

# Axis-Parallel Right Angle Crossing Graphs

Patrizio Angelini ✉ 

John Cabot University, Rome, Italy

Michael A. Bekos ✉ 

Department of Mathematics, University of Ioannina, Greece

Julia Katheder ✉ 

Wilhelm-Schickard-Institut für Informatik, Universität Tübingen, Germany

Michael Kaufmann ✉ 

Wilhelm-Schickard-Institut für Informatik, Universität Tübingen, Germany

Maximilian Pfister ✉ 

Wilhelm-Schickard-Institut für Informatik, Universität Tübingen, Germany

Torsten Ueckerdt ✉ 

Institute of Theoretical Informatics, Karlsruhe Institute of Technology, Germany

---

## Abstract

A RAC graph is one admitting a RAC drawing, that is, a polyline drawing in which each crossing occurs at a right angle. Originally motivated by psychological studies on readability of graph layouts, RAC graphs form one of the most prominent graph classes in beyond planarity.

In this work, we study a subclass of RAC graphs, called axis-parallel RAC (or apRAC, for short), that restricts the crossings to pairs of axis-parallel edge-segments. apRAC drawings combine the readability of planar drawings with the clarity of (non-planar) orthogonal drawings. We consider these graphs both with and without bends. Our contribution is as follows: (i) We study inclusion relationships between apRAC and traditional RAC graphs. (ii) We establish bounds on the edge density of apRAC graphs. (iii) We show that every graph with maximum degree 8 is 2-bend apRAC and give a linear time drawing algorithm. Some of our results on apRAC graphs also improve the state of the art for general RAC graphs. We conclude our work with a list of open questions and a discussion of a natural generalization of the apRAC model.

**2012 ACM Subject Classification** Mathematics of computing → Graph algorithms; Theory of computation → Computational geometry

**Keywords and phrases** Graph drawing, RAC graphs, Graph drawing algorithms

**Digital Object Identifier** 10.4230/LIPIcs.ESA.2023.9

**Related Version** *arXiv*: <https://arxiv.org/abs/2306.17073>

**Funding** The work of J. Katheder and M. Kaufmann is supported by DFG grant Ka 812-18/2.

## 1 Introduction

Planar graphs form a fundamental graph class in algorithms and graph theory. This is due to the fact that planar graphs have many useful properties, e.g., they are closed under minors and have a linear number of edges. Several decision problems, which are NP-complete for general graphs, become polynomial-time tractable, when restricted to planar inputs, e.g. [28]. As a result, the corresponding literature is tremendously large.

A recent attempt to extend this wide knowledge from planar to non-planar graphs was made in the context of *beyond-planarity*, informally defined as a generalization of planarity encompassing several graph-families that are close-to-planar in some sense (e.g., by imposing structural restrictions on corresponding drawings). Notable examples are the classes of



© Patrizio Angelini, Michael A. Bekos, Julia Katheder, Michael Kaufmann, Maximilian Pfister, and Torsten Ueckerdt;

licensed under Creative Commons License CC-BY 4.0

31st Annual European Symposium on Algorithms (ESA 2023).

Editors: Inge Li Gørtz, Martin Farach-Colton, Simon J. Puglisi, and Grzegorz Herman; Article No. 9; pp. 9:1–9:15

Leibniz International Proceedings in Informatics



Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

(i)  $k$ -planar graphs [31], in which each edge cannot be crossed more than  $k$  times, (ii)  $k$ -quasi-planar graphs [1], which disallow  $k$  pairwise crossing edges, and (iii)  $k$ -gap planar graphs [9], in which each crossing is assigned to one of the two involved edges such that each edge is assigned at most  $k$  of its crossings. For an overview refer to the recent textbook [29].

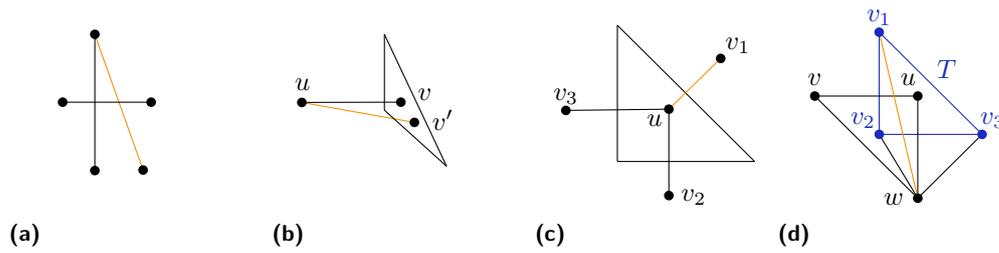
While all of the aforementioned graph-classes are topological, meaning that the actual geometry of the graph's elements is not important, there is a single class proposed in the literature that is purely geometric. The motivation for its study primarily stems from cognitive experiments indicating that the negative effect of edge crossings in a graph drawing tends to be eliminated when the angles formed at the edge crossings are large [30]. In that aspect, the class of *right-angle-crossing* (RAC) graphs forms the optimal case in this scenario, where all crossing angles occur at  $90^\circ$ . Formally, it was introduced by Didimo, Eades and Liotta [22] a decade ago, and since then it has been a fruitful subject of intense research [5, 17, 19, 23, 25].

Generally speaking, the research on RAC graphs has focused on two main research directions depending on whether bends are allowed along the edges or not. Formally, in a  $k$ -bend RAC drawing of a graph each edge is a polyline with at most  $k$  bends and the angle between any two crossing edge-segments is  $90^\circ$ . Accordingly, a  $k$ -bend RAC graph is one admitting such a drawing. A 0-bend RAC graph (or simply RAC graph) with  $n$  vertices has at most  $4n - 10$  edges [22], that is, at most  $n - 4$  edges more than those of a corresponding maximal planar graph. The edge-density bounds for 1- and 2-bend RAC graphs are  $5.5n - 10$  [2] and  $74.2n$  [8], respectively, while for  $k \geq 3$  it is known that every graph is  $k$ -bend RAC [25]. The research on RAC graphs, however, is not limited to edge-density bounds. Several algorithmic and combinatorial results [5, 7, 6, 18, 21, 25], as well as relationships with other graph classes [10, 13, 15, 16, 23, 14] are known; see [20] for a survey.

In this work, we continue the study of RAC graphs along a new and intriguing research line. Inspired by several well-established models for representing graphs (including the widely-used orthogonal model [12, 26, 27]), we introduce and study a natural subfamily of  $k$ -bend RAC graphs, which restricts all edge segments involved in crossings to be axis-parallel. We call this class  $k$ -bend apRAC. We expect that this restriction will further enhance the readability of the obtained drawings, as these combine the simple nature of the planar drawings with the clarity of the (non-planar) orthogonal drawings by allowing non axis-parallel edge segments, only when those are crossing-free. We further expect that our restriction will lead to new results of algorithmic nature. As a matter of fact, almost all algorithms that have been already proposed in the literature about  $k$ -bend RAC graphs in fact yield  $k$ -bend apRAC drawings [11, 22, 25]; e.g., every Hamiltonian degree-3 graph is 0-bend apRAC [6], while degree-4 and degree-6 graphs are 1- and 2-bend apRAC, respectively [3, 5].

Our contribution is as follows:

- In Section 2 we study preliminary properties of 0-bend apRAC graphs in order to prove that recognizing 0-bend apRAC graphs is NP-hard (see Theorem 3).
- We study whether  $k$ -bend apRAC graphs form a proper subclass of  $k$ -bend RAC graphs: For  $k = 0$ , we establish a strict inclusion relationship with  $K_6$  minus one edge being the smallest graph separating the two classes. Further, our edge-density result for 1-bend apRAC graphs establishes a strict inclusion relationship for  $k = 1$ , see Corollary 5. The case  $k = 2$  is more challenging (due to the degrees of freedom introduced by bends) and we leave it as an open problem. For  $k \geq 3$ , the two classes coincide, as the construction establishing that every graph is 3-bend RAC [22] can be converted to 3-bend apRAC by a rotation of  $45^\circ$ .



■ **Figure 1** Forbidden configurations by Properties 1–4.

- We establish bounds on the edge density of  $n$ -vertex  $k$ -bend apRAC graphs: For  $k = 0$ , we prove an upper bound of  $4n - \sqrt{n} - 6$  and give a corresponding lower bound construction with  $4n - 2\lfloor\sqrt{n}\rfloor - 7$  edges (see Theorem 1). For  $k \in \{1, 2\}$ , we give linear upper bounds that are tight up to small additive constants (see Theorems 4 and 6). Notably, for  $k = 2$  our lower-bound construction is a graph with  $n$  vertices and  $10n - \mathcal{O}(1)$  edges. This bound extends to general 2-bend RAC graphs and improves the previous best one of  $7.83n - \mathcal{O}(\sqrt{n})$  [8], answering an open question in [2].
- We show that every graph with maximum degree 8 is 2-bend apRAC and give a linear time drawing algorithm (see Theorem 8) improving the previous best known result stating that 7-edge colorable degree-7 graphs are 2-bend (ap)RAC [3].
- Inspired by the slope-number of graphs, in Section 7 we discuss a natural generalization of apRAC drawings where each edge segment involved in a crossing is parallel or perpendicular to a line having one out of  $s$  different slopes.

## 2 Preliminaries

Throughout this paper, basic graph drawing concepts are used as found in [29, 32]. Let  $G$  be a graph and  $\Gamma$  be a polyline drawing of  $G$  and let  $e = (u, v)$  be an edge of  $G$ . We say that  $e$  uses a horizontal (vertical) port at  $u$  if the edge-segment of  $e$  that is incident to  $u$  is parallel to the  $x$ -axis (to the  $y$ -axis) in  $\Gamma$ . If  $e$  uses neither a vertical nor a horizontal port at  $u$ , then it uses an *oblique* port at  $u$ . In particular, we denote the four orthogonal ports (i.e., the vertical and the horizontal ports) as  $\{N, E, S, W\}$ -ports according to compass directions. In a polyline drawing, vertices and bends are placed on grid-points, whereby the area of the drawing is determined by the smallest rectangular bounding box that contains the drawing. In the following, we recall two properties that hold for 0-bend RAC drawings.

► **Property 1** (Didimo, Eades and Liotta [22]). *In a 0-bend RAC drawing no edge is crossed by two adjacent edges (see Fig. 1a).*

► **Property 2** (Didimo, Eades and Liotta [22]). *A 0-bend RAC drawing does not contain a triangle  $T$  formed by edges of the graph and two edges  $(u, v)$  and  $(u, v')$ , such that  $u$  lies outside  $T$  while both  $v$  and  $v'$  lie inside  $T$  (see Fig. 1b).*

Next, we establish two properties limited to 0-bend apRAC drawings.

► **Property 3.** *A 0-bend apRAC drawing does not contain a triangle  $T$  formed by edges of the graph and three vertices  $v_1, v_2, v_3$  adjacent to a vertex  $u$ , such that  $v_1, v_2, v_3$  lie outside  $T$  and  $u$  lies inside  $T$  (see Fig. 1c).*

**Proof.** Assuming the contrary, Property 1 implies that no two edges adjacent to  $u$  cross the same boundary edge of  $T$ . Hence,  $T$  consists of three axis-parallel edges; a contradiction. ◀

► **Property 4.** *Let  $\Gamma$  be a 0-bend apRAC drawing containing a triangle  $T$  formed by edges of the graph and two adjacent vertices  $u$  and  $v$  such that  $u$  is contained inside  $T$  while  $v$  is outside  $T$ . Then,  $\Gamma$  does not contain a vertex  $w$  adjacent to  $u$ ,  $v$  and all vertices of  $T$  (see Fig. 1d).*

**Proof.** For the sake of contradiction, assume there is a vertex  $w$  adjacent to  $u$ ,  $v$  and all vertices of  $T$ . If  $w$  is inside  $T$  in  $\Gamma$ , then  $(v, u)$  and  $(v, w)$  violate Property 2; a contradiction. Otherwise, since  $(u, v)$  and  $(u, w)$  cross  $T$ , by Property 1, it follows that  $T$  is a right-angled triangle whose legs are axis parallel. W.l.o.g., let  $(v_1, v_2)$  and  $(v_2, v_3)$  be the legs of  $T$  crossed by  $(u, v)$  and  $(u, w)$ , respectively, such that  $(v_1, v_2)$  is horizontal and  $(v_2, v_3)$  is vertical. It follows that the edge  $(v_2, v_3)$  of  $T$  is crossed by  $(u, w)$  and  $(w, v_1)$  violating Property 1; a contradiction. ◀

In Theorem 3 we leverage the following property shown in [7] of the so-called *augmented square antiprism graph*. The gadget used in the NP-hardness proof of Theorem 3 is depicted in Fig. 2a, while the vertex-colored subgraph in Fig. 2a corresponds to the augmented square antiprism graph.

► **Property 5** (Argyriou, Bekos, Symvonis [7]). *Any straight-line RAC drawing of the augmented square antiprism graph has two combinatorial embeddings.*

### 3 0-bend apRAC graphs

In this section, we focus on properties of 0-bend apRAC graphs. We start with an almost tight bound on the edge-density of 0-bend apRAC graphs - for comparison, recall that  $n$ -vertex 0-bend RAC graphs have at most  $4n - 10$  edges [22].

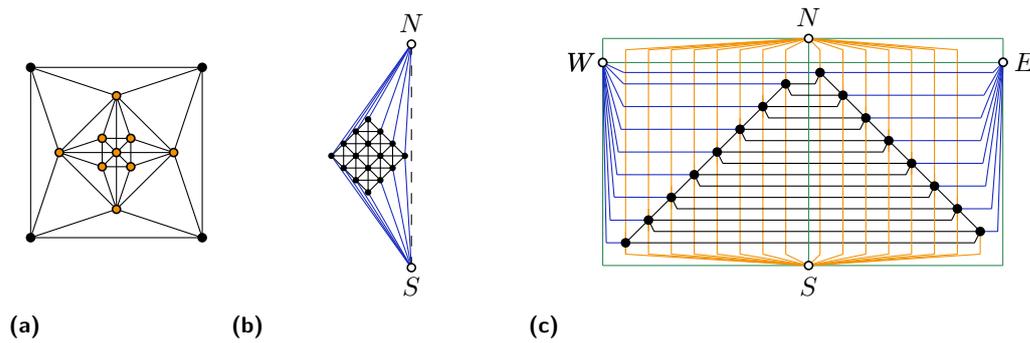
► **Theorem 1.** *A 0-bend apRAC graph with  $n$  vertices has at most  $4n - \sqrt{n} - 6$  edges. Also, there is an infinite family of graphs with  $4n - 2\lfloor\sqrt{n}\rfloor - 7$  edges that admit 0-bend apRAC drawings.*

**Proof.** For the upper bound consider any 0-bend apRAC drawing  $\Gamma$  of a graph  $G$  with  $n$  vertices. As a  $(k \times k)$ -grid has only  $k^2$  grid points, we may assume without loss of generality that the vertices of  $G$  use at least  $\sqrt{n}$  different  $y$ -coordinates in  $\Gamma$ . It follows that the subgraph  $G_h$  of  $G$  defined by the set  $E_h$  of all horizontal edges of  $\Gamma$  is a forest of paths with at least  $\sqrt{n}$  components; at least one for each used  $y$ -coordinate. Thus  $|E_h| \leq n - \sqrt{n}$ . As  $G - E_h$  is crossing-free in  $\Gamma$ , it has at most  $3n - 6$  edges, giving the desired upper bound of  $4n - \sqrt{n} - 6$  edges for  $G$ .

For the lower bound, consider the construction shown in Fig. 2b. For any even  $k > 0$ , construct a  $k \times k$  grid graph  $H_k$  which contains a pair of crossing edges in every quadrangular face. Let  $G_k$  be the graph obtained from  $H_k$  by adding two extremal adjacent vertices  $N$  and  $S$  connected to  $2k - 1$  consecutive boundary vertices of  $H_k$  each (refer to the blue edges in Fig. 2b and observe that the edge between  $N$  and  $S$  can be added by moving  $N$  upwards and to the right and  $S$  downwards and to the right of  $H_k$ ). If we denote by  $n$  the number of vertices of  $G_k$ , then  $n = k^2 + 2$ ,  $k = \sqrt{n - 2}$  and thus  $m = 4n - 2\lfloor\sqrt{n}\rfloor - 7$ . ◀

Since there exist  $n$ -vertex 0-bend RAC graphs with  $4n - 10$  edges, Corollary 2 follows from Theorem 1. In [4], we show that  $K_6$  minus one edge is the smallest graph that is 0-bend RAC but not 0-bend apRAC.

► **Corollary 2.** *The class of 0-bend apRAC graphs is properly contained in the class of 0-bend RAC graphs.*



**Figure 2** (a) Graph used in Theorem 3. (b) Lower bound construction for 0-bend apRAC. (c) Lower bound construction for 1-bend apRAC.

We conclude this section by studying the recognition problem of whether a graph is 0-bend apRAC. Due to space reasons, we only sketch the idea of the proof; the complete proof can be found in [4].

► **Theorem 3.** *It is NP-hard to decide whether a given graph is 0-bend apRAC.*

**Sketch.** We adjust the NP-hardness reduction (from 3-SAT) for the general case of RAC graphs introduced in [7], whose main gadgets use a building block  $\mathcal{H}$ . We determine a 0-bend apRAC graph  $H$  as a substitute for this building block which has the same properties as  $\mathcal{H}$ : (i)  $H$  has a unique embedding, (ii) there are four vertices properly contained in its interior, which can be connected to vertices in its exterior by crossing a single boundary edge, (iii) no edge can (completely) pass through  $H$  without forming a fan crossing, and (iv)  $H$  can be extended horizontally or vertically maintaining properties (i) – (iii). The graph shown in Fig. 2a satisfies all these criteria and can therefore be used for the reduction. ◀

#### 4 1-bend apRAC graphs

In this section, we will establish an upper bound and an almost matching lower bound for the class of 1-bend apRAC graphs. Recall that  $n$ -vertex 1-bend RAC graphs have at most  $5.5n - 10$  edges [2].

► **Theorem 4.** *A 1-bend apRAC graph with  $n$  vertices has at most  $5n - 8$  edges. Also, there is an infinite family of graphs with  $5n - 16$  edges that admit 1-bend apRAC drawings.*

**Proof.** For the upper bound, consider a 1-bend apRAC drawing  $\Gamma$  of an  $n$ -vertex graph  $G$ . Each edge segment in  $\Gamma$  is either horizontal (h), vertical (v) or oblique (o). For  $x, y \in \{h, v, o\}$ , let  $E_{xy}$  be the edges of  $G$  with two edge segments of type  $x$  and  $y$ . Then,  $E_{hv}, E_{ho}, E_{vo}$  and  $E_{oo}$  form a partition of the edge-set of  $G$ , assuming that edges that consist of only one  $h$ -,  $v$ - or  $o$ -segment are counted towards  $E_{ho}, E_{vo}$  and  $E_{oo}$ , respectively. By construction, any crossing involves exactly one vertical and one horizontal segment. Hence, the subgraph of  $G$  induced by  $E_{ho} \cup E_{oo}$  is planar and contains at most  $3n - 6$  edges. Further, as every segment is incident to a vertex and since any vertex is incident to at most two vertical segments, we have  $|E_{vo} \cup E_{hv}| \leq 2n$ . We can assume that the topmost vertex  $v_t$  is incident to at most one vertical edge-segment, since the edge segment incident to  $v_t$  that points upwards cannot be involved in a crossing with a horizontal edge-segment. Otherwise, the endpoint incident to this edge segment would contradict the fact that  $v_t$  is topmost in  $\Gamma$ . Hence, it can be replaced by a steep oblique edge-segment without introducing new crossings. Analogous observations can be made for the bottommost vertex in  $\Gamma$ , which implies that  $|E_{vo} \cup E_{hv}| \leq 2n - 2$ . Thus,  $|E| = |E_{ho}| + |E_{vo}| + |E_{hv}| + |E_{oo}| \leq 5n - 8$ .

Our lower bound construction is as follows; see Fig. 2c. For  $n \geq 7$ , we arrange  $n - 4$  vertices forming a cycle along the two legs of an isosceles triangle with a horizontal base (outer black edges), such that the left leg has  $\lfloor \frac{n-4}{2} \rfloor$  vertices while the right one has  $\lceil \frac{n-4}{2} \rceil$ . These  $n - 4$  vertices are further joined by a  $y$ -monotone path of  $n - 7$  edges (inner black edges). Two extremal vertices  $N$  and  $S$  above and below the triangle are connected to all  $n - 4$  vertices (orange edges). Similarly, two extremal vertices  $W$  and  $E$  to the left and right of the triangle are connected to all vertices of the left and right legs of the triangle respectively (blue edges); the topmost vertex of the right leg is also connected to  $W$ . Finally, we add six edges between the extremal vertices, which gives  $n - 4 + n - 6 + 3(n - 4) + 6 = 5n - 16$  edges. ◀

Since there exists 1-bend RAC graphs with  $5.5n - 72$  edges [2], the following corollary is immediate.

► **Corollary 5.** *The class of 1-bend apRAC graphs is a proper subclass of the one of 1-bend RAC graphs.*

## 5 2-bend apRAC graphs

In Theorem 6, we provide an upper-bound for the edge density of 2-bend apRAC graphs together with a lower-bound construction which is tight up to an additive constant. Our result provides a stark contrast to the one for 2-bend RAC graphs, where the current best upper-bound on the number of edges of  $n$ -vertex graphs is  $74.2n$  [8], while the previous best lower bound-construction contained only  $7.83n - \mathcal{O}(\sqrt{n})$  [8] edges.

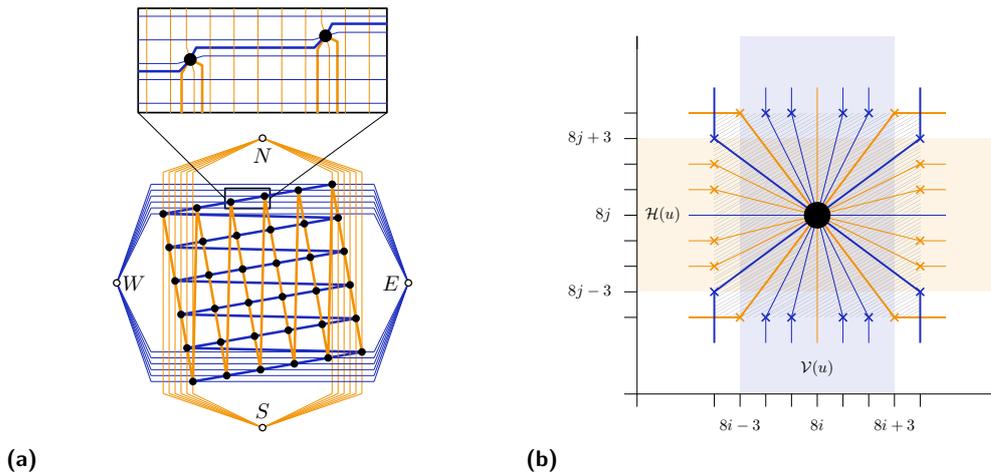
► **Theorem 6.** *A 2-bend apRAC graph with  $n$  vertices has at most  $10n - 12$  edges. Also, there is an infinite family of graphs with  $10n - 46$  edges that admit 2-bend apRAC drawings.*

**Proof.** Consider a 2-bend apRAC drawing  $\Gamma$  of an  $n$ -vertex graph  $G$ . Each edge segment in  $\Gamma$  is either horizontal (h), vertical (v) or oblique (o). Denote by  $S$  the set of edges that contain at least one segment in  $\{h, v\}$  incident to a vertex. Since any vertex is incident to at most two vertical and at most two horizontal segments, it follows that  $|S| \leq 4n$ . Let  $E_h, E_v$  and  $E_o$  be the set of edges of  $E \setminus S$  whose middle part is  $h, v$  and  $o$ , respectively. Assuming that an edge of  $E \setminus S$  consisting of less than three segments belongs to  $E_o$ , it follows that  $E_h, E_v$  and  $E_o$  form a partition of  $E \setminus S$ . Observe that the edges of  $E_o$  cannot be involved in any crossing in  $\Gamma$ , as all of its segments are oblique. Further, no two edges of  $E_h$  or of  $E_v$  can cross. Hence, the subgraphs induced by  $E_h \cup E_o$  and  $E_v \cup E_o$  are planar and contain at most  $3n - 6$  edges each. Recall that  $|S| \leq 4n$  and thus  $|E| \leq |S| + |E_h| + |E_v| + 2|E_o| \leq 4n + 3n - 6 + 3n - 6 = 10n - 12$ .

Refer to Fig. 3a for a schematization of the upcoming lower-bound construction and to Fig. 4 for a concrete example. Fix an integer  $k \geq 6$  and consider a set  $P$  of  $k^2$  points of a  $k \times k$  square grid in the plane but rotated very slightly, say counterclockwise, so that the points in each column have consecutive  $x$ -coordinates (consequently the points in each row have consecutive  $y$ -coordinates). For two points  $p, q \in P$  let their  $x$ -distance  $\text{dist}_x(p, q)$  be the number of points in  $P$  having their  $x$ -coordinate between  $p$  and  $q$ . Similarly define the  $y$ -distance  $\text{dist}_y(p, q)$ . The crucial property of point set  $P$  is the following.

$$\text{For any } p \neq q \in P \text{ we have } \text{dist}_x(p, q) + \text{dist}_y(p, q) \geq k - 1 \geq 5. \tag{1}$$

Between any pair  $p, q \in P$  with consecutive  $x$ -coordinates, i.e.,  $\text{dist}_x(p, q) = 0$ , we add a 2-bend edge with vertical middle segment by starting and ending with a very short oblique segment at  $p$  respectively  $q$ . Similarly, we add a 2-bend edge with horizontal middle segment when  $\text{dist}_y(p, q) = 0$ . Note that these are in total  $2k^2 - 2$  edges, no two of which connect the same pair of points, due to (1).



**Figure 3** (a) Illustration of the construction in Theorem 6 with  $k = 6$ . Edges with two oblique segments are indicated in orange for vertical middle segments and in blue for horizontal middle segments. Edges using the horizontal or vertical ports are omitted for readability. (b) Edge routing in the  $8 \times 8$  box  $B(u)$  of a vertex  $u$ . Blue ports are exclusively used by edges of  $F_1$  and  $F_3$  and orange ports by  $F_2$  and  $F_4$ . Note that the ports illustrated by bold lines are reserved for oblique-2 edges. Bends on the border of the box are emphasized by a cross.

Next we add four additional points  $N, E, S, W$  to the top, right, bottom, and left of all points in  $P$ , respectively. For every point  $p$  we add a 2-bend edge with vertical middle segment between  $p$  and  $N$  starting with a very short oblique segment at  $p$  and ending with an almost horizontal (but still oblique) segment at  $N$ . Similarly, we add a 2-bend edge with vertical middle segment between  $p$  and  $S$ , as well as one with horizontal middle segment to each of  $E, W$ . Note that these are in total  $4k^2$  edges, and that all oblique segments can be chosen such that all crossings involve middle segments only.

Next we add for (almost) each point  $p \in P$  four more 2-bend edges. First, consider for  $p$  the point  $q \in P$  to the right of  $p$  with  $\text{dist}_x(p, q) = 1$ , unless  $p$  is one of the two rightmost points in  $P$ . We draw a 2-bend edge from  $p$  to  $q$  by starting with a horizontal segment at  $p$  to almost the  $x$ -coordinate of  $q$ , continuing with a vertical segment to almost the  $y$ -coordinate of  $q$ , and ending with a very short oblique segment at  $q$ . Similarly, we use the left horizontal port at  $p$  for an edge to the point  $q$  left of  $p$  with  $\text{dist}_x(p, q) = 2$ . (We take  $x$ -distance 2 instead of 1 to avoid introducing a parallel edge.) Symmetrically, we draw two edges using the vertical ports at  $p$ . Note that these are in total  $4k^2 - 10$  edges, and that all crossings involve horizontal and vertical segments only.

Finally, we easily add six edges to create a  $K_4$  on vertices  $N, E, S, W$ . To conclude, we have constructed a 2-bend apRAC graph with  $n = k^2 + 4$  vertices and  $(2k^2 - 2) + 4k^2 + (4k^2 - 10) + 6 = 10k^2 - 6 = 10n - 46$  edges. ◀

## 6 Every graph with maximum degree 8 is 2-bend apRAC

In the following, we prove that graphs with maximum degree 8 admit 2-bend apRAC drawings of quadratic area which can be computed in linear time. We leverage the following result in order to decompose the input graph.

► **Lemma 7** (Eades, Symvonis, Whitesides [24]). *Let  $G = (V, E)$  be an  $n$ -vertex undirected graph of degree  $\Delta$  and let  $d = \lceil \Delta/2 \rceil$ . Then, there exists a directed multigraph  $G' = (V, E')$  with the following properties:*

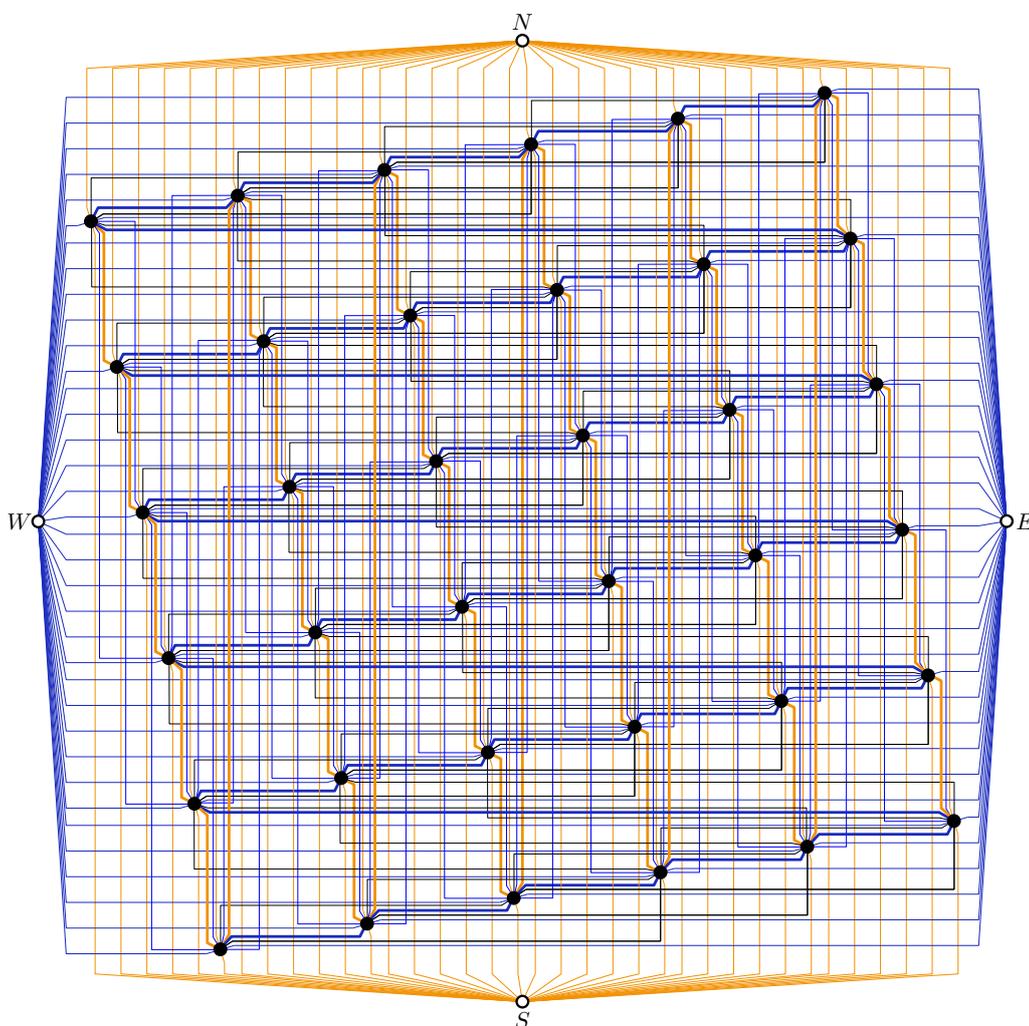
9:8 Axis-Parallel Right Angle Crossing Graphs

1. each vertex of  $G'$  has indegree  $d$  and outdegree  $d$ ;
  2.  $G$  is a subgraph of the underlying undirected graph of  $G'$ ; and
  3. the edges of  $G'$  can be partitioned into  $d$  edge-disjoint directed 2-factors (where a 2-factor is a spanning subgraph of  $G'$  consisting of vertex disjoint cycles, called cycle cover in [24]).
- The directed graph  $G'$  and the  $d$  2-factors can be computed in  $\mathcal{O}(\Delta^2 n)$  time.

Now, we are ready to state the main result.

► **Theorem 8.** *Given a graph  $G$  with maximum degree 8 and  $n$  vertices, it is possible to compute in  $\mathcal{O}(n)$  time a 2-bend apRAC drawing of  $G$  with  $\mathcal{O}(n^2)$  area.*

**Proof.** Let  $G$  be a simple graph with maximum degree 8 and  $n$  vertices. We apply Lemma 7 to augment  $G$  to a **directed** 8-regular multigraph having four edge-disjoint 2-factors  $F_1, F_2, F_3$  and  $F_4$ . Before we present our algorithm in full detail, we sketch an outline of the necessary steps. We want to stress that in the following, the direction of an edge  $(u, v)$  plays an important role and hence we consider it as a directed edge with source  $u$  and target  $v$ .



■ **Figure 4** Illustration of the construction in Theorem 6 with  $k = 6$ . The  $K_4$  on the vertices  $N, E, S, W$  is omitted due to space reasons.

## 6.1 Outline of the algorithm

In the first step, we will construct two total orders  $\prec_x$  and  $\prec_y$  of the vertices of  $G$  which will determine the  $x$ - and  $y$ -coordinates of the vertices in the final drawing. In particular, if vertex  $u$  of  $G$  has the  $i$ -th position in  $\prec_x$  and the  $j$ -th position in  $\prec_y$ , then  $u$  will be placed at point  $(8i, 8j)$  in the final drawing. We will construct these two orders independently such that  $\prec_x$  is defined by  $F_1 \cup F_3$  and  $\prec_y$  is defined by  $F_2 \cup F_4$ . After the computation of  $\prec_x$  and  $\prec_y$ , which finalizes the position of the vertices in our resulting drawing  $\Gamma$ , it remains to draw the edges which are fully characterized by the placement of the respective bend-points. Every edge will be drawn with exactly three segments, which are either horizontal, vertical or oblique. To ensure that all crossings in  $\Gamma$  occur between horizontal and vertical segments, we will restrict oblique segments to be “short” (a precise definition follows below) and require that they are incident to a vertex. To this end, we will define, for each vertex  $u$  of  $G$ , a closed box  $B(u)$  centered at  $u$  of size  $8 \times 8$ , such that the oblique segments incident to  $u$  are fully contained inside  $B(u)$ . Note that by construction, the interior of two boxes do not overlap (they may touch at a corner). Since the  $x$ -coordinate of two consecutive vertices  $u$  and  $v$  of  $\prec_x$  differs by exactly 8, there is a vertical line that is (partially) contained inside both  $B(u)$  and  $B(v)$  (analogous for a horizontal line and consecutive vertices in  $\prec_y$ ). This allows us to join  $u$  and  $v$  by an edge that consists of two oblique segments, which is called an *oblique-2* edge. If the unique orthogonal segment of an oblique-2 edge is vertical (horizontal), we will refer to it as a vertical (horizontal) oblique-2 edge. An edge that contains exactly one oblique segment will analogously be called an *oblique-1* edge.

In the second step, we will classify every edge of  $G$  as either an oblique-1 or an oblique-2 edge - again this classification is done independently for  $F_1 \cup F_3$  and  $F_2 \cup F_4$ ; we focus on the description of  $F_1 \cup F_3$ , the other one is symmetric. Let  $e = (u, v)$  be an edge of  $F_1 \cup F_3$ . If  $u$  and  $v$  are consecutive in  $\prec_x$ , then  $e$  is classified as a vertical oblique-2 edge. Otherwise,  $e$  is classified as an oblique-1 edge such that the (unique) oblique segment is incident to the target  $v$ , while the orthogonal segment at  $u$  uses the  $E$ -port at  $u$  if  $u \prec_x v$ , otherwise it uses the  $W$ -port.

In the final step, we will specify the exact coordinates of the bend-points. At a high level, oblique segments (which are by construction all incident to vertices) will end at the boundary of the corresponding box, see Fig. 3b. The bend-points between vertical and horizontal segments are then naturally defined by the intersections of their corresponding lines.

The final drawing  $\Gamma$  will then satisfy the following two properties.

- (i) No bend-point of an edge lies on another edge and
- (ii) the edges are drawn with two bends each so that only the edge segments that are incident to  $u$  are contained in the interior of  $B(u)$ , while all the other edge segments are either vertical or horizontal.

This will guarantee that the resulting drawing is 2-bend RAC; for an example see Fig. 5. Note that (i) guarantees that no two segments have a non-degenerate overlap.

## 6.2 Computing $\prec_x$ and $\prec_y$

We will now describe how to construct  $\prec_x$  and  $\prec_y$  explicitly. We focus on the construction of  $\prec_x$  which is based on  $F_1$  and  $F_3$ , the order  $\prec_y$  can be constructed analogously. Let  $C_1, C_2, \dots, C_k$  be an arbitrary ordering of the components of  $F_1$ . Recall that by definition, each such  $C_i$  is a directed cycle. Let  $S$  be a set of vertices that contains exactly one arbitrary vertex from each cycle in  $F_1$  and let  $P_1, P_2, \dots, P_k$  be the resulting directed paths obtained by restricting the cycles to  $V \setminus S$ . Note that this may yield paths that are empty, i.e., when the

corresponding cycle consists of a single vertex. We construct  $\prec_x$  (limited to  $V \setminus S$ ) such that the vertices of each path appear consecutively defined by the unique directed walk from one endpoint of the path to the other. The relative order between paths is  $P_1 \prec_x P_2 \prec_x \dots \prec_x P_k$ . Hence it remains to insert the vertices of  $S$  into  $\prec_x$ . Throughout the algorithm, we will maintain the following invariant which will ensure the correctness of our approach.

**I.1** Let  $u \in S$  be a vertex of cycle  $C_i$ . If  $|C_i| > 1$ , then  $u$  is placed next to at least one vertex of  $P_i$ . Otherwise,  $u$  is placed directly after the last vertex of  $C_{i-1}$  (or as first vertex if  $i = 1$ ) in  $\prec_x$ .

If I.1 is maintained, we can guarantee the following observation.

► **Observation 1.** Let  $u \in C_i$  and  $v \in C_j$  be two vertices of  $G$  with  $i \neq j$ . Then, the relative order of  $u$  and  $v$  in  $\prec_x$  is known.

Assume that each vertex in  $S$  that belongs to  $C_1, \dots, C_{i-1}$  has been inserted in  $\prec_x$ . Let  $u \in S$  be the vertex that belongs to  $C_i \setminus P_i$ . If  $|C_i| \leq 2$ , then we place  $u$  immediately after the last vertex of  $C_{i-1}$  in  $\prec_x$  if  $i > 1$ , otherwise  $u$  is the first vertex of  $\prec_x$  which maintains I.1. Hence, in the remainder we can assume that  $C_i$  consists of at least three vertices. Let  $a, b$  and  $c$  be the vertices of  $G$  such that  $(u, a), (b, u) \in F_1$  and  $(u, c) \in F_3$ . Even though  $G$  is a multigraph, we have that  $a \neq b$  since  $C_i$  contains at least three vertices. Hence, by construction we have  $a \prec_x b$  - in particular,  $a$  is the first vertex of  $P_i$  in  $\prec_x$ , while  $b$  is the last one. Let  $C_j$  (possibly  $j = i$ ) be the cycle that contains  $c$ . Note that it is possible that  $c \in S$ , i.e.,  $c$  is not part of  $\prec_x$  initially. However, as this can only happen if  $i \neq j$ , we know the relative position of  $u$  and  $c$  by Observation 1. We distinguish between the following cases based on the relative order of cycle  $C_i$  (which contains  $u$ ) and cycle  $C_j$  (which contains  $c$ ) in  $\prec_x$ .

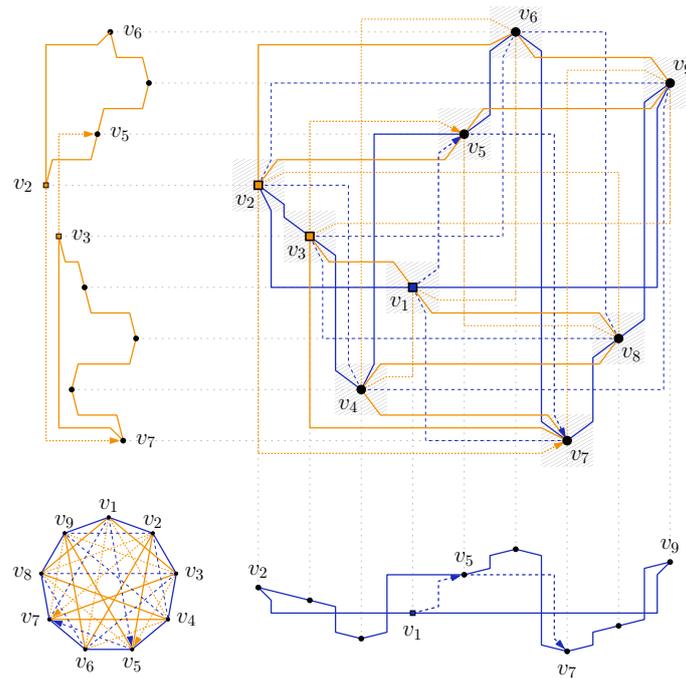
1.  $j < i$ . We insert  $u$  immediately before  $a$  in  $\prec_x$  such that it is the first vertex of  $C_i$ , see Fig. 6a. Clearly, this maintains I.1.
2.  $i < j$ . This case is symmetric to the previous one - we insert  $u$  immediately after  $b$  in  $\prec_x$  such that it is the last vertex of  $C_i$ , see Fig. 6b, which again maintains I.1.
3.  $i = j$ . In this case, we have that  $c$  also belongs to  $C_i$  (in particular,  $c$  belongs to  $P_i$  and thus is already part of  $\prec_x$ ). If  $c = a$  or  $c = b$ , we simply omit the edge  $(u, c)$  and proceed as in the first case, i.e., we place  $u$  as the first vertex of  $C_i$ . Otherwise, we insert  $u$  directly before or directly after  $c$  in  $\prec_x$  based on the edge  $(c, d) \in F_3$ . The relative order of  $c$  and  $d$  in  $\prec_x$  is known by Observation 1 unless  $d \in C_i$ . If  $d \in P_i$ , the relative order between  $c$  and  $d$  is also known (as both are already present in  $\prec_x$ ). If  $d \notin P_i$ , then  $d = u$  and we can omit the edge  $(u, c) \in F_3$  (because it is a copy of  $(c, d) \in F_3$ ), in which case we can again proceed as in the first case. Hence,  $d \neq u$  holds. If  $c \prec_x d$ , we insert  $u$  directly before  $c$  in  $\prec_x$ , see Fig. 6c, otherwise we insert  $u$  directly after  $c$  in  $\prec_x$ . In both cases, we maintain I.1.

This concludes our construction of  $\prec_x$ .

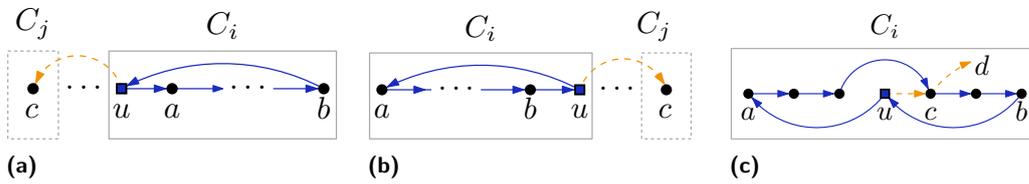
### 6.3 Classification of the edges and port assignment

We focus on the classification of the edges of  $F_1 \cup F_3$  and their port assignment, the classification of the edges of  $F_2 \cup F_4$  is analogous. Our classification will maintain the following invariants.

- I.2** The endpoints of each vertical oblique-2 edge are consecutive in  $\prec_x$ .
- I.3** Each oblique-1 edge  $(u, v) \in F_1 \cup F_3$  is assigned the  $W$ -port at its source vertex  $u$ , if  $v \prec_x u$ ; otherwise, if  $u \prec_x v$ , it is assigned the  $E$ -port at  $u$ .
- I.4** Every horizontal port is assigned at most once.



■ **Figure 5** A 2-bend apRAC drawing of  $K_9$ ;  $F_1$  and  $F_3$  are blue;  $F_2$  and  $F_4$  are orange. Below the drawing of  $K_9$  there is a illustration of the cycles in  $F_1$  and the relevant edges in  $F_3$  for positioning  $v_1 \in S$  according to Case 3 in the construction of  $\prec_x$ . Similarly, a visualization of the cycles in  $F_2$  and the relevant edges in  $F_4$  is displayed to the left.



■ **Figure 6** Illustration of the construction of  $\prec_x$ , Case 1 is shown in (a), Case 2 in (b) and Case 3 in (c). Blue edges belong to  $F_1$ , while dashed orange edges belong to  $F_3$ .

Let us consider an edge  $e \in F_1 \cup F_3$  between vertices  $u$  and  $v$ . If  $u$  and  $v$  are consecutive in  $\prec_x$ , then we classify  $e$  as a vertical oblique-2 edge. If  $u$  and  $v$  are not consecutive in  $\prec_x$ , we will classify  $e$  as an oblique-1 edge, which therefore guarantees I.2. For any oblique-1 edge, we will, in an initial phase, assign the ports precisely as stated in I.3. In a subsequent step, we will create a unique assignment of the horizontal ports by reorienting some edges of  $F_1 \cup F_3$  in order to guarantee I.4. Suppose that after the initial assignment, there exists a vertex  $u$  such that one of its orthogonal ports is assigned to two oblique-1 edges. Assume first the  $W$ -port of  $u$  is assigned to edges  $(u, a)$  and  $(u, b)$ . By construction,  $u$  has exactly one outgoing edge in  $F_1$ , say  $(u, a)$ , and exactly one outgoing edge in  $F_3$ , say  $(u, b)$ . Let  $C_i$  be the cycle of  $F_1$  that contains both  $u$  and  $a$  (which implies that  $|C_i| > 1$ , as we omit self-loops) and let  $C_j$  be the cycle that contains  $b$  (possibly  $i = j$ ). Recall that by construction, the vertices of  $P_i$  appear consecutively in  $\prec_x$  before the insertion of the vertex  $v \in C_i \setminus P_i$ . Since  $(u, a)$  is an oblique-1 edge, we have that  $u$  and  $a$  are not consecutive in  $\prec_x$ . If  $|C_i| = 2$ , one of  $u$  or  $a$  coincides with  $v$ , but then  $u$  and  $a$  are consecutive in  $\prec_x$  and thus the edge  $(u, a)$  is

an oblique-2 edge. Hence,  $|C_i| > 2$  holds and we either have  $u = v$ ,  $a = v$  or  $v$  was inserted directly in between  $a$  and  $u$ . In the following, we will refer to Cases 1 - 3 of Section 6.2, where we computed the total order  $\prec_x$ .

1.  $u = v$ . Assume first that  $C_i \neq C_j$ . Then, since  $(u, b)$  is assigned the  $W$ -port at  $u$ , we have  $b \prec_x u$  by I.3 which implies  $j < i$  and hence we placed  $u$  according to Case 1, i.e., as the first vertex of  $C_i$  in  $\prec_x$ . But since  $a \in C_i$ , we then have  $u \prec_x a$  and thus  $(u, a)$  would use the  $E$ -port at  $u$ , a contradiction.  
Hence assume that  $C_i = C_j$ , i.e.,  $b \in C_i$ . Then we are in Case 3. In particular, we placed  $u$  such that  $u$  and  $b$  are consecutive, thus  $(u, b)$  is classified as an oblique-2 edge, again we obtain a contradiction.
2.  $a = v$ . Since  $(u, a)$  uses the  $W$ -port at  $u$  by assumption, we have that  $a \prec_x u$  by I.3 and thus  $a$  cannot be the last vertex of  $C_i$  in  $\prec_x$ , and so we are in Case 1 or 3. In Case 1,  $a$  is placed as the first vertex of  $C_i$  since there exists a vertex  $a'$  with  $a' \prec_x a$  such that  $(a, a') \in F_3$ . Further,  $a$  is placed next to vertex  $v'$  (i.e., the first vertex of  $P_i$  in  $\prec_x$ ) with  $(a, v') \in F_1$  by construction. Then, we can redirect the edge  $(u, a) \in F_1$  such that we can assign  $(a, u)$  the  $E$ -port at  $a$  which solves the conflict at  $u$  and does not introduce a conflict at  $a$  which guarantees I.4. In Case 3,  $a$  was placed consecutive to vertex  $a' \in P_i$  with  $(a, a') \in F_3$ . As  $(u, a)$  uses the  $W$ -port at  $u$ ,  $u$  is necessarily the last vertex of  $C_i$  in  $\prec_x$ . Since the other neighbor of  $a$  in  $F_1$  different from  $u$  is the first vertex of  $C_i$  in  $\prec_x$ , i.e., it precedes  $a$  in  $\prec_x$ , we can again reorient the edge  $(u, a)$  and assign the edge  $(a, u)$  to the free  $E$ -port of  $a$ , solving the conflict at  $u$  which guarantees I.4.
3.  $v$  was inserted directly in between  $a$  and  $u$ . In this case, we have that both  $a$  and  $u$  belong to  $P_i$ . Since we assume that  $(u, a)$  uses the  $W$ -port at  $u$ , it follows that  $a \prec_x u$  holds. But then by construction, the edge of  $F_1$  that joins  $a$  and  $u$  is directed from  $a$  to  $u$  and we obtain a contradiction.

The case where the  $E$ -port of  $u$  is assigned to two edges can be solved in a similar way; refer to [4] for details.

Observe that if an edge  $(u, v)$  was redirected, then both  $u$  and  $v$  belong to the same cycle  $C_i$  of  $F_1$  and since this operation has to be performed at most once per cycle, it follows that they can be considered independently. So far, we have computed  $\prec_x$  and classified every edge of  $F_1 \cup F_3$  guaranteeing Invariants 1-I.4. Symmetrically, we can compute  $\prec_y$  and classify every edge of  $F_2 \cup F_4$  guaranteeing corresponding versions of Invariants 1-I.4; see [4] for details.

## 6.4 Bend placement

We begin by describing how to place the bends of the edges on each side of the box  $B(u)$  of an arbitrary vertex  $u$  based on the type of the edge that is incident to  $u$ , refer to Fig. 3b. Let  $(x_u, y_u)$  be the coordinates of  $u$  in  $\Gamma$  that are defined by  $\prec_x$  and  $\prec_y$ . Recall that the box  $B(u)$  has size  $8 \times 8$ . Let  $e$  be an edge incident to  $u$ . We focus on the case in which  $e \in F_1 \cup F_3$ , the other case in which  $e$  belongs to  $F_2 \cup F_4$  is handled symmetrically by simply exchanging  $x$  with  $y$ , “top/bottom” with “right/left” and “vertical” with “horizontal” from the following description. By definition,  $e$  is either an oblique-1 edge or a vertical oblique-2 edge. Suppose first that  $e$  is an oblique-1 edge. If  $e = (u, v)$ , i.e.,  $e$  is an outgoing edge of  $u$  in  $F_1 \cup F_3$ , then by Invariant I.3 edge  $e$  uses either the  $W$ - or  $E$ -port at  $u$ . In the former case, the segment of  $e$  incident to  $u$  passes through point  $(y_u, y_u - 4)$ , while in the latter case it passes through point  $(y_u, y_u + 4)$ . For an example, refer to the outgoing edge  $(v_3, v_6)$  of  $v_3$  in Fig. 5. If  $e = (v, u)$ , i.e.,  $e$  is an incoming edge of  $u$  in  $F_1 \cup F_3$ , then by Invariant I.3  $e$  uses a horizontal port at  $v$  and by the fact that every edge consists of exactly three segments, the

vertical segment of  $e$  ends at the top or the bottom side of  $B(u)$ . Since any vertex has at most three incoming edges in  $F_1 \cup F_3$  by construction, we can place the respective bends at  $x$ -coordinate  $x_u + i$  with  $i \in \{-2, -1, 1, 2\}$  and  $y$ -coordinate  $y_u + 4$  ( $y_u - 4$ ) for the top (bottom) side such that the assigned  $i$ -value is unique, refer to the incoming edge  $(v_4, v_9)$  of  $v_9$  in Fig. 5, where  $i = -1$ . Finally, the other bend-point of  $e$  is uniquely defined as  $(x_u + i, y_v)$ , since it connects a vertical with a horizontal segment by construction.

Suppose now that  $e$  is a vertical oblique-2 edge. By I.2,  $u$  and  $v$  are consecutive in  $\prec_x$ . If  $v \prec_x u$  the  $x$ -coordinate of the bend point is  $x_u - 4$ , otherwise it is  $x_u + 4$ ; e.g., refer to the edges  $(v_2, v_3)$  and  $(v_3, v_4)$  of  $v_3$  in Fig. 5, respectively. In order to define the  $y$ -coordinate of the bend point, we have to consider the relative position of  $u$  and  $v$  in  $\prec_y$ . If  $v \prec_y u$  the  $y$ -coordinate of the bend point of  $e$  is  $y_u - 3$  and otherwise it is  $y_u + 3$ . I.2 implies that any vertex has at most two vertical oblique-2 edges since no vertex has more than two direct neighbors in  $\prec_x$ . From the description of the bend-points, the observation follows:

► **Observation 2.** *Let  $b$  be a bend-point that delimits an oblique segment  $s$  which belongs to an edge  $e$ . If  $s$  is incident to  $u$ , then  $b$  does not lie on any other edge incident to  $u$ .*

## 6.5 Proof of correctness

The fact that the obtained drawing is 2-bend apRAC is proved in [4]. To complete the proof of Theorem 8, we discuss the time complexity and the required area. We apply Lemma 7 to  $G$  to obtain  $F_1, F_2, F_3$  and  $F_4$  in  $\mathcal{O}(n)$  time. For each cycle of  $F_1$  and  $F_2$ , an appropriate ordering of its internal vertices, the classification of the incident edges and the assignment of the orthogonal ports can be computed in time linear in the size of the cycle. Clearly, computing the bend-points can be done in linear time as well. Hence we can conclude that the drawing can be computed in  $\mathcal{O}(n)$  time. For the area, we can observe that the size of the grid defined by the boxes is  $8n \times 8n$  and by construction, any vertex and any bend point is placed on a distinct point on the grid. ◀

## 7 Conclusion and Open Problems

In this paper, we introduced the class of  $k$ -bend apRAC graphs, gave edge-density bounds, studied inclusion relationships with the general  $k$ -bend RAC graphs, and concluded with an algorithmic result for graphs with maximum degree 8. A natural extension is to allow drawings where each crossing edge-segment is parallel or perpendicular to a line having one out of  $s$  different slopes. We denote the class of graphs which admit such a drawing as  $k$ -bend  $s$ -apRAC, and w.l.o.g. we assume that the horizontal slope is among the  $s$  ones. Observe that for  $s = 1$ , the derived class coincides with the class of  $k$ -bend apRAC graphs. By joining several copies of the graph supporting Property 5 that all share a common vertex, we show that 0-bend  $s$ -apRAC graphs form a proper subclass of 0-bend RAC graphs for any  $s \in o(n)$ ; see [4] for details. We also adjust the proof of Theorem 6 to derive an upper bound on the edge density of 2-bend  $s$ -apRAC graphs, which is better than the one of [8] that holds for general 2-bend RAC graphs for values of  $s$  up to 17. We conclude with the following open problems.

- Are there 2-bend RAC graphs that are not 2-bend apRAC?
- For  $k \in \{1, 2\}$ , our edge-density bounds do not relate to the simplicity of the drawings. Are bounds different for simple drawings, as in the general 1-bend RAC case [2]?
- For  $k \in \{1, 2\}$ , does the class of  $k$ -bend  $s$ -apRAC graphs on  $n$  vertices coincide with the corresponding class of  $k$ -bend RAC graphs, when  $s \in o(n)$ ?

- Based on our exploration of 2-bend apRAC graphs, we conjecture that the edge density of general 2-bend RAC graphs on  $n$  vertices is  $10n - \mathcal{O}(1)$ .
- Another important open problem in the field is to settle the complexity of the recognition of general 1-bend RAC graphs. What if we restrict ourselves to the axis-parallel setting?
- From a practical perspective, can an advantage in the readability of  $k$ -bend apRAC drawings over  $k$ -bend RAC drawings be demonstrated in user studies?

---

## References

- 1 Pankaj K. Agarwal, Boris Aronov, János Pach, Richard Pollack, and Micha Sharir. Quasi-planar graphs have a linear number of edges. *Comb.*, 17(1):1–9, 1997. doi:10.1007/BF01196127.
- 2 Patrizio Angelini, Michael A. Bekos, Henry Förster, and Michael Kaufmann. On RAC drawings of graphs with one bend per edge. *Theor. Comput. Sci.*, 828-829:42–54, 2020. doi:10.1016/j.tcs.2020.04.018.
- 3 Patrizio Angelini, Michael A. Bekos, Julia Katheder, Michael Kaufmann, and Maximilian Pfister. RAC drawings of graphs with low degree. In Stefan Szeider, Robert Ganian, and Alexandra Silva, editors, *MFCS*, volume 241 of *LIPICs*, pages 11:1–11:15. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022. doi:10.4230/LIPICs.MFCS.2022.11.
- 4 Patrizio Angelini, Michael A. Bekos, Julia Katheder, Michael Kaufmann, Maximilian Pfister, and Torsten Ueckerdt. Axis-parallel right angle crossing graphs. *CoRR*, abs/2306.17073, 2023. doi:10.48550/arXiv.2306.17073.
- 5 Patrizio Angelini, Luca Cittadini, Walter Didimo, Fabrizio Frati, Giuseppe Di Battista, Michael Kaufmann, and Antonios Symvonis. On the perspectives opened by right angle crossing drawings. *J. Graph Algorithms Appl.*, 15(1):53–78, 2011. doi:10.7155/jgaa.00217.
- 6 Evmorfia N. Argyriou, Michael A. Bekos, Michael Kaufmann, and Antonios Symvonis. Geometric RAC simultaneous drawings of graphs. *J. Graph Algorithms Appl.*, 17(1):11–34, 2013. doi:10.7155/jgaa.00282.
- 7 Evmorfia N. Argyriou, Michael A. Bekos, and Antonios Symvonis. The straight-line RAC drawing problem is NP-hard. *J. Graph Algorithms Appl.*, 16(2):569–597, 2012. doi:10.7155/jgaa.00274.
- 8 Karin Arikushi, Radoslav Fulek, Balázs Keszegh, Filip Moric, and Csaba D. Tóth. Graphs that admit right angle crossing drawings. *Comput. Geom.*, 45(4):169–177, 2012. doi:10.1016/j.comgeo.2011.11.008.
- 9 Sang Won Bae, Jean-François Baffier, Jinhee Chun, Peter Eades, Kord Eickmeyer, Luca Grilli, Seok-Hee Hong, Matias Korman, Fabrizio Montecchiani, Ignaz Rutter, and Csaba D. Tóth. Gap-planar graphs. *Theor. Comput. Sci.*, 745:36–52, 2018. doi:10.1016/j.tcs.2018.05.029.
- 10 Michael A. Bekos, Walter Didimo, Giuseppe Liotta, Saeed Mehrabi, and Fabrizio Montecchiani. On RAC drawings of 1-planar graphs. *Theor. Comput. Sci.*, 689:48–57, 2017. doi:10.1016/j.tcs.2017.05.039.
- 11 Michael A. Bekos, Thomas C. van Dijk, Philipp Kindermann, and Alexander Wolff. Simultaneous drawing of planar graphs with right-angle crossings and few bends. *J. Graph Algorithms Appl.*, 20(1):133–158, 2016. doi:10.7155/jgaa.00388.
- 12 Therese C. Biedl and Goos Kant. A better heuristic for orthogonal graph drawings. *Comput. Geom.*, 9(3):159–180, 1998. doi:10.1016/S0925-7721(97)00026-6.
- 13 Franz J. Brandenburg, Walter Didimo, William S. Evans, Philipp Kindermann, Giuseppe Liotta, and Fabrizio Montecchiani. Recognizing and drawing IC-planar graphs. *Theor. Comput. Sci.*, 636:1–16, 2016. doi:10.1016/j.tcs.2016.04.026.
- 14 Steven Chaplick, Henry Förster, Myroslav Kryven, and Alexander Wolff. Drawing graphs with circular arcs and right-angle crossings. In Susanne Albers, editor, *17th Scandinavian Symposium and Workshops on Algorithm Theory, SWAT 2020, June 22-24, 2020, Tórshavn, Faroe Islands*, volume 162 of *LIPICs*, pages 21:1–21:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020. doi:10.4230/LIPICs.SWAT.2020.21.

- 15 Steven Chaplick, Fabian Lipp, Alexander Wolff, and Johannes Zink. Compact drawings of 1-planar graphs with right-angle crossings and few bends. *Comput. Geom.*, 84:50–68, 2019. doi:10.1016/j.comgeo.2019.07.006.
- 16 Hooman Reisi Dehkordi and Peter Eades. Every outer-1-plane graph has a right angle crossing drawing. *Int. J. Comput. Geom. Appl.*, 22(6):543–558, 2012. doi:10.1142/S021819591250015X.
- 17 Emilio Di Giacomo, Walter Didimo, Peter Eades, and Giuseppe Liotta. 2-layer right angle crossing drawings. *Algorithmica*, 68(4):954–997, 2014. doi:10.1007/s00453-012-9706-7.
- 18 Emilio Di Giacomo, Walter Didimo, Luca Grilli, Giuseppe Liotta, and Salvatore Agostino Romeo. Heuristics for the maximum 2-layer RAC subgraph problem. *Comput. J.*, 58(5):1085–1098, 2015. doi:10.1093/comjnl/bxu017.
- 19 Emilio Di Giacomo, Walter Didimo, Giuseppe Liotta, and Henk Meijer. Area, curve complexity, and crossing resolution of non-planar graph drawings. *Theory Comput. Syst.*, 49(3):565–575, 2011. doi:10.1007/s00224-010-9275-6.
- 20 Walter Didimo. Right angle crossing drawings of graphs. In Seok-Hee Hong and Takeshi Tokuyama, editors, *Beyond Planar Graphs*, pages 149–169. Springer, 2020. doi:10.1007/978-981-15-6533-5\_9.
- 21 Walter Didimo, Peter Eades, and Giuseppe Liotta. A characterization of complete bipartite RAC graphs. *Inf. Process. Lett.*, 110(16):687–691, 2010. doi:10.1016/j.ipl.2010.05.023.
- 22 Walter Didimo, Peter Eades, and Giuseppe Liotta. Drawing graphs with right angle crossings. *Theor. Comput. Sci.*, 412(39):5156–5166, 2011. doi:10.1016/j.tcs.2011.05.025.
- 23 Peter Eades and Giuseppe Liotta. Right angle crossing graphs and 1-planarity. *Discret. Appl. Math.*, 161(7-8):961–969, 2013. doi:10.1016/j.dam.2012.11.019.
- 24 Peter Eades, Antonios Symvonis, and Sue Whitesides. Three-dimensional orthogonal graph drawing algorithms. *Discret. Appl. Math.*, 103(1-3):55–87, 2000. doi:10.1016/S0166-218X(00)00172-4.
- 25 Henry Förster and Michael Kaufmann. On compact RAC drawings. In Fabrizio Grandoni, Grzegorz Herman, and Peter Sanders, editors, *ESA*, volume 173 of *LIPICs*, pages 53:1–53:21. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020. doi:10.4230/LIPICs.ESA.2020.53.
- 26 Ulrich Föbmeier and Michael Kaufmann. Drawing high degree graphs with low bend numbers. In Franz-Josef Brandenburg, editor, *GD*, volume 1027 of *LNCS*, pages 254–266. Springer, 1995. doi:10.1007/BFb0021809.
- 27 Ashim Garg and Roberto Tamassia. On the computational complexity of upward and rectilinear planarity testing. *SIAM J. Comput.*, 31(2):601–625, 2001. doi:10.1137/S0097539794277123.
- 28 F. Hadlock. Finding a maximum cut of a planar graph in polynomial time. *SIAM J. Comput.*, 4(3):221–225, 1975.
- 29 Seok-Hee Hong and Takeshi Tokuyama, editors. *Beyond Planar Graphs*. Springer, 2020. doi:10.1007/978-981-15-6533-5.
- 30 Weidong Huang, Peter Eades, and Seok-Hee Hong. Larger crossing angles make graphs easier to read. *J. Vis. Lang. Comput.*, 25(4):452–465, 2014. doi:10.1016/j.jvlc.2014.03.001.
- 31 János Pach and Géza Tóth. Graphs drawn with few crossings per edge. *Comb.*, 17(3):427–439, 1997. doi:10.1007/BF01215922.
- 32 Roberto Tamassia, editor. *Handbook on Graph Drawing and Visualization*. Chapman and Hall/CRC, 2013. URL: <https://www.crcpress.com/Handbook-of-Graph-Drawing-and-Visualization/Tamassia/9781584884125>.