A 2-Approximation for the Height of Maximal Outerplanar Graph Drawings *

Therese Biedl and Philippe Demontigny

David R. Cheriton School of Computer Science, University of Waterloo, Waterloo, Ontario N2L 1A2, Canada.
biedl@uwaterloo.ca, phdemontigny@gmail.com

Abstract In this paper, we study planar drawings of maximal outerplanar graphs with the objective of achieving small height. (We do not necessarily preserve a given planar embedding.) A recent paper gave an algorithm for such drawings that is within a factor of 4 of the optimum height. In this paper, we substantially improve the approximation factor to become 2. The main ingredient is to define a new parameter of outerplanar graphs (the *umbrella depth*, obtained by recursively splitting the graph into graphs called umbrellas). We argue that the height of any poly-line drawing must be at least the umbrella depth, and then devise an algorithm that achieves height at most twice the umbrella depth.

1 Introduction

Graph drawing is the art of creating a picture of a graph that is visually appealing. In this paper, we are interested in drawings of so-called *outerplanar graphs*, i.e., graphs that can be drawn in the plane such that no two edges have a point in common (except at common endpoints) and all vertices are incident to the outerface. All drawings are required to be planar, i.e., to have no crossing. The drawing model used is that of flat visibility representations where vertices are horizontal segments and edges are horizontal or vertical segments, but any such drawing can be transformed into a poly-line drawing (or even a straight-line drawing if the width is of no concern) without adding height [6].

Every planar graph with n vertices has a straight-line drawing in an $n \times n$-grid [19,9]. Minimizing the area is NP-complete [17], even for outerplanar graphs [7]. In this paper, we focus on minimizing just one direction of a drawing (we use the height; minimizing the width is equivalent after rotation). It is not known whether minimizing the height of a planar drawing is NP-hard (the closest related result concerns minimizing the height if edges must connect adjacent rows [16]). Given the height H, testing whether a planar drawing of height H exists is fixed parameter tractable in H [12], but the run-time is exceedingly large in H. As such, approximation algorithms for the height of planar drawings are of interest.

* TB supported by NSERC. Part of this work appeared as PD's Master's thesis [10].

F. Ellen et al. (Eds.): WADS 2017, LNCS 10389, pp. 145–156, 2017.
DOI: 10.1007/978-3-319-62127-2_13

It is known that any graph G with a planar drawing of height H has $pw(G) \leq H$ [13], where $pw(G)$ is the so-called pathwidth of G. This makes the pathwidth a useful parameter for approximating the height of a planar graph drawing. For a tree T, Suderman gave an algorithm to draw T with height at most $\lceil \frac{3}{2} pw(T) \rceil$ [20], making this an asymptotic $\frac{3}{2}$-approximation algorithm. It was discovered later that optimum-height drawings can be found efficiently for trees [18]. Approximation-algorithms for the height or width of order-preserving and/or upward tree drawing have also been investigated [1,2,8].

For outerplanar graphs, the first author gave two results that will be improved upon in this paper. In particular, every maximal outerplanar graph has a drawing of height at most $3 \log n - 1$ [3], or alternatively of height $4pw(G) - 3$ [5]. Note that the second result gives a 4-approximation on the height of drawing outerplanar graphs, and improving this "4" is the main objective of this paper. A number of results for drawing outerplanar graphs have been developed since paper [3]. In particular, any outerplanar graph with maximum degree Δ admits a planar straight-line drawing with area $O(\Delta n^{1.48})$ [15], or with area $O(\Delta n \log n)$ [14]. The former bound was improved to $O(n^{1.48})$ area [11]. Also, every so-called balanced outerplanar graph can be drawn in an $O(\sqrt{n}) \times O(\sqrt{n})$-grid [11].

In this paper, we present a 2-approximation algorithm for the height of planar drawings of maximal outerplanar graphs. The key ingredient is to define the so-called *umbrella depth* $ud(G)$ in Section 3. In Section 4, we show that any outerplanar graph G has a planar drawing of height at most $2ud(G) + 1$. This algorithm is a relatively minor modification of the one in [5], albeit described differently. The bulk of the work for proving a better approximation factor hence lies in proving a better lower bound, which we do in Section 5: Any maximal outerplanar graph G with a planar drawing of height H has $ud(G) \leq H - 1$.

2 Preliminaries

Throughout this paper, we assume that G is a simple graph with $n \geq 3$ vertices that is *maximal outerplanar*. Thus, G has a *standard planar embedding* in which all vertices are in the *outer face* (the infinite connected region outside the drawing) and form an n-cycle, and all *interior faces* are triangles. We call an edge (u, v) of G a *cutting edge* if $G - \{u, v\}$ is disconnected, and a *non-cutting edge* otherwise. In a maximal outerplanar graph, any cutting edge (u, v) has exactly two *cut-components*, i.e., there are two maximal outerplanar subgraphs G_1, G_2 of G such that $G_1 \cap G_2 = \{u, v\}$ and $G_1 \cup G_2 = G$.

The *dual tree* T of G is the weak dual graph of G in the standard embedding, i.e., T has a vertex for each interior face of G, and an edge between two vertices iff their corresponding faces in G share an edge. An *outerplanar path* P is a maximal outerplanar graph whose dual tree is a path. P *connects edges* e *and* e' if e is incident to the first face and e' is incident to the last face of the path that is the dual tree of P. An outerplanar path P with $n = 3$ is a triangle and connects any pair of its edges. Since any two interior faces are connected by a path in T, any two edges e, e' of G are connected by some outerplanar path.

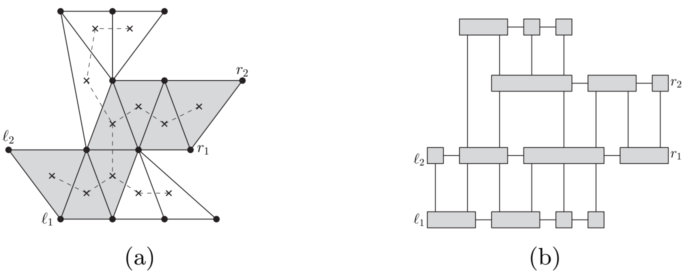


Figure 1: (a) A straight-line drawing in the standard embedding, including the dual tree (dashed edges) and an outerplanar path (shaded) connecting (ℓ_1, ℓ_2) with (r_1, r_2). (b) A flat visibility representation. Both drawings have height 4.

Graph drawing: A *drawing* of a graph assigns to each vertex a point or an axis-aligned box, and to each edge a polygonal curve connecting its endpoints. We only consider *planar drawings* where none of the points, boxes, or curves intersect unless the corresponding elements do in the graph. In this paper, a planar drawing is not required to reflect a graph's given planar embedding. We require that all defining features (points, endpoints of segments, bends) are placed at points with integer y-coordinates. A *layer* (or *row*) is a horizontal line with integer y-coordinate that intersects elements of the drawing, and the *height* is the number of layers.

In a *flat visibility representation* vertices are horizontal line segments, and edges are vertical or horizontal straight-line segments. (For ease of reading, we draw vertices as boxes of small height in our illustrations.) In a *poly-line drawing* vertices are points and edges are polygonal curves, while in a *straight-line drawing* vertices are points and edges are line segments. In this paper, we only study planar flat visibility representations, but simply speak of a *planar drawing*, because it is known that any planar flat visibility representation can be converted into a planar straight-line drawing of the same height and vice versa [6].

3 Umbrellas, bonnets and systems thereof

In this section, we introduce a method of splitting maximal outerplanar graphs into systems of special outerplanar graphs called *umbrellas* and *bonnets*.

Definition 1. *Let G be a maximal outerplanar graph, let U be a subgraph of G with $n \geq 3$, and let (u, v) be a non-cutting edge of G. We say that U is an* umbrella with cap (u, v) *if*

1. *U contains all neighbours of u and v,*
2. *there exists a non-empty outerplanar path $P \subseteq U$ (the* handle*) that connects (u, v) to some non-cutting edge of G, and*
3. *any vertex of U is either in P or a neighbour of u or v.*

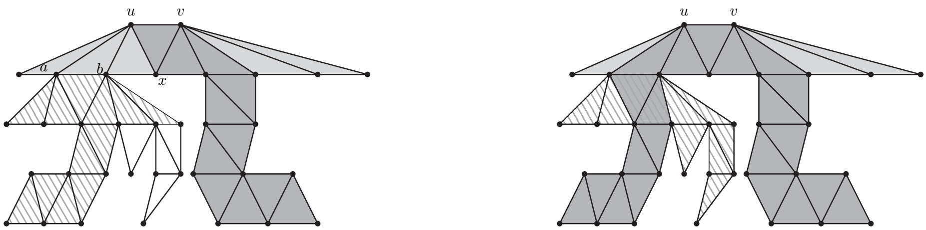
See also Figure 2(a). For such an umbrella U, the *fan at* u is the outerplanar path that starts at an edge (u, x) of the handle P, contains all neighbours of u, and that is minimal with respect to these constraints. If all neighbours of u belong to P, then the fan at u is empty. Define the *fan at* v similarly, using v.

Any edge (a, b) of U that is a cutting edge of G, but not of U, is called an *anchor-edge* of U in G. (In the standard embedding, such edges are on the outerface of U but not on the outerface of G.) The *hanging subgraph with respect to anchor-edge* (a, b) *of* U *in* G is the cut-component $S_{a,b}$ of G with respect to cutting-edge (a, b) that does not contain the cap (u, v) of U. We often omit "of U in G" when umbrella and super-graph are clear from the context.

Definition 2. *Let G be a maximal outerplanar graph with $n \geq 3$, and let (u, v) be a non-cutting edge of G. An* umbrella system $\mathcal{U}$ on G with root-edge (u, v) *is a collection $\mathcal{U} = \mathcal{U}_0 \cup \mathcal{U}_1 \cup \cdots \cup \mathcal{U}_k$ of subgraphs of G for some $k \geq 0$ that satisfy the following:*

1. *$\mathcal{U}_0$ contains only one subgraph U_0 (the* root umbrella*), which is an umbrella with cap (u, v).*
2. *U_0 has k anchor-edges. We denote them by (u_i, v_i) for $i = 1, \ldots, k$, and let S_i be the hanging subgraph with respect to (u_i, v_i).*
3. *For $i = 1, \ldots, k$, $\mathcal{U}_i$ (the* hanging umbrella system*) is an umbrella system of S_i with root-edge (u_i, v_i).*

The depth *of such an umbrella system is defined recursively to be $d(\mathcal{U}) := 1 + \max_{1 \leq i \leq k} d(\mathcal{U}_i)$; in particular $d(\mathcal{U}) = 1$ if $k = 0$.*



(a) (b)

Figure 2: (a) An umbrella system of depth 3. The root umbrella is shaded, with its handle darker shaded. (b) The same graph has a bonnet system of depth 2, with the root bonnet shaded and its ribbon darker shaded.

See also Figure 2(a). A graph may have many different umbrella systems with the same root-edge. Define $ud(G; u, v)$ (the *(rooted) umbrella depth* of G) to be the minimum depth over all umbrella systems with root-edge (u, v). Note that the umbrella depth depends on the choice of the root-edge; define the *free umbrella depth* $ud(G) := ud^{free}(G)$ to be the minimum umbrella depth over all possible root-edges. (One can show that the free umbrella depth is at most one unit less than the rooted umbrella depth for any choice of root-edge; see [10].)

Bonnets: A *bonnet* is a generalization of an umbrella that allows two handles, as long as they go to different sides of the interior face at (u, v). Thus, condition (2) of the definition of an umbrella gets replaced by

2'. There exists a non-empty outerplanar path $P \subseteq U$ (the *ribbon*) that connects two non-cutting edges and contains u, v and their common neighbour.

Other than that, bonnets are defined exactly like umbrellas. See also Figure 2(b). We define *bonnet system*, *root bonnet*, etc., exactly as for an umbrella system, except that "bonnet" is substituted for "umbrella" everywhere. Let $bd(G; u, v)$ (the *rooted bonnet-depth of* G) be the minimum possible depth of a bonnet system with root-edge (u, v), and let $bd^{free}(G) = bd(G)$ be the minimum bonnet-depth over all choices of root-edge. Since any umbrella is a bonnet, we have $bd(G) \leq ud(G)$.

By definition the root bonnet U_0 must contain *all* edges incident to the ends u, v of the root-edge. If follows that no edge incident to u or v can be an anchor-edge of U_0, else the hanging subgraph at it would contain further neighbours of u (resp. v). We note this trivial but useful fact for future reference:

Observation 1 *In a bonnet system with root-edge* (u, v)*, no edge incident to* u *or* v *is an anchor-edge of the root bonnet.*

4 From Bonnet System to Drawing

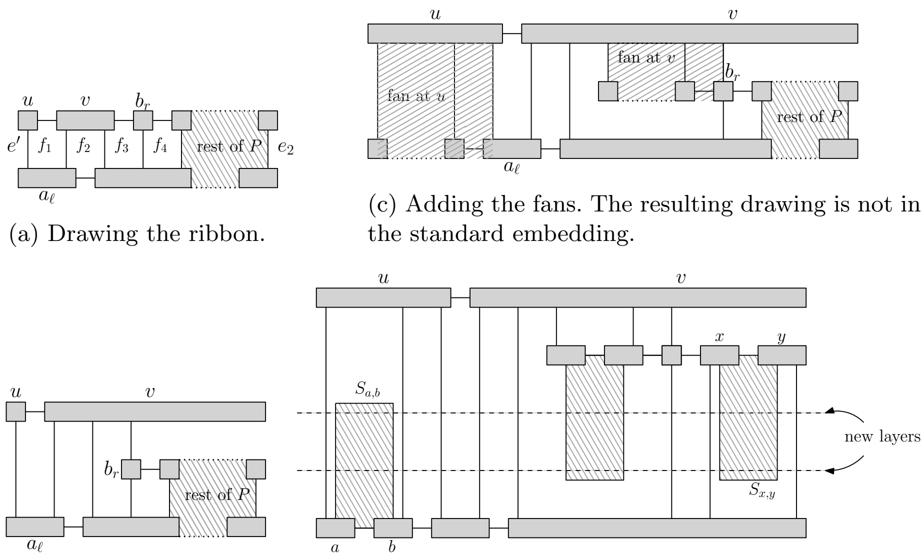
In this section, we show that any outerplanar graph G has a flat visibility representation of height at most $2ud(G) + 1$. We actually show a slightly stronger bound, namely a height of $2bd(G) + 1 \leq 2ud(G) + 1$. So fix a bonnet system of G of depth $bd(G)$ with root-edge (u, v). For merging purposes, we want to draw (u, v) in a special way: It *spans* the top layer, which means that u touches the top left corner of the drawing, and v touches the top right corner, or vice versa (see for example Figure 3(d)). We first explain how to draw the root bonnet U_0.

Lemma 1. *Let* U_0 *be the root bonnet of a bonnet system with root-edge* (u, v)*. Then there exists a flat visibility representation* Γ *of* U_0 *on three layers such that*

1. (u, v) *spans the top layer of* Γ*.*
2. *Any anchor-edge of* U_0 *is drawn horizontally in the middle or bottom layer.*

Proof. As a first step, we draw the ribbon P of U_0 on 2 layers in such a way that (u, v) and all anchor-edges are drawn horizontally; see Figure 3(a) for an illustration. (This part is identical to [5].) To do this, consider the standard embedding of P in which the dual tree is a path, say it consists of faces $f_1, \ldots, f_k$. We draw $k + 1$ vertical edges between two layers, with the goal that the region between two consecutive ones belong to $f_1, \ldots, f_k$ in this order. Place u and v as segments in the top layer, and with an x-range such that they touch all the regions of faces that u and v are incident to. Similarly create segments for all other vertices. The placement for the vertices is uniquely determined by the standard planar embedding, except for the vertices incident to f_1 and f_k. We place those vertices such that the leftmost/rightmost vertical edge is not an anchor-edge. To see that this is possible, recall that P connects two non-cutting edges e_1, e_2 of G that are incident to f_1 and f_k. If $e_1 \neq (u, v)$, then choose the layer for the vertices of f_1 such that e_1 is drawn vertically. If $e_1 = (u, v)$, then one of its ends (say u) is the degree-2 vertex on f_1 and drawn in the top-left corner. The other edge e' incident to u is not an anchor-edge of U by Observation 1, and we draw e' vertically. So the leftmost vertical edge is either a non-cutting edge (hence not an anchor-edge) or edge e' (which is not an anchor-edge). We proceed similarly at f_k so that the rightmost vertical edge is not an anchor-edge. Finally all other vertical edges are cutting edges of U_0 and hence not anchor-edges.

The drawing of P obtained in this first step has (u, v) in the top layer. As a second step, we now *release* (u, v) as in [5]. This operation adds a layer above the drawing, moves (u, v) into it, and re-routes edges at u and v by expanding vertical ones and turning horizontal ones into vertical ones. In the result, (u, v) spans the top layer. See Figure 3(b) for an illustration and [5] for details.



(a) Drawing the ribbon.

(c) Adding the fans. The resulting drawing is not in the standard embedding.

(b) Releasing (u, v).

(d) Merging hanging subgraphs.

Figure 3: From bonnet system to drawing.

As the third and final step, we add the fans. Consider the fan at v, and let (v, b_r) be the edge that it has in common with the ribbon P. Assume first that (v, b_r) was drawn horizontally after the first step, see Figure 3(a). After releasing (u, v) therefore no edge at b_r attaches on its left, see Figure 3(b). Into this space we insert, after adding columns, the remaining vertices of the fan at v, in order in which they appear around v in the standard embedding. See Figure 3(c)).

Else, (v, b_r) was drawn vertically after the first step. (Figure 3(c) does not illustrate this case for v, but illustrates the corresponding case for u.) Since the drawing of the first step is in the standard embedding, and (v, b_r) is on the outerface of the ribbon, therefore (v, b_r) must be the rightmost vertical edge. We can then simply place the vertices of the fan to the right of b_r and extend v.

The fan at u is placed in a symmetric fashion. It remains to show that all anchor-edges are horizontal and in the bottom two layers. We ensured that this is the case in the first step. Releasing (u, v) adds more vertical edges, but all of them are incident to u or v and not anchor-edges by Observation 1. Likewise, all vertical edges added when inserting the fans are incident to u or v. The only horizontal edge in the top layer is (u, v), which is not an anchor-edge. □

Now we explain how to merge hanging subgraphs.

Theorem 1. *Any maximal outerplanar graph G has a planar flat visibility representation of height at most $2bd^{free}(G) + 1$.*

Proof. We show by induction that any graph with a bonnet system $\mathcal{U}$ of depth H has a drawing Γ of height $2H + 1$ where the root-edge (u, v) spans the top layer. This proves the theorem when using a bonnet system $\mathcal{U}$ of depth $bd^{free}(G)$.

Let U_0 be the root bonnet of the bonnet system, and draw U_0 on 3 layers using Lemma 1. Thus (u, v) spans the top and any anchor-edge (a, b) of U_0 is drawn as a horizontal edge in the bottom two layers of Γ_0. If $H = 1$ then there are no hanging subgraphs and we are done. Else add $2H - 2$ layers to Γ_0 between the middle and bottom layers. For each anchor-edge (a, b) of U_0, the hanging subgraph $S_{a,b}$ of U_0 has a bonnet system of depth at most $H - 1$ with root-edge (a, b). By induction $S_{a,b}$ has a drawing Γ_1 on at most $2H - 1$ layers with (a, b) spanning the top layer.

If (a, b) is in the bottom layer of Γ_0, then we can rotate (and reflect, if necessary) Γ_1 so that (a, b) is in the bottom layer of Γ_1 and the left-to-right order of a and b in Γ_1 is the same as their left-to-right order in Γ_0. This updated drawing of Γ_1 can then be inserted in the space between (a, b) in Γ_0. This fits because Γ_1 has height at most $2H-1$, and in the insertion process we can re-use the layer spanned by (a, b). If (a, b) is in the middle layer of U_0, then we can reflect Γ_1 (if necessary) so that (a, b) has the same left-to-right order in Γ_1 as in Γ_0. This updated drawing of Γ_1 can then be inserted in the space between (a, b) in Γ_0. See Figure 3(d). Since we added $2H - 2$ layers to a drawing of height 3, the total height of the final drawing is $2H + 1$ as desired. □

Our proof is algorithmic, and finds a drawing, given a bonnet system, in linear time. One can also show (see [10]) that the rooted bonnet depth, and an associated bonnet system, can be found in linear time using dynamic programming in the dual tree. The free bonnet depth can be found in quadratic time by trying all root-edges, but one can argue [10] that this will save at most one unit of depth and hence barely seems worth the extra run-time.

Comparison to [5]: The algorithm in [5] has only two small differences. The main one is that it does not do the "third step" when drawing the root bonnet, thus it draws the ribbon but not the fans. Thus in the induction step our algorithm always draws at least as much as the one in [5]. Secondly, [5] uses a special construction if $pw(G) = 1$ to save a constant number of levels. This could easily be done for our algorithm as well in the case where $pw(G) = 1$ but $bd(G) = 2$. As such, our construction never has worse height (and frequently it is better).

Comparison to [3]: One can argue that $bd(G) \leq \log(n + 1)$ (see [10]). Since [3] uses $3 \log n - 1$ levels while ours uses $2bd(G) + 1 \leq 2 \log(n + 1) + 1$ levels, the upper bound on the height is better for $n \geq 9$.

5 From Drawing to Umbrella System

The previous section argued that given an umbrella system (or even more generally, a bonnet system) of depth H, we can find a drawing of height at most $2H - 1$. To show that this is within a factor of 2 of the optimum, we show in this section that any drawing of height H gives rise to an umbrella system of depth at most $H - 1$. (Any umbrella system is also a bonnet system, so it also has a bonnet system of depth at most $H - 1$.)

We first briefly sketch the idea. We assume that we have a flat visibility representation, and further, for some non-cutting edge (u, v) we have an "escape path", i.e., a poly-line to the outerface that does not intersect the drawing. Now find an outerplanar path that connects the leftmost vertical edge (x, y) of the drawing with (u, v). This becomes the handle of an umbrella U with cap (u, v). One can now argue that any hanging subgraph of U is drawn with height at most $H - 1$, and furthermore, has an escape path from its anchor-edge. The claim then holds by induction.

We first clarify some definitions illustrated in Figure 4(a). Let Γ be a flat visibility representation, and let B_Γ be a minimum-height bounding box of Γ. A vertex $w \in G$ has a *right escape path* in Γ if there exists a polyline inside B_Γ from w to a point on the right side of B_Γ that is vertex-disjoint from Γ except at w, and for which all bends are on layers. We say that (r_1, r_2) is a *right-free edge* of Γ if it is vertical, and any layer intersected by (r_1, r_2) is empty, except for vertices r_1, r_2, to the right of the edge. In particular, for both r_1 and r_2 the rightward ray on its layer is an escape path. Define *left escape paths* and *left-free edges* symmetrically; an *escape path* is a left escape path or a right escape path.

Observe that in any flat visibility representation any leftmost vertical edge (v, w) is left-free. (Such vertical edges exist, presuming the graph has minimum degree 2, since the leftmost vertex in each layer has at most one incident horizontal edge.) For in any layer spanned by (v, w), no vertical edge is farther left by choice of (v, w), and no vertex can be farther left, else the incident vertical edge of the leftmost of them would be farther left. So (v, w) is left-free.

For the proof of the lower bound, we use as handle an outerplanar path connecting to a left-free edge. Recall that the definition of handle requires that it connects to a non-cutting edge, so we need a left-free edge that is not a cutting

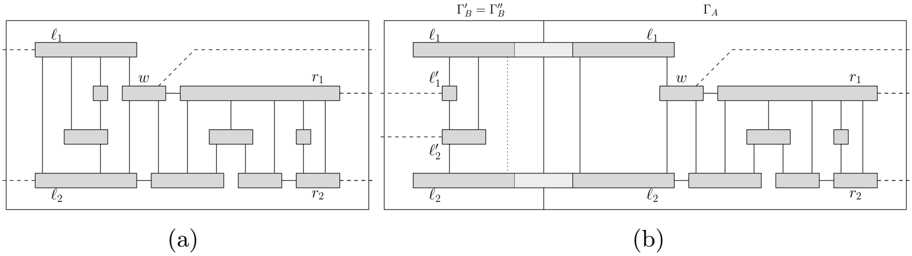


Figure 4: w has a right escape path, (ℓ_1, ℓ_2) is left-free and (r_1, r_2) is right-free. After flipping the cutting component at (ℓ_1, ℓ_2), the non-cutting edge (ℓ_1', ℓ_2') becomes left-free.

edge. This does not exist in all drawings (see e.g. Figure 4(a)), but as we show now, we can modify the drawing without increasing the height such that such an edge exists. To be able to apply it later, we must also show that this modification does not destroy a given escape path.

Lemma 2. *Let Γ be a flat visibility representation of a maximal outerplanar graph G.*

1. *Let (r_1, r_2) be a right-free edge of Γ, and let w be a vertex that has a right escape path. Then there exists a drawing Γ' in which w has a right escape path, (r_1, r_2) is a right-free edge, and there exists a left-free edge that is not a cutting edge of G.*
2. *Let (ℓ_1, ℓ_2) be a left-free edge of Γ, and let w be a vertex that has a left escape path. Then there exists a drawing Γ' in which w has a left escape path, (ℓ_1, ℓ_2) is a left-free edge, and there exists a right-free edge that is not a cutting edge of G.*

In either case, the y-coordinates of all vertices in Γ are unchanged in Γ', and in particular both drawings have the same height.

Proof. We prove the claim by induction on n and show only the first claim (the other is symmetric). Let (ℓ_1, ℓ_2) be the leftmost vertical edge of Γ; this is left-free as argued above. If (ℓ_1, ℓ_2) is not a cutting edge of G, then we are done with $\Gamma' = \Gamma$. This holds in particular if $n = 3$ because then G has no cutting edge.

So assume $n \geq 4$ and (ℓ_1, ℓ_2) is a cutting edge of G. Let A and B be the cut-components of (ℓ_1, ℓ_2), named such that $w \in A$. Let Γ_A [resp. Γ_B] be the drawing of A [B] induced by Γ. Edge (ℓ_1, ℓ_2) is left-free for both Γ_A and Γ_B. Reflect Γ_B horizontally (this makes (ℓ_1, ℓ_2) right-free) to obtain Γ_B'. By induction, we can create a drawing Γ_B'' from Γ_B' in which (ℓ_1, ℓ_2) is right-free and there is a left-free edge (ℓ_1', ℓ_2') that is not a cutting edge of B. We have $(\ell_1', \ell_2') \neq (\ell_1, \ell_2)$, because the common neighbour of ℓ_1, ℓ_2 in B forces a vertex or edge to reside to the left of the right-free edge (ℓ_1, ℓ_2). So (ℓ_1', ℓ_2') is not a cutting edge of G either.

As in Figure 4(b), create a new drawing that places Γ_B'' to the left of Γ_A and extends ℓ_1 and ℓ_2 to join the two copies; this is possible since (ℓ_1, ℓ_2) has the same y-coordinates in $\Gamma_A, \Gamma, \Gamma_B$ and Γ''_B, and it is left-free in Γ_A and right-free in Γ''_B. Also delete one copy of (ℓ_1, ℓ_2). The drawing Γ_A is unchanged, so w will have the same right escape path in Γ' as in Γ, and Γ' will have right-free edge (r_1, r_2) and left-free non-cutting edge (ℓ'_1, ℓ'_2), as desired. □

We are now ready to prove the lower bound if there is an escape path.

Lemma 3. *Let Γ be a flat visibility representation of a maximal outerplanar graph G with height H, and let (u, v) be a non-cutting edge of G. If there exists an escape path from u or v in Γ, then G has an umbrella system with root-edge (u, v) and depth at most $H - 1$.*

Proof. We proceed by induction on H. Assume without loss of generality that there exists a right escape path from v (all other cases are symmetric). Using Lemma 2, we can modify Γ without increasing the height so that v has a right escape path, and there is a left-free edge (ℓ_1, ℓ_2) in Γ that is a not a cutting edge of G. Let P be the outerplanar path that connects edge (ℓ_1, ℓ_2) and (u, v). Let U_0 be the union of P, the neighbors of u, and the neighbors of v; we use U_0 as the root umbrella of an umbrella system.

We now must argue that all hanging subgraphs of U_0 are drawn with height at most $H - 1$ and have escape paths from their anchor-edges; we can then find umbrella systems for them by induction and combining them with U_0 gives the umbrella system for G as desired. To prove the height-bound, define "dividing paths" as follows. The outerface of U_0 in the standard embedding contains (ℓ_1, ℓ_2) (since it is not a cutting edge) as well as v. Let P_1 and P_2 be the two paths from ℓ_1 and ℓ_2 to v along this outerface in the standard embedding. Define the *dividing path Π_i* (for $i = 1, 2$) to be the poly-line in Γ that consists of the leftward ray from ℓ_i, the drawing of the path P_i (i.e., the vertical segments of its edges and parts of the horizontal segments of its vertices), and the right escape path from v. See Figure 5.

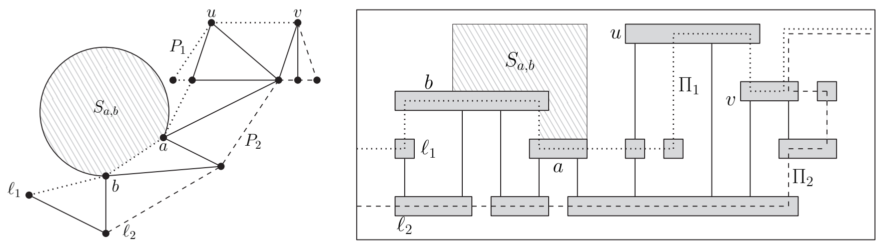


Figure 5: Extracting dividing paths from a flat visibility representation. P_1/Π_1 is dotted while P_2/Π_2 is dashed.

Now consider any hanging subgraph $S_{a,b}$ of U_0 with anchor-edge (a, b). No edge incident to v is an anchor-edge, and neither is (ℓ_1, ℓ_2), since it is not a cutting edge. So (a, b) is an edge of P_1 or P_2 (say P_1) that is not incident to v.

Therefore (a, b) (and with it $S_{a,b}$) is vertex-disjoint from P_2. It follows that the drawing Γ_S of $S_{a,b}$ induced by Γ is disjoint from the dividing path Π_2. Since Π_2 connects a point on the left boundary with a point on the right boundary, therefore Γ_S must be entirely above or entirely below Π_2, say it is above. Since Π_2 has all bends at points with integral y-coordinate, therefore the bottom layer of Γ is not available for Γ_S, and Γ_S has height at most $H-1$ as desired.

Recall that (a, b) belongs to P_1 and is not incident to v. After possible renaming of a and b, we may assume that b is closer to ℓ_1 along P_1 than a. Then the sub-path of P_1 from b to ℓ_1 is interior-disjoint from $S_{a,b}$. The part of Π_1 corresponding to this path is a left escape path from b that resides within the top $H-1$ layers, because it does not contain v and hence is disjoint from Π_2. We can hence apply induction to $S_{a,b}$ to obtain an umbrella system of depth at most $H-2$ with root-edge (a, b). Repeating this for all hanging subgraphs, and combining the resulting umbrella systems with U_0, gives the result. □

Theorem 2. *Let G be a maximal outerplanar graph. If G has a flat visibility representation Γ of height H, then $ud^{free}(G) \leq H-1$.*

Proof. Using Lemma 2, we can convert Γ into a drawing Γ' of the same height in which some edge (u, v) is a right-free non-cutting edge. This implies that there is a right escape path from v, and by Lemma 3 we can find an umbrella system of G with root-edge (u, v) and depth $H-1$. So $ud^{free}(G) \leq ud(G; u, v) \leq H-1$. □

6 Conclusions and Future Work

We presented an algorithm for drawing maximal outerplanar graphs that is a 2-approximation for the optimal height. To this end, we introduced the umbrella depth as a new graph parameter for maximal outerplanar graphs, and used as key result that any drawing of height H implies an umbrella-depth of at least $H-1$. Our result improves the previous best result, which was based on the pathwidth and gave a 4-approximation. We close with some open problems:

- Our result only holds for maximal outerplanar graphs. Can the algorithm be modified so that it becomes a 2-approximation for all outerplanar graphs? Clearly one could apply the algorithm after adding edges to make the graph maximal, but which edges should be added to keep the umbrella depth small?
- The algorithm from Section 4 creates a drawing that does not place all vertices on the outerface. Can we create an algorithm that approximates the optimal height in the standard planar embedding?
- What is the width achieved by the algorithm from Section 4 if we enforce integral x-coordinates? Any visibility representation can be modified without changing the height so that the width is at most $m+n$, where m is the number of edges and n is the number of vertices [6]. Thus the width is $O(n)$, but what is the constant?

Finally, can we determine the optimal height for maximal outerplanar graphs in polynomial time? This question is of interest both if (as in our algorithm) the embedding can be changed, or if the drawing must be in the standard embedding.

References

1. Md. J. Alam, Md. A.H. Samee, M. Rabbi, , and Md. S. Rahman. Minimum-layer upward drawings of trees. *J. Graph Algorithms Appl.*, 14(2):245–267, 2010.
2. J. Batzill and T. Biedl. Order-preserving drawings of trees with approximately optimal height (and small width), 2016. CoRR 1606.02233 [cs.CG]. In submission.
3. T. Biedl. Drawing outer-planar graphs in $O(n \log n)$ area. In S. Kobourov and M. Goodrich, editors, *Graph Drawing (GD'01)*, volume 2528 of *LNCS*, pages 54–65. Springer-Verlag, 2002. Full version included in [4].
4. T. Biedl. Small drawings of outerplanar graphs, series-parallel graphs, and other planar graphs. *Discrete and Computational Geometry*, 45(1):141–160, 2011.
5. T. Biedl. A 4-approximation algorithm for the height of drawing 2-connected outerplanar graphs. In T. Erlebach and G. Persiano, editors, *Workshop on Approximation and Online Algorithms (WAOA'12)*, volume 7846 of *LNCS*, pages 272–285. Springer-Verlag, 2013.
6. T. Biedl. Height-preserving transformations of planar graph drawings. In C. Duncan and A. Symvonis, editors, *Graph Drawing (GD'14)*, volume 8871 of *LNCS*, pages 380–391. Springer, 2014.
7. T. Biedl. On area-optimal planar grid-drawings. In J. Esparza, P. Fraigniaud, T. Husfeldt, and E. Koutsoupias, editors, *International Colloquium on Automata, Languages and Programming (ICALP '14)*, volume 8572 of *LNCS*, pages 198–210. Springer-Verlag, 2014.
8. T. Biedl. Ideal tree-drawings of approximately optimal width (and small height). *Journal of Graph Algorithms and Applications*, 21(4):631–648, 2017.
9. H. de Fraysseix, J. Pach, , and R. Pollack. How to draw a planar graph on a grid. *Combinatorica*, 10:41–51, 1990.
10. P. Demontigny. A 2-approximation for the height of maximal outerplanar graphs. Master's thesis, University of Waterloo, 2016. See also CoRR report 1702.01719.
11. G. Di Battista and F. Frati. Small area drawings of outerplanar graphs. *Algorithmica*, 54(1):25–53, 2009.
12. V. Dujmovic, M. Fellows, M. Kitching, G. Liotta, C. McCartin, N. Nishimura, P. Ragde, F. Rosamond, S. Whitesides, , and D. Wood. On the parameterized complexity of layered graph drawing. *Algorithmica*, 52:267–292, 2008.
13. S. Felsner, G. Liotta, , and S. Wismath. Straight-line drawings on restricted integer grids in two and three dimensions. *J. Graph Alg. Appl*, 7(4):335–362, 2003.
14. F. Frati. Straight-line drawings of outerplanar graphs in $O(dn \log n)$ area. *Comput. Geom.*, 45(9):524–533, 2012.
15. A. Garg and A. Rusu. Area-efficient planar straight-line drawings of outerplanar graphs. *Discrete Applied Mathematics*, 155(9):1116–1140, 2007.
16. L.S. Heath and A.L. Rosenberg. Laying out graphs using queues. *SIAM Journal on Computing*, 21(5):927–958, 1992.
17. M. Krug and D. Wagner. Minimizing the area for planar straight-line grid drawings. In S. Hong, T. Nishizeki, and W. Quan, editors, *Graph Drawing (GD'07)*, volume 4875 of *LNCS*, pages 207–212. Springer-Verlag, 2007.
18. D. Mondal, Md. J. Alam, , and Md. S. Rahman. Minimum-layer drawings of trees. In N. Katoh and A. Kumar, editors, *Algorithms and Computations (WALCOM 2011)*, volume 6552 of *LNCS*, pages 221–232. Springer, 2011.
19. W. Schnyder. Embedding planar graphs on the grid. In *ACM-SIAM Symposium on Discrete Algorithms (SODA '90)*, pages 138–148, 1990.
20. M. Suderman. Pathwidth and layered drawings of trees. *Intl. J. Comp. Geom. Appl*, 14(3):203–225, 2004.