k-Outerplanar Graphs, Planar Duality, and Low Stretch Spanning Trees
(Extended Abstract)

Yuval Emek*

School of Electrical Engineering, Tel Aviv University, Tel Aviv 69978, Israel
yuvale@eng.tau.ac.il

Abstract. Low distortion probabilistic embedding of graphs into approximating trees is an extensively studied topic. Of particular interest is the case where the approximating trees are required to be (subgraph) spanning trees of the given graph (or multigraph), in which case, the focus is usually on the equivalent problem of finding a (single) tree with low average stretch. Among the classes of graphs that received special attention in this context are k-outerplanar graphs (for a fixed k): Chekuri, Gupta, Newman, Rabinovich, and Sinclair show that every k-outerplanar graph can be probabilistically embedded into approximating trees with constant distortion regardless of the size of the graph. The approximating trees in the technique of Chekuri et al. are not necessarily spanning trees, though.

In this paper it is shown that every k-outerplanar multigraph admits a spanning tree with constant average stretch. This immediately translates to a constant bound on the distortion of probabilistically embedding k-outerplanar graphs into their spanning trees. Moreover, a randomized algorithm is presented for constructing such a low average stretch spanning tree in expected linear time. This algorithm relies on some new insights regarding the connection between low average stretch spanning trees and planar duality.

1 Introduction

The Problem. Consider an n-vertex connected graph $G = (V(G), E(G))$ and let $\ell(e)$ be a positive *length* associated with every edge $e \in E(G)$. For any two vertices $u, v \in V(G)$, let $\delta_G(u, v)$ denote the *distance* between u and v in G, namely, the length, taken with respect to ℓ, of a shortest path connecting u and v in G. Given a spanning tree T of G and some edge $e \in E(G)$, the *stretch* of e in T is defined as $\mathrm{str}_T(e) = \delta_T(e)/\ell(e)$. Spanning trees with low stretch for all edges can be very useful in many applications. However, there exist some trivial graphs for which every spanning tree admits an edge with stretch $\Omega(n)$ (e.g., the n-cycle). This motivates the construction of spanning trees with low *average stretch*, denoted by $\text{av-str}_G(T) = \frac{1}{|E(G)|} \sum_{e \in E(G)} \mathrm{str}_T(e)$ [2].

* Supported in part by the Israel Science Foundation, grants 221/07 and 664/05.

A. Fiat and P. Sanders (Eds.): ESA 2009, LNCS 5757, pp. 203–214, 2009.

The following related notion was introduced in [4]. Given a probability distribution $\mathcal{D}$ over a set $\mathcal{T}$ of spanning trees of G, we say that G is *probabilistically embedded* into $\mathcal{T}$ (under $\mathcal{D}$) with *distortion* α if $\mathbb{E}[\delta_T(u,v)] \leq \alpha \cdot \delta_G(u,v)$ for every two vertices $u, v \in V(G)$, where the expectation is with respect to $T \in_{\mathcal{D}} \mathcal{T}$. It is shown in [2] that a graph G can be probabilistically embedded into its spanning trees with distortion α if and only if every multigraph obtained from G by replicating its edges has a spanning tree with average stretch α. Consequently, in the context of constructing low average stretch spanning trees, one usually considers multigraphs rather than simple graphs. (This can be viewed as taking a weighted average of the edge stretch factors.)

A tree T is called a *dominating tree* of the graph G if $V(T) \supseteq V(G)$ and $\delta_T(u,v) \geq \delta_G(u,v)$ for every two vertices $u, v \in V(G)$. Clearly, every spanning tree of G is also a dominating tree of G; the converse is not true as a dominating tree may have vertices and edges that do not exist in the original graph G, and hence it is not necessarily a subgraph of G. The notion of probabilistic embedding can be redefined by allowing the support $\mathcal{T}$ to contain dominating trees that are not subgraphs of G. For many applications and in particular, for those applications mentioned in [4,5], this does not exhibit any obstacle. However, there exist some applications for which it is impossible to use non-subgraph dominating trees in the support of the probabilistic embedding, most notably in the context of networking, where G represents an existing physical graph (e.g., the *minimum communication spanning tree* problem [13]).

k-Outerplanar Graphs. An *outerplanar* graph (or a 1-*outerplanar* graph) is a graph that can be drawn in the plane with all vertices lying on the unbounded face. A planar graph is said to be *k-outerplanar*, $k \geq 2$, if it can be drawn in the plane such that by removing the vertices on the unbounded face we obtain a $(k-1)$-outerplanar graph. A canonical example for a k-outerplanar graph is the $2k \times n$ grid (containing $2k$ rows of vertices with n vertices in each row) which also serves as a canonical example for a graph with tree width proportional to k. When referring to k-outerplanar graphs, we usually assume that k is fixed. However, every planar graph is k-outerplanar for some k (typically, much smaller than n) and this *outerplanarity factor* plays a key role in many polynomial time approximation schemes for NP-hard optimization problems on planar graphs [3].

Related Work. The problem of constructing spanning trees with low average stretch was first studied in [2], where it is proved that every n-vertex multigraph G admits a spanning tree T which satisfies $\text{av-str}_G(T) = e^{O(\sqrt{\ln n \ln \ln n})}$. They also show that there exist some graphs, the $\sqrt{n} \times \sqrt{n}$ grid being one of them, for which every spanning tree admits average stretch $\Omega(\log n)$ and conjectured that this lower bound is tight. The upper bound of [2] was improved drastically in [8] by introducing a construction of spanning trees with average stretch $O(\log^2 n \log \log n)$. Very recently, [1] presented a further improvement by establishing an almost tight upper bound of $O(\log n \log \log n \log^3 \log \log n)$.

The notion of probabilistic embedding was explicitly introduced in [4] (although, it was implicitly used in [2]), which initiated a series of papers that developed probabilistic embeddings of arbitrary graphs into non-subgraph dominating trees: in [4] it is shown that every n-vertex graph can be probabilistically embedded into dominating trees with distortion $O(\log^2 n)$, while some graphs must suffer a distortion of $\Omega(\log n)$; the upper bound was improved to $O(\log n \log\log n)$ in [5,6]; and a tight $O(\log n)$ upper bound is proved in [11].

Some papers study probabilistic embeddings of specific graph classes. In [16] it is shown that every planar graph can be probabilistically embedded into its (non-subgraph) dominating trees with distortion $O(\log n)$ using the decomposition technique of [15] for graphs excluding small minors. This is generalized in [14] to graphs of bounded genus by showing that such graphs can be probabilistically embedded with constant distortion into planar graphs. In [12] it is proved that while series-parallel graphs can be embedded into ℓ_1 with constant distortion, there exist some unweighted series-parallel graphs that cannot be probabilistically embedded into dominating trees with distortion $o(\log n)$. This lower bound is matched in [10] by showing that unweighted series-parallel graphs can be probabilistically embedded into their spanning trees with distortion $O(\log n)$.

It is also proved in [12] that every outerplanar graph can be probabilistically embedded into its spanning trees with constant distortion. The construction of [12] for (1-)outerplanar graphs is (partially) generalized to k-outerplanar graphs in [7], where a probabilistic embedding with distortion exponential in k (but independent of n) is presented. Note however, that unlike the construction of [12], the technique of [7] constructs (random) dominating trees which are not necessarily spanning trees of the original graph.

Contribution. In this paper we show that every k-outerplanar multigraph G admits a spanning tree T which satisfies $\text{av-str}_G(T) \leq c^k$, where c is an absolute constant. This immediately implies that every k-outerplanar graph can be probabilistically embedded into its spanning trees with distortion depending solely on k, thus enhancing the result of [7]. Our proof is constructive: we present a randomized algorithm that constructs such spanning trees in expected linear time. Due to lack of space, some of the proofs are omitted from this extended abstract and can be found in the full version [9].

Techniques. The backbone of our algorithm is a rather standard peeling-an-onion decomposition (cf. [7]): on input k-outerplanar graph G, we first peel off the vertices on the unbounded face to obtain a $(k-1)$-outerplanar graph G'; we then recursively construct a good spanning tree T' of G'; next, we insert the missing vertices of G back into T' to obtain the graph H; and finally, we construct a good spanning tree T of H. This framework is formally presented in Section 3. The secret ingredient of the algorithm lies in the last step: constructing a good spanning tree T of H.

As observed in [7], the graph H is essentially a *Halin* graph, which can be viewed as a planar embedding of a tree merged with a cycle. Indeed, our main challenge is to construct low stretch spanning trees for (a generalization of) Halin graphs, as opposed to the non-subgraph dominating trees constructed in [7]. Our construction is completely different than the construction of [7] and it relies on reducing the task of constructing a low stretch spanning tree for a planar graph to that of constructing a low stretch spanning tree for its planar dual (see Theorem 3). This reduction is employed in two distinct occasions within a series of graph manipulations presented in Section 4.

2 Preliminaries

Consider an n-vertex connected graph G. Let $V(G)$ and $E(G)$ denote the vertex and edge sets of G, respectively. Each edge $e \in E(G)$ is associated with some *length* $\ell(e) \in \mathbb{R}_{>0}$. The *length* of a path P in the graph is the sum of lengths of the edges in the path, denoted by $\ell(P) = \sum_{e \in E(P)} \ell(e)$. Given two vertices $u, v \in V(G)$, let $\delta_G(u, v)$ denote the *distance* between them in G, namely, the length of a shortest path from u to v. The *degree* of u, denoted $\deg(u)$, is defined as the number of edges incident on u in G. It will be convenient for us to define the reciprocal of the length of edge e as its *width*, denoted by $w(e) = 1/\ell(e)$.

In what follows we do not distinguish between graphs and multigraphs (namely, a graph may have edge multiplicities). We say that the graph G is *simple*[1] if G contains at most one edge with endpoints u and v for every two vertices $u, v \in V(G)$. Edges that share both endpoints are called *replicas*. Replicas are usually assumed to have the same length (and width). Given two vertices $u, v \in V(G)$, the *multiplicity* of u and v in G, denoted by $\mu_G(u, v)$, is defined to be the number of (u, v)-replicas, i.e., the number of edges connecting u and v. The *skeleton* H of G is the graph obtained from G when all replicas $e_1, \dots, e_m$ of the edge $(u, v) \in E(G)$, $m = \mu_G(u, v)$, are identified to a single edge $e_{u,v} \in E(H)$ with $w(e_{u,v}) = \sum_{i=1}^{m} w(e_i)$. Clearly, the skeleton H is a simple graph. Given a class $\mathcal{C}$ of graphs, and assuming that $\mathcal{C}$ is not closed under edge replication, the class *replicated-*$\mathcal{C}$ consists of every graph whose skeleton is in $\mathcal{C}$.

A path $\pi = (v_1, \dots, v_k)$ in G is said to be *isolated* if $\deg(v_1), \deg(v_k) \neq 2$ and $\deg(v_i) = 2$ for every $1 < i < k$. The graph H obtained from G by contracting every isolated path π to a single edge e_π with $\ell(e_\pi) = \ell(\pi)$ is referred to as the *core* of G. It is easy to verify that distances between vertices of degree different than 2 in H agree with those in G. Given a class $\mathcal{C}$ of graphs, and assuming that $\mathcal{C}$ is not closed under edge subdivision, the class *subdivided-*$\mathcal{C}$ consists of every graph whose core is in $\mathcal{C}$.

A graph is called *biconnected* if the removal of any single vertex does not separate it. A *block* is a maximal biconnected subgraph. Clearly, every spanning tree of G can be edge-partitioned into spanning trees of the blocks of G.

Stretch and Load. Consider some spanning tree T of G and let $e = (u, v)$ be some edge in $E(G)$. The *stretch* of e in T with respect to G is defined to be

$$\mathrm{str}_{T,G}(e) = \delta_T(u, v)/\ell(e) \ .$$

(Observe that the stretch of e in the spanning tree T does not depend on the graph G, but our notation mentions G to recall that this is the graph that

[1] Self loops are ignored in this paper.

"hosts" T.) The *total stretch* of T with respect to G is denoted by $\text{tot-str}_G(T) = \sum_{e \in E(G)} \text{str}_{T,G}(e)$ and the *average stretch* of T with respect to G is simply $\text{av-str}_G(T) = \text{tot-str}_G(T)/|E(G)|$.

Let $\text{cut}_T(e) \subseteq V(T) \times V(T)$ be the set of all (unordered) vertex pairs which are connected in T via e (if $e \notin E(T)$, then $\text{cut}_T(e) = \emptyset$). The *load* of e in T with respect to G is defined to be

$$\text{load}_{T,G}(e) = \sum_{e' \in E(G) \cap \text{cut}_T(e)} w(e')/w(e) \ .$$

The *total load* of T with respect to G is denoted by $\text{tot-load}_G(T) = \sum_{e \in E(G)} \text{load}_{T,G}(e)$ and the *average load* of T with respect to G is simply $\text{av-load}_G(T) = \text{tot-load}_G(T)/|E(G)|$. Since $\text{load}_{T,G}(e) = 0$ for every edge $e \in E(G) - E(T)$, we can rewrite $\text{tot-load}_G(T) = \sum_{e \in E(T)} \text{load}_{T,G}(e)$. By a simple change of summation, we obtain the following corollary which implies that we may shift our focus from the construction of low average stretch spanning trees to that of low average load spanning trees.

Corollary 1. *For every graph G and spanning tree T of G, we have $\text{tot-str}_G(T) = \text{tot-load}_G(T)$.*

Consider some graph H and let T be a spanning tree of H. The *load-replication* $\widehat{T}$ of T under H is the graph obtained from T if each edge $e \in E(T)$ is replicated $\lceil \text{load}_{T,H}(e) \rceil$ times, namely, $\mu_{\widehat{T}}(e) = \lceil \text{load}_{T,H}(e) \rceil$. Clearly, the cardinality of the edge set of $\widehat{T}$ serves as an upper bound on (and a good estimation of) the total load of T under H. Taking load-replications of spanning trees is a fundamental step in our construction based on the following lemma.

Lemma 2. *Consider some graph G, a vertex induced subgraph H of G, and a spanning tree T of H. Let $\widehat{T}$ be the load-replication of T under H and let $\check{G}$ be the graph resulting from G if H is replaced by $\widehat{T}$, that is, $V(\check{G}) = V(G)$ and $E(\check{G}) = (E(G) - E(H)) \cup E(\widehat{T})$. Consider some spanning tree $\check{T}$ of $\check{G}$ (by definition, $\check{T}$ is also a spanning tree of G). Then $\text{load}_{\check{T},G}(e) \leq \text{load}_{\check{T},\check{G}}(e)$ for every edge $e \in E(\check{T})$.*

Lemma 2 essentially states that if we can construct a spanning tree T of H with low $\text{tot-load}_H(T)$, then for the sake of analysis, we can replace H in G by the load-replication of T under H (a replicated-tree) and continue from there.

Planar Duality. Consider some planar graph G and fix some planar embedding η of G. The *planar dual* $\tilde{G}$ of G under η is the graph which has a vertex v_ϕ corresponding to each face ϕ in η and an edge $\tilde{e} = (v_\phi, v_{\phi'})$ corresponding to each edge $e \in E(G)$ on the boundary of the faces ϕ and ϕ' in η. The planar embedding η uniquely determines a *dual* planar embedding $\tilde{\eta}$ of $\tilde{G}$. It is well known that G is the planar dual of $\tilde{G}$ under $\tilde{\eta}$. We refer to the vertex $v_\phi \in V(\tilde{G})$ as the *dual* of the face ϕ and to the edge $\tilde{e} \in E(\tilde{G})$ as the *dual* of the edge $e \in E(G)$ with respect to the planar duality $\langle \eta, \tilde{\eta} \rangle$. We associate lengths (and widths) with the dual edges by setting $\ell(\tilde{e}) = w(e)$ (and $w(\tilde{e}) = \ell(e)$) for every $\tilde{e} \in E(\tilde{G})$. Clearly, this definition of dual edge lengths does not violate the bidirectionality of the planar duality $\langle \eta, \tilde{\eta} \rangle$, i.e., it is still true that if $\tilde{G}$ is the dual of G under η, then G is the dual of $\tilde{G}$ under $\tilde{\eta}$.

Consider some spanning tree T of G. The *dual* of T with respect to the planar duality $\langle \eta, \tilde{\eta} \rangle$ is the subgraph $\tilde{T}$ of $\tilde{G}$ defined by setting $V(\tilde{T}) = V(\tilde{G})$ and $E(\tilde{T}) = \{\tilde{e} \in E(\tilde{G}) \mid e \in E(G) - E(T)\}$. It is proved in [17] that $\tilde{T}$ is a spanning tree of $\tilde{G}$. Combined with the notion of load, we extend the technique of [17] to establish the following lemma.

Lemma 3. *The dual $\tilde{T}$ of T with respect to the planar duality $\langle \eta, \tilde{\eta} \rangle$ is a spanning tree of $\tilde{G}$. Moreover, $|\mathrm{str}_{T,G}(e) - \mathrm{load}_{\tilde{T},\tilde{G}}(\tilde{e})| \leq 1$ for every edge $e \in E(G)$, and therefore* $\text{av-load}_{\tilde{G}}(\tilde{T}) \leq \text{av-load}_G(T) + 1$.

Graph Classes. Given a planar embedding η of G, we say that η is *outerplanar* (or 1-*outerplanar*) if all vertices of G are incident on the unbounded (outer) face in η. Inductively, η is said to be k-*outerplanar*, $k \geq 2$, if by removing the vertices incident on the unbounded face (and the edges incident on these vertices), we obtain a $(k-1)$-outerplanar embedding of the remaining graph. The graph G is called k-*outerplanar* if it admits a k-outerplanar embedding. (An *outerplanar* graph is simply a 1-outerplanar graph.) Observe that outerplanar graphs are closed under edge replication and not closed under edge subdivision.

A *bush* H is a planar graph obtained by taking a planar embedding of a simple cycle C, embedding a forest T in the region enclosed by C (C and T are disjoint), and introducing some new edges, each one of them has at least one endpoint in C. In other words, the (planar) bush H is defined by taking $V(H) = V(T) \cup V(C)$, $V(C) \cap V(T) = \emptyset$, and $E(H) = E(C) \cup E(T) \cup D$, where $D \subseteq V(C) \times (V(C) \cup V(T))$. If each vertex of C has degree at most 3 in H (i.e., it is adjacent to at most one vertex other than its two neighbors in the cycle), then we say that the bush H is a *Halin* graph. Observe that bushes (and Halin graphs) are closed under edge subdivision and not closed under edge replication.

Consider some planar graph G. A vertex $u \in V(G)$ is said to be a *dominating* vertex if it is adjacent to all other vertices of G, that is, if $(u, v) \in E(G)$ for every vertex $v \in V(G) - \{u\}$. The graph is called a *dominated* graph if it has a dominating vertex. A vertex $u \in V(G)$ is said to be a *pivot* vertex if all simple cycles in G go via u. The graph is called a *pivot* graph if it has a pivot vertex. Observe that dominated graphs are closed under edge replication and not closed under edge subdivision. In contrast, pivot graphs are closed under edge subdivision and not closed under edge replication.

Suppose that G is a subdivided-dominated graph and let H be its core. H is a dominated graph, thus it admits a dominating vertex v. Clearly, v is also a vertex of G. Moreover, v is connected by an isolated path (in G) to every vertex of degree different than 2 in $V(G) - \{v\}$. We refer to v as a *weak dominating vertex* of G.

Useful Assumptions. Recall that the graph G may have arbitrary edge multiplicities. In particular, we cannot bound the number of edges $|E(G)|$ as a function of the number of vertices $n = |V(G)|$. Let m be the number of edges in the skeleton of G, that is, the number of (unordered) vertex pairs $(u, v) \in V(G) \times V(G)$ with $\mu_G(u, v) > 0$. The following lemma, which is essentially derived from combining Lemma 5.2 in [2] and Corollary 1, shows that it is sufficient to consider graphs that do not have "too many" edges.

Lemma 4. *For every graph G, there exists some subgraph G' of G on the same vertex set such that (1) $|E(G')| \leq 2m$; and (2)* $\text{av-load}_G(T) \leq 2 \cdot \text{av-load}_{G'}(T)$ *for every spanning tree T of G'. Moreover, G' can be obtained from G in linear time.*

As G is planar, we know that $m \leq 3n - 6$. Therefore by employing Lemma 4, we can subsequently assume that $|E(G)| = O(n)$ at the price of losing a factor of 2 in the performance guarantee. Another assumption we will have to make is that each vertex in G is adjacent to at most three other vertices (although it may be incident on more than three edges due to edge multiplicities). In that case we say that G is *tri-adjacent*. For the purpose of making such an assumption, we introduce a linear time transformation (based on standard techniques), referred to as the *spreading* transformation. The spreading transformation depends on a real parameter $\tau > 0$ and its properties are stated in the following lemma.

Lemma 5. *Let G' be the outcome of the spreading transformation when applied to G with parameter τ. Then G' satisfies the following properties: (1) G' is tri-adjacent; (2) if G is k-outerplanar, then so is G'; and (3) every spanning tree T' of G' that satisfies* $\text{av-load}_{G'}(T') \leq \tau$ *can be* translated *in linear time back into a spanning tree T of G such that* $\text{av-load}_G(T) \leq 3\tau$.

Assuming that the input graph G is tri-adjacent, we will construct in the remainder of the paper a spanning tree T of G that satisfies $\text{av-load}_G(T) \leq c^k$. Therefore by employing Lemma 5 with parameter $\tau = c^k$, we may subsequently make this assumption at the price of losing a factor of 3 in the performance guarantee.

3 The Algorithm — Peeling an Onion

Our goal in this section (and in the whole paper) is to prove the following theorem.

Theorem 6. *For every k-outerplanar graph G, there exists a spanning tree T such that* $\text{av-load}_G(T) \leq c^k$*, where c is a universal constant (independent of k and G).*

The proof of Theorem 6 is constructive: we present a randomized algorithm, referred to as the *onion peeling algorithm*, that given a k-outerplanar graph G with a realizing planar embedding η, constructs the desired spanning tree T of G in expected linear time. Recall our previous assumptions that $|E(G)| = O(n)$, where $n = |V(G)|$ (due to Lemma 4) and that G is tri-adjacent (due to

Lemma 5). The onion peeling algorithm is based on a recursive process similar to that presented in [7] (and essentially, to many other recursive processes on k-outerplanar graphs, cf. [3]). However, the main building block of the onion peeling algorithm, namely, the construction of low stretch spanning trees for (replicated) Halin graphs, is entirely different (see Section 4). This also leads to a different type of analysis.

The onion peeling algorithm works as follows (a formal pseudo-code description is deferred to [9]): (i) remove the vertices on the unbounded face of G (and the edges incident on these vertices) to obtain a $(k-1)$-outerplanar graph G'; (ii) recursively construct a "good" spanning tree T' for G'; (iii) insert the vertices (and edges) that were removed in step (i) back into the planar embedding of T' to compose the graph H; and (iv) construct a "good" spanning tree T of H.

Our algorithm relies on two fundamental constructions. First (implicit in the above description), when the recursion reaches its halting condition on a 1-outerplanar graph G, we have to construct a "good" spanning tree T of G. This is done via the randomized construction of [12] that probabilistically embeds a given outerplanar graph G into its spanning trees with constant distortion. As we will see later on, this randomized construction of [12] is employed by our algorithm in several occasions, and it is subsequently referred to as Procedure `GNRS`. Actually, we shall use a variant of Procedure `GNRS` (the procedure's name is kept, though) whose input may be a subdivided-outerplanar graph[2]. The performance guarantee of Procedure `GNRS` is stated in the following theorem.

Theorem 7. *Procedure* `GNRS`*, when invoked on a subdivided-outerplanar graph G with a realizing planar embedding η, runs in expected linear time and returns a spanning tree T of G that satisfies* $\text{av-load}_G(T) \leq c_1$*, where c_1 is a universal constant (independent of G).*

The existential claim of Theorem 7 is essentially established in [12]. It is trivial to design an expected polynomial time implementation of Procedure `GNRS` and the linear bound on the expected running time is due to a slightly more involved implementation that we omit from this version of the paper.

The second fundamental construction on which the onion peeling algorithm relies is the construction of a "good" spanning tree T of H (step (iv)). A crucial observation in this context is that H is a replicated-Halin graph (actually, if G is simple, then H is strictly a Halin graph). This is due to the assumption that G is tri-adjacent (without which, H would have been a replicated-bush). The technique of [7] probabilistically embeds a Halin graph H into a collection of dominating trees with constant distortion, but these dominating trees are not necessarily spanning trees of H. By contrast, we present a procedure, called Procedure `RH`, which guarantees that T is a spanning tree of H. The input of Procedure `RH` is not assumed to be a (simple) Halin graph, but rather a replicated-Halin graph (hence the name). The performance guarantee of Procedure `RH` is stated in the following theorem, proved in Section 4.

[2] By employing a simple technique presented in [12], one can contract isolated paths at the price of increasing the distortion of the probabilistic embedding by at most 2.

Theorem 8. *Procedure* RH*, when invoked on a replicated-Halin graph G with a realizing planar embedding η, runs in expected linear time and returns a spanning tree T of G that satisfies* $\text{av-load}_G(T) \leq c_2$*, where c_2 is a universal constant (independent of G).*

This leads to the question: what do we mean by a "good" spanning tree? In most of the previous works which considered graph composition based on replacing a subgraph H by a tree T (including [7]), the tree was chosen randomly according to some probability distribution (that may be supported on many trees) and the goal was to guarantee low distortion. In this work we use a different approach: we shall construct a single tree T and our goal is to guarantee low total load. (Corollary 1 stating that the total load is equal to the total stretch, implies that our approach can be viewed as a relaxation of the previous approach.)

Recall that Lemma 2 essentially implies that for the sake of analysis, we may replace the graph G' (the outcome of step (i)) with the load replication of T' (the outcome of step (ii)) under G' before inserting back the vertices and edges that were removed in step (i) and continue with the construction from there. The onion peeling process revolves around this phenomenon.

Analysis (Sketch). Theorem 6 is proved by induction on k. We first employ Theorems 7 (induction's base) and 8 (induction's step) to show that $|E(H)| \leq c^{k-1} \cdot |E(G)|$, where $c = c(c_1, c_2)$ is a universal constant. Next, we use Lemma 2 to argue that $\text{tot-load}_G(T) \leq \text{tot-load}_H(T)$. Finally, Theorem 8 guarantees that $\text{tot-load}_H(T) \leq c \cdot |E(H)|$, which completes the analysis as it implies that $\text{tot-load}_G(T) \leq c^k \cdot |E(G)|$. A full detail of this analysis is deferred to [9].

4 Replicated-Halin Graphs

In this section we present Procedure RH and prove Theorem 8. Recall that the input of Procedure RH is a replicated-Halin graph G with a realizing planar embedding η. The procedure returns a spanning tree T of G which satisfies $\text{av-load}_G(T) \leq c_2$, where c_2 is a universal constant. By Lemma 4, we may assume that $|E(G)| = O(n)$. (This assumption is essentially reflected in the constant c_2, being twice as large as what we obtain in the remainder of this section.)

Taking Planar Duals. Taking planar duals of some special classes of graphs is the main ingredient of our construction. Due to the sensitivity of the definition of load to edge multiplicities, we first want to understand how the operation of identifying two replicas in a planar graph affects its planar dual. To this end, suppose that some two replicas e and e' in the planar primal are identified to form a single edge of width $w(e) + w(e')$. In the planar dual this translates to the contraction of the simple path consisting of $\tilde{e}$ and $\tilde{e}'$ into a single edge of length $\ell(\tilde{e}) + \ell(\tilde{e}') = w(e) + w(e')$. The following observation is a direct consequence of this phenomenon.

Observation 9. *Let G and $\tilde{G}$ be two planar graphs with planar embeddings η and $\tilde{\eta}$, respectively. Let η' (respectively $\tilde{\eta}'$) be the planar embedding of the skeleton* *of G (resp., the core of $\tilde{G}$), naturally derived from η (resp. $\tilde{\eta}$). If η and $\tilde{\eta}$ are duals, then so are η' and $\tilde{\eta}'$.*

We study planar dualities between some specific classes of (planar) graphs. Our insights are cast in the following lemma, whose proof is deferred to [9].

Lemma 10. *Consider a biconnected planar graph G with a planar embedding η and let $\tilde{G}$ be the planar dual of G under η.*

1. *If G is outerplanar with η being a realizing planar embedding, then $\tilde{G}$ is a pivot graph.*
2. *If G is a pivot graph, then $\tilde{G}$ is an outerplanar graph.*
3. *If G is a Halin graph with η being a realizing planar embedding, then $\tilde{G}$ is a dominated graph.*
4. *If G is a dominated graph, then $\tilde{G}$ is a bush.*
5. *Each block of the graph obtained by removing a weak dominating vertex from a subdivided-dominated graph is a subdivided-outerplanar graph.*

Low Load Spanning Trees for Replicated-Halin Graphs. We now turn to describe the operation of Procedure RH on a replicated-Halin graph G with a realizing planar embedding η. As usual, we assume that G is biconnected (otherwise, we can break it and construct a separate spanning tree for each block). The procedure works in 8 steps. The outcome of step i is denoted by T^i if it is (surely) a tree; and by G^i if it is a graph that may contain cycles. (The superscript notation should not be confused with graph powers.) In this spirit, we denote the replicated-Halin graph G by G^0. The 8 steps of Procedure RH are as follows (refer to Figure 1 for a schematic illustration).

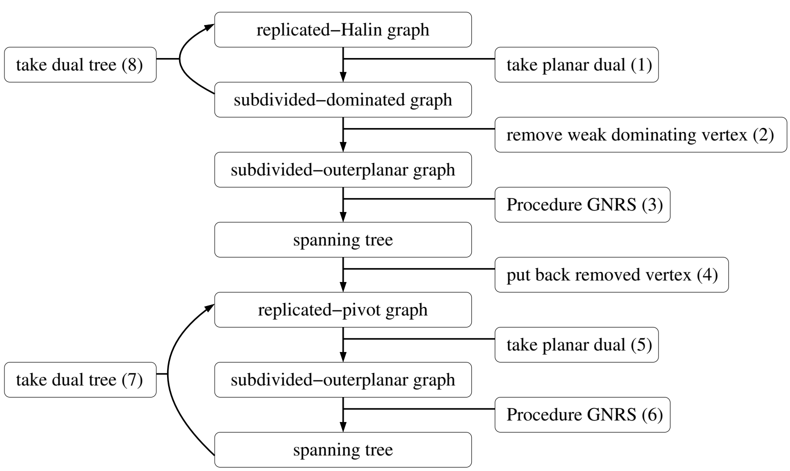

Fig. 1. A schematic illustration of Procedure RH. The step numbers appear in parentheses.

Step 1: Take the planar dual G^1 of G^0 under η. By definition, the skeleton of G^0 is a Halin graph, thus Observation 9 and Lemma 10 imply that G^1 is a subdivided-dominated graph. Let $v \in V(G^1)$ be a weak dominating vertex of G^1.

Step 2: Remove the vertex v and the edges incident on it from G^1 and let G^2 be the remaining graph. Let $G^2_1, \ldots, G^2_m$ be the blocks of G^2. By Lemma 10, G^2_i is a subdivided-outerplanar graph for every $1 \leq i \leq m$.

Step 3: For $i = 1, \ldots, m$, invoke Procedure `GNRS` on G^2_i to generate a spanning tree T^3_i. By Theorem 7, we have $\text{tot-load}_{G^2_i}(T^3_i) \leq c_1 \cdot |E(G^2_i)|$.

Step 4: For $i = 1, \ldots, m$, construct the load-replication $\widehat{T}^3_i$ of T^3_i under G^2_i. Insert the vertex and edges that were removed in step 2 back into the planar embedding of the replicated-trees $\widehat{T}^3_1, \ldots, \widehat{T}^3_m$ to compose the graph G^4. Note that $|E(G^4)| \leq (c_1 + 1) \cdot |E(G^1)|$. Since every simple cycle in the skeleton of G^4 must go via v, we conclude that G^4 is a replicated-pivot graph. Let $G^4_1, \ldots, G^4_{m'}$ be the blocks of G^4 (by definition, each of these blocks is also a replicated pivot graph).

Step 5: For $i = 1, \ldots, m'$, fix some arbitrary planar embedding η'_i of G^4_i and let G^5_i be the planar dual of G^4_i under η'_i. By Observation 9 and Lemma 10, G^5_i is a subdivided-outerplanar graph for every $1 \leq i \leq m'$.

Step 6: For $i = 1, \ldots, m'$, invoke Procedure `GNRS` on G^5_i to generate a spanning tree T^6_i. By Theorem 7, we have $\text{tot-load}_{G^5_i}(T^6_i) \leq c_1 \cdot |E(G^5_i)|$ for every $1 \leq i \leq m'$.

Step 7: For $i = 1, \ldots, m'$, construct the dual T^7_i of the spanning tree T^6_i with respect to the planar duality $\langle \tilde{\eta}'_i, \eta'_i \rangle$, where $\tilde{\eta}'_i$ is the dual planar embedding of η'_i. Lemma 3 guarantees that T^7_i is a spanning tree of G^4_i and by Lemma 3, we have $\text{tot-load}_{G^4_i}(T^7_i) \leq \text{tot-load}_{G^5_i}(T^6_i) + |E(G^5_i)| \leq (c_1+1) \cdot |E(G^5_i)| = (c_1+1) \cdot |E(G^4_i)|$ for every $1 \leq i \leq m'$. Let T^7 be the union of the trees $T^7_1, \ldots, T^7_m$. Note that T^7 is a spanning tree of G^4 and $\text{tot-load}_{G^4}(T^7) \leq (c_1 + 1) \cdot |E(G^4)|$. Since T^7 is also a spanning tree of G^1, we can apply Lemma 2 to deduce that $\text{tot-load}_{G^1}(T^7) \leq \text{tot-load}_{G^4}(T^7) \leq (c_1 + 1) \cdot |E(G^4)| \leq (c_1 + 1)^2 \cdot |E(G^1)|$.

Step 8: Construct the dual T^8 of the spanning tree T^7 with respect to the planar duality $\langle \tilde{\eta}, \eta \rangle$, where $\tilde{\eta}$ is the dual planar embedding of η. Lemma 3 guarantees that T^8 is a spanning tree of G^0 and by Lemma 3, we have $\text{tot-load}_{G^0}(T^8) \leq \text{tot-load}_{G^1}(T^7) + |E(G^1)| \leq ((c_1 + 1)^2 + 1) \cdot |E(G^1)| = ((c_1 + 1)^2 + 1) \cdot |E(G^0)|$.

It follows that upon completion of step 8, we obtain a spanning tree $T = T^8$ which satisfies $\text{av-load}_G(T) \leq c_2$, where $c_2 = (c_1+1)^2+1$ is a universal constant. Theorem 8 follows.

5 Conclusions

We prove that every k-outerplanar graph G admits a spanning tree T such that $\text{av-load}_G(T) \leq c^k$, where c is an absolute constant. The same bound holds for the average stretch of T with respect to G based on the duality of load and stretch. We find it more convenient to bound the (total) load of the trees we construct, mainly due to the (fairly natural) load-replication representation which enables some sort of an iterative graph decomposition. (In previous works, similar approaches were based on probabilistic embeddings.) Planar duality plays a major role in our construction. We hope that some of the tools we develop here will prove useful in other types of embeddings of planar graphs (e.g., into L_1).

Acknowledgments

I would like to thank Robert Krauthgamer, Manor Mendel, and David Peleg for helpful discussions.

References

1. Abraham, I., Bartal, Y., Neiman, O.: Nearly tight low stretch spanning trees. In: 49th IEEE Symp. on Foundations of Computer Science (FOCS), pp. 781–790 (2008)
2. Alon, N., Karp, R.M., Peleg, D., West, D.: A graph-theoretic game and its application to the k-server problem. SIAM J. Comput. 24, 78–100 (1995)
3. Baker, B.S.: Approximation algorithms for NP-complete problems on planar graphs. J. ACM 41(1), 153–180 (1994)
4. Bartal, Y.: Probabilistic approximations of metric spaces and its algorithmic applications. In: 37th IEEE Symp. on Foundations of Computer Science (FOCS), pp. 184–193 (1996)
5. Bartal, Y.: On approximating arbitrary metrics by tree metrics. In: 30th ACM Symp. on the Theory of Computing (STOC), pp. 161–168 (1998)
6. Charikar, M., Chekuri, C., Goel, A., Guha, S., Plotkin, S.A.: Approximating a finite metric by a small number of tree metrics. In: 39th Symp. on Foundations of Computer Science (FOCS), pp. 379–388 (1998)
7. Chekuri, C., Gupta, A., Newman, I., Rabinovich, Y., Sinclair, A.: Embedding k-outerplanar graphs into ℓ_1. SIAM J. Discrete Math. 20(1), 119–136 (2006)
8. Elkin, M., Emek, Y., Spielman, D.A., Teng, S.-H.: Lower-stretch spanning trees. SIAM J. Comput. 38(2), 608–628 (2008)
9. Emek, Y.: k-Outerplanar graphs, planar duality, and low stretch spanning trees, `http://www.eng.tau.ac.il/~yuvale/Publications/k-outerplanar.pdf`
10. Emek, Y., Peleg, D.: A tight upper bound on the probabilistic embedding of series-parallel graphs. In: 17th ACM-SIAM Symp. on Discrete algorithm (SODA), pp. 1045–1053 (2006)
11. Fakcharoenphol, J., Rao, S., Talwar, K.: A tight bound on approximating arbitrary metrics by tree metrics. J. Comput. Syst. Sci. 69(3), 485–497 (2004)
12. Gupta, A., Newman, I., Rabinovich, Y., Sinclair, A.: Cuts, trees and ℓ_1-embeddings of graphs. Combinatorica 24(2), 233–269 (2004)
13. Hu, T.C.: Optimum communication spanning trees. SIAM J. Comput. 3, 188–195 (1974)
14. Indyk, P., Sidiropoulos, A.: Probabilistic embeddings of bounded genus graphs into planar graphs. In: 23rd ACM Symp. on Computational Geometry (SoCG), pp. 204–209 (2007)
15. Klein, P.N., Plotkin, S.A., Rao, S.: Excluded minors, network decomposition, and multicommodity flow. In: 25th ACM Symp. on Theory of Computing (STOC), pp. 682–690 (1993)
16. Konjevod, G., Ravi, R., Salman, F.S.: On approximating planar metrics by tree metrics. Inf. Process. Lett. 80(4), 213–219 (2001)
17. Whitney, H.: On the abstract properties of linear dependence. Amer. J. Math. 57, 509–533 (1935)