Small Area Drawings of Outerplanar Graphs[⋆]
(Extended Abstract)

Giuseppe Di Battista and Fabrizio Frati

Dipartimento di Informatica e Automazione, Università di Roma Tre
gdb@dia.uniroma3.it, fabriziofrati@tiscali.it

Abstract. We show three linear time algorithms for constructing planar straight-line grid drawings of outerplanar graphs. The first and the second algorithm are for balanced outerplanar graphs. Both require linear area. The drawings produced by the first algorithm are not outerplanar while those produced by the second algorithm are. On the other hand, the first algorithm constructs drawings with better angular resolution. The third algorithm constructs outerplanar drawings of general outerplanar graphs with $O(n^{1.48})$ area. Further, we study the interplay between the area requirements of the drawings of an outerplanar graph and the area requirements of a special class of drawings of its dual tree.

1 Introduction

Straight-line drawings of planar graphs have been studied by several authors and constitute one of the main fields of investigation in Graph Drawing. Groundbreaking works of the end of the 20th Century [5, 13, 4] have shown that a planar graph with n vertices has a planar straight-line drawing with integer coordinates ("grid" drawing) with $O(n^2)$ area. Further, it has been shown [12] that there exist graphs that, for such drawings, require quadratic area.

Planar straight-line grid drawings have also been studied for subclasses of planar graphs, looking for subquadratic area bounds. For example a linear area algorithm for drawing binary trees with arbitrary aspect ratio has been shown in [8].

Another subclass of planar graphs that attracted research work in this field is the one of the outerplanar graphs. An outerplanar graph is a planar graph that has a planar drawing such that all its vertices are on the outer face. The dual graph of an outerplanar graph is a tree (but for the outer face). Garg and Rusu [9] proved that an n-vertex outerplanar graph has a planar straight-line grid drawing with $O(d \cdot n^{1.48})$ area, where d is the maximum degree of the vertices of the graph. Biedl [1] conjectured that $O(n \lg n)$ area is sufficient for such graphs.

In [10, 2] are presented algorithms for constructing straight-line drawings with vertices in general position.

Outerplanar graphs have been studied also with respect to other types of drawings. In [1] and in [11] are presented algorithms to construct planar polyline drawings with

[⋆] Work partially supported by EC - Fet Project DELIS - Contract no 001907, by "Project ALGO-NEXT: Algorithms for the Next Generation Internet and Web: Methodologies, Design, and Experiments", MIUR Programmi di Ricerca Scientifica di Rilevante Interesse Nazionale, and by "MAIS Project", MIUR–FIRB.

P. Healy and N.S. Nikolov (Eds.): GD 2005, LNCS 3843, pp. 89–100, 2005.

$O(n \log n)$ area and $O(d \cdot n)$ area, respectively. An algorithm for constructing in three dimensions straight-line drawings with linear volume is presented in [7].

In this paper we present the following results. They always refer to planar straight-line grid drawings. We show (Section 3) a linear time algorithm for constructing non-outerplanar drawings of balanced outerplanar graphs in linear area and with angular resolution $\geq \frac{c}{\sqrt{n}}$, with c constant. A balanced outerplanar graph is such that its dual tree is balanced. We define a new type of drawings of binary trees, called star-shaped drawings (Section 4). We show that, given a drawing of an outerplanar graph it can be found a star-shaped drawing of its dual tree with the same area bound. Conversely, given a star-shaped drawing of a binary tree it can be found a drawing of its dual outerplanar graph with the same area bound but for the placement of two special vertices. Based on such correspondence, we show a linear time algorithm for drawing a balanced outerplanar graph in linear area (Section 4). The drawings obtained with this algorithm are outerplanar, but the angular resolution is worse with respect to the algorithm of Section 3. Again, based on the above correspondence and exploiting a decomposition technique of binary trees presented in [3], we show a linear time algorithm for constructing outerplanar drawings of general outerplanar graphs with $O(n^{1.48})$ area (Section 5).

2 Preliminaries

We assume familiarity with Graph Drawing (see e.g. [6]).

An *outerplanar graph* is a planar graph that has a planar drawing with all its vertices on the same (say outer) face. Such a drawing is called *outerplanar drawing*. In this paper we deal with outerplanar graphs that are also biconnected. However, this is not a limitation since an outerplanar graph can be always augmented with a linear number of extra edges to a biconnected outerplanar graph. Hence, the algorithms and theorems we present can be applied also to general outerplanar graphs after a simple preprocessing step that does not alter the number of vertices of the graph.

We define the *dual graph* of an outerplanar graph G as follows. The vertices of the dual graph are the faces of G, with the exception of the outer face that is not associated to any vertex of the dual of G. Two vertices f_1 and f_2 of the dual graph sharing an edge of G are connected, in the dual graph, by edge (f_1, f_2). The dual graph of an outerplanar graph is always a tree. Hence, in the following we call it *dual tree*.

A *maximal outerplanar graph* is an outerplanar graph such that all its faces but, eventually, the outer face are composed by three edges. Note that any outerplanar graph can be augmented to a maximal outerplanar by adding extra edges. The vertices of the dual graph of a maximal outerplanar graph have degree at most three. From now on, unless otherwise specified, we assume that outerplanar graphs are maximal.

We can select an edge (u, v) of the outer face of an outerplanar graph G and root the dual tree T of G at the internal face r containing (u, v). Let w be the third vertex of r. We call vertices u and v *poles* and vertex w *central vertex*. We also call u *left vertex* and v *right vertex*. Consider a face f of T and suppose that f is composed in G by edges (v_1, v_2), (v_2, v_3), and (v_3, v_1), in this clockwise order around f. Also, suppose that the parent of f in T and f share edge (v_1, v_2) or that (f is the root) $(v_1, v_2) = (u, v)$. The face sharing with f (if any) edge (v_3, v_1) is the *left child* of f, while the face sharing with f (if any) edge (v_2, v_3) is the *right child* of f. We obtain a binary tree.

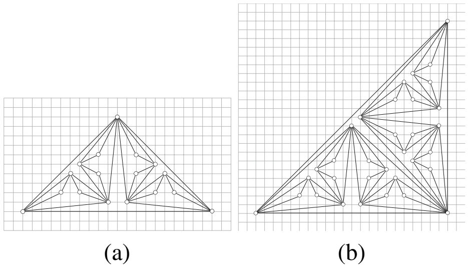
A *balanced outerplanar graph* is an outerplanar graph whose dual tree can be rooted to a balanced binary tree. The *height* of an outerplanar graph is the number of nodes on the longest path of its dual tree from the root to a leaf. A *complete outerplanar graph* is an outerplanar graph whose dual tree is a complete binary tree. A *grid drawing* of a graph is such that all its vertices have integer coordinates. A *straight-line drawing* is such that all edges are rectilinear segments. Let Γ be a straight-line grid drawing and consider the smallest rectangle $B(\Gamma)$ with sides parallel to the x- and y-axes that covers Γ completely. We call $B(\Gamma)$ the *bounding box* of Γ. We denote with $b(\Gamma)$, $t(\Gamma)$, $l(\Gamma)$ and $r(\Gamma)$ the bottom, top, left and right side of $B(\Gamma)$, respectively. The *height* (*width*) of Γ is one plus the height (width) of $B(\Gamma)$. The *area* of Γ is the height of Γ multiplied by its width.

3 Non-outerplanar Drawings of Balanced Outerplanar Graphs

We call G_h a complete outerplanar graph with height h, T_h its dual tree, and Γ_h its planar straight-line grid drawing. Let also u_h, v_h and w_h be the left vertex, the right vertex and the central vertex of G_h, respectively.

We show an inductive algorithm to draw complete outerplanar graphs. **Base case:** if $h = 1$, then place u_1 in $(0,0)$, v_1 in $(1,1)$ and w_1 in $(1,0)$. **Inductive case:** if $h > 1$, suppose you have drawn Γ_{h-1}; let r be the line through v_{h-1} and w_{h-1}, let b be the line through u_{h-1} and v_{h-1} and let a be the line parallel to and at horizontal distance one unit from r, in the opposite side of the drawing with respect to r. Shift u_{h-1} and v_{h-1} along b of one horizontal unit, moving away from Γ_{h-1}. Now mirror the modified drawing Γ_{h-1} with respect to a. Insert the edge from u_{h-1} to its symmetric vertex, say z. Let $u_h = u_{h-1}$, $v_h = z$ and $w_h = v_{h-1}$. Examples of the drawings produced by the algorithm are shown in Fig. 1. Showing the planarity of the obtained drawings is trivially done by induction. Now we analyze their area requirement. Let $height_h$ and $width_h$ be the height and the width of Γ_h, respectively. We distinguish two cases. **h is even:** it's easy to see that $height_{h-1} = 2 \cdot height_{h-2} + 1$ and that $height_h = height_{h-1} + 2$. So we have $height_h = 2 \cdot height_{h-2} + 3$. Hence we obtain:
$height_h = \ldots(((height_2 \underbrace{\cdot 2 + 3) \cdot 2 + 3) \ldots \cdot 2 + 3}_{\frac{h-2}{2} times} = height_2 \cdot 2^{\frac{h-2}{2}} + 3 \cdot 2^{\frac{h-4}{2}} + 3 \cdot 2^{\frac{h-6}{2}} + \ldots + 3$. Let $m = \frac{h-2}{2}$; replacing $height_2$ with its value 4 we obtain: $height_h = 4 \cdot 2^m + 3 \cdot 2^{m-1} + 3 \cdot 2^{m-2} + \ldots + 3 = 4 \cdot 2^m + 3 \cdot (2^{m-1} + 2^{m-2} + \ldots + 1) = 4 \cdot 2^m + 3 \cdot (2^m - 1) = 7 \cdot 2^m - 3 = 7 \cdot 2^{\frac{h-2}{2}} - 3 = \frac{7}{2} \cdot 2^{\frac{h}{2}} - 3 = \frac{7}{2} \cdot 2^{(\lg n)^{\frac{1}{2}}} - 3 = \frac{7}{2} \cdot n^{\frac{1}{2}} - 3 = \frac{7}{2} \cdot \sqrt{n} - 3 = O(n^{\frac{1}{2}})$. It's easy to see that: $width_h = 2 \cdot height_h - 1 = 7 \cdot \sqrt{n} - 7 = O(n^{\frac{1}{2}})$. If **h is odd**, using $height_h = 2 \cdot height_{h-1} + 1$ we obtain: $height_h = \left(\frac{7}{2} \cdot 2^{\frac{h-1}{2}} - 3\right) \cdot 2 + 1 = \frac{7}{\sqrt{2}} \cdot 2^{\frac{h}{2}} - 5 = \frac{7}{\sqrt{2}} \cdot 2^{(\lg n)^{\frac{1}{2}}} - 5 = \frac{7}{\sqrt{2}} \cdot \sqrt{n} - 5 = O(n^{\frac{1}{2}})$. It's easy to see that the width is equal to the height, hence we have: $width_h = \frac{7}{\sqrt{2}} \cdot \sqrt{n} - 5 = O(n^{\frac{1}{2}})$.

About the angular resolution, let u_h be the left vertex of G_h. Recall that $u_{h-1} = u_h$. Passing from G_{h-1} to G_h the number of the neighbours of u_{h-1} increases by one,



(a) (b)

Fig. 1. Applications of the algorithm in *Section 3*. (a) Γ_4. (b) Γ_5.

beacuse of the insertion in Γ_h of the symmetric vertex of u_{h-1}. Let m (t) be the largest odd (even) integer $\leq h$. We can prove by induction that the smallest angle in the drawing is ϕ, the angle between half-lines a and b starting at u_h and passing respectively through v_a, the neighbour of u_h inserted in Γ_m, and through v_b, the neighbour of u_h inserted in Γ_{m-2}. Let c be the half-line starting at u_h and passing through v_c, the neighbour of u_h inserted in Γ_t. Let v_d be the intersection point between a and the line through v_b and orthogonal to a. Let v_e be the intersection point between c and the line through v_b and orthogonal to c. Let R_1 be the triangle whose vertices are u_h, v_b and v_e. We denote with a_h, b_h and c_h the lengths of the segments $\overline{v_b v_e}$, $\overline{u_h v_e}$ and $\overline{u_h v_b}$, respectively. Note that, by construction, $b_h = a_h + 1$. Now suppose h is odd; following the construction of the algorithm we obtain $a_h = 2 \cdot a_{h-2} + 4$ and solving the recurrence equation we obtain $a_h = \frac{7}{2} \cdot 2^{\frac{h-1}{2}} - 4$. Hence $b_h = \frac{7}{2} \cdot 2^{\frac{h-1}{2}} - 3$. Applying the Pythagorean theorem to R_1 we obtain $c_h = \sqrt{a_h^2 + b_h^2} = \sqrt{\frac{49}{4} \cdot 2^h - 49 \cdot 2^{\frac{h-1}{2}} + 25}$. Observing that $\overline{v_b v_d} = \frac{\sqrt{2}}{2}$, for every h, we finally obtain: $\phi \approx \sin \phi = \frac{\sqrt{2}}{2\sqrt{\frac{49}{4} \cdot 2^h - 49 \cdot 2^{\frac{h-1}{2}} + 25}} > c \cdot 2^{-h/2} = \frac{c}{\sqrt{n}}$, where c is a constant. If h is even, in a similar way we obtain:$\phi \approx \sin \phi = \frac{\sqrt{2}}{2\sqrt{\frac{49}{8} \cdot 2^h - \frac{35}{2} \cdot 2^{\frac{h}{2}} + 13}} > c \cdot 2^{-h/2} = \frac{c}{\sqrt{n}}$, where c is a constant. From the above discussion and from the fact that a balanced outerplanar graph can be augmented to complete without altering its height we have:

Theorem 1. *Given an n-vertex balanced outerplanar graph G with height h, there exists an $O(n)$ time algorithm that constructs a planar straight-line grid drawing Γ of G such that: (i) if h is even, then the height of Γ is $\frac{7}{2}\sqrt{n} - 3$ and its width is $7\sqrt{n} - 7$; (ii) if h is odd, then the height of Γ is $\frac{7}{\sqrt{2}}\sqrt{n} - 5$ and its width is $\frac{7}{\sqrt{2}}\sqrt{n} - 5$; (iii) the angular resolution of Γ is greater than $\frac{c}{\sqrt{n}}$, with c constant; (iv) if G is complete, then isomorphic subgraphs of G have congruent drawings in Γ up to a translation and a reflection; and (v) if G is complete, then Γ is axially symmetric.*

4 Outerplanar Drawings and Star-Shaped Drawings

Let T be a binary tree rooted at r. The *leftmost* (*rightmost*) *path* of T is the path $v_0, v_1, \ldots, v_m$ such that $v_0 = r$, v_{i+1} is the left (right) child of v_i, $\forall i$ such that $0 \leq$

$i \leq m-1$, and v_m doesn't have a left (right) child. The *outer-left set* (*outer-right set*) of a planar straight-line drawing Γ of T is the set of points with integer coordinates from which we can draw edges to each one of the nodes of the leftmost (rightmost) path of T without crossing Γ. The *left-right* (*right-left*) *path* of a node $n \in T$ is the path $v_0, v_1, \ldots, v_m$ such that $v_0 = n$, v_1 is the left (right) child of v_0, v_{i+1} is the right (left) child of v_i, $\forall i$ such that $1 \leq i \leq m-1$, and v_m doesn't have a right (left) child. The *left polygon of the neighbours* (*right polygon of the neighbours*) of a node $n \in T$ is the polygon of the segments representing in Γ the edges of the left-right path (of the right-left path) plus an extra segment connecting v_m and v_0.

A planar straight-line order-preserving drawing Γ of T is *star-shaped* if all the following conditions are satisfied. (1) For each node $n \in T$ its left (right) polygon of neighbours $P_l = (n, v_1, \ldots, v_m)$ ($P_r = (n, v_1, \ldots, v_m)$) is a simple polygon and each segment $(n, v_i), 2 \leq i \leq m-1$ belongs to the interior of P_l (P_r), but for its endpoints n and v_i. (2) For each pair of nodes $n_1, n_2 \in T$ the left polygon of neighbours or the right polygon of neighbours of n_1 does not intersect with the left polygon of neighbours or with the right polygon of neighbours of n_2, but, possibly, at common endpoints or at common edges. (3) There exist point p_l in the outer-left set of T and point p_r in the outer-right set of T such that segment (p_l, p_r) doesn't intersect any edge of Γ.

Given a drawing Γ of an outerplanar graph we call *internal subdrawing* the drawing obtained by deleting from Γ its poles and their incident edges.

Lemma 1. *Let G be an n-vertex outerplanar graph such that its dual tree T has a star-shaped drawing with $f(n)$ area. We have that G has an outerplanar straight-line drawing such that the area of its internal subdrawing is $f(n)$.*

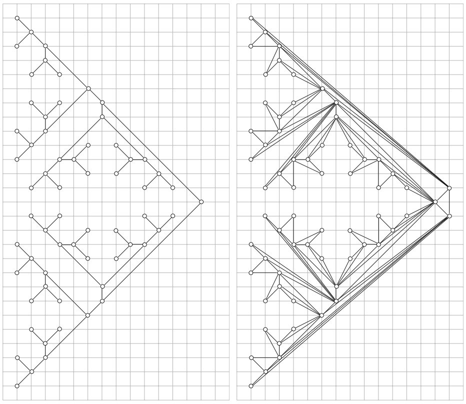
Lemma 2. *Let G be an n-vertex outerplanar graph that has an outerplanar straight-line drawing with $f(n)$ area. We have that its dual tree T has a planar star-shaped straight-line drawing with an area that is at most $f(n)$.*

To prove the above lemmas we first establish a correspondence γ between the vertices of G and the nodes of T, so that for each node $n \in T$ there is one and only one vertex v of G such that $\gamma(n) = v$ and for each vertex $v \in G$, but for the poles, there is one and only one node $n \in T$ such that $\gamma^{-1}(v) = n$. Consider a subtree of T rooted at n. Suppose that (v_l, v_c) is the edge of G dual to the edge connecting n to its left child (if any). Analogously, suppose that (v_r, v_c) is the edge of G dual to the edge connecting n to its right child (if any). We set $\gamma(n) = v_c$. Now, suppose you have a planar star-shaped straight-line grid drawing Γ of T. Map each vertex v of G, but for its poles, to the point where the node n such that $\gamma^{-1}(v) = n$ is drawn. Map the left vertex u_l of G to a point p_l of the outer-left set and the right vertex v_r of G in a point p_r of the outer-right set so that the edge (p_l, p_r) doesn't intersect any of the edges of T. Draw the edges from u_l to each vertex on the leftmost path of T and the edges from v_r to each vertex on the rightmost path of T. Draw the edge (u_l, v_r). By Condition (3) in the definition of star-shaped drawing and by the definitions of outer-left set and of outer-right set, p_l and p_r exist and their incident edges don't intersect Γ. For each node n (and so for each vertex $v = \gamma(n)$) draw edges to each vertex on its left-right path and to each vertex on its right-left path. Because of Condition 1 and 2 in the definition of star-shaped drawing each of such segments doesn't intersect any other segment of the drawing. The drawing obtained after these insertions is an outerplanar straight-line grid drawing of G as a consequence of the construction and of the correspondence between vertices of G and nodes of T. We have also just seen that each step preserves the initial planarity. The area bound of Lemma 1 is easily obtained by observing that the vertices of G (but the poles) and the nodes of T have exactly the same coordinates.

Now we can start from an outerplanar drawing Φ of G, then we can use again the correspondence between vertices of G and nodes of T to obtain a star-shaped drawing of T. Remove from Φ the poles of G. For each vertex v let n be the node of T such that $\gamma^{-1}(v) = n$ and let n_l and n_r be the left and the right child of n, respectively. Remove all edges incident on v, but those whose second endpoint is a vertex z such that $\gamma^{-1}(z) = n_l$ or $\gamma^{-1}(z) = n_r$. We obtain a star-shaped drawing of T: it's easy to see that the drawing is planar, straight-line, grid and order-preserving and that all the conditions of a star-shaped drawing are verified, since the initial drawing Φ is a planar straight-line grid drawing of G. Again, the area bound of Lemma 2 is easily obtained by observing that the vertices of G (but the poles) and the nodes of T have exactly the same coordinates.

We apply the above lemmas to construct a linear area drawing of a complete outerplanar graph. We denote with T_h a complete binary tree, r_h its root, and Γ_h its drawing. What follows is an inductive algorithm to construct a star-shaped drawing of a complete binary tree. **Base case:** if $h = 1$, then place r_1 in $(0, 0)$. **Inductive case:** if $h > 1$, suppose you have drawn Γ_{h-1}. Now we distiguish two subcases. **h is even:** let r be the highest horizontal line such that r intersects Γ_{h-1}. Let a be the line above r parallel to and at vertical distance one unit from r. Let b be the lowest line with slope $\frac{\pi}{4}$ with respect to the x-axis and such that b intersects Γ_{h-1}. Mirror Γ_{h-1} with respect to a. Place r_h at the intersection between a and b. Insert the edges from r_h to its children. If **h is odd** let r be the highest line with slope $\frac{3\pi}{4}$ with respect to the x-axis and such that r intersects Γ_{h-1}. Let a be the line above r parallel to and at vertical distance two units from r. Let b be the lowest line with slope $\frac{\pi}{4}$ with respect to the x-axis and such that b intersects Γ_{h-1}. Mirror Γ_{h-1} respect to a. Translate the new part of the drawing by a vector $(-1, 0)$. Place r_h at the intersection between a and b. Insert the edges from r_h to its children. A drawing produced by the algorithm is shown in Fig. 2.a.

It is easy to see, by induction, that the resulting drawing is star-shaped. Now we analyze the area requirements of the above algorithm. Let $height_h$ and $width_h$ be the height and the width of Γ_h, respectively. We distinguish two cases. **h is even:** it's easy to see that $height_{h-1} = height_{h-2} + 2$ and that $height_h = 2 \cdot height_{h-1} + 1$. So we have $height_h = 2 \cdot height_{h-2} + 5$. Hence we obtain:
$height_h = \ldots(((height_2 \underbrace{\cdot 2 + 5) \cdot 2 + 5) \ldots \cdot 2 + 5}_{\frac{h-2}{2} times} = height_2 \cdot 2^{\frac{h-2}{2}} + 5 \cdot 2^{\frac{h-4}{2}} + 5 \cdot 2^{\frac{h-6}{2}} + \ldots + 5$. Let $m = \frac{h-2}{2}$; replacing $height_2$ with its value 3 we obtain: $height_h = 3 \cdot 2^m + 5 \cdot 2^{m-1} + 5 \cdot 2^{m-2} + \ldots + 5 = 3 \cdot 2^m + 5 \cdot (2^{m-1} + 2^{m-2} + \ldots + 1)$ $= 3 \cdot 2^m + 5 \cdot (2^m - 1) = 8 \cdot 2^m - 5 = 8 \cdot 2^{\frac{h-2}{2}} - 5 = 4 \cdot 2^{\frac{h}{2}} - 5 = 4 \cdot 2^{(\lg n)^{\frac{1}{2}}} - 5 = 4 \cdot n^{\frac{1}{2}} - 5 = 4 \cdot \sqrt{n} - 5 = O(n^{\frac{1}{2}})$. It's easy to see that $width_h = \frac{height_h + 1}{2} = 2 \cdot \sqrt{n} - 2 = O(n^{\frac{1}{2}})$. If **h is odd**, using $height_h = height_{h-1} + 2$ we obtain: $height_h$



(a) (b)

Fig. 2. Application of the algorithm in *Section 4*. (a) Γ_6. (b) The drawing of G_6 built over Γ_6.

$= \left(4 \cdot 2^{\frac{h-1}{2}} - 5\right) + 2 = \frac{4}{\sqrt{2}} \cdot 2^{\frac{h}{2}} - 3 = \frac{4}{\sqrt{2}} \cdot 2^{(\lg n)^{\frac{1}{2}}} - 3 = \frac{4}{\sqrt{2}} \cdot \sqrt{n} - 3 = O(n^{\frac{1}{2}})$. It's easy to see that $width_h = height_h - 1 = \left(\frac{4}{\sqrt{2}} \cdot \sqrt{n} - 3\right) - 1 = \frac{4}{\sqrt{2}} \cdot \sqrt{n} - 4 = O(n^{\frac{1}{2}})$.

We exploit the above algorithm and Lemma 1 to prove the following theorem.

Theorem 2. *Given an n-vertex balanced outerplanar graph G with height h, there exists an $O(n)$ time algorithm that constructs an outerplanar straight-line grid drawing Γ of G such that: (i) if h is even, then the height of Γ is $4\sqrt{n} - 5$ and its width is $2\sqrt{n} - 1$; (ii) if h is odd, then the height of Γ is $\frac{4}{\sqrt{2}}\sqrt{n} - 3$ and its width is $\frac{4}{\sqrt{2}}\sqrt{n} - 3$; (iii) the angular resolution of Γ is less than $\frac{c}{n}$, with c constant; (iv) if G is complete, then isomorphic subgraphs of G have congruent drawings in Γ up to a translation and a reflection; and (v) if G is complete, then Γ is axially symmetric.*

Proof. Γ is constructed as follows. First, we add to G dummy vertices and edges to make it complete without altering h. Second, we draw star-shaped its dual tree T. Third, using the correspondence between the vertices of G and the nodes of T introduced in the proof of Lemmas 1 and 2, we build a drawing Γ' of the internal subgraph of G. Finally, we place the poles of G and their incident edges, obtaining Γ. This is done as follows. We place the left vertex on the same line of $b(\Gamma')$, one unit to the right of $r(\Gamma')$ and we place the right vertex on the same line of $t(\Gamma')$ one unit to the right of $r(\Gamma')$. This placement allows to draw edges from the left vertex to each node of the leftmost path of T and from the right vertex to each node of the rightmost path of T without crossings. Furthermore, this placement increases by one unit the width without altering the height of Γ. Note that similar but different placements of the poles, as the one in Fig. 2.b, are also possible.

The bounds on height and width of Γ descend from the bounds given for star-shaped drawings. Now we analyze the angular resolution. Namely, we show that there is an angle that decreases faster than $\frac{1}{n}$. If h is odd let v_1 be the root of T, else (h even) let v_1 be the left child of the root of T. Let $(v_1, w_0, w_1, \ldots, w_m)$ be the left-right path of

v_1. Let ϕ be the angle between the half-lines a and b starting at v_1 and passing through w_{m-1} and w_m, respectively. From trigonometry we have:

$\sin\phi = \sin(\widehat{w_m v_1 w_0})\cos(\widehat{w_{m-1} v_1 w_0}) - \sin(\widehat{w_{m-1} v_1 w_0})\cos(\widehat{w_m v_1 w_0})$. Observe that $\overline{v_1w_0} = \sqrt{2}$, $\forall h$. Let k be the biggest even integer $\leq h-1$. We have $\overline{w_0w_m} = \sqrt{2}(2\cdot 2^{\frac{k}{2}} - 3)$; moreover $\overline{w_0w_{m-1}} = \overline{w_0w_m} - \sqrt{2}$, since $\overline{w_{m-1}w_m} = \sqrt{2}$, $\forall h$; hence $\overline{w_0w_{m-1}} = \sqrt{2}(2\cdot 2^{\frac{k}{2}} - 4)$. Using the Pythagorean theorem we obtain:

$$\begin{aligned}
\sin\phi = {} & \frac{\overline{w_0w_m}}{\sqrt{\overline{w_0w_m}^2 + \overline{v_1w_0}^2}} \cdot \frac{\overline{v_1w_0}}{\sqrt{\overline{w_0w_{m-1}}^2 + \overline{v_1w_0}^2}} + \\
& + \frac{\overline{v_1w_0}}{\sqrt{\overline{w_0w_m}^2 + \overline{v_1w_0}^2}} \cdot \frac{\overline{w_0w_{m-1}}}{\sqrt{\overline{w_0w_{m-1}}^2 + \overline{v_1w_0}^2}} = \\
& = \frac{\overline{v_1w_0}(\overline{w_0w_m} - \overline{w_0w_{m-1}})}{\sqrt{\overline{w_0w_m}^2 + \overline{v_1w_0}^2}\sqrt{\overline{w_0w_{m-1}}^2 + \overline{v_1w_0}^2}} \\
& = \frac{\sqrt{2}\sqrt{2}}{\sqrt{2(2\cdot 2^{\frac{k}{2}} - 3)^2}\sqrt{2(2\cdot 2^{\frac{k}{2}} - 4)^2}} = \\
& = \frac{2}{2\sqrt{4\cdot 2^k - 12\cdot 2^{\frac{k}{2}} + 10}\sqrt{4\cdot 2^k - 16\cdot 2^{\frac{k}{2}} + 17}}.
\end{aligned}$$

Hence $\phi \approx \sin\phi < c\cdot(2^{-k})$ and since $k = O(h)$, we have $\phi < \frac{c}{n}$, with c constant.

5 Outerplanar Drawings of General Outerplanar Graphs

This section is devoted to the proof of the following theorem. The main ingredients of the proof are: (i) a recursive algorithm for constructing a star-shaped drawing of a binary tree, (ii) Lemma 1, and (iii) Lemma 3 presented by Chan in [3].

Theorem 3. *Given an n-vertex outerplanar graph G, there exists an $O(n)$ time algorithm that constructs an $O(n^{1.48})$ area outerplanar straight-line grid drawing of G.*

Lemma 3. *[3] Let $p = 0.48$. Given any binary tree T of size n, there exists a root-to-leaf path π such that for any left subree α and right subtree β of π, $|\alpha|^p + |\beta|^p \leq (1-\delta)n^p$, for some constant $\delta > 0$.*

First, we show two techniques, called Constructions 1–2, for constructing a star-shaped drawing Γ_i, with $i \in \{1,2\}$, of a general binary tree T with n nodes. Each one is defined in terms of itself and of the other one. In the following we call *spine* a root-to-leaf path $S = (v_0, v_1, \ldots, v_m)$ of T. Let s_i be the non spine child of v_i and let $T(s_i)$ be the subtree of T rooted at s_i. We denote with $W_i(n)$ the width of Γ_i, with $W_{i,l}(n)$ ($W_{i,r}(n)$) the width of the part of Γ_i that is to the left (to the right) of S and with $n(t)$ the number of nodes in the subtree of T rooted at t.

Now we show **Construction** 1. First, we draw each $v_i \in S$ together with $T(s_i)$, obtaining $\Gamma(v_i)$; then we put all the $\Gamma(v_i)$ together to obtain Γ_1. Construction 1 has four subcases, labelled $1xy$, $x \in \{t,b\}$ and $y \in \{l,r\}$. Index x states that S is drawn

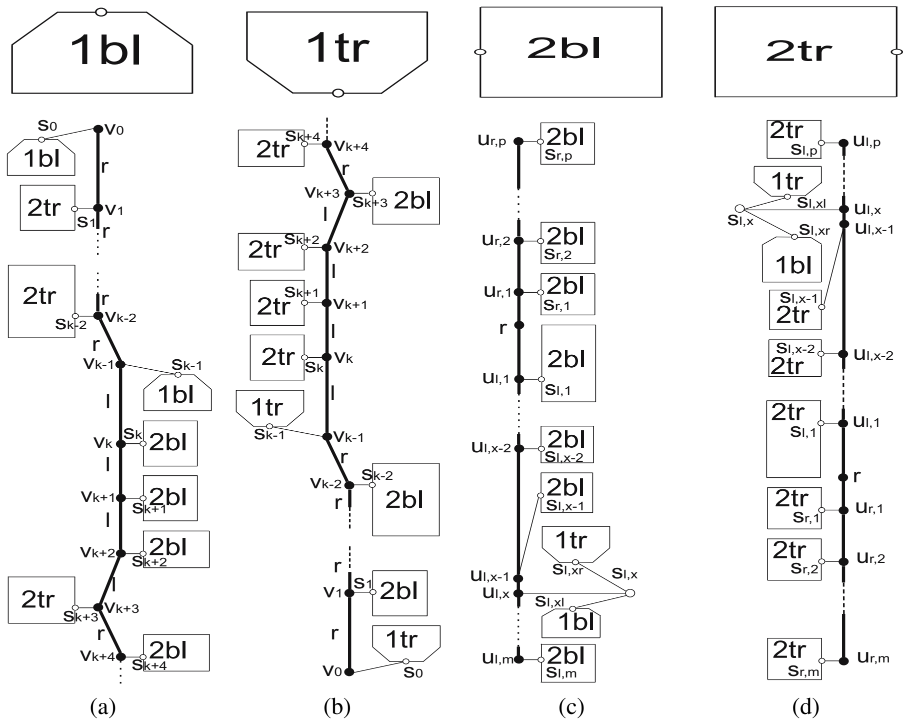


Fig. 3. Constructions (a)$1bl$, (b)$1tr$, (c)$2bl$ and (d)$2tr$. The edges (v_i, v_{i+1}) labelled r (l) are such that v_{i+1} is the right (resp. left) child of v_i. The thick edges show the spine (in Construction 1) and the leftmost and the rightmost paths (in Construction 2) of T.

going towards the top ($x = t$) or towards the bottom ($x = b$) of Γ_1. Index y states that the leftmost path ($y = l$), or the rightmost path ($y = r$), is drawn going towards the left. In the following we show the details of Construction $1bl$, while the others are easily obtained from $1bl$ after a reflection with respect to the x-axis and/or a switch of the left with the right and vice-versa. Constructions $1bl$ and $1tr$ are shown in Fig. 3.

Suppose v_1 is the left (right) child of v_0. Let k be the first index such that v_k is the right (left) child of v_{k-1}. In the following we denote the subtree $T(s_{k-1})$ ($T(s_0)$) also with $T(s_l)$ and we denote the subtree $T(s_0)$ ($T(s_{k-1})$) also with $T(s_r)$. Draw $T(s_0)$ and $T(s_{k-1})$ with Construction $1bl$, obtaining $\Gamma(s_0)$ and $\Gamma(s_{k-1})$, respectively. Draw v_0 one unit above and one unit to the left (right) of $B(\Gamma(s_0))$, obtaining $\Gamma(v_0)$. Draw v_{k-1} one unit above and one unit to the right (left) of $B(\Gamma(s_{k-1}))$, obtaining $\Gamma(v_{k-1})$. Draw any other left (right) subtree with Construction $2tr$ (with Construction $2bl$), obtaining $\Gamma(s_i)$. If s_i is the left (right) child of v_i, draw v_i on the same horizontal channel and one unit to the right (to the left) of s_i, obtaining $\Gamma(v_i)$.

Now we put together all the $\Gamma(v_i)$, $0 \leq i \leq m$ as follows. Place $\Gamma(v_0)$ anywhere in the plane. For $1 \leq i \leq m$, if v_i is the left child (right child) of v_{i-1} and v_{i+1} is the left child (right child) of v_i or v_i is a leaf ($i = m$), then draw $\Gamma(v_i)$ so that v_i is on the same vertical channel of v_{i-1} and so that $b(\Gamma(v_{i-1}))$ is one unit above the $t(\Gamma(v_i))$.

Else (v_i is the left child (right child) of v_{i-1} and v_{i+1} is the right child (left child) of v_i) if v_i is a left child (right child) draw v_i on the vertical channel one unit to the left (to the right) with respect to the vertical channel of v_{i-1} and so that $b(\Gamma(v_{i-1}))$ is one unit above $t(\Gamma(v_i))$.

Property 1. Construction $1bl$ guarantees that all the vertices of the leftmost (rightmost) path of T are visible from any point that is above and to the left (right) of $B(\Gamma_1)$.

Property 2. Suppose that the drawing of Construction $2tr$ is star-shaped and that it places the leftmost and the rightmost paths of the tree on the right side of its bounding box. Suppose also that the drawing of Construction $2bl$ is star-shaped and that it places the leftmost and the rightmost paths of the tree on the left side of its bounding box. We have that the drawing obtained with Construction $1bl$ is star-shaped.

Property 3. $W_{1,l}(n) = \max(W_1(n(s_l)), \max_i(W_2(n(s_i))))$, where i is such that s_i is the left child of v_i. $W_{1,r}(n) = \max(W_1(n(s_r)), \max_i(W_2(n(s_i))))$, where i is such that s_i is the right child of v_i.

Analogous properties hold for Constructions $1br$, $1tl$,and $1tr$.

Construction 2 is as follows. We have four subcases, say $2xy$, where $x \in \{t, b\}$ and $y \in \{l, r\}$. Index x states that the leftmost path is drawn going towards the top ($x = t$) or going towards the bottom ($x = b$) of Γ_2. Index y states that the root is drawn on the right side ($y = r$) or on the left side ($y = l$) of Γ_2. In the following we show Construction $2bl$, while the other cases are easily obtained from $2bl$ after a reflection with respect to the y-axis and/or a switch of the left with the right and vice-versa. Constructions $2bl$ and $2tr$ are shown in Fig. 3.Let r be the root of T, let $C_l = (u_{l,0}, u_{l,1}, \dots, u_{l,m})$ ($C_r = (u_{r,0}, u_{r,1}, \dots, u_{r,p})$) be the leftmost (rightmost) path of T, with $u_{l,0} = u_{r,0} = r$. Let $s_{l,i}$ ($s_{r,i}$) be the right (left) child of a node $u_{l,i} \in C_l$ ($u_{r,i} \in C_r$); we call $T(s_{l,i})$ ($T(s_{r,i})$) the subtree of T rooted in $s_{l,i}$ ($s_{r,i}$). First, we draw each $u_{l,i} \in C_l$ together with $T(s_{l,i})$ and each $u_{r,i} \in C_r$ together with $T(s_{r,i})$, obtaining $\Gamma(u_{l,i})$ and $\Gamma(u_{r,i})$ respectively; then we put all the $\Gamma(u_{l,i})$ and the $\Gamma(u_{r,i})$ together to obtain Γ_2.

Let k and j be two indexes such that $k, j \in \{l, r\}$ and let x such that $1 \leq x \leq m$ if $k = l$ and such that $1 \leq x \leq p$ if $k = r$. Find the heaviest subtree $T(s_{k,x})$ among all the subtrees $T(s_{j,i})$. Let $T(s_{k,xl})$ and $T(s_{k,xr})$ be the left and the right subtree of $s_{k,x}$, with root $s_{k,xl}$ and $s_{k,xr}$, respectively. Draw $T(s_{k,xl})$ with Construction $1bl$ and draw $T(s_{k,xr})$ with Construction $1tr$, obtaining $\Gamma(s_{k,xl})$ and $\Gamma(s_{k,xr})$, respectively. Draw any other subtree $T(s_{j,i})$ with Construction $2bl$, obtaining $\Gamma(s_{j,i})$.

Place $\Gamma(s_{k,xl})$ anywhere in the plane. Place $\Gamma(s_{k,xr})$ so that $b(\Gamma(s_{k,xr}))$ is three vertical units above $t(\Gamma(s_{k,xl}))$ and so that $l(\Gamma(s_{k,xr}))$ is on the same vertical channel of $l(\Gamma(s_{k,xl}))$. Place $s_{k,x}$ one unit above $t(\Gamma(s_{k,xl}))$ and one unit to the right of the rightmost boundary between $r(\Gamma(s_{k,xl}))$ and $r(\Gamma(s_{k,xr}))$. Draw $u_{k,x}$ on the same horizontal channel of $s_{k,x}$, one unit to the left of $l(\Gamma(s_{k,xl}))$. If $k = l$ ($k = r$) draw $u_{k,x-1}$ one unit above (one unit below) $u_{k,x}$. Place $\Gamma(s_{k,x-1})$ so that $l(\Gamma(s_{k,x-1}))$ is on the same vertical channel of $l(\Gamma(s_{k,x}))$ and so that (if $k = l$) $b(\Gamma(s_{k,x-1}))$ is one unit above $t(\Gamma(s_{k,x}))$ or (if $k = r$) $t(\Gamma(s_{k,x-1}))$ is one unit below $b(\Gamma(s_{k,x}))$, obtaining $\Gamma(u_{k,x-1})$. For each $\Gamma(s_{j,i})$, but for $\Gamma(s_{k,x-1})$ and $\Gamma(s_{k,x})$, place $u_{j,i}$ one unit to the left of $s_{j,i}$, obtaining $\Gamma(u_{j,i})$. Finally place all the $\Gamma(u_{j,i})$ (and so also $\Gamma(u_{k,x-1})$)

so that all $u_{j,i}$ are on the same vertical channel, so that $b(\Gamma(u_{r,i}))$ is one unit above $t(\Gamma(u_{r,i-1}))$, $2 \leq i \leq p$, so that $t(\Gamma(u_{l,i}))$ is one unit below $b(\Gamma(u_{l,i-1}))$, $2 \leq i \leq m$, and so that $t(\Gamma(u_{l,1}))$ is one unit below $b(\Gamma(u_{r,1}))$.

Property 4. Construction $2bl$ guarantees that all the vertices of the leftmost (rightmost) path of T are on the left side of the bounding box of Γ_2.

Property 5. Suppose that the drawing of Constructions $1tr$ and $1bl$ are star-shaped. Suppose that Construction $1tr$ is such that the leftmost (the rightmost) path of $T(s_{k,xr})$ is visible from any point that is below and to the right (to the left) of $B(\Gamma(s_{k,xr}))$. Suppose also that Construction $1bl$ is such that the leftmost (the rightmost) path of $T(s_{k,xl})$ is visible from any point that is above and to the left (to the right) of $B(\Gamma(s_{k,xl}))$. We have that the drawing obtained with Construction $2bl$ is star-shaped.

Property 6. $W_2(n) = \max(2+W_1(n(s_{k,xl})), 2+W_1(n(s_{k,xr})), \max(1+W_2(n(s_{j,i}))))$, where $j \in \{l, r\}$ and i is not equal to x.

Analogous properties hold for Constructions $2br$, $2tl$,and $2tr$.

We can use Constructions 1–2 for constructing a star-shaped drawing Γ of a binary tree T as follows. First, we select any spine. Second, we apply Construction $1bl$. Third, we recursively apply all the constructions in the appropriate cases. From the above properties we have that Γ is star-shaped.

At this point we can draw a general outerplanar graph G with dual tree T as follows. First, we draw T with the above algorithm. Second, we apply Lemma 1 to construct an outerplanar drawing of the internal subgraph of G with the same height and width of T. Third, exploiting Property 1 we place the poles of G obtaining a drawing that has the same height and width plus one unit.

Now we analyze the height and the width of Γ. About the height, it's easy to see that there is at least one vertex for each horizontal line that intersects Γ. So we immediately obtain that the height of Γ is $O(n)$. About the width $W(n)$, let n_1 (n_2) be the number of vertices of the heaviest left (right) subtree of the spine S. We want to show that $W(n) \leq W(n_1) + W(n_2) + 6$.

We focus on $W_{1,l}(n)$ to show that $W_{1,l}(n) \leq W(n_1) + 2$. For this purpose we start from the expression of $W_{1,l}(n)$ as a function of $W_1(n)$ and of $W_2(n)$, then we substitute $W_2(n)$ with its definition as function of $W_1(n)$ and of $W_2(n)$. We repeat this substitution until we have obtained that $W_{1,l}(n)$ is defined only in terms of $W_1(n)$.

Let $n(s_j^*)$ be the maximum number of nodes of a subtree recursively drawn with Construction 2, after that j substitutions of $W_2(n)$ with its definition (as a function of $W_1(n)$ and of $W_2(n)$) have been made. Let $n(s_{j,l}^*)$ and $n(s_{j,r}^*)$ be the number of nodes of the left and the right subtrees of s_j^*, respectively.

By Property 3 we have $W_{1,l}(n) = \max(W_1(n(s_l)), \max(W_2(n(s_i))))$, with i such that $T(s_i)$ is the left subtree of a spine node v_i. By applying several times Property 6 to the above equation we have: $W_{1,l}(n) = \max(W_1(n(s_l)), 2 + W_1(n(s_{k,xl})), 2 + W_1(n(s_{k,xr})), 1+W_2(n(s_1^*))) \leq \max(W_1(n(s_l)), 2+W_1(n(s_{k,xl})), 2+W_1(n(s_{k,xr})), 3+W_1(n(s_{1,l}^*)), 3+W_1(n(s_{1,r}^*)), 2+W_2(n(s_2^*))) \leq \max(W_1(n(s_l)), 2+W_1(n(s_{k,xl})), 2+W_1(n(s_{k,xr})), 3+W_1(n(s_{1,l}^*)), 3+W_1(n(s_{1,r}^*)), 4+W_1(n(s_{2,l}^*)), 4+W_1(n(s_{2,r}^*)), 3 + W_2(n(s_3^*))) \leq \ldots \leq \max(W_1(n(s_l)), 2 + W_1(n(s_{k,xl})), 2 + W_1(n(s_{k,xr})), 3 +$

$W_1(n(s^*_{1,l})), 3 + W_1(n(s^*_{1,r})), 4 + W_1(n(s^*_{2,l})), 4 + W_1(n(s^*_{2,r})), 5 + W_1(n(s^*_{3,l})), 5 + W_1(n(s^*_{3,r})), \ldots)$

Observe that $n(s^*_{j+1}) \leq \frac{1}{2}n(s^*_j)$, since we draw the heaviest subtree T' of $T(s^*_j)$ with Construction 1 and a subtree T'' with size greater than $\frac{1}{2}n(s^*_j)$ implies $n(T') + n(T'') > n(s^*_j)$, that is impossible by definition. Hence, assuming $W_1(n) > \lg n$, we obtain $W_{1,l}(n) \leq \max(W_1(n(s_l)), 2 + W_1(n(s_{k,xl})), 2 + W_1(n(s_{k,xr}))) \leq 2 + W_1(n_1)$. With similar arguments we obtain $W_{1,r}(n) \leq 2 + W_1(n_2)$. Observing that S is drawn on two adjacent vertical channels we have $W_1(n) = W_{1,l}(n) + W_{1,r}(n) + 2$, hence we obtain $W(n) = W_1(n) \leq W_1(n_1) + W_1(n_2) + 6 \leq W(n_1) + W(n_2) + 6$. As done in [3], we can choose in linear time a spine of T by maintaining the invariance that ${n_1}^p + {n_2}^p \leq (1 - \delta)n^p$. Observe that $W(n) \leq \max_{n_l^p + n_r^p \leq (1-\delta)n^p}(W(n_l) + W(n_r) + 6)$, for any left (right) subtree of S with n_l (n_r) nodes; by induction this solves to $W(n) = O(n^p)$ and applying Lemma 3, we can complete the analysis of the width of Γ concluding that is possible to get $W(n) = O(n^{0.48})$.

From the results on the height and on the width, we obtain the $O(n^{1.48})$ area bound on Γ. It is easy to see that the algorithm can be implemented to run in linear time.

References

1. T. Biedl. Drawing outer-planar graphs in $O(n \log n)$ area. In M. Goodrich, editor, *Graph Drawing (Proc. GD '02)*, volume 2528 of *Lecture Notes Comput. Sci.*, pages 54–65. Springer-Verlag, 1997.
2. P. Bose. On embedding an outer-planar graph in a point set. In G. Di Battista, editor, *Graph Drawing (Proc. GD '97)*, volume 1353 of *Lecture Notes Comput. Sci.*, pages 25–36. Springer-Verlag, 1997.
3. T.M. Chan. A near-linear area bound for drawing binary trees. *Algorithmica*, 34(1), 2002.
4. M. Chrobak and T. H. Payne. A linear-time algorithm for drawing a planar graph on a grid. *IPL*, 54(4):241–246, 1995.
5. H. de Fraysseix, J. Pach, and R. Pollack. How to draw a planar graph on a grid. *Combinatorica*, 10(1):41–51, 1990.
6. G. Di Battista, P. Eades, R. Tamassia, and I. G. Tollis. *Graph Drawing*. Prentice Hall, Upper Saddle River, NJ, 1999.
7. S. Felsner, G. Liotta, and S. K. Wismath. Straight-line drawings on restricted integer grids in two and three dimensions. *J. Graph Algorithms Appl.*, 7(4):363–398, 2003.
8. A. Garg and A. Rusu. Straight-line drawings of binary trees with linear area and arbitrary aspect ratio. In M. Goodrich, editor, *Graph Drawing (Proc. GD '02)*, volume 2528 of *LNCS*, pages 320–331. Springer-Verlag, 2002.
9. A. Garg and A. Rusu. Area-efficient planar straight-line grid drawings of outerplanar graphs. In G.Liotta, editor, *Graph Drawing (Proc. GD '03)*, volume 2912 of *Lecture Notes Comput. Sci.*, pages 129–134. Springer-Verlag, 2003.
10. P. Gritzmann, B. Mohar, J. Pach, and R. Pollack. Embedding a planar triangulation with vertices at specified points. *Amer. Math. Monthly*, 98(2):165–166, 1991.
11. C. E. Leiserson. Area-efficient graph layouts (for VLSI). In *Proc. 21st Annu. IEEE Sympos. Found. Comput. Sci.*, pages 270–281, 1980.
12. S. Malitz and A. Papakostas. On the angular resolution of planar graphs. *SIAM J. Discrete Math.*, 7:172–183, 1994.
13. W. Schnyder. Embedding planar graphs on the grid. In *Proc. 1st ACM-SIAM Sympos. Discr. Alg.*, pages 138–148, 1990.