On RAC Drawings of 1-Planar Graphs☆

Michael Bekos[a], Walter Didimo[b], Giuseppe Liotta[b], Saeed Mehrabi[c], Fabrizio Montecchiani[b,*]

[a] *Wilhelm-Schickard-Institut für Informatik, Universität Tübingen, Germany*
[b] *Dipartimento di Ingegneria, Università degli Studi di Perugia*
[c] *David R. Cheriton School of Computer Science, University of Waterloo, Canada*

Abstract

A drawing of a graph is *1-planar* if each edge is crossed at most once. A graph is *1-planar* if it has a 1-planar drawing. A *k-bend RAC (Right Angle Crossing) drawing* of a graph is a polyline drawing where each edge has at most k bends and edges cross only at right angles. A graph is *k-bend RAC* if it has a k-bend RAC drawing. A 0-bend RAC graph (drawing) is also called a *straight-line RAC graph (drawing)*. The relationships between 1-planar and k-bend RAC graphs have been partially studied in the literature. It is known that there are both 1-planar graphs that are not straight-line RAC and straight-line RAC graphs that are not 1-planar. The existence of 1-planar straight-line RAC drawings has been proven only for restricted families of 1-planar graphs. Two of the main questions still open are: (i) What is the complexity of deciding whether a graph has a drawing that is both 1-planar and straight-line RAC? (ii) Does every 1-planar graph have a drawing that is both 1-planar and 1-bend RAC? In this paper we answer these two questions. Namely, we prove an NP-hardness result for the first question, and we positively answer the second question by describing a drawing algorithm for 1-planar graphs.

Keywords: 1-Planarity, RAC drawings, NP-hardness

---

1. Introduction

An emerging line of research in Graph Drawing studies families of non-planar graphs that can be drawn so that crossing edges fulfill some desired properties. This topic is informally known as "beyond planarity" (see e.g. [17, 23, 26]). Different types of properties give rise to different families of beyond planar graphs. Among them,

☆A preliminary extended abstract of the results contained in this paper has been presented at the 24th International Symposium on Graph Drawing, GD 2016.

*Principal corresponding author

Email addresses: bekos@informatik.uni-tuebingen.de (Michael Bekos), walter.didimo@unipg.it (Walter Didimo), giuseppe.liotta@unipg.it (Giuseppe Liotta), smehrabi@uwaterloo.ca (Saeed Mehrabi), fabrizio.montecchiani@unipg.it (Fabrizio Montecchiani)

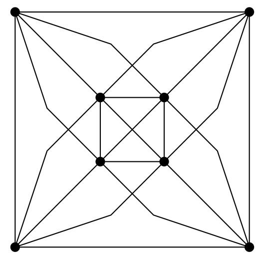


Figure 1: A 1-planar 1-bend RAC drawing.

particular attention has been devoted to *1-planar graphs* (see, e.g., [28, 32]; and refer to [24] for a recent survey) and *k-bend Right Angle Crossing (RAC) graphs* (see, e.g., [12, 13]). A drawing of a graph is *1-planar* if each edge is crossed at most once. A graph is *1-planar* if it has a 1-planar drawing. A *k-bend RAC drawing* of a graph is a polyline drawing where each edge has at most k bends and edges cross only at right angles. A graph is *k-bend RAC* if it has a k-bend RAC drawing. A 0-bend RAC graph (drawing) is also called a *straight-line RAC graph (drawing).* From an application point of view, the study of these two families is motivated by several cognitive experiments, suggesting that the readability of a layout is negatively correlated to the number of crossings [30, 31, 35] and that user task performances are not affected too much if edges cross at large angles [19, 20, 21]. Also, users often prefer straight-line drawings or layouts whose edges have few bends [29], and several algorithms are designed to optimize this aesthetic criterion [9].

For the reasons above, it is interesting to study what graphs can be drawn with at most one crossing per edge, right angle crossings, and few bends per edge at the same time. For example, Figure 1 shows a 1-planar drawing in which each edge has at most one bend and crossings occur at right angles; note that the depicted graph does not admit a straight-line RAC drawing [14]. Recall that n-vertex 1-planar graphs have at most $4n - 8$ edges [28] and that straight-line 1-planar drawings have at most $4n-9$ edges [11], and that both these bounds are tight. Also, straight-line RAC graphs have at most $4n - 10$ edges [12], while 1-bend RAC graphs and 2-bend RAC graphs have at most $6.5n - 13$ and $74.2n$ edges, respectively [4]. Finally, every graph is 3-bend RAC [12]. These results immediately imply that there are 1-planar graphs not admitting 1-planar straight-line drawings and 1-planar graphs not admitting straight-line RAC drawings. The existence of straight-line RAC graphs that are not 1-planar has also been proven [14].

In this scenario, two of the main questions still open are: (i) What is the complexity of deciding whether a graph has a drawing that is both 1-planar and straight-line RAC? (ii) Does every 1-planar graph have a drawing that is both 1-planar and 1-bend RAC?

Contribution. In this paper we answer both questions (i) and (ii). More precisely:

- With respect to question (i), we prove that deciding whether a graph has a 1-planar straight-line RAC drawing is NP-hard in general, even if the order of the edges around the vertices is given as part of the input and cannot be changed.

- We positively answer question (ii) and describe a linear-time algorithm that takes as input a 1-planar graph G along with a 1-planar embedding (see Section 2 for definitions) and computes a 1-planar 1-bend RAC drawing of G. Note that our algorithm may change the initial 1-planar embedding of G.

We remark that a characterization of the 1-planar graphs that can be drawn with straight-line edges was given by Thomassen in 1988 [34]. The characterization is described in terms of the existence of a 1-planar embedding that does not contain two primitive forbidden configurations. This result immediately implies that every 1-planar graph admits a 1-planar drawing with at most one bend per edge (which is not necessarily RAC); it is sufficient to subdivide each crossing edge of any given 1-planar embedding with a dummy vertex, so to remove any possible forbidden configuration. Dummy vertices will correspond to bends in the final drawing. Moreover, Alam *et al.* [1], proved that every 3-connected 1-planar graph can be drawn with straight-line edges, except for at most one edge that may require one bend, in linear time. We also recall that 1-planar straight-line RAC drawings always exist for *IC-planar graphs* [6], which are graphs that admit a 1-planar drawing where no two crossed edges share an end-vertex, and for *outer-1-planar graphs* [8], i.e., those graphs that have a 1-planar drawing in which all vertices belong to the outer face.

The reminder of the paper is structured as follows: Section 2 introduces basic terminology used throughout the paper. The NP-hardness result about question (i) is given in Section 3. Section 4 is concerned with question (ii) and describes the linear-time algorithm that computes 1-planar 1-bend RAC drawings of 1-planar graphs. Conclusions and open problems are discussed in Section 5.

2. Preliminaries

A drawing Γ of a graph G maps each vertex of G to a point of the plane and each edge of G to a Jordan arc connecting its two endpoints. A drawing is *simple* if any two edges have at most one point in common (which is either a common endpoint or a common interior point where the two edges properly cross each other). If not otherwise specified, in the following we shall consider simple drawings. A *k-bend drawing* Γ of a graph is a drawing where each edge is represented as a polyline with at most $k \geq 0$ bends. If $k = 0$, Γ is also called a *straight-line drawing*. A graph G is *planar* if it admits a planar (i.e., crossing-free) drawing. Such a drawing subdivides the plane into topologically connected regions, called *faces*. The infinite region is the *outer face*. The number of vertices encountered in a closed walk along the boundary of a face f is the *degree* of f. Note that, if G is not 2-connected, a vertex may be encountered more than once, thus contributing with more than one unit to the degree of f. A *planar embedding* of G is an equivalence class of planar drawings of G having the same set of faces and the same outer face. A *plane graph* is a planar graph with a given planar embedding. We recall that a planar embedding is uniquely defined by the clockwise order of the edges around each vertex and by the choice of the outer face (see [27] for more details on planar graphs and embeddings).

A *1-planar drawing* of a graph is a drawing where each edge is crossed at most once. The concept of planar embedding is extended to non-planar drawings as follows.

Given a non-planar drawing Γ, interpret every crossing as a vertex. The resulting planarized drawing has a planar embedding. An *embedding* of a (non-planar) graph G is an equivalence class of drawings whose planarized versions have the same planar embedding. A *1-plane* graph is a 1-planar graph with a given *1-planar embedding*, i.e., an embedding where each edge is crossed at most once. Each face of a 1-planar embedding is composed of both vertices and/or crossings, and its degree is the number of vertices and crossings encountered in a closed walk along its boundary (with their multiplicity). An embedding is uniquely defined by the clockwise order of the edges around each vertex, by the clockwise order of the (pieces of) edges around each crossing, and by the choice of the outer face. A *rotation system* R of a graph G defines the clockwise order of the edges around each vertex of G, but does not specify any information on crossings and outer face. A *kite* is a 1-plane graph isomorphic to K_4 with an embedding such that all the vertices are on the boundary of the outer face, the four edges on the boundary are crossing-free, and the remaining two edges cross each other. Given a 1-plane graph G and a kite $K = \{a, b, c, d\}$ such that $K \subseteq G$, we say that K is *empty* if it does not contain any vertex of G inside the 4-cycle $\langle a, b, c, d\rangle$ (that is, it contains only the crossing point). A pair of crossing edges of G *forms an empty kite* if their four end-vertices induce an empty kite.

A *k-bend RAC drawing* is a k-bend drawing where edges cross only at right angles. A *straight-line RAC drawing* is a k-bend RAC drawing where $k = 0$. A *1-planar k-bend RAC drawing* is a k-bend RAC drawing where each edge is crossed at most once. A *1-planar straight-line RAC drawing* is a 1-planar k-bend RAC drawing where $k = 0$.

3. Recognizing 1-planar Straight-line RAC Drawable Graphs

In this section we prove that the problem of deciding whether a graph G admits a 1-planar straight-line RAC drawing is NP-hard. To simplify the presentation, we will first assume that a rotation system R of G is given, which must be preserved in the drawing. However, as we will note later in this section our result holds even without this assumption.

Theorem 1. *It is NP-hard to decide whether a graph G with a rotation system R admits a 1-planar straight-line RAC drawing that preserves R.*

Proof. We reduce from the problem 3-PARTITION, which is known to be NP-hard in the strong sense [15]. An instance of 3-PARTITION consists of a set $A = \{a_1, a_2, \dots, a_{3m}\}$ of $3m$ integers, each of which is strictly between $B/4$ and $B/2$, where $B = \frac{1}{m}\sum_{i=1}^{3m} a_i$. Problem 3-PARTITION asks whether A can be partitioned into a m subsets $A_1, A_2, \dots, A_m$, each of cardinality 3, such that the sum of the integers in each subset is B.

Let A be an instance of 3-PARTITION. We will construct in polynomial time a graph G and a rotation system R of G, such that A can be partitioned if and only if G admits a 1-planar straight-line RAC drawing Γ preserving R. Central in our construction is the so-called *augmented square antiprism graph* (or *ASA graph* for short); see Figure 2a. Argyriou et al. [3] have shown that the ASA graph admits only two embeddings that can be realized as straight-line RAC drawings (and are also 1-planar), illustrated in Figures 2a and 2b respectively. However, if one appropriately *glues* two (or more) ASA graphs as in Figure 2c, then there exists only a single embedding that can be realized as a straight-line RAC drawing [3].

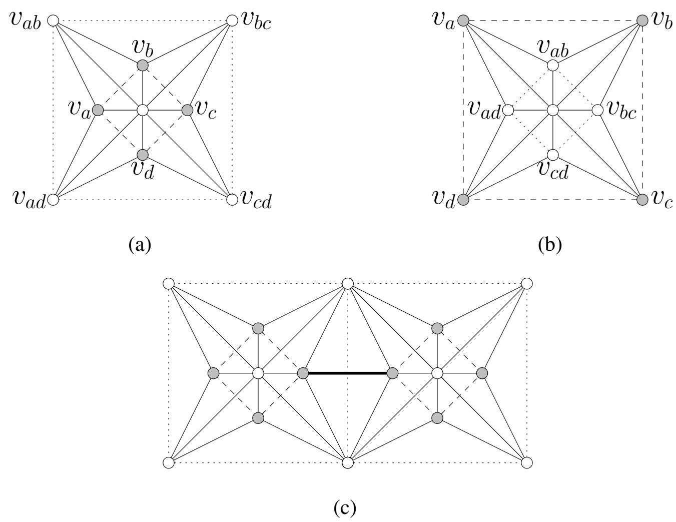
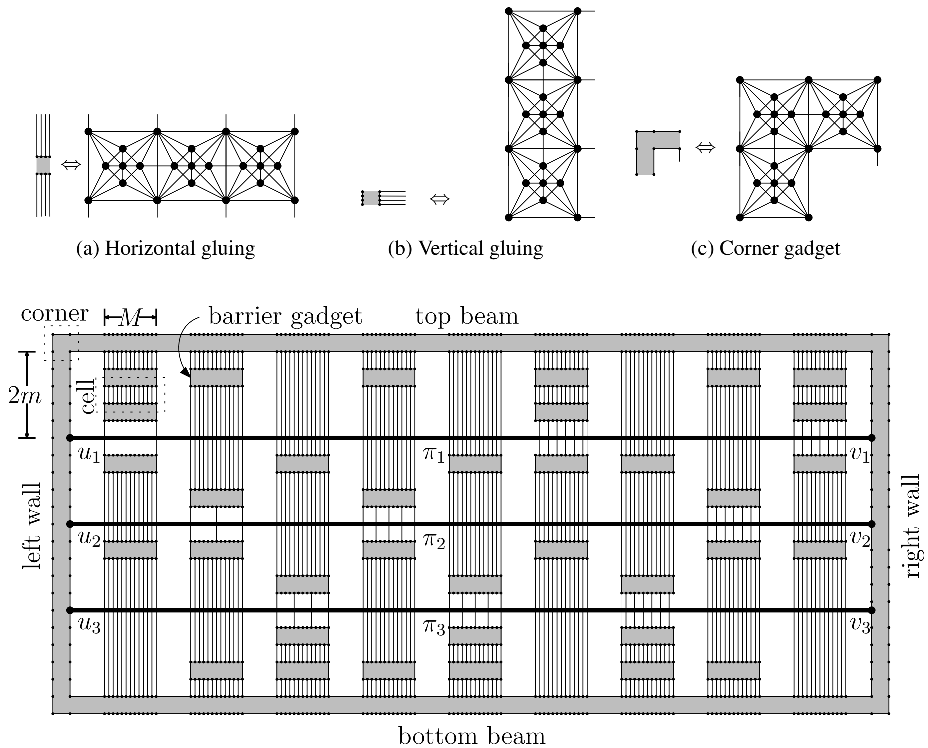


Figure 2: (a),(b) Two different RAC drawings of the ASA graph with different embeddings. (c) The result of *gluing* two ASA graphs yielding a new graph with unique embedding realizable as a straight-line RAC drawing.

Let M be a large integer such that $M > a_i$ for each $i = 1, 2, \dots, 3m$, e.g., $M = \lceil B/2 \rceil + 1$. To construct graph G, we first create a quite rigid structure, called *ring barrier*, consisting of four components: the *top beam*, the *right wall*, the *bottom beam* and the *left wall*; see Figure 3d. Each of the top and bottom beams consists of $3mM$ copies of the ASA graph that are horizontally glued one next to the other; see also Figure 3a. On the other hand, each of the left and right walls consists of $2m^2$ copies of the ASA graph that are vertically glued one above or below the other; see also Figure 3b. The ring barrier is formed by gluing in a circular arrangement the endpoints of the top beam, right wall, bottom beam and left wall with four new copies of the ASA graph that act as *corners*; see also Figure 3c.

We proceed by connecting the top and bottom beams by a set of $3m$ *columns*; see Figure 3d. Each column contains a stack of $2m - 2$ vertically-aligned copies of the so-called *barrier-gadget*, which is formed by horizontally gluing $M - 1$ copies of the ASA graph. Within each column, consecutive copies of the barrier-gadget are connected by pairwise disjoint edges. The topmost and the bottomost copy of the barrier-gadget of each column is connected to the top and bottom beam, respectively, also by pairwise disjoint edges. The edges that realize these connections are called *vertical edges* and form the so-called *cells*. In particular, there are $m - 1$ *topmost cells*, one *central cell* and $m - 1$ *bottomost cells*. The central cell of the i-th column is *sparse* containing only a_i vertical edges; the remaining (topmost or bottomost) ones are *dense* containing M edges each; recall that $M > a_i$.

We conclude the construction of graph G by introducing m pairwise internally



(a) Horizontal gluing (b) Vertical gluing (c) Corner gadget

(d)

Figure 3: Illustration of: (a) horizontal gluing (b) vertical gluing (c) corner gadget (d) the reduction from 3-PARTITION, where $m = 3$, $A = \{2, 3, 4, 5, 5, 6, 6, 7, 7\}$ and $B = 15$. Paths π_1, π_2 and π_3 are routed according to the following solution of 3-PARTITION: $A_1 = \{2, 6, 7\}$, $A_2 = \{3, 5, 7\}$ and $A_3 = \{4, 5, 6\}$.

disjoint paths, $\pi_1, \pi_2, \dots, \pi_m$, called the *transversal paths* of G. Each transversal path has exactly $(3m-3)M + B$ edges. For $j = 1, 2, \dots, m$, let u_j be the bottom-right vertex of the $2mj$-th copy of the ASA graph on the left wall. Symmetrically, let v_j be the bottom-left vertex of the $2mj$-th copy of the ASA graph on the right wall; see Figure 3d. Path π_j connects vertex u_j of the left wall to vertex v_j of the right wall.

In order to define the rotation system R, we will assume that the cyclic order of the edges around each vertex is compatible with the one of Figure 3d. This implies that the ring barrier as well as the barrier gadgets have unique embeddings, because the gluing of two or more ASA graphs has [3]. Also, 1-planarity implies that the vertical edges of each cell cannot cross with each other or with vertical edges of other cells. Now it is straightforward to see that an instance of 3-PARTITION can be transformed into an instance of the 1-planar straight-line RAC drawing problem in time polynomial in m, B and M.

Now, assume that graph G admits a 1-planar straight-line RAC drawing Γ preserving R and consider a transversal path π_j in Γ, $j = 1, 2, \dots, m$. Since π_j connects a vertex of the left wall to a vertex of the right wall and since π_j cannot pass through the ring barrier (due to 1-planarity), it follows that π_j must cross all columns of G. The length of each transversal path, however, guarantees that each of the transversal paths must cross exactly three sparse cells and $3m-3$ dense ones, while 1-planarity ensures that no two transversal paths pass through the same cell. With the latter two properties in mind, we are now able to present how to construct a solution $A_1, A_2, \ldots, A_m$ of the instance A of 3-PARTITION. More precisely, if path π_j crosses the κ-th, λ-th and μ-th columns of G through sparse cells, where $1 \leq \kappa, \lambda, \mu \leq 3m$, then the j-th partition A_j of instance A of 3-PARTITION will contain integers $\{a_\kappa, a_\lambda, a_\mu\}$. Observe that, due to the length of π_j, $a_\kappa + a_\lambda + a_\mu = B$.

To complete the description of our NP-hardness reduction, assume that instance A of 3-PARTITION admits a partition into subsets $A_1, A_2, \ldots, A_m$ each of cardinality three. We construct a 1-planar straight-line RAC drawing of G preserving R as follows. First, we draw each copy of the ASA graph in a 1×1 enclosing box so that the boundary edges of it coincide with the corresponding edges of its enclosing box. This implies that the ring barrier is drawn as a rectangle of width $3mM+2$ and height $2m^2+2$, while each barrier gadget is drawn as a rectangle of width M and unit height. Additionally, the endpoints u_i and v_i of a transversal path π_j are drawn in the same y-coordinate along the left and right walls, respectively, $j = 1, 2, \ldots, m$; see Figure 3d. This implies that we can draw transversal path π_j along the horizontal line-segment connecting u_j and v_j. In addition, all vertical edges of each column of G can be drawn as pairwise crossing-free vertical line-segments of minimum length one. This implies a partial drawing Γ in which the barrier gadgets of each column are allowed to "float" within their columns affecting neither the 1-planarity nor the RAC drawability of Γ (as long as their partial order from top to bottom is preserved).

To complete the drawing, it remains to describe how to appropriately embed transversal paths $\pi_1, \pi_2, \ldots, \pi_m$ of G in the partial drawing Γ constructed so far violating neither the 1-planarity nor the RAC drawability of Γ. To do so, we will exploit solution $\{A_1, A_2, \ldots, A_m\}$ of 3-PARTITION for instance A. The idea is to route $\pi_1, \pi_2, \ldots, \pi_m$ in such a way that: (R1) they do not cross each other, (R2) they do not cross any barrier-gadget, (R3) each cell is traversed by at most one path, (R4) each path passes through exactly 3 sparse cells and $3m-3$ dense cells. Consider a subset A_j of the solution of instance A of 3-PARTITION and assume without loss of generality that $A_j = \{a_\kappa, a_\lambda, a_\mu\}$, where $1 \leq \kappa, \lambda, \mu \leq 3m$. Then, in drawing Γ path π_j will cross the κ-th, λ-th and μ-th columns of G through sparse cells. Path π_j will cross the remaining columns of G through dense cells. Hence, requirement (R4) is satisfied. Consider now the routing of the remaining transversal paths through the κ-th column; the corresponding routings though the λ-th and μ-th columns of G are symmetric. By construction, there must exist exactly $m-1$ available cells above and exactly $m-1$ available cells below the sparse cell of the κ-th column; all of which are dense. This implies that there exist at least as many available dense cells as transversal paths to route at each side of the sparse cell of the κ-th column. Hence, we can route the remaining transversal paths through the κ-th column so that requirements (R1)-(R3) are also satisfied. This completes our NP-hardness proof. □

Note that our reduction is similar to the one proposed by Bekos et al. [5] in order to prove that the fan-planar drawing problem [22] with a fixed rotation system is NP-hard.

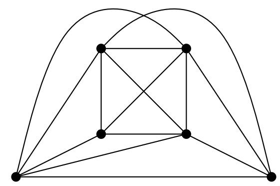


Figure 4: A triangulated 1-plane graph with two pairs of crossing edges. One of these two pairs forms an empty kite, while the other one does not since its crossing is part of the outer face.

However, although the general idea is similar in both reductions, ours required different approaches as it relies more on geometric techniques than the more combinatorial ones given in [5]. Now, observe that our reduction is quite rigid. In particular, since gluing two or more ASA graphs yields a graph with unique embedding realizable as a straight-line RAC drawing [3], it follows that the rotation system of the ring barrier and all copies of the barrier gadgets has to be uniquely defined. On the other hand, 1-planarity guarantees that all columns and all transversal paths are completely drawn in the interior of the ring barrier, and no two vertical edges can cross each other. Hence, R is uniquely implied by these properties. The above discussion can be summarized as follows.

Theorem 2. *It is NP-hard to decide whether a graph admits a 1-planar straight-line RAC drawing.*

4. 1-planar 1-bend RAC Drawings

In this section we prove that every 1-planar graph admits a 1-planar 1-bend RAC drawing.

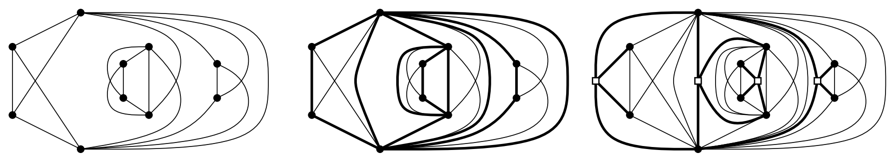
Theorem 3. *Let G be an n-vertex 1-planar graph. Then G admits a 1-planar 1-bend RAC drawing Γ. Also, if a 1-planar embedding of G is given as part of the input, Γ can be computed in $O(n)$ time.*

A 1-plane graph G, possibly containing parallel edges, is *triangulated* if each face is a triangle, formed by either three vertices or by one crossing and two vertices. Clearly, a triangulated 1-plane graph is 2-connected. The next observation immediately follows from the definition of triangulated 1-plane graph.

Observation 1. *Let G be a triangulated 1-plane graph. Every pair of crossing edges of G forms an empty kite, except for at most one pair of crossing edges if their crossing point is on the outer face of G.*

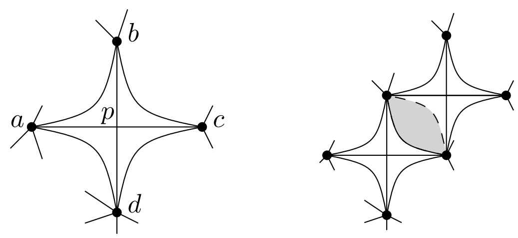
For example, Figure 4 shows a triangulated 1-plane graph with two pairs of crossing edges. One of these two pairs forms an empty kite, while the other one does not since its crossing is part of the outer face.

We are now ready to prove Theorem 3. The proof is based on an algorithm that takes as input a (simple) 1-plane graph G with n vertices (see, e.g., Figure 5a), and



(a) G (b) G_2 (c) G^+

Figure 5: Illustration for the augmentation step of the algorithm. In each subfigure, the thick edges are those added by the augmentation step (note that some of them exist already in G).



(a) (b) G^+

Figure 6: (a) Crossing augmentation; (b) The removal of an edge (the dashed one) from a face of degree two.

computes a 1-planar 1-bend RAC drawing Γ of G in $O(n)$ time. The assumption that G comes with a given 1-planar embedding is motivated by the fact that deciding whether a graph admits such an embedding is NP-complete [16, 25]. Moreover, we shall assume that G is connected, as otherwise we can draw independently each connected component. The high-level idea of the algorithm is as follows. First augment G and (if needed) modify its embedding to get a triangulated 1-plane graph, possibly containing parallel edges. Then, execute a *hierarchical contraction* of the graph and apply a recursive technique to compute a 1-planar 1-bend RAC drawing. Roughly speaking, a hierarchical contraction is a recursive operation that computes from G a hierarchy of coarser graphs by contracting specific subgraphs into single edges. These steps are described in more detail in the following sections.

4.1. Augmentation

The first step of the algorithm transforms G into a triangulated 1-plane graph G^+ by adding edges and vertices. The 1-planar embedding of G^+ may be different from that of G for the common part.

Let (a, c) and (b, d) be two edges of G that cross in a point p. Let $\{a, b, c, d\}$ be the circular order of the vertices around p. For each such pair of crossing edges, we add an edge (a, b), and draw[1] it such that it follows the curves (a, p) and (p, b). Similarly, we draw the three edges (b, c), (c, d) and (d, a) (see Figure 6a). This operation ensures that each pair of crossing edges forms an empty kite. Also, this operation does not introduce edge crossings but it may create parallel edges. We denote by G_1 the resulting

[1]For ease of description, here we are interpreting an embedding as a drawing.

(multi)graph. For each pair of parallel edges e_1 and e_2 of G_1, such that $e_1 \in G$ and $e_2 \in G_1$, we remove e_1 from G_1. Since all edges in $G_1 \setminus G$ are drawn planar, and since we removed all pairs of parallel edges e_1 and e_2 of G_1 such that $e_1 \in G$ and $e_2 \in G_1$, it follows that no parallel edge is crossed in G_1. Then, for each face of degree two (which consists of two parallel edges), we remove one of its edges (see Figure 6b). Note that, if the outer face has degree two, then each of its two parallel edges is part of an empty kite. Thus, we remove one of these two edges to make the degree of the outer face equal to three (it will be formed by two vertices and one crossing).

Let G_2 be the resulting graph, which can be easily computed in $O(n)$ time, since G has $O(n)$ crossings (see, e.g., [33]). Observe that if there exist two parallel edges in G_2, then their two end-vertices form a separation pair. Figure 5b shows the graph G_2 obtained from the graph G of Figure 5a. We remark that a similar operation has been used by Alam et al. [1] in order to compute a straight-line drawable 1-planar embedding of a 3-connected 1-planar graph. Alam et al. [1], however, consider only 3-connected graphs. This ensures that the augmented graph does not contain parallel edges. We do not have any restriction on the connectivity of G, which poses additional issues in the construction of a suitable 1-planar embedding.

We now transform G_2 into a triangulated 1-plane graph. Recall that by construction each face of G_2 is either a triangle or it has degree greater than three. Let f be a face of G_2 that is not a triangle. Such a face contains only vertices on its boundary, since each crossing is shared by exactly four triangular faces due to the empty kite property. We add an *extra vertex* v_f inside f and connect it to all vertices on the boundary of f. Figure 5c shows the graph G^+ obtained from the graph G_2 in Figure 5b. Since G_2 has $O(n)$ faces, G^+ has $O(n)$ vertices and it can be computed in $O(n)$ time. The next lemma follows from the above discussion.

Lemma 4. *Graph G^+ is a triangulated 1-plane (multi)graph.*

4.2. Hierarchical contraction

For ease of description, we define a hierarchical contraction of G^+ inspired by $SPQR$-trees [10], but simpler and more direct for our purposes (that is, we use a simpler data structure and a more concise terminology). We start by proving the following key property of triangulated 1-plane graphs.

Lemma 5. *Let G be a triangulated 1-plane graph, and let $\{u, v\}$ be a separation pair of G. Then there exist two parallel edges e, e' between u and v such that $\{u, v\}$ is not a separation pair for the graph obtained by removing from G all vertices inside the cycle $\langle e, u, e', v\rangle$.*

Proof. Refer to Figure 7. Let $C_0, \ldots, C_{k-1}$ be the $k > 1$ connected components obtained by removing u and v from G. Observe first that there is no pair of crossing edges e_i, e_j, such that $e_i \in C_i$ and $e_j \in C_j$, with $0 \leq i \neq j \leq k-1$, as otherwise, by Observation 1, C_i and C_j would not be distinct components. After a possible relabeling, we can assume that $C_0, \ldots, C_{k-1}$ are such that for every pair of indices i and j, with $0 \leq i < j \leq k-1$, C_i is encountered before C_j in the counterclockwise order around u, starting from some fixed edge of C_0 incident to u.

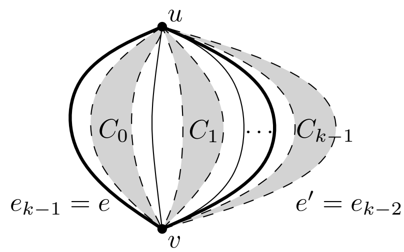
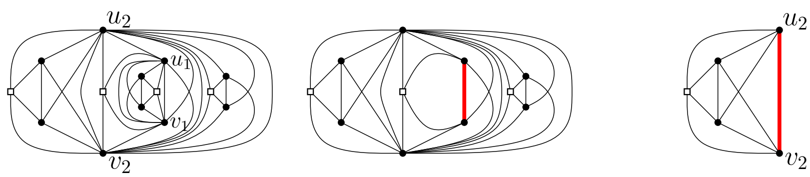


Figure 7: Illustration for the proof of Lemma 5.



(a) G^+ (b) (c) G^*

Figure 8: Illustration for the hierarchical contraction technique. Thick edges are drawn thicker (and red).

Since all faces of G are triangles, each face incident to u and v is also incident either to a third vertex (distinct from u and v), or to a crossing. Thus, for each pair of consecutive components C_i and C_{i+1} (for $i = 0, \ldots, k-1$) there is a parallel edge e_i between u and v that is incident to a face of C_i and to a face of C_{i+1} (indices taken modulo k). We let $e = e_{k-1}$ and $e' = e_{k-2}$ (they are bold in Figure 7). Then, the graph obtained by removing all vertices inside the cycle $\langle e, u, e', v \rangle$ corresponds either to C_0, or to C_{k-1}, and in both cases the statement holds. □

By Lemma 5, for each separation pair $\{u, v\}$ of G^+, there exists a set of $k > 1$ parallel edges $\{e_0, \ldots, e' = e_{k-2}, e = e_{k-1}\}$ between u and v, such that the cycle $\langle e, u, e', v \rangle$ encloses all other parallel edges between u and v in its interior. We call the subgraph G_{uv} of G^+ whose outer face is $\{e, u, e', v\}$, *inner graph* of (u, v), and each subgraph of G_{uv} whose outer face is $\{u, e_{i-1}, v, e_i\}$, for $i = 0, \ldots, k-2$ (where $e_{-1} = e_{k-1}$), *inner component* of (u, v). Let G_{uv} be an inner graph of G^+ that does not contain any inner graph as a subgraph. Replace G_{uv} with a *thick edge* between u and v; the resulting graph is still a triangulated 1-plane graph. Iterate this procedure until there are no more inner graphs to be replaced. Since the end-vertices of each set of parallel edges form a separation pair, and since we removed all subgraphs enclosed by parallel edges, this results in a simple triangulated 3-connected 1-plane graph G^*. Figure 8c shows the graph G^* obtained from the graph G^+ in Figure 8a, through the intermediate step in Figure 8b. Namely, in the first step a thick edge (u_1, v_1) is added to replace $G_{u_1v_1}$ and the resulting graph is shown in Figure 8b. Then a thick edge is added to replace $G_{u_2v_2}$; the resulting graph is shown in Figure 8c and does not contain other separation pairs. The next lemma follows.

Lemma 6. *Graph G^* is a simple 3-connected triangulated 1-plane graph.*

Note that graph G^* can be computed in $O(n)$ time since all separation pairs of the graph can be computed in $O(n)$ time (see, e.g., [18]).

4.3. Drawing

The overview of this third step is as follows. Start with a 1-planar 1-bend RAC drawing of G^*, and then recursively replace thick edges with a 1-planar 1-bend RAC drawing of the corresponding inner graphs. Finally, delete the edges and vertices added in the augmentation step to get a 1-planar 1-bend RAC drawing of G.

To compute a 1-planar 1-bend RAC drawing of G^*, first remove from G^* all pairs of crossing edges and denote by H^* the resulting plane graph (see Figure 9a). Note that thick edges are never crossed and all faces of H^* have either degree three or degree four. We prove the following.

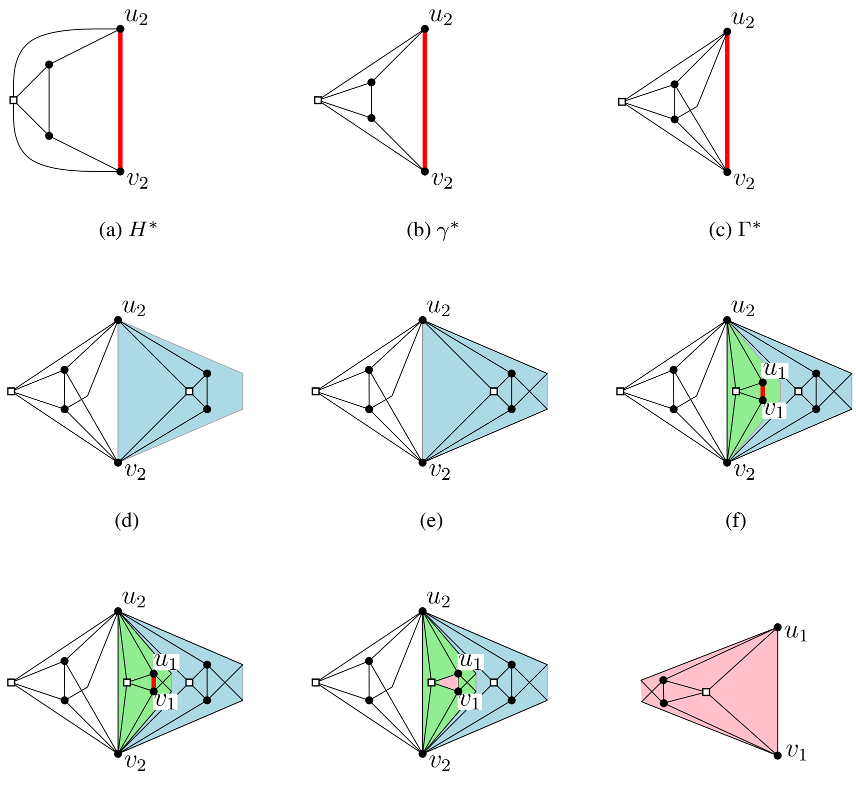
Lemma 7. *Graph H^* is 3-connected.*

Proof. Clearly, H^* is connected, as every pair of crossing edges in G^* forms an empty kite and the removal of the two crossing edges cannot disconnect the graph. Also, observe that each face of H^* has degree either three or four.

Suppose, for a contradiction, that H^* contains a cut vertex c. Then there is a face f of H^* such that c is encountered at least twice in a closed walk $\mathcal{C}$ along the boundary of f. Consider two consecutive occurrences, denoted by c_1 and c_2, of c in $\mathcal{C}$. If in $\mathcal{C}$ no further vertex is encountered between c_1 and c_2, then c has a self-loop, which is not possible. If only one vertex v is encountered, then either v has degree one in H^*, or there are two parallel edges between c and v, and both cases are not possible. Hence, between any two consecutive occurrences of c in $\mathcal{C}$ there must be at least two distinct vertices, and thus the degree of f is at least six, a contradiction to the fact that all faces of H^* have degree either three or four.

It remains to show that H^* contains no separation pair. Suppose, for a contradiction, that H^* contains a separation pair $\{u, v\}$. Then there are at least two faces of H^*, denoted by f_1 and f_2, such that: (i) f_1 and f_2 share no edge and contain both u and v; (ii) either both f_1 and f_2 have degree four, or one has degree three and the other one has degree four. Since every degree-4 face of H^* corresponds to a kite in G^*, in both cases G^* contains at least two parallel edges between u and v, a contradiction to the fact that G^* is simple. □

Compute a planar straight-line drawing γ^* of H^* where all faces are strictly convex and the outer face is a prescribed convex polygon P (see Figure 9b); this can be done by applying the linear-time algorithm by Chiba et al. [7]. We remark that having convex inner faces and a prescribed polygon as outer face are two key ingredients for our construction. Thus, the algorithm by Alam *et al.* [1] cannot be directly used in this context. If the outer face of H^* has degree four, we let P be a trapezoid, else P is a triangle. Since H^* is triconnected and all faces are either triangles or quadrangles, we can avoid three collinear vertices by slight perturbations. Namely, since all faces have degree at most four and are drawn convex, no more than three vertices can be simultaneously collinear and in the same face. Suppose that three vertices are drawn collinear and belong to the same face f. Since the graph is triconnected, none of these three vertices has degree two. Also, the three vertices are incident to faces drawn convex,



(a) H^* (b) γ^* (c) Γ^*

(d) (e) (f)

(g) (h) (i)

Figure 9: Illustration of the drawing technique. The graph depicted in (i) is drawn inside the triangular region incident to (u_1, v_1) (pink).

and thus we can slightly move one of them such that all faces remain convex and no collinearity introduced in the drawing. To reinsert the crossing edges, we distinguish between the inner faces and the outer face of H^*.

Two crossing edges can be easily reinserted in an inner face, just drawing one of the two with no bend and the other with one bend, such that they cross at right angles (see, e.g., [2] and Figure 9c).

To reinsert two crossing edges e_1, e_2 in the outer face of H^* so that they form a right angle, we can draw e_1 and e_2 with one bend each (see also Figure 10). Namely, P is a trapezoid by construction. Assume that the minor base m and the greater base M of P are aligned with the horizontal axis. The first segment of e_1 is such that its rightmost endpoint p_1 coincides with the rightmost endpoint of m, and its leftmost endpoint q_1 is b units above the leftmost endpoint of m, where b is equal to the length of m. The second segment of e_1 has q_1 as rightmost endpoint, and its leftmost endpoint r_1 coincides with the leftmost endpoint of M. Edge e_2 is drawn symmetrically. This construction makes the first segments of e_1 and e_2 intersect in a right angle.

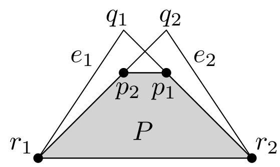


Figure 10: Reinserting the pair of crossing edges on the outer face of H^*.

Consider now a thick edge (u, v) of G^* and its inner graph G_{uv}. Recall that G_{uv} consists of $k-1 \geq 1$ inner components $C_0, \ldots, C_{k-2}$. Each C_i ($i = 0, \ldots, k-2$) has two parallel edges e_{i-1}, e_i as outer face (where $e_{-1} = e_{k-1}$). Also, analogously to G^*, $C_i^- = C_i \setminus \{e_i\}$ is a simple 3-connected triangulated 1-plane graph (it is an induced subgraph of G^+ and all its inner graphs have been replaced by thick edges). Remove all crossing edges of C_k^- and let H_k^- be the resulting 3-connected plane graph. Compute a planar straight-line drawing γ_k of H_k^- such that all faces are strictly convex polygons and the outer face is a prescribed polygon P. We distinguish two cases, depending on the degree of the outer face of H_k^-.

If the outer face of H_k^- has degree 3, P is a triangle whose side with corners u and v has length equal to the length of the thick edge (u, v) in Γ^*, and its height is small enough so that the thick edge (u, v) can be replaced with P without introducing crossings.

If the outer face of H_k^- has degree 4, P is a trapezoid such that its greater base has u and v as corners and the same length as the thick edge (u, v) in Γ^*. The height of P is such that the thick edge (u, v) can be replaced with P without introducing crossings. Also, the minor base of P is made sufficiently short such that the pair of crossing edges on the outer face of H_k^- can be reinserted without introducing crossings in Γ^*, as described for H^* (see Figure 9d). By the same argument used for H^*, all pairs of crossing edges can be reinserted so to form right angle crossings and have at most 1 bend each (see Figure 9e). If $k - 1 > 1$, we iterate this procedure and compute a drawing Γ_i^- for each C_i^-, for $i = k-2, \ldots, 0$. The polygon representing the outer face of each Γ_i is suitably chosen so to fit inside the face containing edge e_i in Γ_{i+1}. The union of all such drawings is a 1-planar 1-bend RAC drawing Γ_{uv} of G_{uv} (see Figures 9f and 9g). All parallel edges $e_0, \ldots, e_{k-1}$ are represented by overlapping segments between u and v, thus all of them but one can be removed from the drawing.

Repeat this procedure for each thick edge of G^*, and recursively apply the same technique for each inner graph of G^*; see Figures 9h and 9i for an illustration. The resulting drawing Γ is a 1-planar 1-bend RAC drawing of G^+. Removing dummy vertices and edges, we get the desired drawing of G. In terms of time complexity, each planar straight-line drawing with (strictly) convex faces is computed in linear time in the size of the input graph [7], and in linear time we can reinsert the crossing edges. Thus the whole procedure takes $O(n)$ time. This concludes the proof of Theorem 3.

5. Conclusions and Open Problems

We proved that deciding if a graph admits a 1-planar straight-line RAC drawing is NP-hard. Note that maximally dense RAC graphs (i.e., RAC graphs with n vertices and exactly $4n - 10$ edges) are 1-planar [14]. So, an interesting open problem is to establish the complexity of deciding whether an n-vertex graph with $4n - 10$ edges is straight-line RAC drawable. On the positive side, it is known that IC-planar graphs and outer-1-planar graphs are straight-line RAC [6, 8]. It would be interesting to find other subfamilies of 1-planar graphs that are straight-line RAC. In particular, we ask whether *NIC-planar graphs* are straight-line RAC. We recall that a graph is NIC-planar if it admits a 1-planar drawing in which any two pairs of crossing edges share at most two vertices (see, e.g., [36]).

We presented a drawing algorithm to compute 1-planar 1-bend RAC drawings of 1-planar graphs. This algorithm may produce drawings with exponential area: Is this area requirement necessary for some family of 1-planar graphs? Also, our algorithm may change the embedding of the input graph. For example, the edge (u_2, v_2) is on the outer face in the original embedding of the input graph G (Figure 5a), while it is not drawn on the outer face in the drawing produced by our algorithm (Figure 9h). Are there 1-planar embeddings that cannot be realized as 1-bend RAC drawings?

Acknowledgments

We thank the anonymous referees of this paper for their valuable suggestions.

References

[1] M. J. Alam, F. J. Brandenburg, and S. G. Kobourov. Straight-line grid drawings of 3-connected 1-planar graphs. In S. K. Wismath and A. Wolff, editors, *GD 2013*, volume 8242 of *LNCS*, pages 83–94. Springer, 2013.

[2] P. Angelini, L. Cittadini, W. Didimo, F. Frati, G. D. Battista, M. Kaufmann, and A. Symvonis. On the perspectives opened by right angle crossing drawings. *J. Graph Algorithms Appl.*, 15(1):53–78, 2011.

[3] E. N. Argyriou, M. A. Bekos, and A. Symvonis. The straight-line RAC drawing problem is NP-hard. *J. Graph Algorithms Appl.*, 16(2):569–597, 2012.

[4] K. Arikushi, R. Fulek, B. Keszegh, F. Moric, and C. D. Tóth. Graphs that admit right angle crossing drawings. *Comput. Geom.*, 45(4):169–177, 2012.

[5] M. A. Bekos, S. Cornelsen, L. Grilli, S. Hong, and M. Kaufmann. On the recognition of fan-planar and maximal outer-fan-planar graphs. In C. A. Duncan and A. Symvonis, editors, *GD 2014*, volume 8871 of *LNCS*, pages 198–209. Springer, 2014.

[6] F. J. Brandenburg, W. Didimo, W. S. Evans, P. Kindermann, G. Liotta, and F. Montecchiani. Recognizing and drawing IC-planar graphs. *Theor. Comput. Sci.*, 636:1–16, 2016.

[7] N. Chiba, T. Yamanouchi, and T. Nishizeki. Linear algorithms for convex drawings of planar graphs. In J. Bondy and U. S. R. Murty, editors, *Progress in graph theory*, pages 153–173. Academic Press, Toronto, 1984.

[8] H. R. Dehkordi and P. Eades. Every outer-1-plane graph has a right angle crossing drawing. *Int. J. Comput. Geometry Appl.*, 22(6):543–558, 2012.

[9] G. Di Battista, P. Eades, R. Tamassia, and I. G. Tollis. *Graph Drawing*. Prentice Hall, Upper Saddle River, NJ, 1999.

[10] G. Di Battista and R. Tamassia. On-line planarity testing. *SIAM J. Comput.*, 25(5):956–997, 1996.

[11] W. Didimo. Density of straight-line 1-planar graph drawings. *Inf. Process. Lett.*, 113(7):236–240, 2013.

[12] W. Didimo, P. Eades, and G. Liotta. Drawing graphs with right angle crossings. *Theor. Comput. Sci.*, 412(39):5156–5166, 2011.

[13] W. Didimo and G. Liotta. The crossing angle resolution in graph drawing. In J. Pach, editor, *Thirty Essays on Geometric Graph Theory*. Springer, 2012.

[14] P. Eades and G. Liotta. Right angle crossing graphs and 1-planarity. *Discrete Appl. Math.*, 161(7-8):961–969, 2013.

[15] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979.

[16] A. Grigoriev and H. L. Bodlaender. Algorithms for graphs embeddable with few crossings per edge. *Algorithmica*, 49(1):1–11, 2007.

[17] S.-H. Hong and T. Tokuyama. Algorithmics For Beyond Planar Graphs, 2016. http://shonan.nii.ac.jp/seminar/089/.

[18] J. E. Hopcroft and R. E. Tarjan. Dividing a graph into triconnected components. *SIAM J. Comput.*, 2(3):135–158, 1973.

[19] W. Huang. Using eye tracking to investigate graph layout effects. In *APVIS 2007*, pages 97–100, 2007.

[20] W. Huang, P. Eades, and S. Hong. Larger crossing angles make graphs easier to read. *J. Vis. Lang. Comput.*, 25(4):452–465, 2014.

[21] W. Huang, S. Hong, and P. Eades. Effects of crossing angles. In *PacificVis 2008*, pages 41–46. IEEE, 2008.

[22] M. Kaufmann and T. Ueckerdt. The density of fan-planar graphs. *CoRR*, abs/1403.6184, 2014.

[23] S. G. Kobourov, S.-H. Hong, M. Kaufmann, and J. Pach. Beyond-Planar Graphs: Algorithmics and Combinatorics, 2016. http://www.dagstuhl.de/16452.

[24] S. G. Kobourov, G. Liotta, and F. Montecchiani. An annotated bibliography on 1-planarity. *CoRR*, abs/1703.02261, 2017.

[25] V. P. Korzhik and B. Mohar. Minimal obstructions for 1-immersions and hardness of 1-planarity testing. *J. Graph Theory*, 72(1):30–71, 2013.

[26] G. Liotta. Graph drawing beyond planarity: some results and open problems. In *ICTCS 2014*, volume 1231 of *CEUR Workshop Proceedings*, pages 3–8. CEUR-WS.org, 2014.

[27] T. Nishizeki and N. Chiba. *Planar graphs: Theory and algorithms*, volume 32. Elsevier, 1988.

[28] J. Pach and G. Tóth. Graphs drawn with few crossings per edge. *Combinatorica*, 17(3):427–439, 1997.

[29] H. C. Purchase. Which aesthetic has the greatest effect on human understanding? In G. Di Battista, editor, *GD 1997*, volume 1353 of *LNCS*, pages 248–261. Springer, 1997.

[30] H. C. Purchase. Effective information visualisation: a study of graph drawing aesthetics and algorithms. *Interacting with Computers*, 13(2):147–162, 2000.

[31] H. C. Purchase, D. A. Carrington, and J. Allder. Empirical evaluation of aesthetics-based graph layout. *Empirical Software Engineering*, 7(3):233–255, 2002.

[32] G. Ringel. Ein Sechsfarbenproblem auf der Kugel. *Abh. Math. Semin. Univ. Hambg.*, 29(1-2):107–117, 1965.

[33] Y. Suzuki. Re-embeddings of maximum 1-planar graphs. *SIAM J. Discrete Math.*, 24(4):1527–1540, 2010.

[34] C. Thomassen. Rectilinear drawings of graphs. *J. Graph Theory*, 12(3):335–341, 1988.

[35] C. Ware, H. C. Purchase, L. Colpoys, and M. McGill. Cognitive measurements of graph aesthetics. *Information Visualization*, 1(2):103–110, 2002.

[36] X. Zhang. Drawing complete multipartite graphs on the plane with restrictions on crossings. *Acta Mathematica Sinica, English Series*, 30(12):2045–2053, 2014.