Discrete Applied Mathematics 195 (2015) 104–109

Contents lists available at ScienceDirect

Discrete Applied Mathematics

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

Tree t-spanners in outerplanar graphs via supply demand partition

N.S. Narayanaswamy [a], G. Ramakrishna [a,b,*]

[a] Department of Computer Science and Engineering, Indian Institute of Technology Madras, India
[b] Indian Institute of Information Technology Sri City, India

ARTICLE INFO

Article history:
Received 16 October 2013
Received in revised form 27 October 2014
Accepted 4 November 2014
Available online 18 March 2015

Keywords:
Tree t-spanner
Minimum stretch spanning tree
Supply–demand tree partition
Outerplanar graphs

ABSTRACT

A *tree t-spanner* of an unweighted graph G is a spanning tree T such that for every two vertices their distance in T is at most t times their distance in G. Given an unweighted graph G and a positive integer t as input, the Tree t-spanner problem is to compute a tree t-spanner of G if one exists. This decision problem is known to be NP-complete even in the restricted class of unweighted planar graphs. We present a linear-time reduction from Tree t-spanner in outerplanar graphs to the supply–demand tree partition problem. Based on this reduction, we obtain a linear-time algorithm to solve Tree t-spanner in outerplanar graphs. Consequently, we show that the minimum value of t for which an input outerplanar graph on n vertices has a tree t-spanner can be found in $O(n \log n)$ time.

1. Introduction

The area of finding sparse data structures to maintain approximate distance information in a graph is a deep one with far-reaching importance in both engineering (computer networks) and mathematics (metric embeddings). The sparsest possible data structures that give relevant distance information are spanning trees, and it is natural to approximate distance information within a desired factor, denoted by the parameter t in this paper. A *tree t-spanner* of an unweighted graph G is a spanning tree T such that for every two vertices their distance in T is at most t times their distance in G. Here the parameter t is called *stretch*. Furthermore, a *minimum stretch spanning tree* of G is a tree t-spanner of G for the smallest possible value of t. Finding a *minimum stretch spanning tree* is a classical optimization problem in algorithmic graph theory that has several applications in networks, distributed systems and biology [2,4,20]. The Tree t-spanner problem is to decide the existence of a tree t-spanner in a graph and find a tree t-spanner if one exists. The results in the literature on the Tree t-spanner problem fall into two classes: when t is a fixed number, and when t is given as part of the input. In this paper, we present results on the case when t is part of the input, and on finding a minimum stretch spanning tree. Furthermore, our focus is on outerplanar graphs. Outerplanar graphs are well-studied due to their rich combinatorial and topological properties. We use the fact that 2-connected outerplanar graphs have a unique outerplane embedding and that their weak dual is a tree. For problems like vertex cover on planar graphs, polynomial time approximation schemes, based on Baker's technique [3], are obtained via algorithms on k-outerplanar graphs (defined in Section 2). These algorithms have a running time that is exponential in k, but polynomial in the graph size. 1-outerplanar graphs are essentially outerplanar graphs, and we believe that our results are a first step towards such algorithms for Tree t-spanner on k-outerplanar graphs. We hope that this will eventually lead to $o(\log n)$ approximation algorithms for Tree t-spanner on planar graphs, when t is part of the input. Outerplanar graphs

* Corresponding author at: Department of Computer Science and Engineering, Indian Institute of Technology Madras, India.
E-mail addresses: swamy@cse.iitm.ac.in (N.S. Narayanaswamy), rama.vikram@gmail.com (G. Ramakrishna).

http://dx.doi.org/10.1016/j.dam.2014.11.001

also appear in several applications including computational drug design, bioinformatics, and telecommunications [13]. For instance, 94.3% of elements in the popular NCI data set are outerplanar [13].

Related work. For each fixed $t \geq 4$, Cai and Corneil [9] showed that Tree t-spanner is NP-complete. Tree 2-spanner is polynomial-time solvable, and the status of Tree 3-spanner is open. The problem has been studied for many graph classes. For planar graphs, Tree 3-spanner was shown to be polynomial time solvable by Fekete and Kremer [12]. For each fixed t, Fomin et al. showed that Tree t-spanner restricted to planar graphs and apex-minor free graphs can be solved in polynomial time. They showed this by designing an FPT algorithm with t as the parameter in planar graphs and apex-minor free graphs [11]. Also, for any fixed $t \geq 4$, the Tree t-spanner is NP-complete even on chordal graphs and chordal bipartite graphs [7,8]. Interval graphs, permutation graphs, regular bipartite graphs and distance hereditary graphs admit a tree 3-spanner [16,17] that can be found in polynomial time. As a consequence, a minimum stretch spanning tree in such graphs can be found in polynomial time due to the fact that Tree 2-spanner is polynomial-time solvable.

While the results in the previous paragraph are for the case when t is fixed, our results are for the case when t is part of the input. When t is part of the input, Fekete and Kremer [12] showed that Tree t-spanner in planar graphs is NP-complete. A minimum stretch spanning tree in grid graphs can be found by a polynomial-time algorithm [6]. Therefore, when t is part of the input, on grid graphs the Tree t-spanner problem can be solved in polynomial time. To the best of our knowledge, when t is part of the input, there is no published result on Tree t-spanner in any subclass of planar graphs, except for grid graphs.

Our results. Our main result is a linear time algorithm for Tree t-spanner when t is part of the input in outerplanar graphs. For any t, the property that a graph has a tree t-spanner can be described as a formula in Monadic Second Order Logic (MSOL) [11]. It must be mentioned here that since t is part of the input, the treewidth of the graph being two *does not* result in a linear time algorithm by a direct application of the well known Courcelle's theorem [10]. The reason is that the length of the MSOL formula depends on the parameter t, which is not *fixed* in the Tree t-spanner problem that we consider. In other words, the size of the formula is *not* a constant. Actually, it is believed that the length of the smallest MSOL formula would have an exponential tower dependency on t that is of the form, $2^{2^{2^{2^t}}}$, since the MSOL formulation has 4 alternating quantifiers. See [15] on the length of the MSOL formula as a function of the number of alternating quantifiers. Another important property of the treewidth being two is that an optimal tree decomposition can be obtained in linear time. It is not clear how the standard approaches for solving NP-hard problems on graphs of bounded treewidth can be made to work for Tree t-spanner on outerplanar graphs when t is part of the input. However, we exploit the topological structure of outerplanar graphs to obtain a linear time algorithm.

We first introduce the basic notation of outerplanar graphs and preliminary results on the structure of outerplanar graphs in Section 2. We then present a linear-time reduction from Tree t-spanner in outerplanar graphs to the Tree S-partition problem (defined below) in Section 3.1. Later, we describe a linear-time reduction from Tree S-partition to the well-studied Supply–demand tree partition problem [14] in Section 3.2. Finally, we use the linear-time algorithm in [14] for solving Supply–demand tree partition, which produces a supply–demand tree partition, if one exists. We now state our main results formally.

Theorem 1. *Given an outerplanar graph G on n vertices and a positive integer t, a tree t-spanner of G, if one exists, can be found in $O(n)$ time.*

From Theorem 1, by using binary search on the value of t, we have the following corollary.

Corollary 2. *Given an outerplanar graph G on n vertices, a minimum stretch spanning tree of G can be found in $O(n \log n)$ time.*

Tree partition problems: To the best of our knowledge, there is no literature on Tree S-partition which we have defined as follows.

Tree S-partition
Instance: A tree T, a weight function $w : V(T) \to \mathbb{N}$, a set $S \subseteq V(T)$ of special vertices, and $t \in \mathbb{N}$.
Question: Does there exist a partition of $V(T)$ into sets $V_1, \ldots, V_{|S|}$ such that
for $1 \leq i \leq |S|$,
$T[V_i]$ (the graph induced on V_i) is connected, $|V_i \cap S| = 1$ and $\sum_{v \in V_i} w(v) \leq t$?

A partition $\{V_1, \ldots, V_{|S|}\}$ of $V(T)$ is a *tree S-partition* of T, if for each $1 \leq i \leq |S|$, $T[V_i]$ is a tree and V_i has exactly one vertex from S. The Supply–demand tree partition problem is defined as follows.

Supply–demand tree partition [14]
Instance: A tree T such that $V(T) = V_s \uplus V_d$, a supply function $s : V_s \to \mathbb{Z}$, and a demand function $d : V_d \to \mathbb{N}$.
Question: Does there exist a partition of $V(T)$ into sets $V_1, \ldots, V_k$, where $k = |V_s|$, such that
for $1 \leq i \leq k$, $T[V_i]$ is connected, V_i contains exactly one vertex $u \in V_s$, $\sum_{v \in V_i \setminus V_s} d(v) \leq s(u)$, and $s(u) > 0$?

In Supply–demand tree partition, each element u in V_s is referred to as supply vertex and $s(u)$ denotes the supply value of u. Similarly, each element u in V_d is referred to as demand vertex and $d(u)$ denotes the demand value of u. A partition $\{V_1, \dots, V_k\}$ of $V(T)$ is a *supply–demand tree partition* of T, if for each $1 \le i \le k$, V_i satisfies all the constraints described above.

2. Preliminaries

Graph theoretic preliminaries: In this paper, we consider only simple, finite, connected and undirected graphs. We refer to [21] for standard graph theoretic terminology. Let $G = (V(G), E(G))$ be a graph on vertex set $V(G)$ and edge set $E(G)$. For a set $S \subseteq V(G)$, $G[S]$ denotes the graph induced on the set S and $G - S$ denotes $G[V(G) - S]$. A set $\{V_1, \dots, V_r\}$ of pairwise disjoint non-empty subsets of a set V is a *partition* if $\bigcup_{i=1}^{r} V_i = V$; each such V_i is referred to as a *part* of the partition. The number of edges in a shortest path between the vertices u and v in G is called the *distance* between u and v.

Planar graph preliminaries: A graph is said to be *planar* if it can be drawn in the plane without crossing edges; otherwise it is *nonplanar*. A *plane graph* is a planar graph with a fixed planar embedding. A planar graph is said to be *outerplanar* if it can be embedded on the plane such that all vertices lie on the boundary of its exterior region. Such an embedding is called an *outerplanar embedding*. Let G be a plane graph. The regions defined by the planar embedding of G are *faces* of G and the set of faces of G is denoted by Faces(G). For each face $f \in$ Faces(G), $V(f)$ and $E(f)$ denote the sets of vertices and edges of f, respectively. Two faces $f, f' \in$ Faces(G) are *adjacent* if $E(f) \cap E(f') \neq \emptyset$. In every planar embedding, there is a unique unbounded face called *exterior* face and it is denoted by f_{ext}. An edge e is *external* if $e \in E(f_{ext})$, otherwise it is *internal*. All the bounded faces of G are *interior*. Interior faces that are adjacent to the exterior face are called E-faces and other interior faces are called I-faces. The set of interior faces, I-faces and E-faces of G are denoted by Int-faces(G), I-faces(G), and E-faces(G), respectively. So, Faces(G) = $\{f_{ext}\} \cup$ Int-faces(G) and Int-faces(G) = E-faces(G) $\cup$ I-faces(G). The *dual graph* of G is $G^* := (V(G^*), E(G^*))$, where $V(G^*) := \{v_f \mid f \in \text{Faces}(G)\}$, $E(G^*) := \{\tilde{e} = (v_f, v_g) \mid f, g \in \text{Faces}(G), e \in E(f) \cap E(g)\}$. Here, there is a bijective correspondence between $E(G)$ and $E(G^*)$. An outerplanar graph is also known as 1-*outerplanar graph*. A planar graph G is *k-outerplanar* if the graph obtained by removing the vertices lie on the exterior boundary of G is $(k-1)$-outerplanar, where $k \ge 2$.

Lemma 3 (*Theorem 1.1 in [9]*). *A spanning tree T of an arbitrary graph G is a tree t-spanner if and only if for each edge $(x, y) \in E(G)$, the distance between x and y in T is at most t.*

Let T be a spanning tree of a graph G. For any edge $e \in E(G)$ with end vertices u and v, the *stretch* of e in T is defined as the distance between u and v in T. By Lemma 3, the stretch of T is max{stretch of e in $T \mid e \in E(G)$}. For vertices $u, v \in V(T)$, $P_T(u, v)$ denotes the path between u and v in T. An edge $e \in E(G)$ is a *non-tree* edge with respect to T if $e \notin E(T)$. For a non-tree edge (u, v) with respect to T, the cycle formed by (u, v) and $P_T(u, v)$ is referred to as a *fundamental* cycle. A fundamental cycle is said to be *external* if the associated non-tree edge is external in G.

If an outerplanar graph G has cut vertices, a minimum stretch spanning tree of G can be obtained by performing a union on the edges of minimum stretch spanning trees of the maximal 2-vertex-connected components of G. So without loss of generality, we consider 2-vertex-connected outerplanar graphs. Every 2-connected outerplanar graph has a unique outerplane embedding [19]. A planar embedding of a planar graph can be obtained in linear time [18]. Therefore, we assume that an outerplanar graph is associated with the fixed outerplanar embedding. The dual graph of an outerplanar graph can be obtained in linear time [5]. Also the faces of an outerplanar graph can be easily extracted in linear time. The following result on outerplanar graphs will be used in the next section.

Lemma 4 (*[19]*). *Every 2-connected outerplanar graph has a Hamiltonian cycle.*

For a 2-connected outerplanar graph G on n vertices, $|V(f_{ext}(G))| = |E(f_{ext}(G))| = n$. Thus $f_{ext}(G)$ is a Hamiltonian cycle. In other words, the set of external edges in a 2-connected outerplanar graph is a Hamiltonian cycle.

3. Tree t-spanner to supply–demand tree partition via tree S-partition

In the rest of the paper, G denotes a 2-vertex-connected outerplanar graph. We refer to the internal and external edges with respect to the graph G. The same holds for E-faces, I-faces and Int-faces.

3.1. Tree t-spanner to tree S-partition

In this section, we present a linear-time reduction from Tree t-spanner in outerplanar graphs to Tree S-partition.

Reduction. For a given instance $\langle G, t\rangle$ of Tree t-spanner, an instance $\langle \tilde{T}, w, S, t-1\rangle$ of Tree S-partition is constructed as follows: $V(\tilde{T}) = \{v_f \mid f \in \text{Int-faces}(G)\}$, $E(\tilde{T}) = \{(v_f, v_g) \mid f, g \in \text{Int-faces}(G), |E(f) \cap E(g)| = 1\}$. The weight function $w : V(\tilde{T}) \to \mathbb{N}$ is defined as $w(v_f) = |E(f)| - 2$, for each $v_f \in V(\tilde{T})$. The set of special vertices S is $\{v_f \mid f \in \text{E-faces}(G)\}$. The graph $\tilde{T}$ is the *weak dual* of G. The weak dual of G can be obtained in linear time from G^* (dual of G) by removing the vertex that corresponds to the unbounded face in G.

We use the following lemmas to prove formally, in Theorem 10, that the described reduction is correct.

Lemma 5 (*Lemma 2.4 in [1]*). *An outerplanar graph G is 2-connected if and only if the weak dual of G is a tree.*

Lemma 6. *Let $\tilde{T}$ be the weak dual of G and C be a cycle in G. Let $X = V(C)$, and $\tilde{X} = \{v_f \mid f \in \text{Int-faces}(G[X])\}$. The outerplanar graph $G[X]$ is 2-connected if and only if $\tilde{T}[\tilde{X}]$ is a tree.*

Proof. We first prove the claim that $\tilde{T}[\tilde{X}]$ is the weak dual of $G[X]$. Clearly, the vertex set $\tilde{X}$ is same as the vertex set of the weak dual of $G[X]$. For any two interior faces f and g in $G[X]$, f and g are adjacent in G if and only if they are adjacent in $G[X]$. Hence, (v_f, v_g) is an edge in $\tilde{T}[\tilde{X}]$ if and only if it is an edge in the weak dual of $G[X]$. Hence the claim holds. This claim together with Lemma 5 implies that, $G[X]$ is 2-connected if and only if $\tilde{T}[\tilde{X}]$ is a tree. □

Lemma 7. *Let H be an induced subgraph of G. Then* $\text{Int-faces}(H) \subseteq \text{Int-faces}(G)$.

Proof. For an outerplanar graph G', we first prove the claim that every induced cycle in G' is an interior face. Consider an induced cycle C in the outerplanar embedding of G'. There are no vertices in the interior region of C, because G' is outerplanar. As C is an induced cycle, no edge in G' joins two non-adjacent vertices in C. Thereby no edge lies in the interior region of C. Hence C is an interior face.

For an outerplanar graph G', we now prove the claim that every interior face in the outerplanar embedding of G' is an induced cycle. Let $f \in \text{Int-faces}(G')$. Let $u, v \in V(f)$ be two non-adjacent vertices in f. As f is a face, (u, v) does not lie in the interior region of f. Also, (u, v) does not lie in the exterior region of f, because all the vertices of f lie on the exterior boundary of G'. Thus, no edge in G' joins any two non-adjacent vertices in f. Therefore f is an induced cycle.

Since H is an induced subgraph, every induced cycle in H is an induced cycle in G. This observation along with the above two claims follows that every interior face in H is an interior face in G. □

For a subset $\tilde{X} \subseteq V(\tilde{T})$, we define $\text{cost}(\tilde{X}) = \sum_{v\in\tilde{X}} w(v)$.

Lemma 8. *Let $\tilde{T}$ be the weak dual of G and C be a cycle in G. Let $\tilde{X}$ be the set of vertices in $\tilde{T}$ that corresponds to the interior faces in $G[V(C)]$. Then* $\text{cost}(\tilde{X}) = |E(C)| - 2$.

Proof. Let H be the induced graph of G on $V(C)$ and $\{f_1, \dots, f_r\}$ be the set of interior faces in H. Since H is a 2-connected outerplanar graph consisting of r interior faces, the number of internal edges in H is $r-1$. Also $E(C)$ equals the set of external edges in H. In the sum $\sum_{i=1}^{r} |E(f_i)|$, each internal edge of H is counted twice and each external edge of H is counted once. Thus we have $|E(C)| = (\sum_{i=1}^{r} |E(f_i)|) - 2(r-1)$. Further, we have defined that for each vertex $v_f \in \tilde{X}$, $w(v_f) = |E(f)| - 2$. Consequently, $(\sum_{i=1}^{r} |E(f_i)|) - 2(r-1) = (\sum_{i=1}^{r} |E(f_i)| - 2) + 2 = (\sum_{v\in\tilde{X}} w(v)) + 2 = \text{cost}(\tilde{X}) + 2$. Therefore $|E(C)| = \text{cost}(\tilde{X}) + 2$. □

Observation 9. *Let T be a spanning tree of G. Let $\{C_1, \dots, C_k\}$ be the set of external fundamental cycles with respect to T. For each $1 \le i \le k$, let $G_i = G[V(C_i)]$. Then*

a. $\text{Int-faces}(G) = \text{Int-faces}(G_1) \uplus \cdots \uplus \text{Int-faces}(G_k)$.
b. *The set of non-tree edges in G with respect to T is a disjoint union of the set of non-tree edges in $G_1, \dots, G_k$ with respect to T.*

Proof. Let r be the number of edges in T that are internal edges in G. The proof is by induction on r. Suppose $r = 0$, then all the edges in T are external edges in G. Then T has exactly one external fundamental cycle, which is also an exterior face of G. Consequently, the two claims in this lemma hold. Consider the case when $r \ge 1$. Then there exists an edge $(u, v) \in E(T)$, such that (u, v) is internal in G. We decompose G into graphs H_1 and H_2 such that $V(G) = V(H_1) \cup V(H_2)$, $E(G) = E(H_1) \cup E(H_2)$, $V(H_1) \cap V(H_2) = \{u, v\}$ and $E(H_1) \cap E(H_2) = \{(u, v)\}$. This decomposition is possible as the end vertices of any internal edge in an outerplanar graph disconnect the graph exactly into two components. Observe that *Int-faces*$(G) =$ *Int-faces*$(H_1) \uplus$ *Int-faces*(H_2).

For $i \in \{1, 2\}$, let $T_i = T[V(H_i)]$. Let $i \in \{1, 2\}$ and let $x, y \in V(T_i)$ such that $\{x, y\} \neq \{u, v\}$, we now show that there is a path between x and y in T_i. Let $P(x, y)$ be the path between x and y in T. If all the vertices in $P(x, y)$ are in T_i, then we are done. Consider the case that there is a vertex z in $P(x, y)$, such that z is in T_j but not in T_i, where $i \neq j$. Since the vertices $x, y \in V(T_i)$ and $z \in V(T_j)$, and we know that $\{u, v\}$ is a minimal vertex separator of G, it follows that there is a subpath $P(u, v) \subseteq P(x, y)$ in T, such that $P(u, v)$ has the vertex z. Therefore, $P(u, v)$ has at least two edges in T. As (u, v) is in T, $P(u, v) + (u, v)$ forms a cycle in T. This contradicts that T is an acyclic graph. As a result, for $i \in \{1, 2\}$, T_i is a spanning tree of G_i. Since $E(H_1) \cap E(H_2)$ is (u, v), the set of non-tree edges with respect to T is a disjoint union of the set of non-tree edges with respect to T_1 in H_1 and the set of non-tree edges with respect to T_2 in H_2. It implies that the set of external fundamental cycles with respect to T is a disjoint union of the set of external fundamental cycles with respect to T_1 and the set of external fundamental cycles with respect to T_2. Without loss of generality, assume that $C_1, \dots, C_l$ be the external fundamental cycles with respect to T_1 and $C_{l+1}, \dots, C_k$ be the external fundamental cycles with respect to T_2.

By the induction hypothesis, *Int-faces*$(H_1) =$ *Int-faces*$(G_1) \uplus \cdots \uplus$ *Int-faces*(G_l) and *Int-faces*$(H_2) =$ *Int-faces*$(G_{l+1}) \uplus \cdots \uplus$ *Int-faces*(G_k). As a result, *Int-faces*$(G) =$ *Int-faces*$(G_1) \uplus \cdots \uplus$ *Int-faces*(G_k). Also by the induction hypothesis, the set

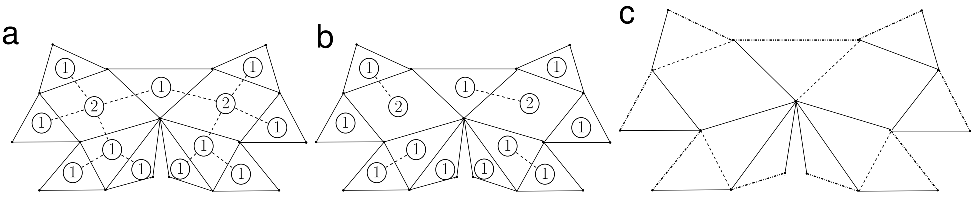

Fig. 1. (a) An outerplanar graph G is in solid edges and the weak dual $\tilde{T}$ of G is in dashed edges. (b) A tree S-partition $\pi = \{\tilde{T}_1, \ldots, \tilde{T}_9\}$ of $\tilde{T}$ with cost at most 3. (c) From π, a tree 4-spanner of G is obtained by deleting the dashed edges (dual edges that correspond to the edges in $\tilde{T}_1, \ldots, \tilde{T}_9$) and dash dotted edges (exactly one external edge from each E-face).

of non-tree edges with respect to T_1 in H_1 is a disjoint union of the set of non-tree edges with respect to T_1 in $G_1, \ldots, G_l$ and the set of non-tree edges with respect to T_2 in H_2 is a disjoint union of the set of non-tree edges with respect to T_2 in $G_{l+1}, \ldots, G_k$. Consequently, the set of non-tree edges with respect to T in G is a disjoint union of the set of non-tree edges with respect to T in $G_1, \ldots, G_k$. □

For a partition $\pi = \{\tilde{V}_1, \ldots, \tilde{V}_k\}$ of $V(\tilde{T})$, cost(π) is defined as $\max\{\text{cost}(\tilde{V}_i) \mid \tilde{V}_i \in \pi\}$.

Theorem 10. *G admits a tree t-spanner if and only if $\tilde{T}$ has a tree S-partition of cost at most $t-1$. Furthermore, from a tree S-partition π of $\tilde{T}$, such that cost(π) $\leq t-1$, a tree t-spanner of G can be obtained in linear time.*

Proof. Necessity. Let T be a tree t-spanner of G. We first obtain a partition of $V(\tilde{T})$, such that the induced graph of $\tilde{T}$ on each part in the partition is connected and its cost is at most $t-1$. For each external non-tree edge $e = (u, v)$ of G with respect to T, let $C_e = P_T(u, v) + (u, v)$ be the external fundamental cycle with respect to T, let $G_e = G[V(C_e)]$, and let $\tilde{V}_e$ be the set of vertices in $\tilde{T}$ that corresponds to Int-faces(G_e). Let $\pi = \{\tilde{V}_e \mid e$ is an external non-tree edge in G with respect to $T\}$. For each part $\tilde{V}_e \in \pi$, we have the following observations.

As the distance between the end vertices of e in T is at most t, $|E(C_e)| \leq t+1$. From Lemma 8, cost($\tilde{V}_e$) $= |E(C_e)| - 2$. Thus, cost($\tilde{V}_e$) $\leq t-1$. Since $G[V(C_e)]$ is 2-connected, by Lemma 6, $\tilde{T}[\tilde{V}_e]$ is connected. As C_e is an external fundamental cycle, G_e contains at least one E-face of G and thus $\tilde{V}_e$ contains at least one S-vertex. From Observation 9.a, π is a partition of $V(\tilde{T})$.

Each part $\tilde{V}_e \in \pi$ consisting of $k \geq 2$ S-vertices, can be further partitioned into k parts in such a way that the induced graph of $\tilde{T}$ on each part is connected and contains exactly one S-vertex. Hence $\tilde{T}$ has a tree S-partition of cost at most $t-1$.

Sufficiency. Let $\pi = \{\tilde{V}_1, \ldots, \tilde{V}_q\}$ be a tree S-partition of $\tilde{T}$ such that cost(π) $\leq t-1$, where $|S| = q$. As there is a bijective mapping from Int-faces(G) to $V(\tilde{T})$, by Lemmas 5 and 6, there is a bijective mapping from the set of 2-connected subgraphs of G to the set of subtrees of $\tilde{T}$. From π, we construct a tree t-spanner of G. For each $1 \leq i \leq q$, let G_i be a 2-connected outerplanar graph, where $V(G_i) = \bigcup_{v_f \in \tilde{V}_i} V(f)$ and $E(G_i) = \bigcup_{v_f \in \tilde{V}_i} E(f)$. As G_i is a subgraph of G, clearly G_i is outerplanar. Further by Lemma 6, G_i is 2-connected, because $\tilde{T}[\tilde{V}_i]$ is a tree. Thus G_i is a 2-connected outerplanar graph. From Lemma 4, G_i has a Hamiltonian cycle, say C_i. Let X_i be the set of internal edges in G_i. Because the set of external edges in a 2-connected outerplanar graph forms a Hamiltonian cycle, $E(G_i)$ is a disjoint union of $E(C_i)$ and X_i. Since $\tilde{V}_1, \ldots, \tilde{V}_q$ is a partition of $V(\tilde{T})$, observe that Int-faces(G_1), …, Int-faces(G_q) is a partition of Int-faces(G). We obtain the 2-connected outerplanar graph G' by removing the edges $X_1 \cup \cdots \cup X_q$ from G. Since G' does not contain internal edges from any G_i, we have I-faces(G') $= \emptyset$ and E-faces(G') $= \{C_1, \ldots, C_q\}$. Let $e_i \in E(C_i)$ be an external edge in G'. We remove the set $\{e_1, \ldots, e_q\}$ of edges from G' and obtain T. Since we have deleted exactly one external edge from each E-face of G' and there are no I-faces in G', T is a spanning tree. It remains to show that T is a tree t-spanner of G. For each $1 \leq i \leq q$, cost($\tilde{V}_i$) $\leq t-1$, because $\tilde{V}_i \in \pi$; from Lemma 8, $|E(C_i)| = \text{cost}(\tilde{V}_i) + 2$, and thus $|E(C_i)| \leq t+1$. It follows that for each $1 \leq i \leq q$, stretch of e_i in T is at most t. We can observe that $e_1, \ldots, e_q$ are the only external non-tree edges in G with respect to T. Due to Observation 9.b, for each internal non-tree edge (u, v) in G with respect to T, there is an external fundamental cycle C_i, such that (u, v) is an internal edge in $G[V(C_i)]$; thereby stretch of (u, v) in T is at most $t-1$. Therefore, by Lemma 3, T is a tree t-spanner of G.

In order to obtain the tree t-spanner T from G, we have to delete the edges $X_1 \cup \cdots \cup X_q$ and exactly one external edge from all the E-faces of G. For each $1 \leq i \leq q$, $e \in X_i$ if and only if $\tilde{e} \in E(\tilde{T}[\tilde{V}_i])$ and $\tilde{T}[\tilde{V}_1], \ldots, \tilde{T}[\tilde{V}_q]$ can be constructed in linear time. Therefore a tree t-spanner of G can be constructed in linear time from π. This process is illustrated in Fig. 1. □

3.2. Tree S-partition to supply–demand tree partition

In this section, we present a linear time reduction from Tree S-partition to Supply–demand tree partition. Supply–demand tree partition can be solved in linear time [14], and the composition of all these linear time procedures results in a linear time algorithm for Tree t-spanner in outerplanar graphs.

Let $\langle T, S \subseteq V(T), w, t\rangle$ be an input instance of Tree S-partition. We now describe the construction of an instance $\langle T', s : V_s(T') \to \mathbb{R}, d : V_d(T') \to \mathbb{R}\rangle$ of Supply–demand tree partition from $\langle T, S \subseteq V(T), w, t\rangle$. Let $V_s(T') = \{u' \mid u$

$\in S\}$, $V_d(T') = \{u' \mid u \in V(T) \setminus S\}$, $E(T') = \{(u', v') \mid (u, v) \in E(T)\}$. Further, for each $u' \in V_s(T')$, assign $s(u') = t - w(u)$ and for each $u' \in V_d(T')$, assign $d(u') = w(u)$.

Theorem 11. *$\langle T, S \subseteq V(T), w, t\rangle$ is a YES instance of* Tree S-partition *if and only if $\langle T', s : V_s(T') \to \mathbb{R}, d : V_d(T') \to \mathbb{R}\rangle$ is a YES instance of* Supply–demand tree partition.

Proof. Let $\pi = \{V_1, \ldots, V_{|S|}\}$ be a set of subsets of $V(T)$. Let $\pi' = \{V'_1, \ldots, V'_{|S|}\}$, where for each i, $V'_i = \{u' \in V(T') \mid u \in V_i\}$. Then π is a partition of $V(T)$ if and only if π' is a partition of $V(T')$. Also, V_i contains a special vertex v_i if and only if V'_i contains a supply vertex v'_i. Furthermore, $T[V_i]$ is a tree if and only if $T[V'_i]$ is a tree. The total weight of all the vertices in V_i, which includes the special vertex v_i, is at most t if and only if the total demand of V'_i is at most $s(v'_i)$, because $s(v'_i) = t - w(v_i)$. Therefore, π is a tree S-partition of T if and only if π' is a supply–demand partition of T'. □

Acknowledgements

We would like to thank the anonymous reviewers for their feedback, which helped us to significantly shorten and improve the presentation of our results.

References

[1] G. Agnarsson, M.M. Halldórsson, On colorings of squares of outerplanar graphs, in: Proc. 15th ACM–SIAM Symposium on Discrete Algorithms, 2004, pp. 242–253.
[2] B. Awerbuch, A. Baratz, D. Peleg, Efficient broadcast and light-weight spanners. Manuscript, 1992.
[3] Brenda S. Baker, Approximation algorithms for NP-complete problems on planar graphs, J. ACM 41 (1) (1994) 153–180.
[4] H.J. Bandelt, Reconstructing the shape of a tree from observed dissimilarity data, Adv. in Appl. Math. 7 (1986) 309–343.
[5] Hans L. Bodlaender, Fedor V. Fomin, Approximation of pathwidth of outerplanar graphs, J. Algorithms 43 (2002) 190–200.
[6] P. Boksberger, F. Kuhn, R. Wattenhofer, On the approximation of the minimum maximum stretch tree problem, Tech. Report 409, ETH Zürich, Switzerland, 2003.
[7] A. Brandstädt, F.F. Dragan, H. Le, V.B. Le, Tree spanners on chordal graphs: complexity and algorithms, Theoret. Comput. Sci. 310 (1–3) (2004) 329–354.
[8] A. Brandstädt, F.F. Dragan, H. Le, V.B. Le, R. Uehara, Tree spanners for bipartite graphs and probe interval graphs, Algorithmica 47 (2007) 27–51.
[9] L. Cai, D. Corneil, Tree spanners, SIAM J. Discrete Math. 8 (3) (1995) 359–387.
[10] B. Courcelle, The monadic second-order logic of graphs I: recognizable sets of finite graphs, Inform. and Comput. 85 (1) (1990) 12–75.
[11] F.F. Dragan, F.V. Fomin, P.A. Golovach, Spanners in sparse graphs, J. Comput. System Sci. 77 (2011) 1108–1119.
[12] S.P. Fekete, J. Kremer, Tree spanners in planar graphs, Discrete Appl. Math. 108 (1) (2001) 85–103.
[13] Tamás Horváth, Jan Ramon, Stefan Wrobel, Frequent subgraph mining in outerplanar graphs, Data Min. Knowl. Discov. 21 (3) (2010) 472–508.
[14] T. Ito, X. Zhou, T. Nishizeki, Partitioning trees of supply and demand, in: ISAAC, 2002, pp. 612–623.
[15] Joachim Kneis, Alexander Langer, A practical approach to Courcelle's theorem, Electron. Notes Theor. Comput. Sci. 251 (2009) 65–81.
[16] D. Kratsch, H. Le, H. Müller, E. Prisner, D. Wagner, Additive tree spanners, SIAM J. Discrete Math. 17 (2004) 332–340.
[17] M.S. Madanlal, G. Venkatesan, C. Pandu Rangan, Tree 3-spanners on interval, permutation and regular bipartite graphs, Inform. Process. Lett. 59 (1996) 97–102.
[18] Kurt Mehlhorn, Petra Mutzel, On the embedding phase of the Hopcroft and Tarjan planarity testing algorithm, Algorithmica 16 (1994) 233–242.
[19] Maciej M. Sysło, Characterizations of outerplanar graphs, Discrete Math. 26 (1) (1979) 47–53.
[20] D. Peleg, Distributed Computing: A Locality-Sensitive Approach, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2000.
[21] D.B. West, Introduction to Graph Theory, second ed., Prentice Hall, 2001.