

# Geometric RAC Simultaneous Drawings of Graphs\*

Evmorfia Argyriou<sup>1</sup>, Michael Bekos<sup>2</sup>, Michael Kaufmann<sup>2</sup>, and Antonios Symvonis<sup>1</sup>

<sup>1</sup> School of Applied Mathematical & Physical Sciences,  
National Technical University of Athens, Greece  
{fargyriou, symvonis}@math.ntua.gr

<sup>2</sup> Institute for Informatics, University of Tübingen, Germany  
{bekos, mk}@informatik.uni-tuebingen.de

**Abstract.** In this paper, we study the *geometric RAC simultaneous drawing problem*: Given two planar graphs that share a common vertex set but have disjoint edge sets, a geometric RAC simultaneous drawing is a straight-line drawing in which (i) each graph is drawn planar, (ii) there are no edge overlaps, and, (iii) crossings between edges of the two graphs occur at right-angles. We first prove that two planar graphs admitting a geometric simultaneous drawing may not admit a geometric RAC simultaneous drawing. We further show that a cycle and a matching always admit a geometric RAC simultaneous drawing, which can be constructed in linear time.

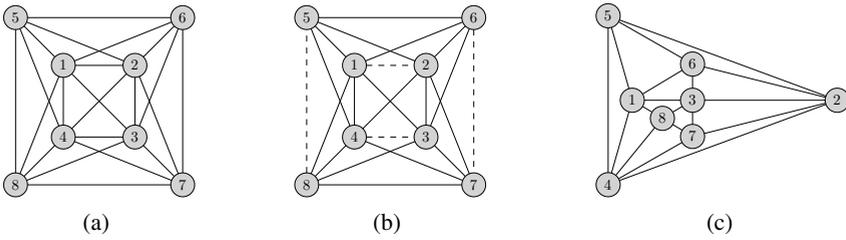
We also study a closely related problem according to which we are given a planar embedded graph  $G$  and the main goal is to determine a geometric drawing of  $G$  and its dual  $G^*$  (without the face-vertex corresponding to the external face) such that: (i)  $G$  and  $G^*$  are drawn planar, (ii) each vertex of the dual is drawn inside its corresponding face of  $G$  and, (iii) the primal-dual edge crossings form right-angles. We prove that it is always possible to construct such a drawing if the input graph is an outerplanar embedded graph.

## 1 Introduction

A *geometric right-angle crossing drawing* (or *geometric RAC drawing*, for short) of a graph is a straight-line drawing in which every pair of crossing edges intersects at right-angle. A graph which admits a geometric RAC drawing is called *right-angle crossing graph* (or *RAC graph*, for short). Motivated by cognitive experiments of Huang et al. [17], which indicate that the negative impact of an edge crossing on the human understanding of a graph drawing is eliminated in the case where the crossing angle is greater than seventy degrees, RAC graphs were recently introduced in [10] as a response to the problem of drawing graphs with optimal crossing resolution.

---

\* The work of E.N. Argyriou has been co-financed by the European Union (European Social Fund - ESF) and Greek national funds through the Operational Program “Education and Lifelong Learning” of the National Strategic Reference Framework (NSRF) - Research Funding Program: Heracleitus II. Investing in knowledge society through the European Social Fund. The work of M.A. Bekos is implemented within the framework of the Action “Supporting Postdoctoral Researchers” of the Operational Program “Education and Lifelong Learning” (Action’s Beneficiary: General Secretariat for Research and Technology), and is co-financed by the European Social Fund (ESF) and the Greek State.



**Fig. 1.** (a) A graph with 8 vertices and 22 edges which does not admit a RAC drawing [11]. (b) A decomposition of the graph of Fig.1a into a planar graph (solid edges; a planar drawing is given in Fig.1c) and a matching (dashed edges), which implies that a planar graph and a matching do not always admit a GRacSim drawing; their union is not RAC.

*Simultaneous graph drawing* deals with the problem of drawing two (or more) planar graphs on the same set of vertices on the plane, such that each graph is drawn planar<sup>1</sup> (i.e., only edges of different graphs are allowed to cross). The *geometric* version restricts the problem to straight-line drawings. Besides its independent theoretical interest, this problem arises in several application areas, such as software engineering, databases and social networks, where a visual analysis of evolving graphs, defined on the same set of vertices, is useful.

Both problems mentioned above are active research topics in the graph drawing literature and positive and negative results are known for certain variations (refer to Section 2). In this paper, we study the *geometric RAC simultaneous drawing problem* (or *GRacSim* drawing problem, for short), i.e., a combination of both problems. Formally, the GRacSim drawing problem can be stated as follows: Let  $G_1 = (V, E_1)$  and  $G_2 = (V, E_2)$  be two planar graphs that share a common vertex set but have disjoint edge sets, i.e.,  $E_1 \subseteq V \times V$ ,  $E_2 \subseteq V \times V$  and  $E_1 \cap E_2 = \emptyset$ . The main task is to place the vertices on the plane so that, when the edges are drawn as straight-lines, (i) each graph is drawn planar, (ii) there are no edge overlaps and (iii) crossings between edges in  $E_1$  and  $E_2$  occur at right-angles. Let  $G = (V, E_1 \cup E_2)$  be the graph induced by the union of  $G_1$  and  $G_2$ . Observe that  $G$  should be a RAC graph, which implies that  $|E_1 \cup E_2| \leq 4|V| - 10$  [10]. We refer to this relationship as the *RAC-size constraint*.

Note that, in the ordinary geometric simultaneous drawing problem the input graphs are allowed to share edges, i.e.,  $E_1 \cap E_2$  is non-empty in general. For instance, it is known that there exists a planar graph and a matching that do not admit a geometric simultaneous drawing [7]. However, this does not immediately imply that a planar graph and a matching do not admit a GRacSim drawing either (since the graphs utilized in the proof of the corresponding theorem in [7] share a common edge). Fig.1 depicts an alternative and simpler technique to prove such negative results for GRacSim drawings, which is based on the fact that not all graphs that obey the RAC-size constraint are eventually RAC graphs. On the other hand, as we will shortly see, two planar graphs admitting a geometric simultaneous drawing may not admit a GRacSim drawing.

<sup>1</sup> In the graph drawing literature, the problem is known as “simultaneous graph drawing with mapping”. For simplicity, we use the term “simultaneous graph drawing”.

The GRacSim drawing problem is of interest since it combines two current research topics in graph drawing. Our motivation to study this problem rests on the work of Didimo et al. [10] who proved that the crossing graph of a geometric RAC drawing is bipartite<sup>2</sup>. Thus, the edges of a geometric RAC drawing of a graph  $G = (V, E)$  can be partitioned into two sets  $E_1$  and  $E_2$ , such that no two edges of the same set cross. So, the problem we study is, in a sense, equivalent to the problem of finding a geometric RAC drawing of an input graph (if one exists), given its crossing graph.

A closely related problem to the GRacSim drawing problem, referred to as *geometric Graph-Dual RAC simultaneous drawing problem* (or *GDual-GRacSim* for short), is the following: *Given a planar embedded graph  $G$ , determine a geometric drawing of  $G$  and its dual  $G^*$  (without the face-vertex corresponding to the external face) such that: (i)  $G$  and  $G^*$  are drawn planar; (ii) each vertex of the dual is drawn inside its corresponding face of  $G$  and, (iii) the primal-dual edge crossings form right-angles.*

This paper is structured as follows: In Section 2, we review relevant previous research. In Section 3, we demonstrate that two planar graphs admitting a geometric simultaneous drawing may not admit a GRacSim drawing. In Section 4, we prove that a cycle and a matching always admit a GRacSim drawing, which can be constructed in linear time. In Section 5, we study the GDual-GRacSim drawing problem. We show that given a planar embedded graph, a GDual-GRacSim drawing of the planar graph and its dual does not always exist. If the input graph is an outerplanar embedded graph, we present an algorithm that constructs a GDual-GRacSim drawing of the outerplanar graph and its dual. We conclude in Section 6 with open problems.

## 2 Related Work

Didimo et al. [10] were the first to study the geometric RAC drawing problem and proved that any graph with  $n \geq 3$  vertices that admits a geometric RAC drawing has at most  $4n - 10$  edges. Arikushi et al. [4] presented bounds on the number of edges of polyline RAC drawings with at most one or two bends per edge. Angelini et al. [1] presented acyclic planar digraphs that do not admit upward geometric RAC drawings and proved that the corresponding decision problem is  $\mathcal{NP}$ -hard. Argyriou et al. [3] proved that it is  $\mathcal{NP}$ -hard to decide whether a given graph admits a geometric RAC drawing (i.e., the upwardness requirement is relaxed). Di Giacomo et al. [8] presented tradeoffs on the maximum number of bends per edge, the required area and the crossing angle resolution. Didimo et al. [9] characterized classes of complete bipartite graphs that admit geometric RAC drawings. Van Kreveld [18] showed that the quality of a planar drawing of a planar graph (measured in terms of area required, edge-length and angular resolution) can be improved if one allows right-angle crossings. Eades and Liotta [11] proved that a *maximally dense RAC graph* (i.e.,  $|E| = 4|V| - 10$ ) is also 1-planar, i.e., it admits a drawing in which every edge is crossed at most once.

Regarding the geometric simultaneous graph drawing problem, Brass et al. [5] presented algorithms for drawing simultaneously (a) two paths, (b) two cycles and, (c) two caterpillars. Estrella-Balderrama et al. [14] proved that the problem of determining

<sup>2</sup> This can be interpreted as follows: “If two edges of a geometric RAC drawing cross a third one, then these two edges must be parallel.”

whether two planar graphs admit a geometric simultaneous drawing is  $\mathcal{NP}$ -hard. Erten and Kobourov [13] showed that a planar graph and a path cannot always be drawn simultaneously. Geyer, Kaufmann and Vrt'o [16], showed that a geometric simultaneous drawing of two trees does not always exist. Angelini et al. [2] proved the same result for a path and a tree. Cabello et al. [7] showed that a geometric simultaneous drawing of a matching and (a) a wheel, (b) an outerpath or, (c) a tree always exists. For a quick overview of known results refer to Table 1 of [15].

Brightwell and Scheinermann [6] proved that the GDual-GRacSim drawing problem always admits a solution if the input graph is a triconnected planar graph. To the best of our knowledge, this is the only result which incorporates the requirement that the primal-dual edge crossings form right-angles. Erten and Kobourov [12], presented an  $O(n)$  time algorithm that results into a simultaneous drawing but, unfortunately, not a RAC drawing of a triconnected planar graph and its dual on an  $O(n^2)$  grid, where  $n$  is the number of vertices of  $G$  and  $G^*$ .

Before we proceed with the description of our results, we introduce some necessary notation. Let  $G = (V, E)$  be a simple, undirected graph drawn on the plane. We denote by  $\Gamma(G)$  the drawing of  $G$ . By  $x(v)$  and  $y(v)$ , we denote the  $x$ - and  $y$ -coordinate of  $v \in V$  in  $\Gamma(G)$ . We refer to the vertex (edge) set of  $G$  as  $V(G)$  ( $E(G)$ ). Given two graphs  $G$  and  $G'$ , we denote by  $G \cup G'$  the graph induced by the union of  $G$  and  $G'$ .

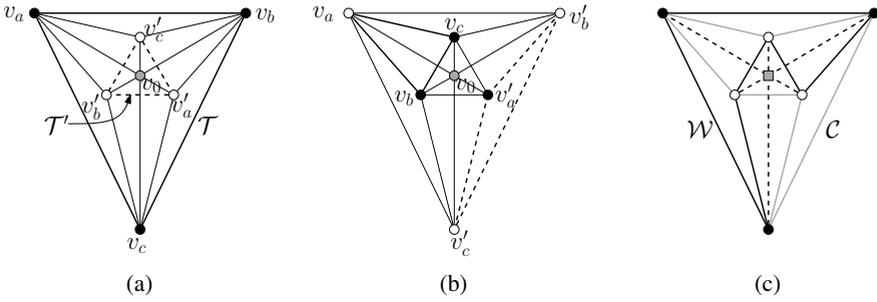
### 3 A Wheel and a Cycle: A Negative Result

In this section, we show that there exists a pair of planar graphs that admits a geometric simultaneous drawing, their union meets the RAC size constraint and they do not admit a GRacSim drawing (i.e, the class of graphs that admit GRacSim drawings is a subset of the class of graphs for which a simultaneous drawing is possible). We achieve this by showing that there exists a wheel and a cycle which do not admit a GRacSim drawing. Cabello et al. [7] have shown that a geometric simultaneous drawing of a wheel and a cycle always exists.

Our proof utilizes the *augmented triangle antiprism graph* [3,10], depicted in Fig.2a. The augmented triangle antiprism graph contains two triangles  $\mathcal{T}$  and  $\mathcal{T}'$  (refer to the dashed and bold drawn triangles in Fig.2a) and a ‘‘central’’ vertex  $v_0$  incident to the vertices of  $\mathcal{T}$  and  $\mathcal{T}'$ . If we delete the central vertex, the remaining graph corresponds to the skeleton of a triangle antiprism and it is commonly referred to as *triangle antiprism graph*. Didimo et al. [10] used the augmented triangle antiprism graph as an example of a maximally dense RAC graph (i.e.,  $|E| = 4|V| - 10$ ).

**Lemma 1.** *The geometric RAC drawings of the augmented triangle antiprism graph define exactly eight combinatorial embeddings.*

*Sketch of proof.* Due to space constraints we omit the detailed proof of this lemma. We simply note that (i)  $v_a$  and  $v'_a$ , (ii)  $v_b$  and  $v'_b$ , and, (iii)  $v_c$  and  $v'_c$  share the same neighbors. So, the eight combinatorial embeddings are implied by the drawing of Fig. 2a by mutually exchanging the positions of (i)  $v_a$  and  $v'_a$ , (ii)  $v_b$  and  $v'_b$ , and, (iii)  $v_c$  and  $v'_c$ . For an example refer to Fig. 2b.  $\square$



**Fig. 2.** (a)-(b) Two different RAC drawings of the augmented triangle antiprism graph with different combinatorial embeddings. (c) The union of wheel  $\mathcal{W}$  (solid and dashed black edges) and cycle  $\mathcal{C}$  (gray edges) is the augmented triangle antiprism graph.

**Theorem 1.** *There exists a wheel and a cycle which do not admit a GRacSim drawing.*

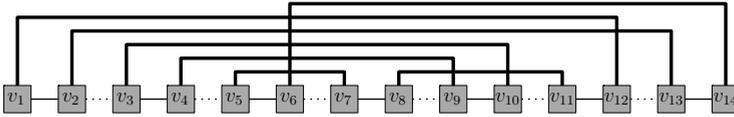
*Proof.* We denote the wheel by  $\mathcal{W}$  and the cycle by  $\mathcal{C}$ . The counterexample is depicted in Fig.2c. The center of  $\mathcal{W}$  is marked by a box, the spokes of  $\mathcal{W}$  are drawn as dashed line-segments, while the rim of  $\mathcal{W}$  is drawn in bold. Cycle  $\mathcal{C}$  is drawn in gray. The graph induced by the union of  $\mathcal{W}$  and  $\mathcal{C}$  (which in a GRacSim drawing of  $\mathcal{W}$  and  $\mathcal{C}$  should be drawn with right-angle crossings) is the augmented triangle antiprism graph, which, by Lemma 1, has exactly eight RAC combinatorial embeddings. However, in none of them wheel  $\mathcal{W}$  is drawn planar. This completes the proof.  $\square$

### 4 A Cycle and a Matching: A Positive Result

In this section, we first prove that a path and a matching always admit a GRacSim drawing and then we show that a cycle and a matching always admit a GRacSim drawing as well. Note that the union of a path and a matching is not necessarily a planar graph. Cabello et al. [7] provide an example of a path and a matching, which form a subdivision of  $K_{3,3}$ . We denote the path by  $\mathcal{P}$  and the matching by  $\mathcal{M}$ . Let  $v_1 \rightarrow v_2 \rightarrow \dots \rightarrow v_n$  be the edges of  $\mathcal{P}$  (see Fig.3). In order to keep the description of our algorithm simple, we will initially assume that  $n$  is even and  $|E(\mathcal{M})| = n/2$ . Later on this section, we will describe how to cope with the cases where  $n$  is odd or  $|E(\mathcal{M})| < n/2$ . Recall that by the definition of the GRacSim drawing problem,  $\mathcal{P}$  and  $\mathcal{M}$  do not share any edge, i.e.,  $E(\mathcal{P}) \cap E(\mathcal{M}) = \emptyset$ .

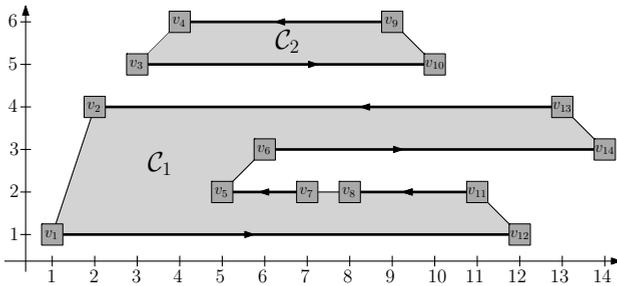
The basic idea of our algorithm is to identify in the graph induced by the union of  $\mathcal{P}$  and  $\mathcal{M}$  a set of cycles  $\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_k, k \leq n/4$ , such that: (i)  $|E(\mathcal{C}_1)| + |E(\mathcal{C}_2)| + \dots + |E(\mathcal{C}_k)| = n$ , (ii)  $\mathcal{M} \subseteq \mathcal{C}_1 \cup \mathcal{C}_2 \cup \dots \cup \mathcal{C}_k$ , and, (iii) the edges of cycle  $\mathcal{C}_i, i = 1, 2, \dots, k$  alternate between edges of  $\mathcal{P}$  and  $\mathcal{M}$ . Note that properties (i) and (ii) imply that the cycle collection will contain half of  $\mathcal{P}$ 's edges and all of  $\mathcal{M}$ 's edges. In our drawing, these edges will not cross with each other. The remaining edges of  $\mathcal{P}$  will introduce only right-angle crossings with the edges of  $\mathcal{M}$ .

Let  $\mathcal{P}_{odd}$  be a subgraph of  $\mathcal{P}$  which contains each second edge of  $\mathcal{P}$ , starting from its first edge, i.e.,  $E(\mathcal{P}_{odd}) = \{(v_i, v_{i+1}); 1 \leq i < n, i \text{ is odd}\}$ . In Fig.3, the edges



**Fig. 3.** An example of a path  $\mathcal{P}$  and a matching  $\mathcal{M}$ . The path appears at the bottom of the figure. The edges of  $\mathcal{M}$  are drawn bold, with two bends each. The edges of path  $\mathcal{P}$  form two matchings, i.e.,  $\mathcal{P}_{odd}$  and  $\mathcal{P} - \mathcal{P}_{odd}$ . The edges of  $\mathcal{P}_{odd}$  are drawn solid, while the edges of  $\mathcal{P} - \mathcal{P}_{odd}$  dotted.

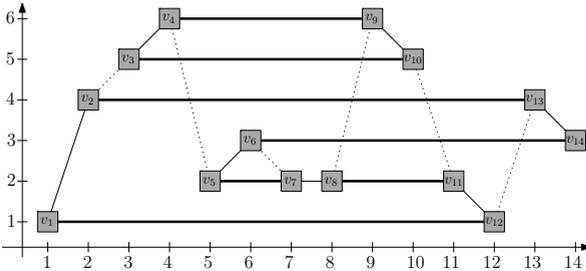
of  $\mathcal{P}_{odd}$  are drawn solid. Clearly,  $\mathcal{P}_{odd}$  is a matching. Since we have assumed that  $n$  is even,  $\mathcal{P}_{odd}$  contains exactly  $n/2$  edges. Hence,  $|E(\mathcal{P}_{odd})| = |E(\mathcal{M})|$ . In addition,  $\mathcal{P}_{odd}$  covers all vertices of  $\mathcal{P}$ , and,  $E(\mathcal{P}_{odd}) \cap E(\mathcal{M}) = \emptyset$ . The later equation trivially follows from our initial hypothesis, which states that  $E(\mathcal{P}) \cap E(\mathcal{M}) = \emptyset$ . We conclude that  $\mathcal{P}_{odd} \cup \mathcal{M}$  is a 2-regular graph. Thus, each connected component of  $\mathcal{P}_{odd} \cup \mathcal{M}$  corresponds to a cycle of even length, which alternates between edges of  $\mathcal{P}_{odd}$  and  $\mathcal{M}$ . This is the cycle collection mentioned above (see Fig.4).



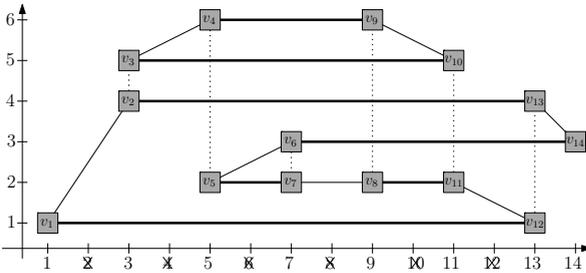
**Fig. 4.**  $\mathcal{P}_{odd} \cup \mathcal{M}$  (of Fig.3) consists of cycles  $\mathcal{C}_1$  and  $\mathcal{C}_2$ . The edges of  $\mathcal{P}_{odd}$  are drawn solid, while the edges of  $\mathcal{M}$  are drawn bold.

Initially, we fix the  $x$ -coordinate of each vertex of  $\mathcal{P}$  by setting  $x(v_i) = i$ ,  $1 \leq i \leq n$ . This ensures that  $\mathcal{P}$  is  $x$ -monotone and hence planar. Later on, we will slightly change the  $x$ -coordinate of some vertices of  $\mathcal{P}$  (without affecting  $\mathcal{P}$ 's monotonicity). Note that the algorithm can be adjusted so that the  $x$  and  $y$  coordinates of each vertex are computed at the same time. We have chosen to compute them separately in order to simplify the presentation. The  $y$ -coordinate of each vertex of  $\mathcal{P}$  is established by considering the cycles of  $\mathcal{P}_{odd} \cup \mathcal{M}$ .

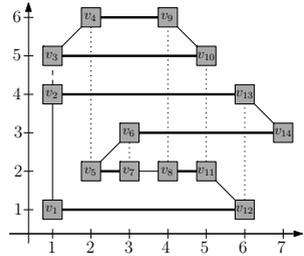
We draw each of these cycles in turn. More precisely, assume that zero or more cycles have been completely drawn and let  $\mathcal{C}$  be the cycle in the cycle collection which contains the leftmost vertex, say  $v_i$ , of  $\mathcal{P}$  that has not been drawn yet (initially,  $v_i$  is identified by  $v_1$ ). Then, vertex  $v_i$  should be an odd-indexed vertex and thus  $(v_i, v_{i+1})$  belongs to  $\mathcal{C}$ . Orient cycle  $\mathcal{C}$  so that vertex  $v_i$  is the first vertex of cycle  $\mathcal{C}$  and  $v_{i+1}$  is the last (see Fig.4). Based on this orientation, we will draw the edges of  $\mathcal{C}$  in a snake-like fashion, starting from vertex  $v_i$  and reaching vertex  $v_{i+1}$  last. The first edge to be drawn



(a) A drawing obtained by incorporating the edges of  $\mathcal{P} - \mathcal{P}_{odd}$  into the drawing of Fig.4.



(b) A drawing obtained by moving the even-indexed vertices of  $\mathcal{P}$  in the drawing of Fig.5a one unit to the right.



(c) A compact GRacSim drawing

**Fig. 5.** In all drawings, the edges of  $\mathcal{P}_{odd}$  are drawn solid, while the edges of  $\mathcal{P} - \mathcal{P}_{odd}$  dotted. The edges of  $\mathcal{M}$  are drawn bold.

is incident to vertex  $v_i$  and belongs to  $\mathcal{M}$ . We draw it as a horizontal line-segment at the bottommost available layer in the produced drawing (initially,  $L_1 : y = 1$ ). Since cycle  $\mathcal{C}$  alternates between edges of  $\mathcal{P}_{odd}$  and  $\mathcal{M}$ , the next edge to be drawn belongs to  $\mathcal{P}_{odd}$  followed by an edge of  $\mathcal{M}$ . If we can draw both of them in the current layer without introducing edge overlaps, we do so. Otherwise, we employ an additional layer. We continue in the same manner, until edge  $(v_i, v_{i+1})$  is reached in the traversal of cycle  $\mathcal{C}$ . This edge connects two consecutive vertices of  $\mathcal{P}$  that are the leftmost in the drawing of  $\mathcal{C}$ . Therefore, edge  $(v_i, v_{i+1})$  can be added in the drawing of  $\mathcal{C}$  without introducing any crossings. Thus, cycle  $\mathcal{C}$  is drawn planar.

So far, we have drawn all edges of  $\mathcal{M}$  and half of the edges of  $\mathcal{P}$  (i.e.,  $\mathcal{P}_{odd}$ ) and we have obtained a planar drawing in which all edges of  $\mathcal{M}$  are drawn as horizontal, non-overlapping line segments. In the worst case, this drawing occupies  $n/2$  layers.

We proceed to incorporate the remaining edges of  $\mathcal{P}$ , i.e, the ones that belong to  $\mathcal{P} - \mathcal{P}_{odd}$ , into the drawing (refer to the dotted drawn edges of Fig.5a). Since  $x(v_i) = i, i = 1, 2, \dots, n$ , the edges of  $\mathcal{P}$  do not cross with each other and therefore  $\mathcal{P}$  is drawn planar. In contrast, an edge of  $\mathcal{P} - \mathcal{P}_{odd}$  may cross multiple edges of  $\mathcal{M}$ , and, these crossings do not form right-angles (see Fig.5a). In order to fix these crossings, we suggest to move each even-indexed vertex of  $\mathcal{P}$  one unit to the right (keeping its  $y$ -coordinate unchanged), except for the last vertex of  $\mathcal{P}$ . Then, the endpoints of the edges of  $\mathcal{P} - \mathcal{P}_{odd}$  have exactly the same  $x$ -coordinate and cross at right-angles the edges of

$\mathcal{M}$  which are drawn as horizontal line-segments. The path remains  $x$ -monotone (but not strictly anymore) and hence planar. In addition, it is not possible to introduce vertex overlaps, since in the produced drawing each edge of  $\mathcal{M}$  has at least two units length (recall that  $E(\mathcal{P}) \cap E(\mathcal{M}) = \emptyset$ ). Since the vertices of the drawing do not occupy even  $x$ -coordinates, the width of the drawing can be reduced from  $n$  to  $n/2 + 1$  (see Fig.5b). We can further reduce the width of the produced drawing by merging consecutive columns that do not interfere in  $y$ -direction into a common column (see Fig.5c). However, this post-processing does not result into a drawing of asymptotically smaller area.

In order to complete the description of our algorithm, it remains to consider the cases where  $n$  is odd or  $|E(\mathcal{M})| < n/2$ . Both cases can be treated similarly. If  $n$  is odd or  $|E(\mathcal{M})| < n/2$ , there exist vertices of  $\mathcal{P}$  which are not covered by matching  $\mathcal{M}$ . As long as there exist such vertices, we can momentarily remove them from the path by contracting each subpath consisting of degree-2 vertices into a single edge. By this procedure, we obtain a new path  $\mathcal{P}'$ , so that  $\mathcal{M}$  covers all vertices of  $\mathcal{P}'$ . If we draw  $\mathcal{P}'$  and  $\mathcal{M}$  simultaneously, then it is easy to incorporate the removed vertices in the produced drawing, since they do not participate in  $\mathcal{M}$ . The following theorem summarizes our result.

**Theorem 2.** *A path and a matching always admit a GRacSim drawing on an  $(n/2 + 1) \times n/2$  integer grid. Moreover, the drawing can be computed in linear time.*

*Proof.* Finding the cycles of  $\mathcal{P}_{odd} \cup \mathcal{M}$  can be done in  $O(n)$  time, where  $n$  is the number of vertices of  $\mathcal{P}$ ; we identify the leftmost vertex of each cycle and then we traverse it. Having computed the cycle collection of  $\mathcal{P}_{odd} \cup \mathcal{M}$ , the coordinates of the vertices are computed in  $O(n)$  total time by a traversal of the cycle.  $\square$

We extend the algorithm that produces a GRacSim drawing of a path and a matching to also cover the case of a cycle  $\mathcal{C}$  and a matching  $\mathcal{M}$ . Obviously, if we remove an edge from the input cycle, the remaining graph is a path  $\mathcal{P}$ . Then, we apply the developed algorithm and obtain a GRacSim drawing of  $\mathcal{P}$  and  $\mathcal{M}$ , in which the first vertex of  $\mathcal{P}$  is drawn at the bottommost layer (hence its incident edge in  $\mathcal{M}$  is not crossed), and, the last vertex of  $\mathcal{P}$  is drawn rightmost. With these two properties, we can add the removed edge, between the first and the last vertex of  $\mathcal{P}$  without introducing new crossings. To achieve this, we move the first vertex of  $\mathcal{P}$  at most  $n/2 + 2$  units downwards (keeping its  $x$ -coordinate unchanged) and the last vertex of  $\mathcal{P}$  at most  $n/2 + 1$  units rightwards (keeping its  $y$ -coordinate unchanged). Then, the insertion in the drawing of the edge that closes the cycle does not introduce any crossings, as desired.

**Theorem 3.** *A cycle and a matching always admit a GRacSim drawing on an  $(n + 2) \times (n + 2)$  integer grid. Moreover, the drawing can be computed in linear time.*

**Theorem 4.** *Let  $G$  be a simple connected graph that can be decomposed into a matching and either a path or a cycle. Then,  $G$  is a RAC graph.*

*Proof.* The argument trivially holds in the case where the input path (or cycle) is hamiltonian. If it is not hamiltonian, vertices that are not covered by the path (or cycle) are only incident to edges of the matching. Thus, they can be momentarily removed, compute a drawing of the remaining graph and easily insert them into the resulting drawing, since they are of degree 1.  $\square$

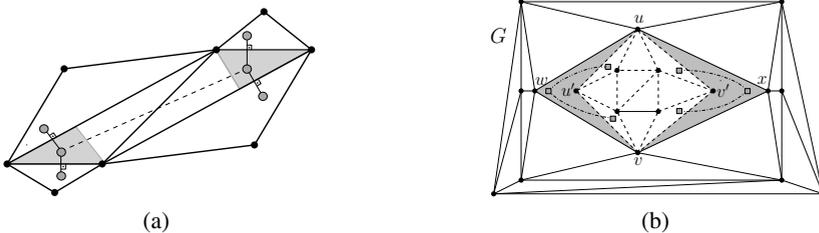
### 5 A Planar Graph and Its Dual: An Interesting Variation

In this section, we examine the GDual-GRacSim drawing problem. This problem can be considered as a variation of the GRacSim drawing problem, where the first graph (i.e., the planar graph) determines the second one (i.e., the dual) and places restrictions on its layout. As already stated in Section 2, Brightwell and Scheinermann [6] proved that the GDual-GRacSim problem always admits a solution if the input graph is a triconnected planar graph. For the general case of planar graphs, we demonstrate by an example that it is not always possible to compute such a drawing, and thus, we concentrate our study in the more interesting case of outerplanar graphs.

Initially, we consider the case where the planar drawing  $\Gamma(G)$  of graph  $G$  is specified as part of the input and it is required that it remains unchanged in the output. We demonstrate by an example that it is not always feasible to incorporate  $G^*$  into  $\Gamma(G)$  and obtain a GDual-GRacSim drawing of  $G$  and  $G^*$ . The example is illustrated in Fig.6a. In the following, we prove that if the input graph is a planar embedded graph, then the GDual-GRacSim drawing problem does not always admit a solution, as well.

**Theorem 5.** *Given a planar embedded graph  $G$ , a GDual-GRacSim drawing of  $G$  and its dual  $G^*$  does not always exist.*

*Proof.* We prove a slightly stronger result by investigating all possible planar embeddings of a particular  $G$  for which we prove that a GDual-GRacSim drawing of  $G$  and its dual  $G^*$  does not exist. Graph  $G$  used to establish the theorem is depicted in Fig.6b, where the vertices drawn as boxes belong to  $G^*$ . Observe that the subgraph drawn with dashed edges is triconnected planar. Thus, it has a unique planar embedding (up to a reflection). If we replace this subgraph by an edge, the remaining primal graph is also triconnected. Hence, the graph of our example is a subdivision of a triconnected graph, which implies that it has two planar combinatorial embeddings obtained by reflections of the triconnected planar subgraph, at  $u$  and  $v$ , i.e., either  $u'$  is to the “left” of  $v'$ , or to its “right”. Now, observe that the dual graph should have two vertices within the gray-colored faces of Fig.6b (refer to the vertices drawn as boxes). Each of these two vertices is incident to two vertices of the dual that lie within the triangular faces of the dashed



**Fig. 6.** (a) The input planar drawing of the primal graph  $G$  is sketched with black colored vertices and bold edges and should remain unchanged in the output. The vertices of the dual  $G^*$  are colored gray. Then, the dual’s dashed drawn edge will inevitably introduce a non right-angle crossing. (b) An example of a planar embedded graph  $G$  for which the GDual-GRacSim does not admit a solution. The problematic faces are drawn in gray.

drawn subgraph of  $G$ , incident to the two gray-colored faces. Observe that in any RAC drawing of  $G$  and  $G^*$  both quadrilaterals  $uu'vw$  and  $uv'vx$  must be convex, which is impossible.  $\square$

**Theorem 6.** *Given an outerplane embedding of an outerplanar graph  $G$ , it is always possible to determine a GDual-GRacSim drawing of  $G$  and its dual  $G^*$ .*

*Proof.* The proof is given by a recursive geometric construction which computes a GDual-GRacSim drawing of  $G$  and its dual  $G^*$ . Consider an arbitrary edge  $(u, v)$  of the outerplanar graph that does not belong to its external face and let  $f$  and  $g$  be the faces to its left and the right side, respectively, as we move along  $(u, v)$  from vertex  $u$  to vertex  $v$ . Then,  $(f, g)$  is an edge of the dual graph  $G^*$ . Since the dual of an outerplanar graph is a tree, the removal of edge  $(f, g)$  results in two trees  $T_f$  and  $T_g$  that can be considered to be rooted at vertices  $f$  and  $g$  of  $G^*$ , respectively. For the recursive step of our algorithm, we assume that we have already produced a GDual-GRacSim drawing for  $T_f$  and its corresponding subgraph of  $G$  that satisfies the following invariant properties:

- I-P1: *Edge  $(u, v)$  is drawn on the external face of the GDual-GSimRAC drawing constructed so far. Let  $u$  and  $v$  be drawn at points  $p_u$  and  $p_v$ , respectively. Denote by  $\ell_{u,v}$  the line defined by  $p_u$  and  $p_v$ .*
- I-P2: *Let the face-vertex  $f$  be drawn at point  $p_f$ . The perpendicular from point  $p_f$  to line  $\ell_{u,v}$  intersects the line segment  $p_u p_v$ . Let  $p$  be the point of intersection.*
- I-P3: *There exists two parallel semi-lines  $\ell_u$  and  $\ell_v$  passing from  $p_u$  and  $p_v$ , respectively, that define a semi-strip to the right of segment  $p_u p_v$  that does not intersect the drawing constructed so far. Denote this empty semi-strip by  $R_{u,v}$ .*

We proceed to describe how to recursively produce a drawing for tree  $T_g$  and its corresponding subgraph of  $G$  so that the overall drawing is a GDual-GRacSim drawing for  $G$  and  $G^*$ . Refer to Fig.7a. Let  $p_g$  be a point in semi-strip  $R_{u,v}$  that also belongs to the perpendicular line to line-segment  $p_u p_v$  that passes from point  $p$ . Thus, the segment corresponding to the edge  $(f, g)$  of the dual crosses at right-angle the segment corresponding to the edge  $(u, v)$  of  $G$ , as required. If  $g$  is a leaf (i.e., all edges of  $g$  except  $(u, v)$  are edges of the external face), we can draw the remaining edges of face  $g$  as a polyline of appropriate number of points that goes around  $p_g$  and connects  $p_u$  and  $p_v$ .

Consider now the more interesting case where  $g$  is not a leaf in  $G^*$ . In this case, we draw two circles, say  $C_g$  and  $C'_g$ , centered at  $p_g$  such that both lie entirely within semi-strip  $R_{u,v}$  and do not touch neither  $\ell_u$  nor  $\ell_v$ . Assume that circle  $C'_g$  is the external of the two circles. From  $p_u$  draw the tangent to circle  $C_g$  and let  $a$  be the point it touches  $C_g$  and  $a'$  be the point to the right of  $a$  where the tangent intersects circle  $C'_g$  (see Fig.7a). Similarly, we define points  $b$  and  $b'$  based on the tangent from  $p_v$  to  $C_g$ .

Let  $k \geq 4$  be the number of vertices defining face  $g$ . The case where  $k = 3$  will be examined later. Draw  $k - 4$  points on the  $(a', b')$  arc, which is furthest from segment  $p_u p_v$ . These points, say  $\{p_i \mid 1 \leq i \leq k - 4\}$ , together with points  $p_u, p_v, a'$  and  $b'$  form face  $g$ . Observe that from point  $p_g$ , we can draw perpendicular lines towards each edge of the face. Indeed, line segments  $p_g a$  and  $p_g b$  are perpendicular to  $p_u a'$  and  $p_v b'$ , respectively. The remaining edges of the face are chords of circle  $C'_g$  and thus, we can always draw perpendicular lines to their midpoints from the center  $p_g$  of the circle.

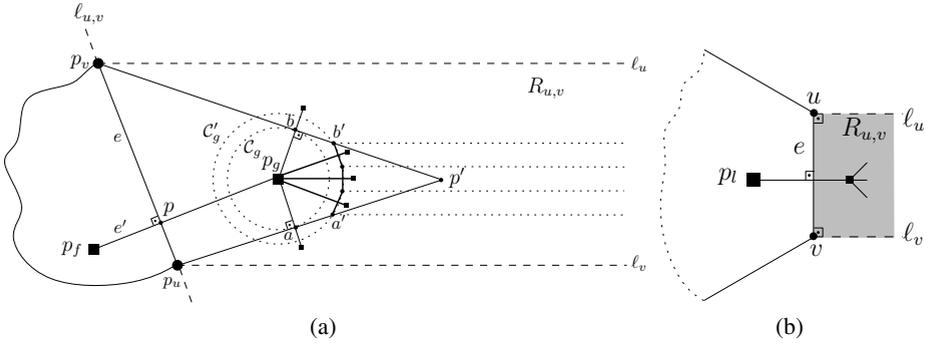


Fig. 7. (a) The recursion step of our algorithm, (b) The initial step of our algorithm

Now, from each of the newly inserted points of face  $g$  draw a semi-line that is parallel to semi-line  $\ell_u$  and lies entirely in the semi-strip  $R_{u,v}$ . We observe all invariant properties stated above hold for each child of face  $g$  in the subtree  $T_g$  of the dual of  $G$ . Thus, our algorithm can be applied recursively. In the case where  $k = 3$ , we use the intersection of the two tangents, say  $p'$ , as the third point of the triangular face. We have to be careful so that  $p'$  lies inside the semi-strip. However, we can always select a point  $p_g$  close to segment  $p_u p_v$  and an appropriately small radius for circle  $C_g$ , so that  $p'$  is inside  $R_{u,v}$ .

Now that we have described the recursive step, it remains to define how the recursion begins (see Fig.7b). We start from any face of  $G$  that is a leaf at its dual tree, say face  $l$ . We draw the face as regular polygon, with face-vertex  $l$  mapped at its center, say  $p_l$ . Let  $e = (u, v)$  be the only edge of the face that is internal to the outerplane embedding of  $G$ . W.l.o.g. assume that  $e$  is drawn vertically. Then, draw the horizontal semi-lines  $\ell_u$  and  $\ell_v$  from the endpoints of  $e$  in order to define semi-strip  $R_{u,v}$ . From this point on, the algorithm can recursively draw the remaining faces of  $G$  and  $G^*$ .

Note that the produced GDual-GRacSim drawing of  $G$  and its dual proves that producing such drawings is possible. The drawing is not particularly appealing since the height of the strips quickly becomes very small. However, it is a starting point towards algorithms that produce better layouts. Also note that the algorithm performs a linear number of “point computations” since for each face-vertex of the dual tree the performed computations are proportional to the degree of the face-vertex. However, the coordinates of some points may be non-rational numbers.  $\square$

## 6 Conclusion – Open Problems

In this paper, we introduced and examined geometric RAC simultaneous drawings. Our study raises several open problems. Among them are the following: (1) What other non-trivial classes of graphs, besides a matching and either a path or a cycle, admit a GRacSim drawing? (2) We showed that if two graphs admit a geometric simultaneous drawing, it is not necessary that they admit a GRacSim drawing. Finding a class of graphs (instead of a particular graph) with this property would strengthen this result. (3) A quite similar problem to the GRacSim drawing problem is the problem of drawing

two (or more) graphs on the same vertex set on the plane, such that each graph is drawn RAC (i.e., only edges of different graphs may introduce non-right angle crossings). Note that the class of graphs that admit such drawings contains the class of graphs for which a simultaneous drawing is possible. (4) Obtain more appealing GDual-GRacSim drawings for an outerplanar graph and its dual. Study the required drawing area.

## References

1. Angelini, P., Cittadini, L., Di Battista, G., Didimo, W., Frati, F., Kaufmann, M., Symvonis, A.: On the perspectives opened by right angle crossing drawings. *JGAA* 15(1), 53–78 (2011)
2. Angelini, P., Geyer, M., Kaufmann, M., Neuwirth, D.: On a Tree and a Path with No Geometric Simultaneous Embedding. In: Brandes, U., Cornelsen, S. (eds.) *GD 2010*. LNCS, vol. 6502, pp. 38–49. Springer, Heidelberg (2011)
3. Argyriou, E.N., Bekos, M.A., Symvonis, A.: The Straight-Line RAC Drawing Problem Is NP-Hard. In: Černá, I., Gyimóthy, T., Hromkovič, J., Jefferey, K., Kráľovič, R., Vukolić, M., Wolf, S. (eds.) *SOFSEM 2011*. LNCS, vol. 6543, pp. 74–85. Springer, Heidelberg (2011)
4. Arikushi, K., Fulek, R., Keszegh, B., Morić, F., Tóth, C.D.: Graphs that Admit Right Angle Crossing Drawings. In: Thilikos, D.M. (ed.) *WG 2010*. LNCS, vol. 6410, pp. 135–146. Springer, Heidelberg (2010)
5. Brass, P., Cenek, E., Duncan, C.A., Efrat, A., Erten, C., Ismailescu, D., Kobourov, S.G., Lubiw, A., Mitchell, J.S.B.: On simultaneous planar graph embeddings. *Computational Geometry: Theory and Applications* 36(2), 117–130 (2007)
6. Brightwell, G., Scheinerman, E.R.: Representations of planar graphs. *SIAM Journal Discrete Mathematics* 6(2), 214–229 (1993)
7. Cabello, S., van Kreveld, M.J., Liotta, G., Meijer, H., Speckmann, B., Verbeek, K.: Geometric simultaneous embeddings of a graph and a matching. *JGAA* 15(1), 79–96 (2011)
8. Di Giacomo, E., Didimo, W., Liotta, G., Meijer, H.: Area, Curve Complexity, and Crossing Resolution of Non-planar Graph Drawings. In: Eppstein, D., Gansner, E.R. (eds.) *GD 2009*. LNCS, vol. 5849, pp. 15–20. Springer, Heidelberg (2010)
9. Didimo, W., Eades, P., Liotta, G.: A characterization of complete bipartite graphs. *Information Processing Letters* 110(16), 687–691 (2010)
10. Didimo, W., Eades, P., Liotta, G.: Drawing graphs with right angle crossings. *Theor. Comp. Sci.* 412(39), 5156–5166 (2011)
11. Eades, P., Liotta, G.: Right angle crossing graphs and 1-planarity. In: *27th European Workshop on Computational Geometry* (2011)
12. Erten, C., Kobourov, S.G.: Simultaneous embedding of a planar graph and its dual on the grid. *Theory Computing* 38(3), 313–327 (2005)
13. Erten, C., Kobourov, S.G.: Simultaneous embedding of planar graphs with few bends. *JGAA* 9(3), 347–364 (2005)
14. Estrella-Balderrama, A., Gassner, E., Jünger, M., Percan, M., Schaefer, M., Schulz, M.: Simultaneous Geometric Graph Embeddings. In: Hong, S.-H., Nishizeki, T., Quan, W. (eds.) *GD 2007*. LNCS, vol. 4875, pp. 280–290. Springer, Heidelberg (2008)
15. Frati, F., Kaufmann, M., Kobourov, S.G.: Constrained simultaneous and near-simultaneous embeddings. *JGAA* 13(3), 447–465 (2009)
16. Geyer, M., Kaufmann, M., Vrto, I.: Two trees which are self-intersecting when drawn simultaneously. *Discrete Mathematics* 309(7), 1909–1916 (2009)
17. Huang, W., Hong, S.H., Eades, P.: Effects of crossing angles. In: *IEEE Pacific Visualization Symp.*, pp. 41–46. IEEE (2008)
18. van Kreveld, M.: The Quality Ratio of RAC Drawings and Planar Drawings of Planar Graphs. In: Brandes, U., Cornelsen, S. (eds.) *GD 2010*. LNCS, vol. 6502, pp. 371–376. Springer, Heidelberg (2011)