

# The Partial Visibility Representation Extension Problem

Steven Chaplick<sup>1(✉)</sup>, Grzegorz Guśpiel<sup>2</sup>, Grzegorz Gutowski<sup>2</sup>,  
Tomasz Krawczyk<sup>2</sup>, and Giuseppe Liotta<sup>3</sup>

<sup>1</sup> Lehrstuhl für Informatik I, Universität Würzburg, Würzburg, Germany  
[steven.chaplick@uni-wuerzburg.de](mailto:steven.chaplick@uni-wuerzburg.de)

<sup>2</sup> Theoretical Computer Science Department,  
Faculty of Mathematics and Computer Science, Jagiellonian University,  
Kraków, Poland

[{guspiel,gutowski,krawczyk}@tcs.uj.edu.pl](mailto:{guspiel,gutowski,krawczyk}@tcs.uj.edu.pl)

<sup>3</sup> Dipartimento di Ingegneria, Università degli Studi di Perugia, Perugia, Italy  
[giuseppe.liotta@unipg.it](mailto:giuseppe.liotta@unipg.it)

**Abstract.** For a graph  $G$ , a function  $\psi$  is called a *bar visibility representation* of  $G$  when for each vertex  $v \in V(G)$ ,  $\psi(v)$  is a horizontal line segment (*bar*) and  $uv \in E(G)$  iff there is an unobstructed, vertical,  $\varepsilon$ -wide line of sight between  $\psi(u)$  and  $\psi(v)$ . Graphs admitting such representations are well understood (via simple characterizations) and recognizable in linear time. For a directed graph  $G$ , a bar visibility representation  $\psi$  of  $G$ , additionally, for each directed edge  $(u, v)$  of  $G$ , puts the bar  $\psi(u)$  strictly below the bar  $\psi(v)$ . We study a generalization of the recognition problem where a function  $\psi'$  defined on a subset  $V'$  of  $V(G)$  is given and the question is whether there is a bar visibility representation  $\psi$  of  $G$  with  $\psi|_{V'} = \psi'$ . We show that for undirected graphs this problem together with closely related problems are NP-complete, but for certain cases involving directed graphs it is solvable in polynomial time.

## 1 Introduction

The concept of a visibility representation of a graph is a classic one in computational geometry and graph drawing and the first studies on this concept date back to the early days of these fields (see, e.g. [16, 17] and [12] for a recent survey).

---

This work was partially supported by ESF project EUROGIGA GraDR and preliminary ideas were formed during HOMONOLO 2014. G. Guśpiel was partially supported by the MNiSW grant DI2013 000443. G. Gutowski was partially supported by the Polish National Science Center grant UMO-2011/03/D/ST6/01370. T. Krawczyk was partially supported by the Polish National Science Center grant UMO-2015/17/B/ST6/01873. G. Liotta was partially supported by the MIUR project AMANDA “Algorithmics for MAssive and Networked DATa”, prot. 2012C4E3KT\_001. The full version of this paper is available on arXiv:1512.00174 [5]. Note: whenever we refer to sections in the *Appendix* we mean the appendix of [arXiv:1512.00174v2](https://arxiv.org/abs/1512.00174v2).

In the most general setting, a visibility representation of a graph is defined as a collection of disjoint sets from an Euclidean space such that the vertices are bijectively mapped to the sets and the edges correspond to unobstructed lines of sight between two such sets. Many different classes of visibility representations have been studied via restricting the space (e.g., to be the plane), the sets (e.g., to be points or line segments) and/or the lines of sight (e.g., to be non-crossing or axis-parallel). In this work we focus on a classic visibility representation setting in which the sets are horizontal line segments (*bars*) in the plane and the lines of sight are vertical. As such, whenever we refer to a visibility representation, we mean one of this type. The study of such representations was inspired by the problems in VLSI design and was conducted by different authors [9, 13, 14] under variations of the notion of visibility. Tamassia and Tollis [16] gave an elegant unification of different definitions and we follow their approach.

A *horizontal bar* is an open, non-degenerate segment parallel to the  $x$ -axis of the coordinate plane. For a set  $\Gamma$  of pairwise disjoint horizontal bars, a *visibility ray* between two bars  $a$  and  $b$  in  $\Gamma$  is a vertical closed segment spanned between bars  $a$  and  $b$  that intersects  $a$ ,  $b$ , and no other bar. A *visibility gap* between two bars  $a$  and  $b$  in  $\Gamma$  is an axis aligned, non-degenerate open rectangle spanned between bars  $a$  and  $b$  that intersects no other bar.

For a graph  $G$ , a *visibility representation*  $\psi$  is a function that assigns a distinct horizontal bar to each vertex such that these bars are pairwise disjoint and satisfy additional visibility constraints. There are three standard visibility models:

- *Weak visibility.* In this model, for each edge  $\{u, v\}$  of  $G$ , there is a visibility ray between  $\psi(u)$  and  $\psi(v)$  in  $\psi(V(G))$ .
- *Strong visibility.* In this model, two vertices  $u, v$  of  $G$  are adjacent if and only if there is a visibility ray between  $\psi(u)$  and  $\psi(v)$  in  $\psi(V(G))$ .
- *Bar visibility.* In this model, two vertices  $u, v$  of  $G$  are adjacent if and only if there is a visibility gap between  $\psi(u)$  and  $\psi(v)$  in  $\psi(V(G))$ .

The bar visibility model is also known as the  $\varepsilon$ -visibility model in the literature.

A graph that admits a visibility representation in any of these models is a planar graph, but the converse does not hold in general. Tamassia and Tollis [16] characterized the graphs that admit a visibility representation in these models as follows. A graph admits a weak visibility representation if and only if it is planar. A graph admits a bar visibility representation if and only if it has a planar embedding with all cut-points on the boundary of the outer face. For both of these models, Tamassia and Tollis [16] presented linear time algorithms for the recognition of representable graphs, and for constructing the appropriate visibility representations. The situation is different for the strong visibility model. Although the planar graphs admitting a strong visibility representation are characterized in [16] (via strong  $st$ -numberings), Andreae [1] proved that the recognition of such graphs is NP-complete. Summing up, from a computational point of view, the problems of recognizing graphs that admit visibility representations and of constructing such representations are well understood.

Recently, a lot of attention has been paid to the question of extending partial representations of graphs. In this setting a representation of some vertices of

the graph is already fixed and the task is to find a representation of the whole graph that extends the given partial representation. Problems of this kind are often encountered in graph drawing and are sometimes computationally harder than testing for existence of an unconstrained drawing. The problem of extending partial drawings of planar graphs is a good illustration of this phenomenon. On the one hand, by Fáry's theorem, every planar graph can be drawn in the plane so that each vertex is represented as a point, and edges are pairwise non-crossing, straight-line segments joining the corresponding points. Moreover, such a drawing can be constructed in linear time. On the other hand, testing whether a partial drawing of this kind (i.e., an assignment of points to some of the vertices) can be extended to a straight-line drawing of the whole graph is NP-hard [15]. However, an analogous problem in the model that allows the edges to be drawn as arbitrary curves instead of straight-line segments has a linear-time solution [2]. A similar phenomenon occurs when we consider contact representations of planar graphs. Every planar graph is representable as a disk contact graph or a triangle contact graph. Every bipartite planar graph is representable as a contact graph of horizontal and vertical segments in the plane. Although such representations can be constructed in polynomial time, the problems of extending partial representations of these kinds are NP-hard [4].

In this paper we initiate the study of extending partial visibility representations of graphs. From a practical point of view, it may be worth recalling that visibility representations are not only an appealing way of drawing graphs, but they are also typically used as an intermediate step towards constructing visualizations of networks in which all edges are oriented in a common direction and some vertices are aligned (for example to highlight critical activities in a PERT diagram). Visibility representations are also used to construct orthogonal drawings with at most two bends per edge. See, e.g. [6] for more details about these applications. The partial representation extension problem that we study in this paper occurs, for example, when we want to use visibility representations to incrementally draw a large network and we want to preserve the user's mental map in a visual exploration that adds a few vertices and edges per time.

Both for weak visibility and for strong visibility, the partial representation extension problems are easily found to be NP-hard. For weak visibility, the hardness follows from results on contact representations by Chaplick et al. [4]. For strong visibility, it follows trivially from results by Andreae [1]. Our contribution is the study of the partial representation extension problem for bar visibility. Hence, the central problem for this paper is the following:

Bar Visibility Representation Extension:

**Input:**  $(G, \psi')$ ;  $G$  is a graph;  $\psi'$  is a map assigning bars to a  $V' \subseteq V(G)$ .

**Question:** Does  $G$  admit a bar visibility representation  $\psi$  with  $\psi|_{V'} = \psi'$ ?

One of our results is the following.

**Theorem 1.** *The Bar Visibility Representation Extension Problem is NP-complete.*

The proof is a standard reduction from PLANARMONOTONE3SAT problem, which is known to be NP-complete thanks to de Berg and Khosravi [3]. The reduction uses gadgets that simulate logic gates and constructs a planar boolean circuit that encodes the given formula. Theorem 1 is proven in Appendix D. We investigate a few natural modifications of the problem. Most notably, we study the version of the problem for directed graphs. We provide some efficient algorithms for extension problems in this setting. A visibility representation induces a natural orientation on edges of the graph – each edge is oriented from the lower bar to the upper one. This leads to the definition of a visibility representation for a directed graph. The function  $\psi$  is a representation of a digraph  $G$  if, additionally to satisfying visibility constraints, for each directed edge  $(u, v)$  of  $G$ , the bar  $\psi(u)$  is strictly below the bar  $\psi(v)$ . Note that a planar digraph that admits a visibility representation also admits an *upward planar drawing* (see e.g., [10]), that is, a drawing in which the edges are represented as non-crossing  $y$ -monotone curves.

A *planar  $st$ -graph* is a planar acyclic digraph with exactly one source  $s$  and exactly one sink  $t$  which admits a planar embedding such that  $s$  and  $t$  are on the outer face. Di Battista and Tamassia [7] proved that a planar digraph admits an upward planar drawing if and only if it is a subgraph of a planar  $st$ -graph if and only if it admits a weak visibility representation. Garg and Tamassia [11] showed that the recognition of planar digraphs that admit an upward planar drawing is NP-complete. It follows that the recognition of planar digraphs that admit a weak visibility representation is NP-complete, and so is the corresponding partial representation extension problem. Nevertheless, as is shown in Lemma 1 (see Appendix A for the proof), the situation might be different for bar visibility.

**Lemma 1.** *Let  $st(G)$  be a graph constructed from a planar digraph  $G$  by adding two vertices  $s$  and  $t$ , the edge  $(s, t)$ , an edge  $(s, v)$  for each source vertex  $v$  of  $G$ , and an edge  $(v, t)$  for each sink vertex  $v$  of  $G$ . A planar digraph  $G$  admits a bar visibility representation if and only if the graph  $st(G)$  is a planar  $st$ -graph.*

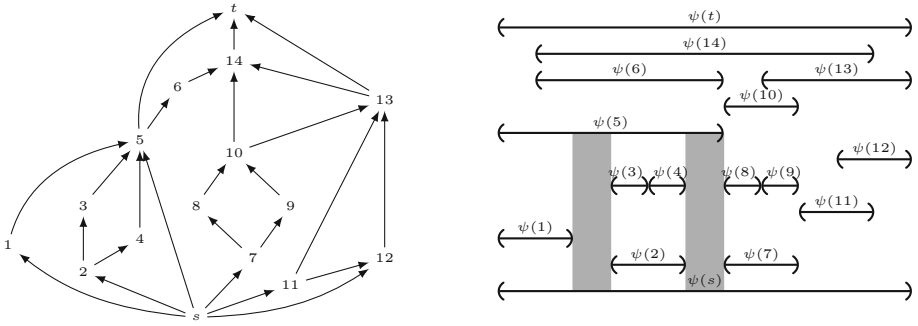
As planar  $st$ -graphs can be recognized in linear time, the same is true for planar digraphs that admit a bar visibility representation. The natural problem that arises is the following:

Bar Visibility Representation Extension for Digraphs:

**Input:**  $(G, \psi')$ ;  $G$  is a digraph;  $\psi'$  is a map assigning bars to a  $V' \subseteq V(G)$ .

**Question:** Does  $G$  admit a bar visibility representation  $\psi$  with  $\psi|_{V'} = \psi'$ ?

Although we do not provide a solution for this problem, we present an efficient algorithm for an important variant. A bar visibility representation  $\psi$  of a directed graph  $G$  is called *rectangular* if  $\psi$  has a unique bar  $\psi(s)$  with the lowest  $y$ -coordinate, a unique bar  $\psi(t)$  with the highest  $y$ -coordinate,  $\psi(s)$  and  $\psi(t)$  span the same  $x$ -interval, and all other bars are inside the rectangle spanned between  $\psi(s)$  and  $\psi(t)$ . See Fig. 1 for an example of a rectangular bar visibility representation of a planar  $st$ -graph.



**Fig. 1.** A planar  $st$ -graph  $G$  and a rectangular bar visibility representation  $\psi$  of  $G$ .

Tamassia and Tollis [16] showed that a planar digraph  $G$  admits a rectangular bar visibility representation if and only if  $G$  is a planar  $st$ -graph. In Sect. 3 we give an efficient algorithm for the following problem:

Rectangular Bar Visibility Representation Extension for  $st$ -graphs:

**Input:**  $(G, \psi')$ ;  $G$  is a planar  $st$ -graph;  $\psi'$  is a map assigning bars to a  $V' \subseteq V(G)$ .

**Question:** Does  $G$  admit a rectangular bar visibility representation  $\psi$  with  $\psi|_{V'} = \psi'$ ?

The main result in this paper is the following.

**Theorem 2.** *The Rectangular Bar Visibility Representation Extension Problem for an  $st$ -graph with  $n$  vertices can be solved in  $O(n \log^2 n)$  time.*

Our algorithm exploits the correspondence between bar visibility representations and  $st$ -orientations of planar graphs, and utilizes the  $SPQR$ -decomposition.

The rest of the paper is organized as follows. Section 2 contains the necessary definitions and description of the necessary tools. Section 3 contains the general ideas for the proof of Theorem 2. The omitted parts of the proof are reported in Appendix C together with some figures illustrating the ideas behind the proofs. Section 4 mentions further results from the full version and open problems.

## 2 Preliminaries

For a horizontal bar  $a$ , functions  $y(a)$ ,  $l(a)$ ,  $r(a)$  give respectively the  $y$ -coordinate of  $a$ , the  $x$ -coordinate of the left end of  $a$ , and the  $x$ -coordinate of the right end of  $a$ . For any bounded object  $Q$  in the plane, we use functions  $X(Q)$  and  $Y(Q)$  to denote the smallest possible, possibly degenerate, closed interval containing the projection of  $Q$  on the  $x$ -, and on the  $y$ -axis respectively. We denote the left end of  $X(Q)$  by  $l(Q)$  and the right end of  $X(Q)$  by  $r(Q)$ . Let  $a$  and  $b$  be two horizontal bars with  $y(a) < y(b)$ . We say that  $Q$  is *spanned between  $a$  and  $b$*  if  $X(Q) \subseteq X(a)$ ,  $X(Q) \subseteq X(b)$ , and  $Y(Q) = [y(a), y(b)]$ .

For a graph  $G$ , we often describe the visibility representation  $\psi$  by providing the values of functions  $y_\psi = y(\psi(v))$ ,  $l_\psi = l(\psi(v))$ ,  $r_\psi = r(\psi(v))$  for any vertex  $v$  of  $G$ . We drop the subscripts when the representation is known from the context.

Let  $G$  be a planar  $st$ -graph. An  $st$ -embedding of  $G$  is any planar embedding with  $s$  and  $t$  on the boundary of the outer face. A planar  $st$ -graph together with an  $st$ -embedding is called a *plane  $st$ -graph*. Vertices  $s$  and  $t$  of a planar (plane)  $st$ -graph are called the *poles* of  $G$ . We abuse notation and we use the term *planar (plane)  $uv$ -graph* to mean a planar (plane)  $st$ -graph with poles  $u$  and  $v$ . An *inner vertex* of  $G$  is a vertex of  $G$  other than the poles of  $G$ . A real valued function  $\xi$  from  $V(G)$  is an  $st$ -valuation of  $G$  if for each edge  $(u, v)$  we have  $\xi(u) < \xi(v)$ .

Tamassia and Tollis [16] showed that the following properties hold for any plane  $st$ -graph:

1. For every inner face  $f$ , the boundary of  $f$  consists of two directed paths with a common origin and a common destination.
2. The boundary of the outer face consists of two directed paths, with a common origin  $s$  and a common destination  $t$ .
3. For every inner vertex  $v$ , edges from  $v$  (to  $v$ ) are consecutive around  $v$ .

Let  $G$  be a plane  $st$ -graph. We introduce two objects associated with the outer face of  $G$ : the *left outer face*  $s^*$  and the *right outer face*  $t^*$ . Properties (1)–(3) allow us to introduce the following standard notions: *left/right face* of an edge and a vertex, *left/right path* of a face, and the *dual*  $G^*$  of  $G$  – a planar  $st$ -graph with vertex set consisting of inner faces of  $G$ ,  $s^*$ , and  $t^*$ . For two faces  $f$  and  $g$  in  $V(G^*)$  we say that  $f$  is to the left of  $g$ , and that  $g$  is to the right of  $f$ , if there is a directed path from  $f$  to  $g$  in  $G^*$ . See Appendix B.2 for the precise definitions which follow the standard definitions given by Tamassia and Tollis [16].

### 3 Rectangular Bar Visibility Representations of $st$ -graphs

In this section we provide an efficient algorithm that solves the rectangular bar visibility representation extension problem for  $st$ -graphs. Our algorithm employs a specific version of the  $SPQR$ -decomposition that allows us to describe all  $st$ -embeddings of a planar  $st$ -graph. See Appendix B.1 for the exact definition which follows the one given by Di Battista and Tamassia [8]. In particular, an  $SPQR$ -tree  $T$  of a planar  $st$ -graph  $G$  consists of nodes of four different types:  $S$  for *series nodes*,  $P$  for *parallel nodes*,  $Q$  for *edge nodes*, and  $R$  for *rigid nodes*. Each node  $\mu$  represents a *pertinent graph*  $G_\mu$ , a subgraph of  $G$  which is an  $st$ -graph with poles  $s_\mu$  and  $t_\mu$ . Additionally,  $\mu$  has an associated directed multigraph  $skel(\mu)$  called the *skeleton* of  $\mu$ . The only difference between our definition of the  $SPQR$ -tree and the one given in [8] is that we do not add an additional edge between the poles of the skeleton of a node. Our definition ensures that we have a one-to-one correspondence between the edges of  $skel(\mu)$  and the children of  $\mu$  in  $T$ . In Sect. 3.1, we use the  $SPQR$ -tree  $T$  of  $G$  to describe how a rectangular bar visibility representation is composed of rectangular bar visibility representations of the pertinent graphs of  $T$ .

The skeleton of a rigid node has only two *st*-embeddings, one being the flip of the other around the poles of the node. The skeleton of a parallel node with  $k$  children has  $k!$  *st*-embeddings, one for every permutation of the edges of the skeleton. The skeleton of a series node or an edge node has only one *st*-embedding.

Section 3.1 presents structural properties of bar visibility representations in relation to an *SPQR*-decomposition. In Sect. 3.2 we present an algorithm that solves this extension problem in quadratic time. In Appendix C.6 we give a refined algorithm that works in  $O(n \log^2 n)$  time for an *st*-graph with  $n$  vertices.

### 3.1 Structural Properties

Let  $\Gamma$  be a collection of pairwise disjoint bars. For a pair of bars  $a, b$  in  $\Gamma$  with  $y(a) < y(b)$  let the *set of visibility rectangles*  $R(a, b)$  be the interior of the set of points  $(x, y)$  in  $\mathbb{R}^2$  where:

1.  $a$  is the first bar in  $\Gamma$  on a vertical line downwards from  $(x, y)$ ,
2.  $b$  is the first bar in  $\Gamma$  on a vertical line upwards from  $(x, y)$ .

Figure 1 shows (shaded area) the set of visibility rectangles  $R(s, 5)$ . Note that there is a visibility gap between  $a$  and  $b$  in  $\Gamma$  iff  $R(a, b)$  is non-empty. If  $R(a, b)$  is non-empty, then it is a union of pairwise disjoint open rectangles spanned between  $a$  and  $b$ .

Let  $G$  be a planar *st*-graph and let  $T$  be the *SPQR*-tree for  $G$ . Let  $\psi$  be a rectangular bar visibility representation of  $G$ . For every node  $\mu$  of  $T$  we define the set  $B_\psi(\mu)$ , called the *bounding box of  $\mu$  with respect to  $\psi$* , as the closure of the following union:

$$\bigcup \{R(\psi(u), \psi(v)) : (u, v) \text{ is an edge of the pertinent digraph } G_\mu\}.$$

If  $\psi$  is clear from the context, then the set  $B_\psi(\mu)$  is denoted by  $B(\mu)$  and is called the *bounding box of  $\mu$* . Let  $B(\psi) = X(\psi(V(G))) \times Y(\psi(V(G)))$  be the minimal closed axis-aligned rectangle that contains the representation  $\psi$ . It follows that:

1.  $B(\psi) = B(\mu)$ , where  $\mu$  is the root of  $T$ ,
2. each point in  $B(\psi)$  is in the closure of at least one set of visibility rectangles  $R(\psi(u), \psi(v))$  for some edge  $(u, v)$  of  $G$ ,
3. each point in  $B(\psi)$  is in at most one set of visibility rectangles.

The following two lemmas describe basic properties of a bounding box.

**Lemma 2 (Q-Tiling Lemma).** *Let  $\mu$  be a Q-node in  $T$  corresponding to an edge  $(u, v)$  of  $G$ . For any rectangular bar visibility representation  $\psi$  of  $G$  we have:*

1.  $B(\mu)$  is a union of pairwise disjoint rectangles spanned between  $\psi(u)$  and  $\psi(v)$ .
2. If  $B(\mu)$  is not a single rectangle, then the parent  $\lambda$  of  $\mu$  in  $T$  is a P-node, and  $u, v$  are the poles of the pertinent digraph  $G_\lambda$ .

The Basic Tiling Lemma presented below describes the relation between the bounding box of an inner node  $\mu$  and the bounding boxes of the children of  $\mu$  in any rectangular bar visibility representation of  $G$ . The next lemma justifies the name *bounding box* for  $B(\mu)$ .

**Lemma 3 (Basic Tiling Lemma).** *Let  $\mu$  be an inner node in  $T$  with children  $\mu_1, \dots, \mu_k, k \geq 2$ . For a rectangular bar visibility representation  $\psi$  of  $G$  we have:*

1.  $\psi(v) \subseteq B(\mu)$  for every inner vertex  $v$  of  $G_\mu$ .
2.  $B(\mu)$  is a rectangle that is spanned between  $\psi(s_\mu)$  and  $\psi(t_\mu)$ .
3. The sets  $B(\mu_1), \dots, B(\mu_k)$  tile the rectangle  $B(\mu)$ , i.e.,  $B(\mu_1), \dots, B(\mu_k)$  cover  $B(\mu)$  and the interiors of  $B(\mu_1), \dots, B(\mu_k)$  are pairwise disjoint.

In the next three lemmas we specialize the Basic Tiling Lemma depending on whether  $\mu$  is a  $P$ -node, an  $S$ -node, or an  $R$ -node. These lemmas allow us to describe all tilings of  $B(\mu)$  by bounding boxes of  $\mu$ 's children. For Lemmas 4, 5, and 7 we let  $\mu_1, \dots, \mu_k$  be  $\mu$ 's children. The next lemma follows from the Basic Tiling Lemma and the Q-Tiling Lemma.

**Lemma 4 (P-Tiling Lemma).** *Let  $\mu$  be a  $P$ -node. For any rectangular bar visibility representation  $\psi$  of  $G$  we have:*

1. If  $(s_\mu, t_\mu)$  is not an edge of  $G$ , then the sets  $B(\mu_1), \dots, B(\mu_k)$  are rectangles spanned between  $\psi(s_\mu)$  and  $\psi(t_\mu)$ .
2. If  $(s_\mu, t_\mu)$  is an edge of  $G$ , then  $\mu$  has exactly one child that is a  $Q$ -node, say  $\mu_1$ , and:
  - For  $i = 2, \dots, k$ ,  $B(\mu_i)$  is a rectangle spanned between  $\psi(s_\mu)$  and  $\psi(t_\mu)$ .
  - $B(\mu_1) \neq \emptyset$  is a union of rectangles spanned between  $\psi(s_\mu)$  and  $\psi(t_\mu)$ .

When  $\mu$  is an  $S$ -node or an  $R$ -node, then there is no edge  $(s_\mu, t_\mu)$ . By the Q-Tiling Lemma and by the Basic Tiling Lemma, each set  $B(\mu_i)$  is a rectangle that is spanned between the bars representing the poles of  $G_{\mu_i}$ .

**Lemma 5 (S-Tiling Lemma).** *Let  $\mu$  be an  $S$ -node. Let  $c_1, \dots, c_{k-1}$  be the cut-vertices of  $G_\mu$  encountered in this order on a path from  $s_\mu$  to  $t_\mu$ . Let  $c_0 = s_\mu$ , and  $c_k = t_\mu$ . For any rectangular bar visibility representation  $\psi$  of  $G$ , for every  $i = 1, \dots, k - 1$ , we have  $X(\psi(c_i)) = X(B(\mu))$ . For every  $i = 1, \dots, k$ ,  $B(\mu_i)$  is spanned between  $\psi(c_{i-1})$  and  $\psi(c_i)$  and  $X(B(\mu_i)) = X(B(\mu))$ .*

The R-Tiling Lemma should describe all possible tilings of the bounding box of an  $R$ -node  $\mu$  that appear in all representations of  $G$ . Since there is a one-to-one correspondence between the edges of  $skel(\mu)$  and the children of  $\mu$ , we abuse notation and write  $B(u, v)$  to denote the bounding box of the child of  $\mu$  that corresponds to the edge  $(u, v)$ . By the Basic Tiling Lemma,  $B(u, v)$  is spanned between the bars representing  $u$  and  $v$ .

Suppose that  $\psi$  is a representation of  $G$ . The tiling  $\tau = (B_\psi(\mu_1), \dots, B_\psi(\mu_k))$  of  $B_\psi(\mu)$  determines a triple  $(\mathcal{E}, \xi, \chi)$ , where:  $\mathcal{E}$  is an  $s_\mu t_\mu$ -embedding of  $skel(\mu)$ ,  $\xi$  is an  $st$ -valuation of  $\mathcal{E}$ , and  $\chi$  is an  $st$ -valuation of  $\mathcal{E}^*$ , that are defined as



follows. Consider the following planar drawing of the  $st$ -graph  $skel(\mu)$ . Draw every vertex  $u$  in the middle of  $\psi(u)$ , and every edge  $e = (u, v)$  as a curve that starts in the middle of  $\psi(u)$ , goes a little above  $\psi(u)$  towards the rectangle  $B_\psi(u, v)$ , goes inside  $B_\psi(u, v)$  towards  $\psi(v)$ , and a little below  $\psi(v)$  to the middle of  $\psi(v)$ . This way we obtain a plane  $st$ -graph  $\mathcal{E}$ , which is an  $st$ -embedding of  $skel(\mu)$ . The  $st$ -valuation  $\xi$  of  $\mathcal{E}$  is just the restriction of  $y_\psi$  to the vertices from  $skel(\mu)$ , i.e.,  $\xi = y_\psi|V(skel(\mu))$ . To define the  $st$ -valuation  $\chi$  of  $\mathcal{E}^*$  we use the following lemma.

**Lemma 6 (Face Condition).**

1. Let  $f$  be a face in  $V(\mathcal{E}^*)$  different than  $t^*$ , and let  $v_0, v_1, \dots, v_p$  be the right path of  $f$ . There is a vertical line  $L_r(f)$  that contains the left endpoints of  $\psi(v_1), \dots, \psi(v_{p-1})$  and the left sides of  $B_\psi(v_0, v_1), \dots, B_\psi(v_{p-1}, v_p)$ .
2. Let  $f$  be a face in  $V(\mathcal{E}^*)$  different than  $s^*$ , and let  $u_0, u_1, \dots, u_m$  be the left path of  $f$ . There is a vertical line  $L_l(f)$  that contains the right endpoints of  $\psi(u_1), \dots, \psi(u_{q-1})$  and the right sides of  $B_\psi(u_0, u_1), \dots, B_\psi(u_{q-1}, u_q)$ .
3. If  $f$  is an inner face of  $\mathcal{E}$  then  $L_l(f) = L_r(f)$ .

The above lemma allows us to introduce the notion of a *splitting line* for every face  $f$  in  $V(\mathcal{E}^*)$ ; namely, it is: the line  $L_l(f) = L_r(f)$  if  $f$  is an inner face of  $\mathcal{E}$ ,  $L_r(f)$  if  $f$  is the left outer face of  $\mathcal{E}$ , and  $L_l(f)$  if  $f$  is the right outer face of  $\mathcal{E}$ . Now, let  $\chi(f)$  be the  $x$ -coordinate of the splitting line for a face  $f$  in  $V(\mathcal{E}^*)$ . To show that  $\chi(f)$  is an  $st$ -valuation of  $\mathcal{E}^*$ , note that for any edge  $(f, g)$  of  $\mathcal{E}^*$  there is an edge  $(u, v)$  of  $\mathcal{E}$  that has  $f$  on the left side and  $g$  on the right side. It follows that  $\chi(f) = l(B_\psi(u, v)) < r(B_\psi(u, v)) = \chi(g)$ , proving the claim.

The representation  $\psi$  of  $G$  determines the triple  $(\mathcal{E}, \xi, \chi)$ . Note that any other representation with the same tiling  $\tau = (B_\psi(\mu_1), \dots, B_\psi(\mu_k))$  of  $B(\mu)$  gives the same triple. To emphasize that the triple  $(\mathcal{E}, \xi, \chi)$  is determined by tiling  $\tau$ , we write  $(\mathcal{E}_\tau, \xi_\tau, \chi_\tau)$ .

Now, assume that  $\mathcal{E}$  is an  $st$ -embedding of  $skel(\mu)$ ,  $\xi$  is an  $st$ -valuation of  $\mathcal{E}$ , and  $\chi$  is an  $st$ -valuation of the dual of  $\mathcal{E}$ . Consider the function  $\phi$  that assigns to every vertex  $v$  of  $skel(\mu)$  the bar  $\phi(v)$  defined as follows:  $y_\phi(v) = \xi(v)$ ,  $l_\phi(v) = \chi(\text{left face of } v)$ ,  $r_\phi(v) = \chi(\text{right face of } v)$ . Firstly, Tamassia and Tollis [16] showed that  $\phi$  is a bar visibility representation of  $skel(\mu)$  and that for  $\tau = (B_\phi(\mu_1), \dots, B_\phi(\mu_k))$ , we have  $(\mathcal{E}_\tau, \xi_\tau, \chi_\tau) = (\mathcal{E}, \xi, \chi)$ . Secondly, there is a representation  $\psi$  of  $G$  that agrees with  $\tau$  on  $skel(\mu)$ , i.e., such that  $\tau = (B_\psi(\mu_1), \dots, B_\psi(\mu_k))$ . To construct such a representation, we take any representation  $\psi$  of  $G$ , translate and scale all bars in  $\psi$  to get  $B_\psi(\mu) = B_\phi(\mu)$ , and represent the pertinent digraphs  $G_{\mu_1}, \dots, G_{\mu_k}$  so that the bounding box of  $\mu_i$  coincides with  $B_\phi(\mu_i)$  for  $i = 1, \dots, k$ . This leads to the next lemma.

**Lemma 7 (R-Tiling Lemma).** *Let  $\mu$  be an  $R$ -node. There is a bijection between the set  $\{(B_\psi(\mu_1), \dots, B_\psi(\mu_k)) : \psi \text{ is a rectangular bar visibility representation of } G\}$  of all possible tilings of the bounding box of  $\mu$  by the bounding boxes of  $\mu_1, \dots, \mu_k$  in all representations of  $G$ , and the set  $\{(\mathcal{E}, \xi, \chi) : \mathcal{E} \text{ is an } st\text{-embedding of } skel(\mu), \xi \text{ is an } st\text{-valuation of } \mathcal{E}, \chi \text{ is an } st\text{-valuation of the dual of } \mathcal{E}\}$ .*

### 3.2 Algorithm

Let  $G$  be an  $n$ -vertex planar  $st$ -graph and let  $\psi'$  be a partial representation of  $G$  with the set  $V'$  of fixed vertices. We present a quadratic time algorithm that tests if there exists a rectangular bar visibility representation  $\psi$  of  $G$  that extends  $\psi'$ . If such a representation exists, the algorithm can construct it in the same time.

In the first step, our algorithm calculates  $y_{\psi'}$ . Namely, the algorithm checks whether  $y_{\psi'} : V' \rightarrow \mathbb{R}$  is extendable to an  $st$ -valuation of  $G$ . When such an extension does not exist, the algorithm rejects the instance  $(G, \psi')$ ; otherwise any extension of  $y_{\psi'}$  can be used as  $y_{\psi}$ . The next lemma verifies this step's correctness.

**Lemma 8.** *Let  $\psi$  be a rectangular bar visibility representation of  $G$  that extends  $\psi'$ .*

1. *The function  $y_{\psi}$  is an  $st$ -valuation of  $G$  that extends  $y_{\psi'}$ ,*
2. *If  $y$  is an  $st$ -valuation of  $G$  that extends  $y_{\psi'}$ , then a function  $\phi$  that sends every vertex  $v$  of  $G$  into a bar so that  $y_{\phi}(v) = y(v)$ ,  $l_{\phi}(v) = l_{\psi}(v)$ ,  $r_{\phi}(v) = r_{\psi}(v)$  is also a rectangular bar visibility representation of  $G$  that extends  $\psi'$ .*

Clearly, checking whether  $y_{\psi'}$  is extendable to an  $st$ -valuation of  $G$ , and constructing such an extension can be done in  $O(n)$  time. In the second step, the algorithm computes the  $SPQR$ -tree  $T$  for  $G$ , which also takes linear time.

Before we describe the last step in our algorithm, we need some preparation. For an inner node  $\mu$  in  $T$  we define the sets  $V'(\mu)$  and  $C(\mu)$  as follows:

$$\begin{aligned}
 V'(\mu) &= \text{the set of fixed vertices in } V(G_{\mu}) \setminus \{s_{\mu}, t_{\mu}\}, \\
 C(\mu) &= \begin{cases} \emptyset, & \text{if } V'(\mu) = \emptyset; \\ \text{the smallest closed rectangle containing } \psi'(u) & \text{for all } u \in V'(\mu), \text{ otherwise.} \end{cases}
 \end{aligned}$$

The set  $C(\mu)$  is called the *core* of  $\mu$ . For a node  $\mu$  whose core is empty, our algorithm can represent  $G_{\mu}$  in any rectangle spanned between the poles of  $G_{\mu}$ . Thus, we focus our attention on nodes whose core is non-empty.

Assume that  $\mu$  is a node whose core is non-empty. We describe the ‘possible shapes’ the bounding box of  $\mu$  might have in a representation of  $G$  that extends  $\psi'$ . The bounding box of  $\mu$  is a rectangle that is spanned between the bars corresponding to the poles of  $G_{\mu}$ . By the Basic Tiling Lemma, if  $C(\mu)$  is non-empty then  $B(\mu)$  contains  $C(\mu)$ . For our algorithm it is important to distinguish whether the left (right) side of  $B(\mu)$  contains the left (right) side of  $C(\mu)$ . This criterion leads to four types of representations of  $\mu$  with respect to the core of  $\mu$ .

The main idea of the algorithm is to decide for each inner node  $\mu$  whose core is non-empty, which of the four types of representation of  $\mu$  are possible and which are not. The algorithm traverses the tree bottom-up and for each node and each type of representation it tries to construct the appropriate tiling using the information about possible representations of its children. The types chosen for different children need to fit together to obtain a tiling of the parent node. In what follows, we present our approach in more detail.

Let  $\mu$  be an inner node in  $T$ . Fix  $\phi' = \psi'|V'(\mu)$ . Function  $\phi'$  gives a partial representation of the pertinent digraph  $G_\mu$  obtained by restricting  $\psi'$  to the inner vertices of  $G_\mu$ . Let  $x, x'$  be two real values. A rectangular bar visibility representation  $\phi$  of  $G_\mu$  is called an  $[x, x']$ -representation of  $\mu$  if  $\phi$  extends  $\phi'$  and  $X(\phi(s_\mu)) = X(\phi(t_\mu)) = [x, x']$ . We say that an  $[x, x']$ -representation of  $\mu$  is:

- left-loose, right-loose (LL), when  $x < l(C(\mu))$  and  $x' > r(C(\mu))$ ,
- left-loose, right-fixed (LF), when  $x < l(C(\mu))$  and  $x' = r(C(\mu))$ ,
- left-fixed, right-loose (FL), when  $x = l(C(\mu))$  and  $x' > r(C(\mu))$ ,
- left-fixed, right-fixed (FF), when  $x = l(C(\mu))$  and  $x' = r(C(\mu))$ .

The next lemma justifies this categorization of representations. It says that if a representation of a given type exists, then every representation of the same type is also realisable.

**Lemma 9 (Stretching Lemma).** *Let  $\mu$  be an inner node whose core is non-empty. If  $\mu$  has an LL-representation, then  $\mu$  has an  $[x, x']$ -representation for any  $x < l(C(\mu))$  and any  $x' > r(C(\mu))$ . If  $\mu$  has an LF-representation, then  $\mu$  has an  $[x, x']$ -representation for any  $x < l(C(\mu))$  and  $x' = r(C(\mu))$ . If  $\mu$  has an FL-representation, then  $\mu$  has an  $[x, x']$ -representation for  $x = l(C(\mu))$  and any  $x' > r(C(\mu))$ .*

The main task of the algorithm is to verify which representations are feasible for nodes that have non-empty cores. We assume that:  $\mu$  is an inner node whose core is non-empty;  $\mu_1, \dots, \mu_k$  are the children of  $\mu$ ,  $k \geq 2$ ;  $\lambda_1, \dots, \lambda_{k'}$  are the children of  $\mu$  with  $C(\lambda_i) \neq \emptyset$ ,  $0 \leq k' \leq k$ ;  $\theta(\lambda_i)$  is the set of feasible types of representations for  $\lambda_i$ ,  $\theta(\lambda_i) \subseteq \{LL, LF, FL, FF\}$ . We process the tree bottom-up and assume that  $\theta(\lambda_i)$  is already computed and non-empty.

Let  $x$  and  $x'$  be two real numbers such that  $x \leq l(C(\mu))$  and  $x' \geq r(C(\mu))$ . We provide an algorithm that tests whether an  $[x, x']$ -representation of  $\mu$  exists. We use it to find feasible types for  $\mu$  by calling it 4 times with appropriate values of  $x$  and  $x'$ . While searching for an  $[x, x']$ -representation of  $\mu$  our algorithm tries to tile the rectangle  $[x, x'] \times [y(s_\mu), y(t_\mu)]$  with  $B(\mu_1), \dots, B(\mu_k)$ . The tiling procedure is determined by the type of  $\mu$ . Note that as the core of a  $Q$ -node is empty, the algorithm splits into three cases:  $\mu$  is an  $S$ -node, a  $P$ -node, and an  $R$ -node. The pseudocode for the algorithms is given in Appendix C.3.

**Case S.  $\mu$  is an  $S$ -node.** In this case we attempt to align the left and right side of the bounding box of each child  $\lambda$  of  $\mu$  to  $x$  and  $x'$  respectively. For example, if the core of  $\lambda$  is strictly contained in  $[x, x']$ , then  $\lambda$  must have an LL-representation. The other cases follow similarly. We also must set the  $x$ -intervals of the bars of the cut vertices of  $G_\mu$  to  $[x, x']$ . The S-Tiling Lemma and the Stretching Lemma imply the correctness of this approach.

**Case P.  $\mu$  is a  $P$ -node.** In this case we attempt to tile the rectangle  $[x, x'] \times [y(s_\mu), y(t_\mu)]$  by placing the bounding boxes of the children of  $\mu$  side by side from left to right. The order of children whose cores are non-empty is determined by the position of those cores. We sort  $\lambda_1, \dots, \lambda_{k'}$  by the left ends of their cores. Let  $l_i = l(C(\lambda_i))$  and  $r_i = r(C(\lambda_i))$ ,  $r_0 = x$ ,  $l_{k'+1} = x'$ , and without loss of generality  $l_1 < \dots < l_{k'}$ .

We need to find enough space to place the bounding boxes of children whose cores are empty. Additionally, if  $(s_\mu, t_\mu)$  is an edge of  $G$ , then we need to leave at least one visibility gap in the tiling for that edge. Otherwise, if  $(s_\mu, t_\mu)$  is not an edge of  $G$ , we need to close all the gaps in the tiling. A more detailed description of the algorithm follows.

If there are  $\lambda_i, \lambda_{i+1}$  such that the interior of the set  $X(C(\lambda_i)) \cap X(C(\lambda_{i+1}))$  is non-empty, then we prove that there is no  $[x, x']$ -representation of  $G_\mu$ . Indeed, by the P-Tiling Lemma and by  $C(\lambda_i) \subseteq B(\lambda_i)$ , the interior of  $B(\lambda_i) \cap B(\lambda_{i+1})$  is non-empty and hence tiling of  $B(\mu)$  with  $B(\mu_1), \dots, B(\mu_k)$  is not possible. Additionally, if  $r(C(\lambda_i)) = l(C(\lambda_{i+1}))$ , then neither a right-loose representation of  $\lambda_i$  nor a left-loose representation of  $\lambda_{i+1}$  can be used, so we delete such types of representations from  $\theta(\lambda_i)$  and  $\theta(\lambda_{i+1})$ . If that leaves some  $\theta(\lambda_i)$  empty, then an  $[x, x']$ -representation of  $\mu$  does not exist. These checks take  $O(k')$  time.

Let  $Q_i = [r_i, l_{i+1}] \times [y(s_\mu), y(t_\mu)]$  for  $i \in [0, k']$ . We say that  $Q_i$  is an *open gap* (after  $\lambda_i$ , before  $\lambda_{i+1}$ ) if  $Q_i$  has non-empty interior. In particular, if  $x = r_0 < l_1$  ( $r_{k'} < l_{k'+1} = x'$ ) then there is an open gap before  $\lambda_1$  (after  $\lambda_{k'}$ ). On the one hand, if there is an edge  $(s_\mu, t_\mu)$  or there is at least one  $\mu_i$  whose core is empty then we need at least one open gap to construct an  $[x, x']$ -representation. On the other hand, if  $(s_\mu, t_\mu)$  is not an edge of  $G$  then we need to close all the gaps in the tiling. There are two ways to close the gaps. Firstly, the representation of each child node whose core is empty can be placed so that it closes a gap. The second way is to use loose representations for children nodes  $\lambda_1, \dots, \lambda_{k'}$ .

Suppose that  $c$  is a function that assigns to every  $\lambda_i$  a feasible type of representation from the set  $\theta(\lambda_i)$ . Whenever  $c(\lambda_i)$  is right-loose or  $c(\lambda_{i+1})$  is left-loose, we can stretch the representation of  $\lambda_i$  or  $\lambda_{i+1}$ , so that it closes the gap  $Q_i$ . We describe a simple greedy approach to close the maximum number of gaps in this way. We process the  $\lambda_i$ 's from left to right and for each one: we close both adjacent gaps if we can (i.e.  $LL \in \theta(\lambda_i)$ ); otherwise, we prefer to close the left gap if it is not yet closed rather than the right gap. This is optimal by a simple greedy exchange argument.

If there are still  $g > 0$  open gaps left and  $(s_\mu, t_\mu)$  is not an edge of  $G$ , then each open gap needs to be closed by placing in this gap a representation of one or more of the children whose core is empty. Thus, it is enough to check that  $k - k' \geq g$ . The correctness of the described algorithm follows by the P-Tiling Lemma, and the Stretching Lemma.

**Case R.  $\mu$  is an R-node.** The detailed discussion of this case is reported in Appendix C.4. Here, we sketch our approach. By the R-Tiling Lemma, the set of possible tilings of  $B(\mu)$  by  $B(\mu_1), \dots, B(\mu_k)$  is in correspondence with the triples  $(\mathcal{E}, \xi, \chi)$ , where  $\mathcal{E}$  is a planar embedding of  $skel(\mu)$ ,  $\xi$  is an  $st$ -valuation of  $\mathcal{E}$ , and  $\chi$  is an  $st$ -valuation of  $\mathcal{E}^*$ . To find an appropriate tiling of  $B(\mu)$  (that yields an  $[x, x']$ -representation of  $\mu$ ) we search through the set of such triples. Since  $\mu$  is a rigid node, there are only two  $st$ -embeddings of  $skel(\mu)$  and we consider both of them separately. Let  $\mathcal{E}$  be one of these planar embeddings. Since the  $y$ -coordinate for each vertex of  $G$  is already fixed, the  $st$ -valuation  $\xi$  is given by the  $y$ -coordinates of the vertices from  $skel(\mu)$ . It remains to find an  $st$ -valuation  $\chi$  of  $\mathcal{E}^*$ , i.e., to determine the  $x$ -coordinate of the splitting line for every face.

We claim that the existence of an  $st$ -valuation  $\chi$  is equivalent to checking the satisfiability of a carefully designed 2-CNF formula. For every child  $\lambda$  of  $\mu$  whose core is non-empty, we introduce two boolean variables that indicate which type (LL, LF, FL, FF) of representation is used for  $\lambda$ . Additionally, for every inner face  $f$  of  $\mathcal{E}$  we introduce two boolean variables: the first (the second) indicates if the splitting line of  $f$  is set to the leftmost (rightmost) possible position determined by the bounding boxes of nodes on the left (right) path of  $f$ . Now, using those variables, we can express that: feasible representations of the children nodes are used, splitting line of a face  $f$  agrees with the choice of representation for the nodes on the boundary of  $f$  (see Face Condition Lemma), the choice of splitting lines gives an  $st$ -valuation of  $\mathcal{E}^*$ .

In Appendix C.4 we present a formula construction that uses a quadratic number of clauses and results in a quadratic time algorithm. In Appendix C.6, we present a different, less direct, approach that constructs smaller formulas for  $R$ -nodes and leads to the  $O(n \log^2 n)$  time algorithm. Therefore, Lemma 8, and the discussion of cases S, P, and R, together with the results in Appendix C imply Theorem 2.

## 4 Concluding Remarks and Open Problems

We considered the representation extension problem for bar visibility representations and provided an efficient algorithm for  $st$ -graphs and showed NP-completeness for planar graphs. An important variant of bar visibility representations is when all bars used in the representation have integral coordinates, i.e., *grid representations*. Any visibility representation can be easily modified into a grid representation. However, this transformation does not preserve coordinates of the given vertex bars. Indeed, we can show (in Appendix D.3) that the (Rectangular) Bar Visibility Representation Extension problem is NP-hard on series-parallel  $st$ -graphs when one desires a grid representation.

We conclude with two natural, interesting open problems. The first one is to decide if there exists a polynomial time algorithm that checks whether a partial representation of a directed planar graph is extendable to a bar visibility representation of the whole graph. Although we show an efficient algorithm for an important case of planar  $st$ -graphs, it seems that some additional ideas are needed to resolve this problem in general. The second one is to decide if there is an efficient algorithm for recognition of digraphs admitting strong visibility representation, and for the corresponding partial representation extension problem.

## References

1. Andreae, T.: Some results on visibility graphs. *Discrete Appl. Math.* **40**(1), 5–17 (1992)
2. Angelini, P., Di Battista, G., Frati, F., Jelínek, V., Kratochvíl, J., Patrignani, M., Rutter, I.: Testing planarity of partially embedded graphs. *ACM Trans. Algorithms* **11**(4), 32:1–32:42 (2015)

3. de Berg, M., Khosravi, A.: Optimal binary space partitions in the plane. In: Thai, M.T., Sahni, S. (eds.) COCOON 2010. LNCS, vol. 6196, pp. 216–225. Springer, Heidelberg (2010). doi:[10.1007/978-3-642-14031-0\\_25](https://doi.org/10.1007/978-3-642-14031-0_25)
4. Chaplick, S., Dorbec, P., Kratochvíl, J., Montassier, M., Stacho, J.: Contact representations of planar graphs: extending a partial representation is hard. In: Kratsch, D., Todinca, I. (eds.) WG 2014. LNCS, vol. 8747, pp. 139–151. Springer, Heidelberg (2014). doi:[10.1007/978-3-319-12340-0\\_12](https://doi.org/10.1007/978-3-319-12340-0_12)
5. Chaplick, S., Guśpiel, G., Gutowski, G., Krawczyk, T., Liotta, G.: The partial visibility representation extension problem (2015). Pre-print [arXiv:1512.00174](https://arxiv.org/abs/1512.00174)
6. Di Battista, G., Eades, P., Tamassia, R., Tollis, I.G.: Graph Drawing: Algorithms for the Visualization of Graphs. Prentice-Hall, Upper Saddle River (1999)
7. Di Battista, G., Tamassia, R.: Algorithms for plane representations of acyclic digraphs. *Theoret. Comput. Sci.* **61**(2–3), 175–198 (1988)
8. Di Battista, G., Tamassia, R.: On-line planarity testing. *SIAM J. Comput.* **25**(5), 956–997 (1996)
9. Duchet, P., Hamidoune, Y.O., Vergnas, M.L., Meyniel, H.: Representing a planar graph by vertical lines joining different levels. *Discrete Math.* **46**(3), 319–321 (1983)
10. Garg, A., Tamassia, R.: Upward planarity testing. *Order* **12**(2), 109–133 (1995)
11. Garg, A., Tamassia, R.: On the computational complexity of upward and rectilinear planarity testing. *SIAM J. Comput.* **31**(2), 601–625 (2001)
12. Ghosh, S.K., Goswami, P.P.: Unsolved problems in visibility graphs of points, segments, and polygons. *ACM Comput. Surv.* **46**(2), 22:1–22:29 (2013)
13. Luccio, F., Mazzone, S., Wong, C.K.: A note on visibility graphs. *Discrete Math.* **64**(2–3), 209–219 (1987)
14. Otten, R., van Wijk, J.G.: Graph representations in interactive layout design. In: Proceedings of IEEE International Symposium on Circuits and Systems, New York, NY, USA, May 1978, pp. 914–918 (1978)
15. Patrignani, M.: On extending a partial straight-line drawing. *Int. J. Found. Comput. Sci.* **17**(5), 1061–1070 (2006)
16. Tamassia, R., Tollis, I.G.: A unified approach to visibility representations of planar graphs. *Discrete Comput. Geom.* **1**(4), 321–341 (1986)
17. Wismath, S.K.: Characterizing bar line-of-sight graphs. In: Proceedings of 1st Annual Symposium on Computational Geometry SCG 1985, Baltimore, MD, USA, June 1985, pp. 147–152 (1985)