



Rook-drawings of Plane Graphs

David Auber¹ Nicolas Bonichon¹ Paul Dorbec¹ Claire Pennarun¹

¹Univ. Bordeaux, LaBRI, UMR 5800, F-33400 Talence, France
CNRS, LaBRI, UMR 5800, F-33400 Talence, France

Abstract

We introduce a new type of graph drawing called “rook-drawing”. A *rook-drawing* of a graph G is obtained by placing the n nodes of G on the intersections of a regular grid, such that each row and column of the grid supports exactly one node. This paper focuses on rook-drawings of planar graphs. We first give a linear algorithm to compute a planar straight-line rook-drawing for outerplanar graphs. We then characterize the maximal planar graphs admitting a planar straight-line rook-drawing, which are unique for a given order. Finally, we give a linear time algorithm to compute a polyline planar rook-drawing for plane graphs with at most $n - 3$ bent edges.

Submitted: November 2015	Reviewed: April 2016	Revised: May 2016	Accepted: August 2016	Final: September 2016
		Published: January 2017		
	Article type: Regular Paper		Communicated by: E. Di Giacomo and A. Lubiw	

This work has been carried out as part of the “REQUEST” project (PIAO18062-645401) supported by the French “Investissement d’Avenir” Program (Big Data - Cloud Computing topic) and has been supported by ANR grant JCJC EGOS ANR-12-JS02-002-01

E-mail addresses: auber@labri.fr (David Auber) bonichon@labri.fr (Nicolas Bonichon) dorbec@labri.fr (Paul Dorbec) claire.pennarun@labri.fr (Claire Pennarun)

1 Introduction

Nowadays, large and dynamic graphs are widely used in the context of Big Data, and their visualization is a classical tool for their analysis. On one hand, when representing dynamic graphs, it is necessary to handle the addition or deletion of nodes or edges easily. On the other hand, when using hierarchical views, the ability to aggregate or de-aggregate sets of nodes is required [1, 11]. When doing such operations, it is important to preserve the mental map of the graph [4], as well as to compute the changes in the representation efficiently, both in order to guarantee a smooth use.

In the following, we define a new graph drawing style on a grid, that we call rook-drawing. In a rook-drawing, we require that the nodes of the graph lie on the intersections of a $(n - 1) \times (n - 1)$ regular grid, in such a way that each row and column hosts exactly one node. Then, the addition or deletion of a node impacts only the row and column it lies on, without interfering with other nodes or other parts of the drawing. In particular, dealing with aggregated data consists in stretching the grid to create enough room for the new appearing nodes (see Fig. 1). These operations clearly preserve orthogonal ordering, which is the first type of mental map defined in [14]. Observe that this technique of having exactly one node per row and per column is also used by Kornaropoulos et al. [12, 13] and by Angelini et al. [3], who represented edges with overlapping orthogonal polylines.

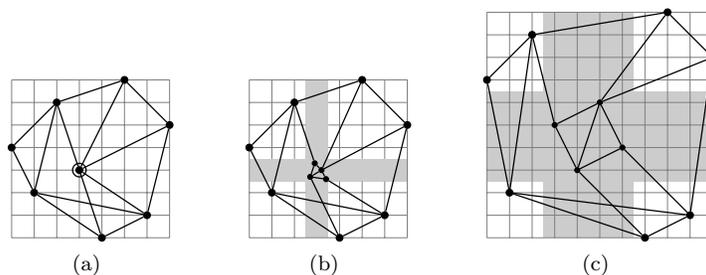


Figure 1: Expansion of an aggregated node in a rook-drawing of a non-planar graph.

Here we explore the existence of rook-drawings for planar graphs. The first question that comes to mind is: *Does every planar graph admit a planar straight-line rook-drawing, i.e. a rook-drawing in which each edge is represented by a segment and no two edges cross?* De Fraysseix et al. showed that every planar graph admits a straight-line drawing on an $(n - 2) \times (2n - 4)$ grid [9]. Schnyder improved this result by proving the existence of such a drawing on an $(n - 2) \times (n - 2)$ -grid [16]. But in such drawings, some columns and rows contain several nodes and some others may be empty. Upward-rightward drawings presented by Di Giacomo et al. [10] are grid drawings in which all nodes have different y -coordinates, but empty rows and several nodes in a same column are allowed.

Contrasting with these results, we show that:

- almost every maximal planar graph admits no planar rook-drawing (see Sect. 3)
- every outerplanar graph admits a planar straight-line rook-drawing computable in linear time (see Sect. 4).
- every planar graph admits a polyline planar rook-drawing, i.e. a planar rook-drawing in which edges are drawn as polylines with bends placed on grid intersections (see Sect. 5). Moreover, this drawing can be computed in linear time, each edge is bent at most once and the total number of bends is at most $n - 3$.

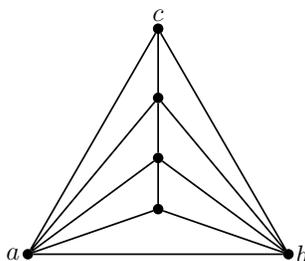
2 Definitions

A *drawing* of a graph G is a mapping of the nodes of G to points on the plane and of the edges of G to curves between their endpoints. The drawing is *straight-line* if the edges are mapped to line segments. It is *polyline* if the edges are series of line segments. A *grid drawing* is a drawing in which the nodes are mapped to intersections of a regular grid. In such a drawing, we use positive coordinates $(x(u), y(u))$ for each node u . A $k \times l$ -grid is a grid of width k and height l . Recall that a *rook-drawing* of a graph with n vertices is a $(n - 1) \times (n - 1)$ -grid drawing, i.e. the functions x and y are bijections from the set of vertices to $\{1, \dots, n\}$. For simplicity, throughout this paper, the term “rook-drawing” denotes a *straight-line rook-drawing* unless otherwise stated.

A *planar graph* is a graph admitting a planar drawing, i.e. a drawing on the plane in which no two edges cross. Such a drawing can be characterized by the collection of circular permutations of the incident edges around each node of the graph, called an *embedding*. A planar graph together with an embedding is called a *plane graph*.

In a plane graph, the edges partition the plane into regions called *faces*. A *rooted plane graph* is a plane graph in which one face (called *outer face*) and one node (called *root*) lying on this face are distinguished. The nodes lying on the outer face are called *outer nodes*, all the other nodes are *inner nodes*. Similarly, *outer edges* are edges belonging to the outer face, the other edges are called *inner edges*. An *outerplane graph* is a rooted plane graph in which every node is on the outer face. A *maximal plane graph* is a plane graph with maximal number of edges, implying that every face is a triangle if the graph has at least three nodes.

A *tree* is a rooted plane graph without cycles. In a tree, a node u is a *descendant* of a node v (or v is an *ancestor* of u) if v is on the path from the root to u . Moreover, if v is connected to u , we say that v is the *parent* of u (and u is a *child* of v). Two nodes are said to be *unrelated* if one is neither ancestor nor descendant of the other. A *leaf* of a tree is a node of the tree without descendants. The *depth* of a tree is the length of the longest path from a leaf to

Figure 2: The tower plane graph \mathcal{T}_6 .

the root in the tree. For a tree T and a node u , the *subtree* of u , denoted $T(u)$, is the tree induced by u and all of its descendants.

The clockwise *preorder* of a tree T is a list of the nodes of T in the order of a clockwise depth-first search algorithm on T . The clockwise *postorder* of a tree T is a list of the nodes of T in the order of their last visit in a clockwise depth-first search algorithm of T . Counterclockwise preorder and postorder are defined similarly.

3 Existence of a planar rook-drawing

We define the *tower plane graph* \mathcal{T}_n of order $n \geq 3$ as the plane join graph $K_2 + P_{n-2}$ (i.e. a complete graph K_2 and a path on $n - 2$ nodes P_{n-2} together with all the edges joining nodes from K_2 to nodes of P_{n-2}) drawn in such a way that the nodes of K_2 are on the outer face (see Fig. 2 for a drawing of \mathcal{T}_6). The tower plane graph can also be seen as a planar 3-tree where the internal nodes induce a path.

Theorem 1 *There exists a unique maximal plane graph on $n \geq 3$ nodes admitting a planar rook-drawing, namely the tower plane graph \mathcal{T}_n .*

Proof: Suppose we have a planar rook-drawing of a maximal plane graph G . We prove that G is the tower plane graph \mathcal{T}_n .

Let a, b, c be the three outer nodes of G . To maintain planarity, the inner nodes are placed at coordinates inside the area defined by the edges (ab) , (bc) and (ca) . Thus the outer nodes must occupy altogether the four borders of the grid, and one of them has to be placed in a corner. Without loss of generality, assume that a occupies the bottom-left corner.

Consider the positions of the two other outer nodes of G . Suppose one of them is in the top-right corner (without loss of generality, say b). If the third node c is placed below the edge (ab) (see Fig. 3a), then the second column on the left can not contain a node: the coordinates $(k, 2)$ are outside the area delimited by the edges (ab) , (bc) and (ca) for all $k > 2$. The point $(2, 2)$ is covered by (ab) and the point $(2, 1)$ can not contain a node because a is already on the first row. If c is above (ab) , then for similar reasons the column left to b can not contain

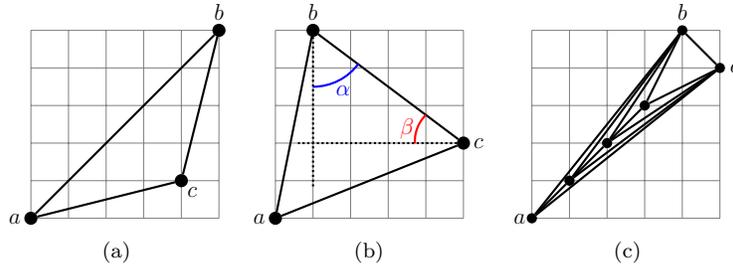


Figure 3: (a) and (b) Illustrations of the proof of Theorem 1. (c) A planar rook-drawing of \mathcal{T}_6 .

a node. Thus b is not in a corner. Without loss of generality, assume b is on the top row and c on the rightmost column of the grid.

Now consider the positions of the inner nodes of G . Let α be the angle between the column containing b and the edge (bc) and β be the angle between the row containing c and the edge (bc) (see Fig.3b). Consider the row just below b : the angle between the edge (ab) and the column containing b is less or equal to 45° thus no nodes can be placed at the left of b on the row below it. No node can be placed on the same column as b either. No node can be placed at the right of the intersection between the edge (bc) and the row below b . Thus for the row under b to contain a node we must have $\alpha \geq 45^\circ$. With similar arguments, for the column on the left of c to contain a node, we infer $\beta \geq 45^\circ$. Since $\alpha + \beta = 90^\circ$, we infer $\alpha = \beta = 45^\circ$. Suppose c is not the node placed on the row below b . Then the edge (bc) would prevent this row to be occupied. Thus, c is the node placed on the row below b and b is placed on the column left to c , i.e. $x(b) = y(c) = n - 1$. Finally, the inner nodes must be placed on coordinates (i, i) for $2 \leq i \leq n - 2$, i.e. along a diagonal of the grid (see Fig. 3c).

Now the positions of the nodes are determined and there is only one way to complete the drawing into a maximal plane graph, forming the graph \mathcal{T}_n . \square

4 Planar rook-drawing for outerplane graphs

In this section, we prove the following theorem:

Theorem 2 *Every outerplane graph admits a planar rook-drawing. This drawing can be computed in linear time.*

To prove Theorem 2, we use a partition of the edges of an outerplane graph introduced in [5]:

Theorem 3 (Bonichon, Gavaille and Hanusse [5]) *Let G be an outerplane graph rooted in r . There exists a unique partition of the edges of G into two sets T and S such that:*

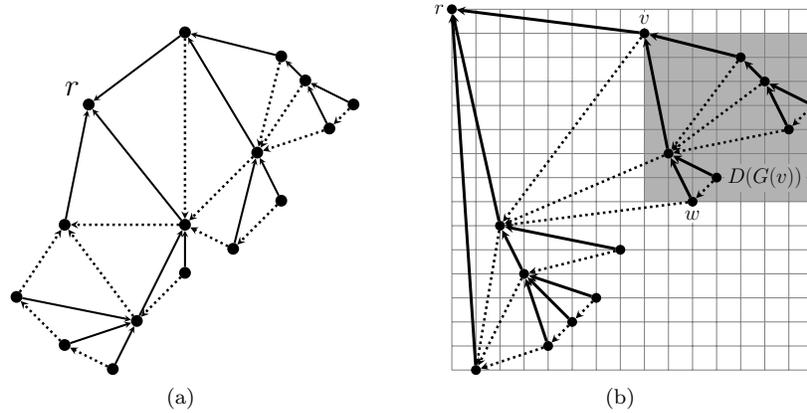


Figure 4: (a) The decomposition of an outerplane graph G rooted in r into T (solid edges) and S (dotted edges). (b) The rook-drawing of G produced by applying Lemma 1. The subtree of T rooted in v is drawn in the gray area $\mathcal{D}(G(v))$. The left branch of v is the path between v and w .

- T is a spanning tree rooted in r
- edges of S join a node u to the first unrelated node after u in the clockwise preorder of T .

Such a partition can be computed in linear time.

Remark that the clockwise preorder of T is exactly the reverse order of the counterclockwise postorder of T . We denote by $y(v)$ the index of v in a counterclockwise postorder of T . We consider an orientation of the edges of T and S such that all edges of T are oriented towards the root r and the edges (uv) of S are oriented from u to v if $y(u) > y(v)$.

The tree T can be computed by Algorithm 1 presented in [5]. A call $\text{Traversal}(G, \emptyset, r)$ returns the tree T of G rooted in r , the second parameter stands for the current set of edges of the tree during the execution.

Algorithm 1: $\text{Traversal}(G, T, u)$

```

begin
   $C \leftarrow \{(uv) \in G \mid v \notin T\}$ 
   $T \leftarrow T \cup C$ 
  for all edges  $(uv) \in C$  taken in the clockwise order around  $u$  do
     $T \leftarrow \text{Traversal}(H, T, v)$ 
  return  $T$ 

```

For each node v of the outerplane graph G , we denote $x(v)$ its index in counterclockwise preorder of T . Recall that $y(v)$ is its index in counterclockwise postorder of T .

Lemma 1 *Placing each node v of G at coordinates $(x(v), y(v))$ produces a planar rook-drawing of G .*

Proof: By construction, the drawing $\mathcal{D}(G)$ obtained is a rook-drawing. It remains to show that this drawing is planar.

For a node v of G , let T_v be the subtree of T rooted in v . Let $G(v)$ be the subgraph of G induced by the nodes of T_v . Let $\mathcal{D}(G(v))$ be the drawing induced by the edges and nodes of $G(v)$. The *left branch* of v in T_v denotes the path between v and the first leaf found in a counterclockwise postorder of T_v (see Fig.4).

Let u and v be two nodes of G . The following observations are direct consequences of the definition of the x - and y -coordinates:

- (i) If (uv) is an edge of S from u to v , then v is the first unrelated node of T after u in clockwise preorder of T , and thus the first node unrelated to u with $y(v) < y(u)$.
- (ii) If u is parent of v in T , then $x(u) < x(v)$ and $y(u) > y(v)$.
- (iii) Let (uv) be an edge of S with $y(u) > y(v)$. Then v is before u in counterclockwise postorder of T . Since v and u are unrelated, v is also before u in counterclockwise preorder of T , i.e. $x(v) < x(u)$. Thus the edges of S are going down and to the left.

We now want to prove by induction the following proposition : $\mathcal{D}(G(u))$ is planar and drawn in the subgrid $[x(u), x(u) + |T_u| - 1] \times [y(u) - |T_u| + 1, y(u)]$.

When T_u is reduced to a single node, the proposition clearly holds.

Now assume the proposition holds for nodes having a subtree of depth at most k . Let u be a node with a subtree T_u of depth $k + 1$. Denote by u_1, \dots, u_m the children of u in clockwise order. Their subtrees in T are denoted T_{u_1}, \dots, T_{u_m} . By induction hypothesis, the subtrees T_{u_1}, \dots, T_{u_m} are placed in disjoint areas (see Fig. 4). Then $\mathcal{D}(G(u_i))$ and $\mathcal{D}(G(u_j))$ with $i \neq j$ do not intersect. Thus T_u is planarly drawn in the sub-grid $[x(u), x(u) + |T_u| - 1] \times [y(u) - |T_u| + 1, y(u)]$.

We now prove that the edges of S joining nodes belonging to different subtrees do not create any crossing in $\mathcal{D}(G(u))$. Let v and w be nodes from different subtrees linked by an edge of S from v to w . By remark (i), w is the first node unrelated to v with $y(w) < y(v)$. So v and w are in consecutive trees, say T_{u_i} and $T_{u_{i+1}}$, and $w = u_{i+1}$. Thus all edges of S joining T_i to T_{i+1} have u_{i+1} as an end: edges of S join nodes of the left branch of u_i to u_{i+1} . Then by remarks (ii) and (iii), the edges of S can not cross each other or edges of the tree T .

Thus $\mathcal{D}(G(u))$ is planar and this concludes the proof. \square

Remark that as Andrews [2] showed, a strictly convex grid drawing of a cycle of n nodes requires area $\Omega(n^3)$. Since a rook-drawing uses area $O(n^2)$, it's not possible to produce strictly convex rook-drawing for the cycle for large n .

Also note that the existence of n nodes both in rook position and in general position, i.e. such that no three nodes are colinear (see definition in [15]), would imply an algorithm for generating a rook-drawing of outerplane graphs (from [8]). We do not know how to prove whether such a configuration exists. Remark though that the algorithm in [8] has complexity $O(n \log^3(n))$, while the algorithm presented here is linear.

5 Polyline rook-drawing for planar graphs

As we proved that some plane graphs do not admit a planar rook-drawing with straight lines, we now relax the straight-line constraint and look at planar polyline rook-drawings. We first recall the definition of Schnyder woods.

5.1 Properties of Schnyder woods

Definition 1 (Schnyder [16]) *A Schnyder wood of a maximal plane graph G is a partition of the inner edges of G into three directed trees T_0, T_1, T_2 with the following properties:*

- each tree T_i is rooted on a distinct outer node v_i ;
- the edges of each tree are directed toward the root;
- each inner node u of G has one parent in each T_i , denoted $P_i(u)$;
- in counterclockwise order around each inner node, the outgoing edges are in T_0 then T_1 then T_2 ;
- each ingoing edge belonging to the tree T_i is placed after the outgoing edge in $T_{i+1 \bmod 3}$ and before the outgoing edge in $T_{i-1 \bmod 3}$ in counterclockwise order around an inner node.

Let \bar{T}_i be the tree T_i plus the edges (v_{i-1}, v_i) and (v_{i+1}, v_i) . The orientation of edges around an inner node is shown in Fig. 5, where \bar{T}_0 is drawn solid, T_1 is dotted, and T_2 is dotted-dashed. Throughout the paper, we call a 0-edge (respectively 1-edge, 2-edge) an edge belonging to the tree T_0 (resp. T_1, T_2).

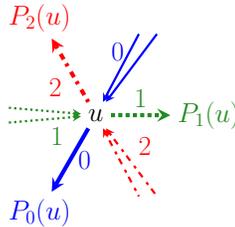


Figure 5: Orientation around an inner node u in a Schnyder wood.

Let us first recall a useful property of Schnyder woods:

Lemma 2 ([6, Lemma 4]) *For every inner node u , $P_{i-1}(u)$ and $P_{i+1}(u)$ are unrelated to u in \bar{T}_i . Moreover $P_{i-1}(u)$ (resp. $P_{i+1}(u)$) appears before (resp. after) u in the clockwise preorder and postorder of \bar{T}_i .*

5.2 Polyline rook-drawing algorithm

We here describe an algorithm to produce a planar polyline rook-drawing of a maximal plane graph of order n . The algorithm is inspired by an algorithm for polyline drawings proposed in [7]. The original algorithm was designed to minimize the grid size and thus many rows and columns support several nodes. This new algorithm shares with the former the edge bending strategy, but the node placement is different.

Theorem 4 *Every maximal plane graph G with n nodes admits a polyline planar rook-drawing $\mathcal{D}(G)$, which can be computed with Algorithm 2 in linear time. This drawing has $n - 3$ bends.*

In Algorithm 2 and later, $ll_0(u)$ denotes the last descendant met in a clockwise preorder of u in T_0 .

Algorithm 2: Planar polyline rook-drawing for a maximal plane graph G

```

 $(T_0, T_1, T_2) \leftarrow$  Schnyder wood of  $G$ 
column order  $C \leftarrow$  clockwise preorder of  $\bar{T}_0$ 
row order  $R \leftarrow$  clockwise postorder of  $\bar{T}_1$ 
for  $u$  node of  $G$  do
    |  $(x(u), y(u)) = (C(u), R(u))$ 
for  $u$  inner node of  $G$  do
    | if  $x(u) \neq x(P_0(u)) + 1$  then
    | | Bend edge  $(uP_0(u))$  at  $(x(u), y(P_0(u)) + 1)$ 
    | | if  $u \neq ll_0(u)$  then
    | | | Bend edge  $(uP_1(u))$  at  $(ll_0(u), y(u))$ 
    Bend  $e = (v_1v_0)$  at  $(x(v_1), 1)$ 
    
```

An example of the result of Algorithm 2 on a maximal plane graph is presented in Fig. 6b.

We first make the following observations on the placement of nodes after applying Algorithm 2:

- Since the nodes are placed according to their position in a preorder and a postorder, each row and column contains exactly one node. Thus $\mathcal{D}(G)$ is a rook-drawing.
- When u is a leaf of T_0 , then $ll_0(u) = u$ and this is the only case when the edge from u to $P_1(u)$ is drawn straight.

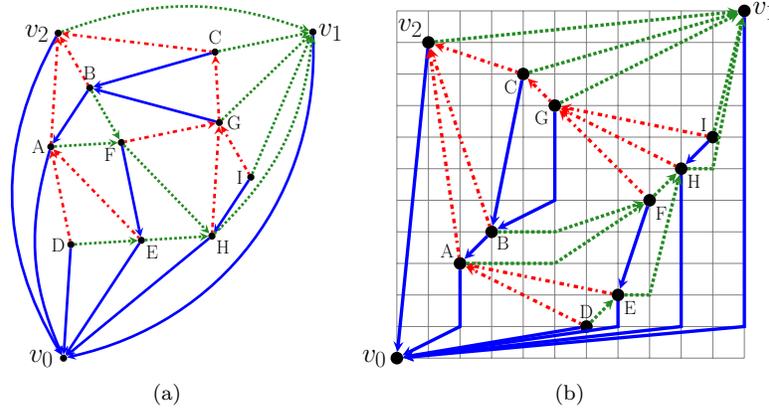


Figure 6: (a) Schnyder wood of a maximal plane graph G . (b) The polyline drawing $\mathcal{D}(G)$ obtained by Algorithm 2.

5.2.1 Number of bends.

Let k be the number of leaves in T_0 . By construction, T_0 , T_1 and T_2 contain each $n - 3$ edges and $n - 1$ nodes. The edges of T_0 are all bent, except one for each non-leaf node in T_0 . Thus $n - 3 - (n - 1 - k)$ 0-edges are bent. The edges of T_1 are all bent, except $k - 1$ (v_2 is a leaf in T_0 but is not a node of T_1). Thus $n - 3 - (k - 1)$ 1-edges are bent. Finally, the edges of T_2 are never bent. The edges (v_0v_2) and (v_2v_1) are not bent, and the edge (v_0v_1) is bent. Thus there are exactly $n - 3 - (n - 1 - k) + (n - 3 - (k - 1)) + 1 = n - 3$ bends in the drawing of G .

5.2.2 Planarity.

We first describe in the following some structural properties of the drawing with Lemmas 3 and 4.

Lemma 3 *In $\mathcal{D}(G)$, for each inner node u :*

- $x(P_0(u)) < x(u)$ and $y(P_0(u)) < y(u)$: $P_0(u)$ is left and below u .
- $x(P_1(u)) > x(u)$ and $y(P_1(u)) > y(u)$: $P_1(u)$ is right and above u .
- $x(P_2(u)) < x(u)$ and $y(P_2(u)) > y(u)$: $P_2(u)$ is left and above u .

Proof: Recall that the x -coordinates are given by a clockwise preorder of T_0 and that the y -coordinates are given by a clockwise postorder of T_1 . Hence we have that $x(P_0(u)) < x(u)$ and $y(P_1(u)) > y(u)$. Applying Lemma 2 with $i = 1$ we get that $P_0(u), u$ and $P_2(u)$ appear in that order in the clockwise postorder of T_1 . Hence $y(P_0(u)) < y(u)$ and $y(P_2(u)) > y(u)$. Applying again Lemma 2 but this time with $i = 0$ we get that $P_2(u), u$ and $P_1(u)$ appear in

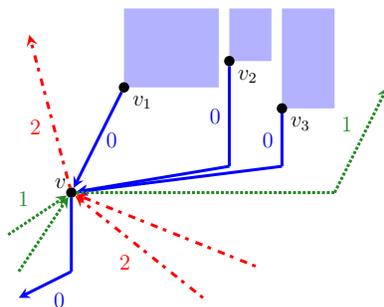


Figure 7: Edges orientation around an inner node v in $\mathcal{D}(G)$. The area at the top-right hand side of v_i is the area in which the subtree of v_i in T_0 is drawn.

that order in the clockwise preorder of T_0 . Hence we have that $x(P_2(u)) < x(u)$ and $x(P_1(u)) > x(u)$. \square

From Lemma 3 and the coordinates of bends chosen for the edges in Algorithm 2, we observe that the configuration around an inner node follows the scheme illustrated in Fig. 7.

Lemma 4 *For every inner node u we have the following statements:*

- (i) every node v such that $x(P_0(u)) < x(v) < x(u)$ is a descendant of $P_0(u)$ in T_0 .
- (ii) every node v such that $x(u) < x(v) < x(P_1(u))$ is either a descendant of u in T_0 or $y(v) < y(u)$ in $\mathcal{D}(G)$.
- (iii) every node w such that $x(P_2(u)) < x(w) < x(u)$ is either a descendant of $P_2(u)$ in T_0 or $y(w) < y(u)$ in $\mathcal{D}(G)$.

Proof: The first property is a direct consequence of the fact that the x -coordinates are given by the clockwise preorder of T_0 .

Let v be a node such that $x(u) < x(v) < x(P_1(u))$. By definition of $x(v)$, either v is a descendant of u in T_0 , or it is in the area \mathcal{A} delimited in the original graph G by the T_0 -paths (u, v_0) and $(P_1(u), v_0)$, and the edge $(uP_1(u))$ in T_1 (see Fig. 8a). Note that v may belong to the path $(P_1(u), v_0)$. We consider the path P in T_2 from v to v_2 . Considering the relative orientations of the edges in the Schnyder wood, since v_2 is out of \mathcal{A} , P has to intersect the border (u, v_0) of \mathcal{A} to reach v_2 (even if v belongs to the T_0 -path $(P_1(u), v_0)$, its outgoing edge in T_2 heads into \mathcal{A}). Let t denote the node at the intersection. There is a path in T_2 from v to t and a path in T_0 from u to t . By Lemma 3, $y(v) < y(t)$ and $y(t) < y(u)$, and thus $y(v) < y(u)$.

A similar argument, exchanging the roles of T_1 and T_2 (see Fig. 8b), proves the third statement. \square

We now prove that the edges drawn do not cross. Lemmas 5 and 6 state such for the edges inside each tree \bar{T}_0 and T_1 . Then we prove that edges from

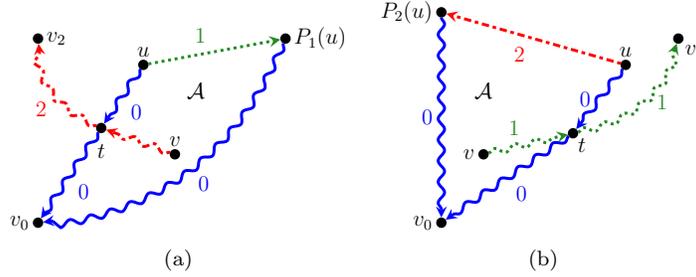


Figure 8: Illustrations of the proof of Lemma 4.(ii) and Lemma 4.(iii).

different trees do not cross with Lemmas 7, 8 and 9. After that, we show that edges from T_2 do not cross pairwise with Lemma 10. Finally, drawing the edge (v_1v_2) with a straight line does not create any crossing, since all nodes and bends are placed on the same side of the straight-line defined by the nodes v_1 and v_2 .

For each node u of G , recall that we denote $ll_0(u)$ the last descendant met in the clockwise preorder of u in \bar{T}_0 .

Lemma 5 *The edges of \bar{T}_0 do not cross in $\mathcal{D}(G)$.*

Proof: We prove by induction on the depth k of subtrees in \bar{T}_0 the following proposition: in $\mathcal{D}(G)$, the subtree of a node u in \bar{T}_0 , denoted $\bar{T}_0(u)$, is drawn planarly in the subgrid $[x(u), x(u) + |\bar{T}_0(u)| - 1] \times [y(u), n]$ (see Fig. 9).

The proposition clearly holds when $k = 0$.

Assume the proposition holds for subtrees of depth at most k in \bar{T}_0 . Let u be a node of G such that $\bar{T}_0(u)$ is of depth $k+1$. Let u_1, \dots, u_m be the children of u (taken in the clockwise order). Nodes of $\bar{T}_0(u)$ are consecutive in the clockwise preorder of \bar{T}_0 and by Lemma 3, $y(v) > y(u)$ for every node v of $\bar{T}_0(u)$. Thus the drawing of the union of subtrees $\bar{T}_0(u_\ell)$ lies in $[x(u), x(u) + |\bar{T}_0(u)| - 1] \times [y(u), n]$.

By the induction hypothesis the edges of these subtrees are drawn in disjoint areas, hence the drawing of the union of the subtrees is planar. Moreover, by construction the edges joining u to u_1, \dots, u_m do not cross each other and lie in a region free of edges from the subtrees (each node u_ℓ lies in the bottom row of its dedicated area). Hence the drawing of $\bar{T}_0(u)$ is planar. By construction of the bends for the edges joining u to u_1, \dots, u_m , $\bar{T}_0(u)$ is also drawn in the subgrid $[x(u), x(u) + |\bar{T}_0(u)| - 1] \times [y(u), n]$. \square

Lemma 6 *The edges of T_1 do not cross in $\mathcal{D}(G)$.*

Proof: We prove by induction on the depth k of subtrees in T_1 the following proposition: in $\mathcal{D}(G)$, the subtree of a node u in T_1 , denoted $T_1(u)$, is drawn planarly in the subgrid $[1, x(u)] \times [y(u) - |T_1(u)| + 1, y(u)]$ (see Fig. 10).

The proposition clearly holds when $k = 0$.

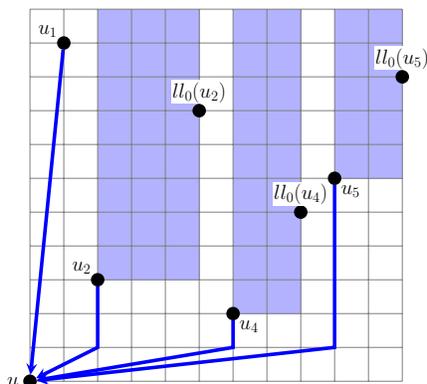


Figure 9: Illustration of the proof of Lemma 5. The area at the top-right hand side of u_i is the area in which the subtree of u_i in \overline{T}_0 is drawn.

Assume the proposition holds for subtrees of depth at most k in T_1 . Let u be a node of G such that $T_1(u)$ is of depth $k+1$. Let u_1, \dots, u_m be the children of u (taken in the clockwise order). Nodes of $T_1(u)$ are consecutive in the clockwise postorder of T_1 and by Lemma 3, $x(v) < x(u)$ for every node v of $T_1(u)$. Thus the drawing of the union of subtrees $T_1(u_i)$ lies in $[1, x(u)] \times [y(u) - |T_1(u)| + 1, y(u)]$.

By the induction hypothesis, the edges of these subtrees are drawn in disjoint areas, hence the drawing of their union is planar.

Let now prove that the edges joining u and u_1, \dots, u_m do not cross each other. Recall that $(u_\ell u)$ is bent at coordinates $(x(ll_0(u_\ell)), y(u_\ell))$. Let u_ℓ and $u_{\ell+1}$ be two consecutive children of u . Let us show that the bends (uu_ℓ) and $(uu_{\ell+1})$ are placed at positions with increasing y -coordinates and decreasing x -coordinates. By Lemma 3, $y(u_\ell) < y(u_{\ell+1})$, thus the bends have increasing y -coordinates. We now prove that $x(ll_0(u_{\ell+1})) \leq x(ll_0(u_\ell))$. By definition of the x -coordinates, all descendants of an inner node t in T_0 are placed on consecutive columns at the right of t . Since $y(u_\ell) < y(u_{\ell+1})$, by contrapositive of Lemma 4.(ii), either $u_{\ell+1}$ is a descendant of u_ℓ in T_0 or $x(u_{\ell+1}) < x(u_\ell)$. Suppose $u_{\ell+1}$ is a descendant of u_ℓ in T_0 . By Lemma 3 and the definition of the x -coordinates, $x(ll_0(u_{\ell+1})) \leq x(ll_0(u_\ell))$. Suppose now $x(u_{\ell+1}) < x(u_\ell)$. Thus all descendants of $u_{\ell+1}$ in T_0 have abscissas between $x(u_{\ell+1})$ and $x(u_\ell)$, hence $x(ll_0(u_{\ell+1})) \leq x(ll_0(u_\ell))$.

Then the bends for the edges (uu_ℓ) are placed at positions with increasing y -coordinates and decreasing x -coordinates. Thus the edges joining u and u_1, \dots, u_m do not cross each other. These edges can not cross edges of a subtree of u_ℓ in T_0 either, as each node u_ℓ lies in the bottom row of the area dedicated to the drawing of $\overline{T}_0(u)$ and the bent of the edge $(u_\ell u)$ lies in the right-most column of this area. Hence the drawing of $T_1(u)$ is planar. \square

Lemma 7 *The edges of \overline{T}_0 and T_1 do not cross in $\mathcal{D}(G)$.*

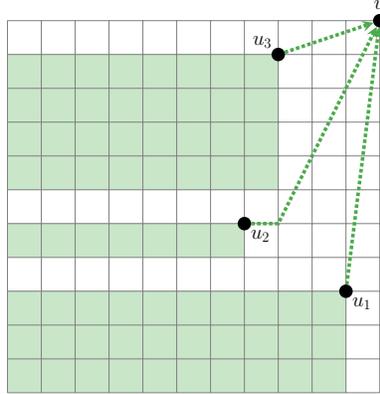


Figure 10: Illustration of the proof of Lemma 6. The area at the bottom-left hand side of u_i is the area in which the subtree of u_i in T_1 is drawn.

Proof: Let (wt) be a 1-edge. By Lemma 3, $x(w) < x(t)$ and $y(w) < y(t)$. Let (uw) be an edge of \overline{T}_0 . If $x(w) > x(u)$ or $y(w) > y(u)$, then the edges (wt) and (uw) can not cross each other. Suppose then that $x(w) < x(u)$ and $y(w) < y(u)$.

Suppose first that $x(t) > x(u)$. Since $x(w) < x(u) < x(t)$ and $y(w) < y(u)$, by Lemma 4.(ii), u is a descendant of w in \overline{T}_0 . Thus v is also a descendant of w in \overline{T}_0 . But then the bend of (wt) avoids any crossing.

Suppose then that $x(t) < x(u)$. If $y(v) < y(w)$, then the bend of uv avoids any crossing. Thus assume $y(v) > y(w)$. By Lemma 4.(i), if $x(v) < x(w)$, then w is a descendant of v in T_0 , contradicting $y(v) > y(w)$. So $x(w) < x(v) < x(t)$ and by Lemma 4.(ii), v is a descendant of w in \overline{T}_0 , and so is u . By definition of the x -coordinates, all descendants of w in \overline{T}_0 are consecutive to w and before t in the x -order, a contradiction with $x(t) < x(u)$. \square

Lemma 8 *The edges of \overline{T}_0 and T_2 do not cross in $\mathcal{D}(G)$.*

Proof: Let (wt) be a 2-edge. By Lemma 3, $x(t) < x(w)$ and $y(t) > y(w)$. Let (uw) be an edge of \overline{T}_0 . If $x(w) < x(v)$ or $y(v) > y(t)$, then the edges (uw) and (wt) do not cross each other. Similarly, if $x(u) < x(t)$ or $y(u) < y(w)$, the edges do not cross either. So assume that $x(v) < x(w)$, $y(v) < y(t)$, $x(t) < x(u)$ and $y(w) < y(u)$.

If $x(u) < x(w)$, by Lemma 4.(iii), u is a descendant of t in \overline{T}_0 . Since v is the parent of u in \overline{T}_0 , v is also a descendant of t in \overline{T}_0 . Hence by Lemma 3, $y(v) > y(t)$; a contradiction.

If $x(u) > x(w)$, then $x(v) < x(w) < x(u)$ and by Lemma 4.(i), w is a descendant of v in \overline{T}_0 . Thus (uw) avoids w and there is no crossing. \square

Lemma 9 *The edges of T_1 and T_2 do not cross in $\mathcal{D}(G)$.*

Proof: Let (uv) be a 2-edge and (wt) a 1-edge. By Lemma 3, $x(u) > x(v)$, $y(u) < y(v)$, $x(t) > x(w)$ and $y(t) > y(w)$. Assume $x(t) > x(v)$, $y(t) > y(u)$, $x(w) < x(u)$ and $y(w) < y(v)$, or (uv) and (wt) may not cross.

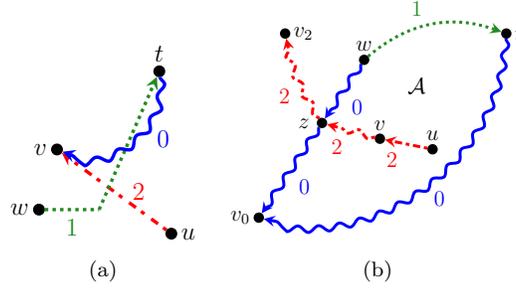


Figure 11: Illustration of the two cases for the proof of Lemma 9.

If $x(t) < x(u)$ (see Fig. 11a), then $x(v) < x(t) < x(u)$. Since $y(t) > y(u)$, by the contrapositive of Lemma 4.(iii), t is a descendant of v in \overline{T}_0 . Then the orientation of the edges around t implies that either (uv) and (wt) do not cross, or (wt) also cross the T_0 -path (t, v) , contradicting Lemma 7.

If $x(u) < x(t)$, then $x(w) < x(u) < x(t)$. u is not a descendant of w in \overline{T}_0 (otherwise the bend of (wt) would prevent the crossing). Thus by definition of $x(u)$, u is in the area \mathcal{A} delimited in the original graph G by the paths (w, v_0) and (t, v_0) in \overline{T}_0 , and the edge (wt) in T_1 (see Fig.11b). Note that u may belong to the path (t, v_0) in \overline{T}_0 . We consider the path \mathcal{P} in T_2 from u to v_2 . Considering the relative orientations of the edges in the Schnyder wood, since v_2 is out of \mathcal{A} , then \mathcal{P} has to intersect the border (w, v_0) of \mathcal{A} to reach v_2 (even if u is on the path (t, v_0) , its outgoing edge in T_2 heads into \mathcal{A}). Let z denote the intersection. By Lemma 3, $y(z) < y(w)$. z is a parent of u in T_2 and thus a parent of v in T_2 (we may have $z = v$). Thus by Lemma 3, $y(z) \geq y(v)$ and thus $y(w) > y(v)$, which is a contradiction. \square

Lemma 10 *The edges of T_2 do not cross in $\mathcal{D}(G)$.*

Proof: Let (uv) and (wt) be edges of T_2 . Suppose they cross each other. Thus $x(w) > x(v)$, $y(w) < y(v)$, $x(t) < x(u)$ and $y(t) > y(u)$. Since the two edges play similar roles, we can suppose without loss of generality that $x(t) > x(v)$. Then $x(v) < x(t) < x(u)$ and $y(t) > y(u)$, and by Lemma 4.(iii), t is a descendant of v in \overline{T}_0 . But then the edge (wt) would cross a 1-edge in the path of T_1 joining v to v_1 , contradicting Lemma 9. \square

6 Conclusion

In this paper, we observed that no maximal planar graph but the tower graphs admit a planar straight-line rook-drawing. On the other hand we showed that every outerplane graph admits a planar straight-line rook-drawing. A natural question is: are there other classes of plane graphs that all admit a planar

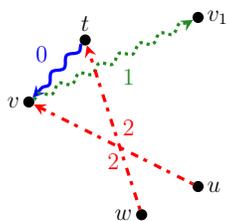


Figure 12: Illustration for the proof of Lemma 10.

straight-line rook-drawing? A plane graph that has a triangular outer face and admits a planar straight-line rook-drawing is necessarily a subgraph of the tower plane graph we described earlier. However, if we consider plane graphs with an outer face with at least 4 vertices, it seems that many of them should admit such a drawing. Then, plane graphs that do not contain non-facial triangles, as, for instance, quadrangulations or 4-connected triangulations with outer face of degree at least 4, are possibly good candidates for admitting a planar rook-drawing.

We also showed that every plane graph admits a planar polyline rook-drawing with at most $n - 3$ bent edges. Even if this number of bends is reasonable, one could ask if a linear number of bends is needed for allowing a planar rook-drawing of any planar graph.

Another interesting question would be to consider *relaxed rook-drawing* in which each row and column contains at most one node (and no longer exactly one node). Clearly every plane graph admits a planar relaxed rook-drawing: it suffices to consider a straight-line planar drawing of the plane graph and add a tiny perturbation to nodes sharing some coordinates. This naive approach produces drawings with a huge number of empty columns and rows, which is not suitable in practice. Hence the good question would be: does every plane graph admit a planar relaxed rook-drawing with a small (i.e. linear or sub-linear) number of empty rows and columns? There are no evidence yet that even a constant number of empty rows and columns would not suffice.

References

- [1] J. Abello, F. Van Ham, and N. Krishnan. ASK-GraphView: A large scale graph visualization system. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):669–676, 2006. doi:10.1109/TVCG.2006.120.
- [2] G. E. Andrews. A lower bound for the volume of strictly convex bodies with many boundary lattice points. *Transactions of the American Mathematical Society*, pages 270–279, 1963. doi:10.2307/1993769.
- [3] P. Angelini, G. Da Lozzo, M. Di Bartolomeo, V. Di Donato, M. Patrignani, V. Roselli, and I. G. Tollis. L-drawings of directed graphs. In *SOFSEM 2016: Theory and Practice of Computer Science*, pages 134–147. Springer, 2016. doi:10.1007/978-3-662-49192-8_11.
- [4] D. Archambault and H. C. Purchase. Mental map preservation helps user orientation in dynamic graphs. In *Graph Drawing*, pages 475–486. Springer, 2013. doi:10.1007/978-3-642-36763-2_42.
- [5] N. Bonichon, C. Gavoille, and N. Hanusse. Canonical decomposition of outerplanar maps and application to enumeration, coding, and generation. *Journal of Graph Algorithms and Applications*, 9(2):185–204, 2005. doi:10.1007/978-3-540-39890-5_8.
- [6] N. Bonichon, C. Gavoille, and N. Hanusse. An information-theoretic upper bound on planar graphs using well-orderly maps. In *Towards an Information Theory of Complex Networks*, pages 17–46. Springer, 2011. doi:10.1007/978-0-8176-4904-3_2.
- [7] N. Bonichon, B. Le Saëc, and M. Mosbah. Optimal area algorithm for planar polyline drawings. In *Graph-Theoretic Concepts in Computer Science*, pages 35–46. Springer, 2002. doi:10.1007/3-540-36379-3_4.
- [8] P. Bose. On embedding an outerplanar graph in a point set. *Computational Geometry*, 23(3):303–312, 2002. doi:10.1016/S0925-7721(01)00069-4.
- [9] H. de Fraysseix, J. Pach, and R. Pollack. Small sets supporting Fáry embeddings of planar graphs. In *Proceedings of the twentieth annual ACM symposium on Theory of computing*, pages 426–433. ACM, 1988. doi:10.1145/62212.62254.
- [10] E. Di Giacomo, W. Didimo, M. Kaufmann, G. Liotta, and F. Montecchiani. Upward-rightward planar drawings. In *Information, Intelligence, Systems and Applications, IISA 2014, The 5th International Conference on*, pages 145–150. IEEE, 2014. doi:10.1109/IISA.2014.6878792.
- [11] P. Eades and Q.-W. Feng. Multilevel visualization of clustered graphs. In *Graph Drawing*, pages 101–112. Springer, 1997. doi:10.1007/3-540-62495-3_41.

- [12] E. M. Kornaropoulos and I. G. Tollis. Overloaded orthogonal drawings. In *Graph Drawing*, pages 242–253. Springer, 2012. doi:10.1007/978-3-642-25878-7_24.
- [13] E. M. Kornaropoulos and I. G. Tollis. DAGView: an approach for visualizing large graphs. In *Graph Drawing*, pages 499–510. Springer, 2013. doi:10.1007/978-3-642-36763-2_44.
- [14] K. Misue, P. Eades, W. Lai, and K. Sugiyama. Layout adjustment and the mental map. *Journal of visual languages and computing*, 6(2):183–210, 1995. doi:10.1006/jvlc.1995.1010.
- [15] J. Pach, P. Gritzmann, B. Mohar, and R. Pollack. Embedding a planar triangulation with vertices at specified points. *American Mathematical Monthly*, 98:165–166, 1991. doi:10.2307/2323956.
- [16] W. Schnyder. Embedding planar graphs on the grid. In *Symposium on Discrete Algorithms*, volume 90, pages 138–148, 1990.