Rook-Drawing for Plane Graphs

David Auber[1,2], Nicolas Bonichon[1,2], Paul Dorbec[1,2],
and Claire Pennarun[1,2](✉)

[1] University of Bordeaux, LaBRI, UMR 5800, 33400 Talence, France
{david.auber,nicolas.bonichon,paul.dorbec,claire.pennarun}@labri.fr
[2] CNRS, LaBRI, UMR 5800, 33400 Talence, France

Abstract. Motivated by visualization of large graphs, 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.

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 the one hand, when representing dynamic graphs, it is necessary to handle easily the addition or deletion of nodes or edges. On the other hand, when using hierarchical views, the ability to aggregate or de-aggregate sets of nodes is required [1,8]. When doing such operations, it is important to preserve the mental map of the graph [3], 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 particular type of graph drawing 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 [12]. Observe that this technique of

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. Di Giacomo and A. Lubiw (Eds.): GD 2015, LNCS 9411, pp. 180–191, 2015.
DOI: 10.1007/978-3-319-27261-0_15

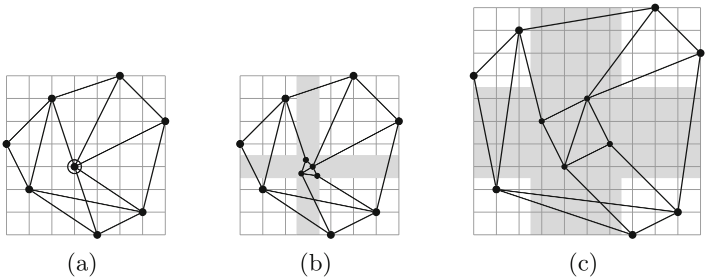

Fig. 1. Expansion of an aggregated node in a rook-drawing of a non-planar graph.

having exactly one node per row and per column is also used by Kornaropoulos et al. [10,11], who represented edges with overlapping orthogonal polylines.

We here 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 [14]. 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. [7] 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 in Sect. 4 that almost every maximal planar graph admits no planar rook-drawing. Yet, every outerplanar graph admits a planar straight-line rook-drawing computable in linear time, as shown in Sect. 3. We then consider *polyline planar rook-drawings*, in which edges are drawn as polylines with bends placed on grid intersections. We show in Sect. 5) that every planar graph admits a polyline planar rook-drawing. 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 of 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 $\{1, \ldots, n\}$. For simplicity, throughout this paper, the term "rook-drawing" denotes a *straight-line rook-drawing* unless otherwise precised.

A *planar graph* is a graph admitting a planar drawing, i.e. a drawing on the plane in which no pair of edges crosses. Such a drawing can be characterized by the collection of circular permutations of incident edges around each node, called *embedding*. A connected 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 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 there are 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 *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 the root in the tree. For a tree T a node u, the *subtree* of u, denoted $T(u)$, is the tree induced on 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 Planar Rook-Drawing for Outerplane Graphs

In this section, we prove the following theorem:

Theorem 1. *Every outerplane graph admits a planar rook-drawing. This drawing can be computed in linear time.*

To prove Theorem 1, we use a partition of the edges of outerplane graphs introduced by Bonichon et al. [4]:

Theorem 2 ([4])**.** *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:*

- *T is a tree rooted in r*
- *edges of S join a node u to the first node after u in the counterclockwise postorder of T.*

Such a partition can be computed in linear time.

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 $y(u) > y(v)$. If G is maximal, then S is a tree rooted in w with $y(w) = 0$ that does not contain the root r of G.

The tree T can be computed by Algorithm 1 due to Bonichon et al. [4]. A call $\mathsf{Traversal}(G, \varnothing, r)$ 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. $\mathsf{Traversal}(G, T, u)$

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

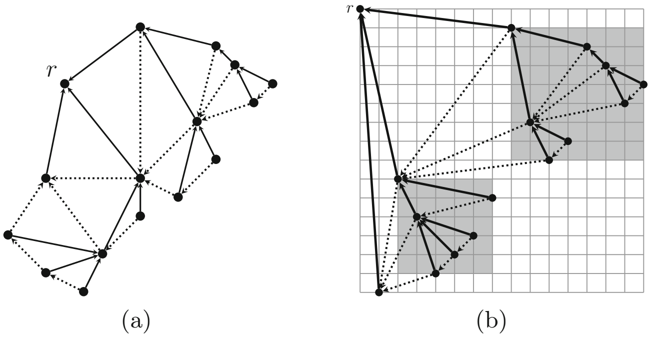

Fig. 2. (a) The decomposition of an outerplane graph G rooted in r into T (solid edges) and S (dotted edges). (b) A rook-drawing of G showing the induction process of the proof for Lemma 1.

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 v a node 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.

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 u is before v in counterclockwise preorder of T (i.e. $x(u) < x(v)$) and they are unrelated, then $y(u) < y(v)$ and for each descendant w of u in T, $x(w) < x(v)$ and $y(w) < y(v)$.
(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 preorder of T (i.e. $x(v) < x(u)$) and as they are unrelated, v is also before u in counterclockwise postorder of T (i.e. $y(v) < y(u)$). Thus the edges of S are going down and to the left.
(iv) The coordinates of the nodes of the left branch of v are x-increasing and y-decreasing.

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 $u_1, ..., u_m$ the children of u in clockwise order. Their subtrees in T are denoted $T_{u_1}, ..., T_{u_m}$. By induction hypothesis, the subtrees $T_{u_1}, ..., T_{u_m}$ are placed in disjoint areas (see Fig. 2). 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, and such that $x(w) < x(v)$. Recall that by definition of S, 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 (iii) and (iv), the edges of S can not cross each other or edges of the tree T.

Thus $\mathcal{D}(G(u))$ is planar. This concludes the proof. □

Remark that as Andrews [2] showed that a strictly convex drawing of a cycle of n nodes with integer coordinates requires area $\Omega(n^3)$, whereas a rook-drawing requires area $\Omega(n^2)$, our algorithm can not produce strictly convex drawings for outerplane graphs 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 [13]) would imply an algorithm for generating a rook drawing of outerplane graphs (from [6]). We do not know how to prove whether such a configuration exists. Remark though that the algorithm in [6] is of complexity $n \log^3(n)$, while the algorithm presented here is linear.

4 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. 3 for a drawing of $\mathcal{T}_6$).

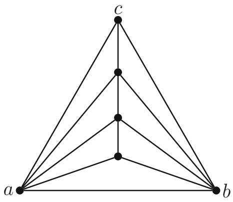

Fig. 3. The tower plane graph $\mathcal{T}_6$.

Theorem 3. *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. 4a), 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 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. 4b). 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 $\alpha \geq 45°$. With similar arguments, for the column on the left of c to contain a node, we must have $\beta \geq 45°$. We have thus $\alpha = \beta = 45°$. Thus c is the node placed on the row below b and b is placed on the column left to c and $x(b) = y(c) = n - 1$. 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. 4c).

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 $\mathcal{T}_n$. □

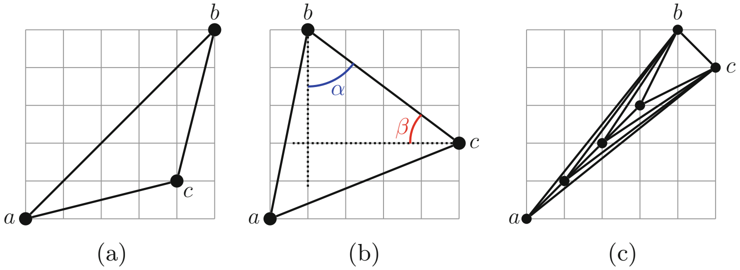

Fig. 4. (a) and (b) Illustrations of the proof of Theorem 3. (c) A planar rook-drawing of $\mathcal{T}_6$.

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 [14]). *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.*

The orientation of edges around an inner node is shown in Fig. 5, where 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).

Two properties of Schnyder woods follow.

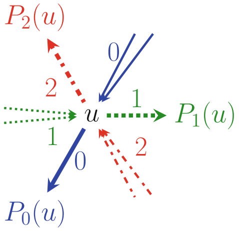

Fig. 5. Orientation around an inner node u in a Schnyder wood.

Proposition 1 (Bonichon et al. [5]). *If u is a descendant of v in T_i, then u is unrelated to v in T_j, $j \neq i$.*

Proposition 2. *If u is the parent of v in T_i, then u is before v in counterclockwise preorder of T_{i-1} and after v in counterclockwise preorder of T_{i+1}.*

Proof. Without loss of generality, assume $i = 2$. In this proof, an i-path denotes a directed path in the tree T_i. Recall that v_i denotes the root of the tree T_i. Let u be the parent of v in T_2 (see Fig. 6).

Suppose that u is after v in the counterclockwise preorder of T_1. By orientation around the node v, the 1-path from v to v_1 has to cross the 2-path from u to v_2. Let t be the intersection of these two paths. Then t is an ancestor of v in T_2 (it is an ancestor of u and thus of v). But t is also an ancestor of v in T_1 because it is on the 1-path from v_1 to v. Though this contradicts Proposition 1. So u is before v in the counterclockwise preorder of T_1, as claimed.

A similar argument proves that u is after v in the counterclockwise preorder of T_0. □

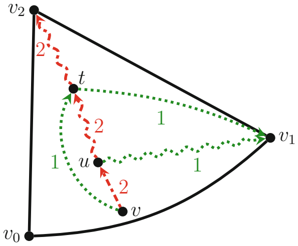

Fig. 6. Illustration of the proof of Proposition 2

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 by Bonichon et al. [5]. 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.*

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

```
$(T_0, T_1, T_2) \leftarrow$ Schnyder wood of $G$
add the oriented edge $(v_1 v_0)$ to $T_0$
add the oriented edge $(v_2 v_0)$ to $T_0$
add the oriented edge $(v_2 v_1)$ to $T_1$
column order $C \leftarrow$ clockwise preorder of $T_0$
row order $R \leftarrow$ clockwise postorder of $T_1$
for $u$ node of $G$ do
    $(x(u), y(u)) = (C(u), R(u))$
draw all $T_2$ edges with straight lines
for $e = (u, P_0(u))$ edge of $T_0$ do
    if $x(u) = x(P_0(u)) + 1$ then draw $e$ with a straight line else Bend $e$ at
    $(x(u), y(P_0(u)) + 1)$
for $e = (u, P_1(u))$ edge of $T_1$ do
    $v \leftarrow ll_0(u)$.
    bend $e$ at $(x(v), y(u))$
```

In Algorithm 2 and later, $ll_0(u)$ denotes the last leaf found in a clockwise preorder of u in T_0. An example of the result of Algorithm 2 on a maximal plane graph is presented in Fig. 7b.

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.

Number of Bends. Let k be the number of leaves in T_0. By construction, T_0 contains $n-1$ edges, T_1 contains $n-2$ edges and T_2 contains $n-3$ edges. The edges of T_0 are all bent, except one for each non-leaf node in T_0. Thus $n-1-(n-k)$ 0-edges are bent. The edges of T_1 are all bent, except k. Thus $n-2-k$ 1-edges are bent. Finally, the edges of T_2 are never bent. Thus, there are exactly $n-3$ bends in the drawing of G.

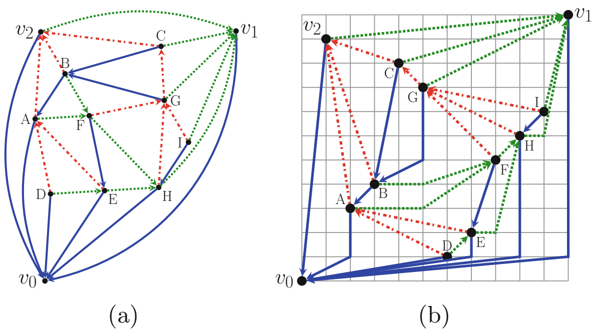

Fig. 7. (a) Schnyder wood of a maximal plane graph G. (b) Result of our polyline algorithm on G.

Planarity. Most of the proofs for the planarity of the drawing are placed in the appendix. We describe in the following some structural properties of the drawing with Lemmas 2, 3 and 4.

Lemma 2. *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*.*

From Lemma 2 and the coordinates of bends chosen for the edges in Algorithm 2, we observe that the configuration around an inner node follow the scheme illustrated in Fig. 8.

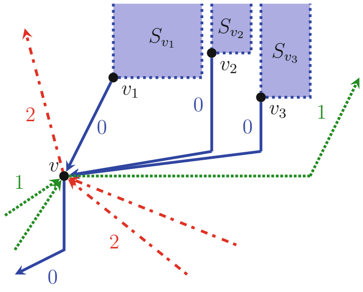

Fig. 8. Edges orientation around an inner node v. S_{v_i} denotes the area in which the subtree of v_i in T_0 is drawn.

This drawing gives a good intuition of why the edges within T_0 do not cross. The detailed proof is not given here, but is based on the following lemmas.

Lemma 3. *For every inner node u, every node v such that $x(P_0(u)) < x(v) < x(u)$ is a descendant of $P_0(u)$ in T_0.*

Proof. This is a direct consequence of the fact that the x-coordinates are given by the clockwise preorder of T_0. □

Lemma 4. *For every inner node u, every node v such that $x(u) < x(v) < x(P_1(u))$ (resp. $x(P_2(u)) < x(v) < x(u)$) is either a descendant of u (resp. $P_2(u)$) in T_0 or $y(v) < y(u)$ (resp. $y(v) < y(P_2(u))$) in $\mathcal{D}(G)$.*

The final step is to explicitly state that the edges drawn do not cross. The proofs are not given due to space limitation. The idea is the following: we first show that edges inside each tree T_0, T_1 and T_2 do not cross. Then we prove that edges from different trees do not cross.

6 Conclusion

In this paper, we observed that all maximal planar graphs but the tower graphs admit no 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 usual classes of plane graphs that all admit a planar 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 admits a planar relaxed rook-drawing with a small (i.e. linear or sublinear) 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. Abello, J., Van Ham, F., Krishnan, N.: ASK-GraphView: a large scale graph visualization system. IEEE Trans. Vis. Comput. Graph. **12**(5), 669–676 (2006)
2. Andrews, G.E.: A lower bound for the volume of strictly convex bodies with many boundary lattice points. Trans. Am. Math. Soc. **106**, 270–279 (1963)

3. Archambault, D., Purchase, H.C.: Mental map preservation helps user orientation in dynamic graphs. In: Didimo, W., Patrignani, M. (eds.) GD 2012. LNCS, vol. 7704, pp. 475–486. Springer, Heidelberg (2013)
4. Bonichon, N., Gavoille, C., Hanusse, N.: Canonical decomposition of outerplanar maps and application to enumeration, coding, and generation. J. Graph Algorithms Appl. **9**(2), 185–204 (2005)
5. Bonichon, N., Le Saëc, B., Mosbah, M.: Optimal area algorithm for planar polyline drawings. In: Kučera, L. (ed.) WG 2002. LNCS, vol. 2573, pp. 35–46. Springer, Heidelberg (2002)
6. Bose, P.: On embedding an outerplanar graph in a point set. Comput. Geom. **23**(3), 303–312 (2002)
7. Di Giacomo, E., Didimo, W., Kaufmann, M., Liotta, G., Montecchiani, F.: Upward-rightward planar drawings. In: The 5th International Conference on Information, Intelligence, Systems and Applications, IISA 2014, pp. 145–150. IEEE (2014)
8. Eades, P., Feng, Q.-W.: Multilevel visualization of clustered graphs. In: North, S.C. (ed.) GD 1996. LNCS, vol. 1190, pp. 101–112. Springer, Heidelberg (1997)
9. de Fraysseix, H., Pach, J., Pollack, R.: Small sets supporting fary embeddings of planar graphs. In: Proceedings of the twentieth annual ACM symposium on Theory of computing, pp. 426–433. ACM (1988)
10. Kornaropoulos, E.M., Tollis, I.G.: Overloaded orthogonal drawings. In: Speckmann, B. (ed.) GD 2011. LNCS, vol. 7034, pp. 242–253. Springer, Heidelberg (2011)
11. Kornaropoulos, E.M., Tollis, I.G.: DAGView: an approach for visualizing large graphs. In: Didimo, W., Patrignani, M. (eds.) GD 2012. LNCS, vol. 7704, pp. 499–510. Springer, Heidelberg (2013)
12. Misue, K., Eades, P., Lai, W., Sugiyama, K.: Layout adjustment and the mental map. J. Visual Lang. Comput. **6**(2), 183–210 (1995)
13. Pach, J., Gritzmann, P., Mohar, B., Pollack, R.: Embedding a planar triangulation with vertices at specified points. Am. Math. Monthly **98**, 165–166 (1991)
14. Schnyder, W.: Embedding planar graphs on the grid. Symp. Discrete Algorithms **90**, 138–148 (1990)