ is *minor-closed* if for every graph $G \in \mathcal{G}$ all minors of G are in $\mathcal{G}$, too.

We say that a graph G is *H-minor-free* when it does not contain H as a minor. We also say that a graph class $\mathcal{G}$ is *H-minor-free* (or, excludes H as a minor) when all its members are H-minor-free. Clearly, all minor-free graph classes are minor-closed.

An *apex graph* is a graph that can be made planar by the removal of a single vertex. The deleted vertex is called an *apex* of the graph. The main property of planar graphs, which apex graphs do not have, is the bounded local treewidth. For example, a $k \times k$ grid with apex adjacent to all vertices of the grid has treewidth $k + 1$ and diameter 2. A graph class is *apex-minor-free* if it does not contain any graph with some fixed apex graph as a minor. Particularly, planar graphs (and bounded-genus graphs) are apex-minor-free graphs.

Eppstein [24,25] characterized all minor-closed graph classes that have bounded local treewidth by proving that they are exactly apex-minor-free graphs. These results were improved by Demaine and Hajiaghayi [16] who proved that all apex-minor-free graphs have linear local treewidth.

3. Partial spanners for graphs of bounded treewidth

In this section, we show how to solve the Minimum Weight Partial t-Spanner problem on graphs with bounded treewidth and with bounded edge-weights.

Let t and W be positive integers, and let G be an n-vertex weighted connected graph such that $w(e) \leq W$ for every $e \in E(G)$. Suppose that $A \subseteq E(G)$. Let also (X, T) be a tree decomposition of G of width k.

Every tree decomposition can be converted (in linear time) into a *nice* tree decomposition [27] of same width (and with a linear size of T) with the rooted binary tree T with the root r, which induces a parent–child relation in the tree, such that nodes of T are of four types:

1. *Leaf* nodes i are leaves of T and have $|X_i| = 1$.
2. *Introduce* nodes i have one child j with $X_i = X_j \cup \{v\}$ for some vertex $v \in V(G)$.
3. *Forget* nodes i have one child j with $X_i = X_j \setminus \{v\}$ for some vertex $v \in V(G)$.
4. *Join* nodes i have two children j_1 and j_2 with $X_i = X_{j_1} = X_{j_2}$.

For any node $i \in V(T)$, we denote by T_i the rooted subtree induced by the descendants of i with the root i. We also define subgraph

$$G_i = G\Bigg[\bigcup_{j\in V(T_i)} X_j\Bigg].$$

Our algorithm follows a classical dynamic programming approach on graphs of bounded treewidth (see e.g. the survey [9]). It constructs for every node $i \in V(T)$, starting from leaves, and tables of data. From the table computed for the root r, we are able to find the spanner.

The following intuition is behind our construction. Let S be a t-spanner of G. Denote by S_i the subgraph of S induced by the vertices of G_i. For every edge $uv \in A$, the length of the shortest u, v-path in S is at most $t \cdot w(uv)$. If $uv \in A \setminus E(G_i)$, then the shortest u, v-path in S can include edges of $G[X_i]$ and segments which are paths with endpoints in X_i and all internal vertices in $V(S_i) \setminus X_i$. Hence, we keep the information about edges of S_i with endpoints in the bag X_i and paths in S_i which join vertices of the bag such that all internal vertices are not included in X_i. Observe that only paths of length at most $W \cdot t$ can be segments of the u, v-paths of lengths at most $t \cdot w(uv) \leq t \cdot W$. Therefore, it is sufficient to store the information about such paths only. In similar way, if $uv \in A \cap E(G_i)$, then the shortest u, v-path in S can include paths with endpoints in X_i and all internal vertices in $V(G) \setminus V(G_i)$ as segments. Hence we keep the corresponding information. Notice again that we are only interested in paths of length at most $W \cdot t$.

Now we describe the tables of data. For $x, y \in X_i, x \neq y$, it is assumed that $d_{x,y}$ and $l_{x,y}$ are either integers from the set $\{1, \ldots, W \cdot t\}$ or $+\infty$. These integers correspond to lengths of x, y-paths in the constructed spanner with all internal vertices in $V(G_i) \setminus X_i$ and in $V(G) \setminus V(G_i)$. Let $E_i \subseteq E(G[X_i])$, $D_i = \{d_{x,y} : x, y \in X_i, x \neq y\}$ and $L_i = \{l_{x,y} : x, y \in X_i, x \neq y\}$. For the triple (E_i, D_i, L_i), the table for the node i keeps the following information. If there is a subgraph S_i of G_i (a partial solution) such that

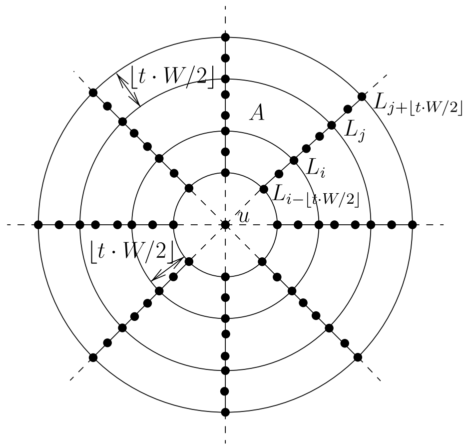

Fig. 1. Construction of graphs G_{ij} and G'_{ij}.

1. $E_i = \{xy \in E(S_i) : x, y \in X_i\}$, i.e. E_i is the set of edges of the partial solutions with endpoints in the bag X_i;
2. for each pair of different vertices $x, y \in X_i$ such that $d_{x,y} < +\infty$, there is an x, y- path in S_i of length at most $d_{x,y}$ with all internal vertices in $V(S_i) \setminus X_i$, i.e. these x, y-paths of length at most $d_{x,y}$ can be used further;
3. if S'_i is the graph obtained by joining pairs of different vertices $x, y \in X_i$ by paths of length $l_{x,y}$ with all internal vertices laying outside S_i (if $l_{x,y} = +\infty$ then it is assumed that there is no path), then for each $uv \in A \cap E(G_i)$, $\mathrm{dist}_{S'_i}(u, v) \le t \cdot w(uv)$, i.e. if such x, y-paths would be added to the partial solution in the further stages of the algorithm then for any $uv \in A \cap E(G_i)$, u, v-path of length at most $t \cdot w(uv)$ could be found in the spanner;

then the table stores the set of edges of such S_i with the minimum possible weight (this set is a partial solution of our problem for G_i). Otherwise the table keeps NO. Observe that each table always stores at least one entry different from NO, since for $E_i = \{xy \in E(S_i) : x, y \in X_i\}$ and $d_{x,y}, l_{x,y} = +\infty, S_i = G_i$ satisfies Conditions 1–3.

Such tables for all possible triples can be constructed for leaves of T in a trivial way (since corresponding bags contain unique vertices), and it can be easily checked that the table for a vertex i can be computed if the tables for children of i are given. If the table for the root r is constructed, then we can find the partial t-spanner for G in the following way. Let $l_{x,y} = +\infty$ for all $x, y \in X_r$ and let $L_r = \{l_{x,y} : x, y \in X_r, x \neq y\}$. Checking all $E_r \subset E(G[X_r])$ and collections $D_r = \{d_{x,y} : x, y \in X_r, x \neq y\}$, we find the partial solution S_r of minimum weight.

The size of each table can be estimated as $2^{\binom{k+1}{2}} \cdot (W \cdot t + 1)^{2\binom{k+1}{2}}$. Therefore, the running time of the algorithm is $(W \cdot t)^{O(k^2)} \cdot |V(G)|$.

We conclude with the following lemma.

Lemma 1. *Let t, k and W be positive integers. Let also G be a connected weighted graph of treewidth at most k such that $w(e) \le W$ for all $e \in E(G)$, and let $A \subseteq E(G)$. The* Minimum Weight Partial t-Spanner *problem on G can be solved in time $(W \cdot t)^{O(k^2)} \cdot |V(G)|$, if a corresponding tree decomposition of G is given.*

Observe in the conclusion of this section, that Arnborg et al. [3] described classes of optimization problems which can be solved in linear time for graphs of bounded treewidth. It is possible to show that the Minimum Weight Partial t-Spanner with bounded weights of edges is in one of these classes. However, the only thing we are able to say about the running time of their algorithm is that it is $O(f(W, k, t) \cdot |V(G)|)$ for some function f. It also should be noted that the dynamic programming algorithm for the case $A = E(G)$ and unweighted graphs with similar running time is due to Makowsky and Rotics [30].

4. Main result

Now we are ready to describe our EPTAS for the Minimum Weight t-Spanner problem for sparse graphs.

Let t and W be positive integers. Let also G be a connected weighted graph such that $w(e) \le W$ for all $e \in E(G)$.

We choose a vertex u of G. For $i \ge 0$, we denote by L_i the ith level of the breadth first search, i.e., the set of vertices at distance i from u. We call the partition of the vertex set $V(G)\mathcal{L}(G, u) = \{L_0, L_1, \dots, L_r\}$ *breadth first search (BFS) decomposition* of G. We assume for convenience that for BFS decomposition $\mathcal{L}(G, u)$, $L_i = \emptyset$ for $i < 0$ or $i > r$, so we can use negative indices and indices that are more than r. It can be easily seen that the BFS decomposition can be constructed by the breadth first search in linear time.

Suppose that $i \le j$ are integers. For $i \le j$ we define

$$G_{ij} = G\left[\bigcup_{s=i}^{j} L_s\right].$$

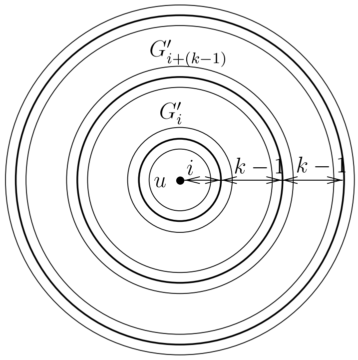

Fig. 2. Graphs G'_j.

The following result is due to Demaine and Hajiaghayi [16] (see also the work of Eppstein [25]). It follows from the result of Demaine and Hajiaghayi that for every apex-minor-free graph G the treewidth of its subgraphs depends linearly on their diameters and the observation that the graph obtained from G by the contraction of all edges of $G[L_0 \cup \cdots \cup L_{i-1}]$ is the minor of G, and thus apex-minor-free.

Lemma 2 *([16]). Let G be an apex-minor-free graph. Then* $\mathbf{tw}(G_{ij}) = O(j - i)$.

Denote by $G'_{ij} = G_{i-\lfloor t\cdot W/2\rfloor, j+\lfloor t\cdot W/2\rfloor}$ (see Fig. 1), and let $A = E(G_{ij})$. Let S be a t-spanner of G and S' be the subgraph of S induced by $V(G'_{ij})$. We need the following lemma.

Lemma 3. *S' is a partial t-spanner for A in G'_{ij}.*

Proof. Let $xy \in A$. Note that $x, y \in V(G_{ij})$, i.e. $x, y \in L_i \cup \cdots \cup L_j$. Since S is a t-spanner of G, we have that there is a x, y-path P in S of length at most $t \cdot w(xy) \le t \cdot W$. We claim that P is a path in S'. Suppose, for the sake of contradiction, that some vertex v of this path does not belong to G'_{ij}. Then $v \in L_q$ for some $q < i - \lfloor t \cdot W/2\rfloor$ or $q > j + \lfloor t \cdot W/2\rfloor$. By the definition of the BFS decomposition, $\mathrm{dist}_G(x, v) > \lfloor t \cdot W/2\rfloor$ and $\mathrm{dist}_G(y, v) > \lfloor t \cdot W/2\rfloor$. But then P has length at least $\mathrm{dist}_G(x, v) + \mathrm{dist}_G(v, y) \ge 2\lfloor t \cdot W/2\rfloor + 2 > t \cdot W$. Hence, all vertices of P are vertices of G'. Since S' is a subgraph of S induced by the set of vertices of G'_{ij}, we have that P is a path in S' and $\mathrm{dist}_{S'}(x, y) \le w(P) \le t \cdot w(xy)$. $\square$

This lemma indicates that for apex-minor-free graphs, it is sufficient to consider "local" spanners. In particular, to approximate distances between vertices in levels $L_i, \ldots, L_j$, it is sufficient to know the behavior of the spanner within levels $L_{i-\lfloor t\cdot W/2\rfloor}, \ldots, L_{j+\lfloor t\cdot W/2\rfloor}$ of the BFS decomposition.

Now we choose a positive integer l and put $k = (2\lfloor t \cdot W/2\rfloor + 1)l + 1$.

If $r \le k$ then a t-spanner S of G of minimum weight is constructed directly. We use the fact that $\mathbf{tw}(G) = O(k)$ and, for example, use Bodlaender's algorithm [7] to construct in linear time a suitable tree decomposition of G. Then, by Lemma 1, a t-spanner of minimum weight for G can be found in linear time.

Suppose that $r > k$. We consequently construct t-spanners S_i of G for $i = 1, 2, \ldots, k - 1$ as follows. Let

$$J_i = \left\{ i - 1 + (k-1)s \colon s \in \left\{0, \ldots, \left\lfloor \frac{r-i}{k-1} \right\rfloor \right\}\right\}.$$

Recall that $L_q = \emptyset$ for $q < 0$ or $q > r$. For every $j \in J_i$, we consider graph $G'_j = G_{j-\lfloor t\cdot W/2\rfloor, j+k+\lfloor t\cdot W/2\rfloor-1}$ and set of edges $A_j = E(G_{j,j+k-1})$. In other words, we "cover" graph G by graphs $G'_{i-(k-1)}, G'_i, G'_{i+(k-1)}, \ldots$, and two consecutive graphs "overlap" by $2\lfloor t \cdot W/2\rfloor + 1$ levels in the BFS decomposition (see Fig. 2). Observe that while the parameter l can be chosen, the integers W and t are given constants, and the number of overlapping levels of two consecutive graphs is bounded and could be made relatively small in comparison with k. It gives us a possibility to apply a shifting approach similar to Bakers's. By Lemma 2, $\mathbf{tw}(G'_j) = O(k + t \cdot W)$. For every graph G'_j we construct a partial t-spanner S_{ij} for A_j in G'_j of minimum weight by making use of Lemma 1. The union of all sets A_j is the set $E(G)$, and we define

$$S_i = \bigcup_{j \in J_i} S_{ij},$$

i.e. we "glue" our local spanners together via the overlapping parts. Finally, we choose among graphs $S_1, S_2, \ldots, S_{k-1}$ the graph with the minimum weight and denote it by S.

The following lemma describes properties of the graph S.

Lemma 4. *Let S be the subgraph of an apex-minor-free graph G obtained by the algorithm described above. Then the following holds.*

1. *S is a t-spanner of G.*
2. *For every t, W and $l > 0$, S can be constructed in time $l(Wt)^{O((Wtl)^2)} \cdot |V(G)|$.*

3. *S has weight at most* $(1+\frac{1}{l})\cdot \mathrm{OPT}(G)$*, where* $\mathrm{OPT}(G)$ *is the weight of the solution of the* Minimum Weight t-Spanner *problem on G.*

Proof. 1. Every S_i is a t-spanner of G. Indeed, for every edge $xy \in E(G)$, there is $j \in J_i$ such that $xy \in A_j$, and $\mathrm{dist}_{S_i}(x,y) \le \mathrm{dist}_{S_{ij}}(x,y) \le t \cdot w(xy)$.
2. The second claim follows from Lemmata 1 and 2.
3. If $k \ge r$ then the claim is obvious. Let $k < r$ and let T be a t-spanner of G of minimum weight $w(T) = \mathrm{OPT}(G)$. Assume that $i \in \{1,2,\ldots,k-1\}$ and $j \in J_i$. Let $T_j = T[V(G'_j)]$. By Lemma 3, T_j is a partial t-spanner for the set A_j in G'_j. Since S_{ij} is a partial t-spanner for the set A_j in G'_j with the minimum weight, we have that

$$w(T_j) \ge w(S_{ij}),$$

and

$$w(S_i) \le \sum_{j\in J_i} w(S_{ij}) \le \sum_{j\in J_i} w(T_j).$$

Because $k \ge t\cdot W + 2$, spanners T_j and $T_{j'}$ have common edges only if j and j' are consecutive integers in J_i, and their intersection contains edges of T with endpoints in $L_{j-\lfloor t\cdot W/2\rfloor} \cup\cdots\cup L_{j+\lfloor t\cdot W/2\rfloor}$. Denote by $E_{i,j}$ the set of edges of $T[L_i\cup\cdots\cup L_j]$ for $i \le j$. Then

$$w(S_i) \le w(T) + \sum_{j\in J_i} w(E_{j-\lfloor t\cdot W/2\rfloor,\, j+\lfloor t\cdot W/2\rfloor}).$$

Hence

$$\begin{aligned} w(S) &= \min_{1\le i\le k-1} w(S_i) \\ &\le w(T) + \min_{1\le i\le k-1} \sum_{j\in J_i} w(E_{j-\lfloor t\cdot W/2\rfloor,\, j+\lfloor t\cdot W/2\rfloor}). \end{aligned}$$

Let $E'_{i,j} = E_{i,j} \cup \{xy \in E(T) : x \in L_{i-1}, y \in L_i\}$. Observe that

$$E(T) = E'_{1,r} = \bigcup_{s=0}^{l-1}\left(\bigcup_{j\in J_{(2\lfloor t\cdot W/2\rfloor+1)s+1}} E'_{j-\lfloor t\cdot W/2\rfloor,\, j+\lfloor t\cdot W/2\rfloor}\right),$$

and thus

$$w(T) = \sum_{s=0}^{l-1}\left(\sum_{j\in J_{(2\lfloor t\cdot W/2\rfloor+1)s+1}} w(E'_{j-\lfloor t\cdot W/2\rfloor,\, j+\lfloor t\cdot W/2\rfloor})\right).$$

It follows that there is $s \in \{0,\ldots,l-1\}$ such that

$$\sum_{j\in J_{(2\lfloor t\cdot W/2\rfloor+1)s+1}} w(E'_{j-\lfloor t\cdot W/2\rfloor,\, j+\lfloor t\cdot W/2\rfloor}) \le \frac{1}{l} w(T).$$

Clearly,

$$\begin{aligned} \min_{1\le i\le k-1} \sum_{j\in J_i} w(E_{j-\lfloor t\cdot W/2\rfloor,\, j+\lfloor t\cdot W/2\rfloor}) &\le \sum_{j\in J_{(2\lfloor t\cdot W/2\rfloor+1)s+1}} w(E_{j-\lfloor t\cdot W/2\rfloor,\, j+\lfloor t\cdot W/2\rfloor}) \\ &\le \sum_{j\in J_{(2\lfloor t\cdot W/2\rfloor+1)s+1}} w(E'_{j-\lfloor t\cdot W/2\rfloor,\, j+\lfloor t\cdot W/2\rfloor}) \le \frac{1}{l} w(T). \end{aligned}$$

We conclude that

$$w(S) \le w(T) + \min_{1\le i\le k-1} \sum_{j\in J_i} w(E_{j-\lfloor t\cdot W/2\rfloor,\, j+\lfloor t\cdot W/2\rfloor}) \le \left(1+\frac{1}{l}\right) w(T). \quad \square$$

For every $\epsilon > 0$ in Lemma 4, we can put $l = \lceil \frac{1}{\epsilon} \rceil$, which implies the main result of this paper.

Theorem 1. *For every* $\epsilon > 0$ *and fixed positive integers* t *and* W*, the* Minimum Weight t-Spanner *problem admits a EPTAS with running time* $\frac{1}{\epsilon}\cdot(W\cdot t)^{O((W\cdot t/\epsilon)^2)}\cdot n$ *for the class of n-vertex apex-minor-free graphs (and, hence, for the planar graphs and for the graphs with bounded genus), if weights of edges of the input graph are integers not greater than* W*.*

5. Conclusion

In this work we have shown that the Minimum Weight t-Spanner has EPTAS on planar, and more generally, on apex-minor-free graphs, when the maximum edge weight is bounded by some constant. Our result can be easily extended to solve the variant of the problem proposed by Kortsarz [28]. In his version each edge $e \in E(G)$ has positive length $l(e)$ and nonnegative weight $w(e)$. The aim is to find a t-spanner of minimum weight, and the distance between two vertices x and y is measured by the sum of lengths of edges of a shortest x, y-path. It is possible to construct EPTAS on apex-minor-free graphs when all edges have positive integer lengths and the maximum edge length is at most some constant (weights of edges can be arbitrary). We leave the following two questions open.

- Our technique does not work when the maximum weight W is some polynomial of the number of vertices of the input graph. What is the approximability status of Minimum Weight t-Spanner on planar graphs with polynomial weights?
- Another question concerns the unweighted case of the problem. Can our approximability results be extended to larger classes of graphs, say, graphs excluding some fixed graph as a minor?

Acknowledgements

We would like to thank the anonymous referees for comments which improved the presentation of this paper. The second author is supported by the Norwegian Research Council and the third author is supported by EPSRC under project EP/G043434/1.

References

[1] I. Althöfer, G. Das, D. Dobkin, D. Joseph, J. Soares, On sparse spanners of weighted graphs, Discrete Comput. Geom. 9 (1993) 81–100.
[2] K. Alzoubi, X.-Y. Li, Y. Wang, P.J. Wan, O. Frieder, Geometric spanners for wireless ad hoc networks, IEEE Trans. Parallel Distrib. Syst. 14 (2003) 408–421.
[3] S. Arnborg, J. Lagergren, D. Seese, Easy problems for tree-decomposable graphs, J. Algorithms 12 (1991) 308–340.
[4] B. Awerbuch, Complexity of network synchronization, J. ACM 32 (1985) 804–823.
[5] B. Awerbuch, D. Peleg, Routing with polynomial communication-space trade-off, SIAM J. Discrete Math. 5 (1992) 151–162.
[6] B.S. Baker, Approximation algorithms for NP-complete problems on planar graphs, J. Assoc. Comput. Mach. 41 (1994) 153–180.
[7] H.L. Bodlaender, A linear-time algorithm for finding tree-decompositions of small treewidth, SIAM J. Comput. 25 (1996) 1305–1317.
[8] H.L. Bodlaender, A partial *k*-arboretum of graphs with bounded treewidth, Theoret. Comput. Sci. 209 (1998) 1–45.
[9] H.L. Bodlaender, A. Koster, Combinatorial optimization on graphs of bounded treewidth, Comput. J. 51 (2008) 255–269.
[10] U. Brandes, D. Handke, *NP*-completeness results for minimum planar spanners, Discrete Math. Theoret. Comput. Sci. 3 (1998) 1–10.
[11] L. Cai, Tree spanners: spanning trees that approximate the distances, Ph.D. Thesis, University of Toronto, 1992.
[12] L. Cai, D.G. Corneil, Tree spanners, SIAM J. Discrete Math. 8 (1995) 359–387.
[13] M. Cesati, L. Trevisan, On the efficiency of polynomial time approximation schemes, Inform. Process. Lett. 64 (1997) 165–171.
[14] B. Chandra, G. Das, G. Narasimhan, J. Soares, New sparseness results on graph spanners, in: Proceedings of the Eighth Annual Symposium on Computational Geometry, ACM Press, 1992, pp. 192–201.
[15] L.P. Chew, There are planar graphs almost as good as the complete graph, J. Comput. System Sci. 39 (1989) 205–219.
[16] E.D. Demaine, M.T. Hajiaghayi, The bidimensionality theory and its algorithmic applications, Comput. J. 51 (2008) 292–302.
[17] D.P. Dobkin, S.J. Friedman, K.J. Supowit, Delaunay graphs are almost as good as complete graphs, Discrete Comput. Geom. 5 (1990) 399–407.
[18] F.F. Dragan, F.V. Fomin, P.A. Golovach, Spanners in sparse graphs, J. Comput. System Sci., doi:10.1016/j.jcss.2010.10.002.
[19] F.F. Dragan, F.V. Fomin, P.A. Golovach, A PTAS for the sparsest spanners problem on apex-minor-free graphs, in: Proceedings of the 33rd International Symposium on Foundations of Computer Science, MFCS 2008, in: Lecture Notes in Comput. Sci., vol. 5162, Springer, 2008, pp. 290–298.
[20] W. Duckworth, N.C. Wormald, M. Zito, A PTAS for the sparsest 2-spanner of 4-connected planar triangulations, J. Discrete Algorithms 1 (2003) 67–76.
[21] M. Elkin, Computing almost shortest paths, ACM Trans. Algorithms 1 (2005) 283–323.
[22] M. Elkin, D. Peleg, The hardness of approximating spanner problems, Theory Comput. Syst. 41 (2007) 691–729.
[23] M. Elkin, D. Peleg, Approximating *k*-spanner problems for $k > 2$, Theoret. Comput. Sci. 337 (2005) 249–277.
[24] D. Eppstein, Subgraph isomorphism in planar graphs and related problems, in: SODA, 1995, pp. 632–640.
[25] D. Eppstein, Diameter and treewidth in minor-closed graph families, Algorithmica 27 (2000) 275–291.
[26] M. Grohe, Local tree-width, excluded minors, and approximation algorithms, Combinatorica 23 (2003) 613–632.
[27] T. Kloks, Treewidth, Computations and Approximations, in: Lecture Notes in Computer Science, vol. 842, Springer, 1994.
[28] G. Kortsarz, On the hardness of approximating spanners, Algorithmica 30 (2001) 432–450.
[29] G. Kortsarz, D. Peleg, Generating sparse 2-spanners, J. Algorithms 17 (1994) 222–236.
[30] J.A. Makowsky, U. Rotics, Optimal spanners in partial *k*-trees, manuscript.
[31] G. Narasimhan, M. Smid, Geometric Spanner Networks, Cambridge, 2007.
[32] D. Peleg, A.A. Schäffer, Graph spanners, J. Graph Theory 13 (1989) 99–116.
[33] D. Peleg, E. Upfal, A trade-off between space and efficiency for routing tables, J. ACM 36 (1989) 510–530.
[34] D. Peleg, J.D. Ullman, An optimal synchronizer for the hypercube, SIAM J. Comput. 18 (1989) 740–747.
[35] S. Pettie, Low distortion spanners, in: Proceedings of the 32nd International Colloquium on Automata, Languages and Programming, ICALP 2007, in: Lecture Notes in Computer Science, Springer, 2007, pp. 78–89.
[36] N. Robertson, P.D. Seymour, Graph minors III: planar tree-width, J. Combin. Theory Ser. B 36 (1984) 49–64.
[37] L. Roditty, M. Thorup, U. Zwick, Deterministic constructions of approximate distance oracles and spanners, in: Proceedings of the 32nd International Colloquium on Automata, Languages and Programming, ICALP 2005, in: Lecture Notes in Computer Science, vol. 3580, Springer, 2005, pp. 261–272.
[38] M. Thorup, U. Zwick, Compact routing schemes, in: Proceedings of the 13th Ann. ACM Symp. on Par. Alg. and Arch, SPAA 2001, ACM, 2001, pp. 1–10.
[39] M. Thorup, U. Zwick, Approximate distance oracles, J. ACM 52 (2005) 1–24.