Integral Unit Bar-Visibility Graphs

Therese Biedl* Ahmad Biniaz* Veronika Irvine* Philipp Kindermann* Anurag Murty Naredla*
Alexi Turcotte*

Abstract

In this paper, we take another look at unit bar-visibility representations, that is bar-visibility representations where every bar has the same width. Motivated by some applications in textile construction, we restrict the graphs further to integral unit bar-visibility representations (IUBVR), that is a bar-visibility representation where the bar of every vertex is a horizontal line segment $[i-1, i]$, for some positive integer i, at some real-value y position.

We study which graph classes do/don't have an IUBVR, both in the weak model and in the strong model. In the weak model, we show that it is NP-hard to test whether a graph has an IUBVR. We also present recursive algorithms to create IUBVRs for some graph classes, such as 2-connected outerplanar graphs with maximum degree 4. In the strong model, we provide a polynomial-time algorithm to test for the existence of a strong IUBVR. In the event of a positive answer, the algorithm also generates such a strong IUBVR.

1 Introduction

The topic of *bar-visibility representations* is well-studied in the graph drawing community. We want to represent a graph by assigning a horizontal line segment (*bar*) to every vertex in such a way that no two bars share a point and for every edge (a, b), the two bars assigned to a and b are visible to each other in the sense that some vertical segment drawn from a reaches b without crossing any other vertices—we call this vertical segment a *line-of-sight*. There are some variations of this idea. Sometimes a line-of-sight is required to exist along a positive-width strip (ε*-visibility*) [11, 3]. Also, in the *strong model*, if a line-of-sight exists between two intervals, then there *must* be a corresponding edge in the graph, whereas in the *weak model* such edges may or may not exist.

It is well-understood which graphs have a bar-visibility representation (we give a detailed review below). Researchers have therefore turned their attention to versions where the bars are further restricted. Of particular interest to us are *unit bar-visibility representations*, where every bar has unit width. Motivated by some applications in textile construction, we take this one step further and study in this paper an *integral unit bar-visibility representation* (IUBVR), which is a bar-visibility representation where the bar of every vertex has the form $[i_v-1, i_v] \times y_v$ for some $i_v \in \mathbb{N}$ and $y_v \in \mathbb{R}$.

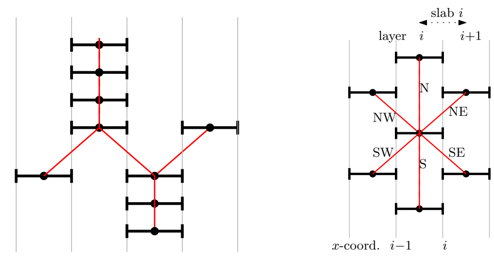
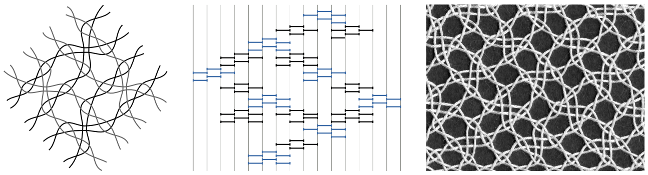


Figure 1: A strong IUBVR of a tree, and six edge-directions at a vertex.

Motivation: We first came across the idea of IUBVR in a problem we studied related to bobbin lace. In bobbin lace, and other methods of creating textiles such as macramé or friendship bracelets, the standard setup is several strings hanging down in parallel. The artist picks up a few (typically in multiples of two) consecutive strings and first braids (or knots) them together, then releases the strings and creates another braid elsewhere. A braid can only be executed if the strings involved hang freely, i.e., the braid must be below all previously executed braids involving any of the strings in this subset. This can be modeled as a graph as follows: Define the vertical line with x-coordinate $i \in \mathbb{N}$ to represent one of the strings. Now draw a horizontal bar $[i-1, i]$ to represent a braid that involves the strings at $i-1$ and i; the y-coordinate of the bar represents the relative order of the braid.

Notice that the strong bar-visibility representation induced by these bars is an IUBVR. If we direct all edges in the resulting graph downward, then the possible topological orders of the resulting digraph correspond exactly to the orders in which we can execute the braids. Because we want to maximize the number of crossings that can be made using the threads (or a subset of the threads) already in the artist's hand, a function of braiding-order, we became interested in the types of graphs that could be represented in such a fash-

*University of Waterloo, Canada. Research of TB, AB and VI supported by NSERC. This research was initiated at the Algorithmic Problem Session group at the University of Waterloo; many thanks to Craig Kaplan and Anna Lubiw for helpful input.



(a) (b) (c)

Figure 2: Bobbin lace motivation: (a) Drawing of one repeat, (b) sIUBVR of pattern, (c) several repeats worked in cotton thread

ion.

Our contributions: In this paper, we study which graph classes do/don't have an IUBVR, both in the weak model and in the strong model. In the weak model, we show that it is NP-hard to test whether a plane graph has an IUBVR; we denote a weak IUBVR by wIUBVR. We also present recursive algorithms to create IUBVRs for some graph classes, such as 2-connected outerplanar graphs with maximum degree 4.

We then turn to the strong model. As a warm-up, we argue exactly which trees have a strong IUBVR, denoted by sIUBVR. Then, we turn to the most intricate result of this paper and provide a polynomial-time algorithm to test for the existence of an sIUBVR. In the event of a positive answer, the algorithm also generates such an sIUBVR.

1.1 Related work

The primary application of bar-visibility graphs is to generate a compact layout for a printed VLSI circuit board. The research in this area covers two main topics: 1) characterizing all graphs that have a bar-visibility representation and 2) determining whether a specific graph supports a bar-visibility representation.

Every planar graph has a weak bar-visibility representation [4] and that representation can be found in linear time [8, 9, 11]. Determining whether a 3-connected planar graph has a strong bar-visibility representation was shown to be NP-complete by Andreae [1]. However, for maximal planar and 4-connected planar graphs, there exist $O(|V|)$ and $O(|V|^3)$ algorithms, respectively, for computing a strong visibility representation [11].

Melnikov introduced the idea of ε-visibility [7] as the model most germane to VLSI layout. Duchet showed that every maximal planar graph has an ε-visibility representation [4] which Thomassen extended to all 3-connected planar graphs [12]. Wismath [14] as well as Tamassia and Tollis [11] independently characterized bar-visibility graphs under the ε-model as planar graphs having all cutpoints on a single face. This can be tested, and the representation constructed, in linear time [11].

In addition to bars, axis-aligned rectangles which admit a horizontal as well as a vertical line-of-sight have been explored [2].

Unit bar-visibility representations The concept of bar-visibility graphs with uniform length bars was first studied by Dean and Veytsel [3] under the ε-model. They characterized the existence of such representations for several graph classes including trees, complete bipartite, outerplanar and triangulated graphs. Wiglesworth [13] characterized graphs with a bar-visibility representation with reach (maximum distance between the left and right bar coordinates) less than 2.

Layered drawings IUBVRs turn out to be closely related to so-called *layered drawings*, see e.g. Suderman [10] and the references therein. A *layered drawing* in the most general sense is a planar straight-line drawing where every vertex is assigned to a *layer* or *level*, i.e., a vertical line with integer x-coordinate.[1] There are a number of different models, depending on what types of edges are allowed. In this paper, we consider *short layered drawings*: every edge (v, w) must satisfy that its ends are either in the same layer or in adjacent layers. Any IUBVR naturally gives rise to a short layered drawing, see below. The second type of layered drawing that we will need is called *proper layered drawing* in [10] but we use the term *leveled-planar drawing* from [6]; here, for every edge (v, w) the two vertices must be on adjacent layers. Heath and Rosenberg [6] showed that it is NP-hard to determine whether a planar graph has a leveled-planar drawing. Suderman [10] studied various models of layered drawings with the objective of obtaining such drawings with few layers for trees. It is also known that for various models of layered drawings minimizing the number of layers is fixed-parameter tractable in the number of layers [5].

2 Preliminaries

Since a bar visibility representation can only exist for a planar graph, we assume throughout the paper that G is a planar graph, i.e., can be drawn without crossings in the plane. We usually assume that G is *plane*, i.e., we have fixed the clockwise order of edges at every vertex (which determines the *faces*) and the outer-face of G.

Fix an IUBVR D of G where, as before, vertex v has bar $[i_v{-}1, i_v] \times y_v$. We can created a drawing Γ from D by placing vertex v at point $(i_v - \frac{1}{2}, y_v)$ and drawing edges as straight-line segments. It is straightforward to verify that Γ is a planar short layered drawing; we call this the *associated layered drawing of D*. (Not all short layered drawings come from an IUBVR.) Based on the associated layered drawing, the following terminology is natural: Vertex v is said to reside in *layer i of D* if $i_v{=}i$.

[1] Some references use horizontal lines instead; this is the same after a rotation.

An edge e *spans slab i of D* if the ends of e are in layers i and $i{+}1$. We furthermore need the following crucial observation (see also Fig. 1):

Observation 1 *In an IUBVR D, every vertex v has at most 6 incident edges which can be classified as follows:*

- *There can be at most one* N-edge *connecting v to a neighbour w with $i_w{=}i_v$ and $y_w{>}y_v$.*
- *There can be at most one* NE-edge *connecting v to a neighbour w with $i_w{=}i_v{+}1$ and $y_w{>}y_v$.*
- *There can be at most one* NW-edge *connecting v to a neighbour w with $i_w{=}i_v{-}1$ and $y_w{>}y_v$.*
- *Symmetrically there can be at most one* S-edge, SE-edge *and* SW-edge *in which the condition "$y_w{>}y_v$" is replaced by "$y_w{<}y_v$".*

Proof. Assume that there are two NE-edges, say (v,w) and (v,x). So $i_w = i_x = i_v + 1$ and (say) $y_v < y_w < y_x$. But then the bar of w blocks all lines of sight from v to x. So there is at most one NE-edge. $\square$

Fix some $\alpha \in \{\mathrm{NW}, \mathrm{N}, \mathrm{NE}, \mathrm{SE}, \mathrm{S}, \mathrm{SW}\}$. In an IUBVR, we say that a directed edge $v \to w$ has *orientation α* if it is the α-edge at v. We use the term also for an undirected edge, meaning that it becomes an α-edge when directed suitably. We also call edges *vertical*, *diagonal*, *upward-diagonal* and *downward-diagonal* in the obvious way. A vertex v may or may not have an α-edge. When creating drawings, we sometimes use the term *α-port* for the possibility of adding an edge at v in that orientation.

We say that two IUBVRs, D and D', are *equivalent* if (possibly after a translation) the layers contain the same vertices in the same top-to-bottom order, and (after imposing arbitrary directions) the edges have the same orientations in D and D'. We say that a graph has a *unique* IUBVR if all its IUBVRs are equivalent up to a rotation by π.

3 Graph classes that admit weak IUBVRs

In this section, we show that all trees and 2-connected outerplanar graphs of maximum degree 4 admit wIUBVR.

Theorem 1 *Every tree T of maximum degree 4 has a wIUBVR.*

Proof. Created a rooted tree from T by selecting any leaf of T as the root. We prove the result for any rooted subtree T_x of T, by induction on the height of T_x. We created two wIUBVRs for T_x, one where the drawing resides within $R^{\urcorner}(x)$ and one that resides within $R^{\lrcorner}(x)$ (see Fig. 3a for the definition of these shapes; they are meant to extend rightward to infinity). We only explain how to construct the wIUBVR in $R^{\urcorner}(x)$; the other construction is symmetric. If x has no children then the

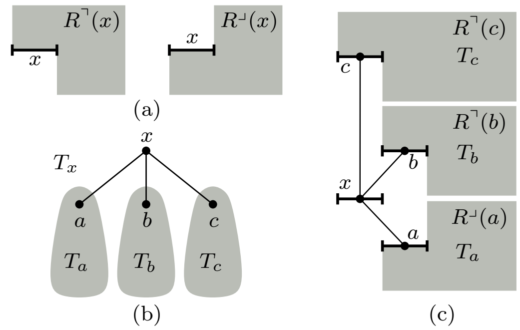
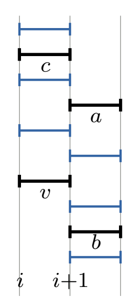


Figure 3: Illustration of the proof of Theorem 1: (a) Regions $R^{\urcorner}(x)$ and $R^{\lrcorner}(x)$, (b) a rooted tree T_x, and (c) a wIUBVR of T_x in $R^{\urcorner}(x)$.

bar of x alone gives the representation, so assume that x has children. Since T has maximum degree 4 and T is a rooted tree, every vertex has at most 3 children; we assume here that x has exactly three children a, b, c (we can pad the tree with some dummy-children if it has fewer). If the given bar for x is in layer i, then place unit bars for a and b in layer $i{+}1$, with $y_a{<}y_x{<}y_b$. Place a unit bar for c in layer i with $y_x{<}y_c$. By the inductive hypothesis we can obtain representations of T_a, T_b, and T_c in $R^{\lrcorner}(a)$, $R^{\urcorner}(b)$, and $R^{\urcorner}(c)$, respectively. Putting these representations together, we obtain a representation of T_x in $R^{\urcorner}(x)$ as depicted in Fig. 3c. $\square$

Theorem 2 *There are trees of maximum degree 5 without a wIUBVR.*



Proof. Consider a tree T with a degree-5 vertex v that is adjacent to five degree-5 vertices. Assume for contradiction that T has a wIUBVR. Of the 6 ports at v (cf. Observation 1), exactly one is unused. Up to symmetry we may assume that the unused port is either the SW-port or the S-port. Let the NE-edge, SE-edge and N-edge be (v,a), (v,b) and (v,c). Observe that a cannot have a NW-edge, because any such neighbour would need to be below c and hence block the line-of-sight for (v,c). It also cannot have an S-edge because such a neighbour, regardless of its position, would either block the line-of-sight for (v,b) or for (v,a). Therefore $\deg(a) \le 4$, a contradiction. $\square$

The following will be useful later:

Corollary 3 *Let G be a graph with an IUBVR D and an edge (v,a) where* $\deg(v){=}6$ *and* $\deg(a) \ge 5$ *and a, v have no common neighbor. Then (v,a) is vertical in D.*

Proof. Assume for contradiction that (v,a) is diagonal, say it is the NE-edge of v. By $\deg(v) = 6$ it has a SE-

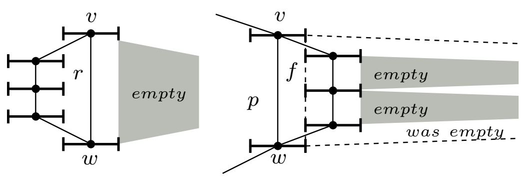


Figure 4: Creating a representation for max degree 3.

edge (v, b) and a N-edge (v, c). As in the previous proof then $\deg(a) \leq 4$, a contradiction. □

Now we turn to 2-connected outer-planar graphs, i.e., planar graphs whose outer-face is a simple cycle that contains all vertices. The *weak dual* G^* of such a graph is obtained by creating a vertex f^* for every inner face f and connecting two such vertices (f^*, g^*) with an edge in G^* if they share an edge (which is necessarily a chord of the outer-face cycle). It is well-known that the weak dual of a 2-connected outer-planar graph is a tree.

Theorem 4 *Every 2-connected outerplanar graph of maximum degree 4 has a wIUBVR.*

Proof. For brevity, we prove the case when the maximum degree is 3; the proof for maximum degree 4 is significantly more involved and is given in the appendix. Root the weak dual tree G^* at a face r^* that is a leaf. We now add the faces of G following their pre-order in the dual tree. We maintain the invariant that any chord e is drawn vertically and, as long as only one of the faces incident to e has been drawn, nothing is drawn to the right of e.

We start with the root r^* and let (v, w) be the unique chord of r (recall that r^* is a leaf of G^*). Draw r as a rotated trapezoid, with (v, w) on the long side in the right layer. Clearly the invariant holds. Now consider some face f and assume that the parent p^* of f^* has already been drawn, with the common chord (v, w) of p and f drawn vertically and without anything to its right. Draw f as a rotated trapezoid, with (v, w) as a unique edge on its left and all other vertices in the layer to the right of (v, w). The created SE-edge of v cannot be a chord because v already has three incident edges and therefore no other inner face can be incident here. Likewise the NE-edge of w cannot be a chord. So the invariant holds, and repeating for all faces gives a wIUBVR. □

4 Recognition in the weak model

In this section, we show that testing whether a plane graph admits a wIUBVR is NP-hard, by reducing from the NP-hard problem [6] of testing whether a given plane graph has a leveled-planar drawing.

In this reduction we will represent edges by a rigid structure. Consider the *rigid block* B depicted in Fig. 5a. One can easily show that B has a unique wIUBVR. Namely, since $\deg(a) = \deg(b) = 6$, by Corollary 3, (a, b) must be drawn vertically, say a above b. All other edge-orientations are then determined by the planar embedding since all ports at a and b are used.

We now combine M rigid blocks $B_1, \ldots, B_M$ to form an *M-tube*, see Fig. 5b. Since each rigid block has a unique wIUBVR, so does the M-tube. Note that an M-tube spans exactly $2M+1$ layers.

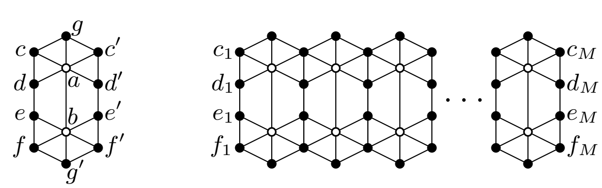
For representing the vertices, we make use of a different structure. Consider the *node block* L depicted in Fig. 6a. A node block can be connected to an M-tube as depicted in Fig. 6b. The wIUBVR of a node block is not unique; in particular it can be "bent" at the white squares.

Now fix a plane graph $G=(V_G, E_G)$ for which we wish to find a leveled-planar drawing. Create a plane graph H for which we wish to find a wIUBVR as follows:

Each edge e in G gives rise to an M-tube M_e in H, where M is chosen sufficiently large ($M=24n$ should do). Each vertex v of G is replaced by a *node gadget* N_v that consists of a cycle of d (where $d=\deg(v)$) node blocks $L_1, \ldots, L_d$ that are connected by identifying each vertex b_i with the vertex a_{i+1}; see Fig. 6c. Enumerating the edges around v as $e_1, \ldots, e_d$, we attach L_i to one end of M-tube M_{e_i} as depicted in Fig. 6b.

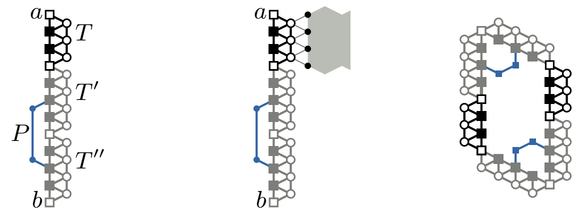
It is quite easy to see that if G has a leveled-planar drawing Γ, then H has an wIUBVR D, essentially by mapping level h of Γ to the layers from $h(2M+8) - 1$ to $h(2M+8) + 5$ in D, arguing that the node gadget N_v can be placed in those layers, and placing the M-tube of edge (v, w) in the $2M+1$ layers that are between the layers of its endpoints; see Fig. 7 for an example. For the other direction, we argue that there is in fact no other way than to lay out the node gadgets in these layers, so we can obtain a level-assignment that gives a leveled-planar drawing of G. The appendix has details. We conclude:

Theorem 5 *It is NP-hard to test whether a given plane graph H has a wIUBVR.*



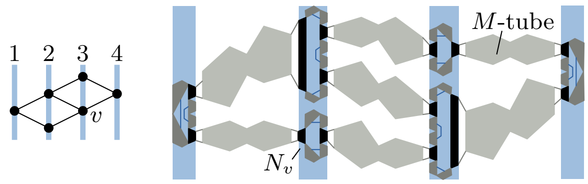
(a) A rigid block (b) An M-tube

Figure 5: The rigid structures used to represent edges.



(a) (b) (c)

Figure 6: (a) A node block (b) connected to an M-tube and (c) the node gadget of a degree-2 vertex.



(a) G (b) H

Figure 7: An example of a drawing of H obtained from a drawing of G. The node gadgets lie completely inside the shaded rectangles and the M-tubes lie in between.

5 Strong model

Now we turn to the strong model, where the existence of a line-of-sight implies that the corresponding edge *must* exist in G. As a simple warm-up result, we have:

Theorem 6 *A tree T has an sIUBVR if and only if it is a subdivision of a caterpillar of maximum degree 3.*

Proof. If T has an sIUBVR, then it also has a strong unit bar-visibility representation. As shown by Dean and Veytsel [3], it then has maximum degree 3 and is a subdivided caterpillar, i.e., it contains a path S (the spine) such that $T - S$ consists of paths (the subdivided legs). Vice versa, for any subdivided caterpillar of maximum degree 3 it is easy to create an sIUBVR; Fig. 1 illustrates the construction. □

The rest of this section is devoted to showing that more generally, we can test for any plane graph G whether it has an sIUBVR. We may assume that G is connected, else test each component separately. We assume for now that G is 2-connected.

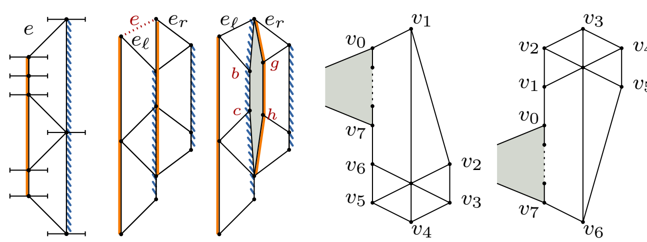
It will be convenient to direct outer-face edges so that the outer-face is to their left. Observe that in any IUBVR, the topmost diagonal edge that spans a slab is on the outer-face, with the outer-face above it; with the above direction therefore its orientation is NE or SE.

Let's start by outlining the idea. We create an auxiliary directed graph H that has a super-source s, a super-sink t, and a vertex $v(e, \alpha)$ for each *configuration* (e, α), where e is an edge on the outer-face of G and $\alpha \in \{\mathrm{SE}, \mathrm{NE}\}$. Vertex $v(e, \alpha)$ expresses the possibility of an sIUBVR where e has orientation α and e is the topmost edge in the slab that it spans. Crucially, fixing (e, α) determines the entire drawing within this slab. We can also define conditions under which two configurations, (e_ℓ, α_ℓ) and (e_r, α_r), could occur in consecutive slabs; if they are met, add an arc $v(e_\ell, \alpha_\ell) \to v(e_r, \alpha_r)$ to H. Likewise we can add arcs $s \to v(e, \alpha)$ or $v(e, \alpha) \to t$ if (e, α) could occur in the leftmost/rightmost slab. Testing whether an sIUBVR exists then amounts to finding a directed path from s to t in H.

To explain the details, we need a few observations.

Lemma 7 *In any sIUBVR D, any internal face f spans exactly one slab of D.*

Proof. In the strong model, the vertices within one layer i form an induced path connecting two vertices on the outer-face. Hence any face is either to the left of i or to the right of i, and so it cannot span two slabs. □



(a) (b) (c) (d)

Figure 8: (a) The sub-drawing $D(e, \mathrm{SE})$. $L(e, \mathrm{SE})$ is orange (bold), $R(e, \mathrm{SE})$ is blue (dashed). (b and c) Two examples of slabs that are not compatible: (b) The combined drawing creates an unwanted edge e. (c) $R(e_\ell, \mathrm{NE}) \cup L(e_r, \mathrm{SE})$ is not an induced path, leading to parts of G (gray) that are not represented. (d) Example that has non-unique sIUBVRs.

In fact, as illustrated in Fig. 8a, inner faces in an sIUBVR have a special form. We say that an inner face f *forms a trapezoid* (in some sIUBVR) if the edges of f can be enumerated (in clockwise or counter-clockwise order) as $e_1, e_2, \dots, e_d$ such that e_1 is upward diagonal, e_2 is vertical, e_3 is downward diagonal, and $e_4, \dots, e_d$, if they exist, are vertical (Note that we allow the trapezoid to degenerate to a triangle).

Lemma 8 *Fix an arbitrary sIUBVR D. Then any internal face f forms a trapezoid.*

Proof. Assume f spans slab i and let e_b, e_t be the bottommost/topmost edges in slab i that belong to f. One can easily argue that e_b and e_t must have different orientations, else the face would be split by another diagonal edge that spans slab i and is between e_b and e_t. Up to symmetry e_t is upward-diagonal and e_b is downward-diagonal. The right ends of e_t and e_b are connected to each other by a path that runs along level $i+1$. If this path contains any vertices other than the right ends of e_t and e_b, these additional vertices would necessarily be below the top end of e_t and above the bottom end of e_b. At least one of these vertices would be adjacent to a vertex on the left side of f (the bottom end of e_t or the top end of e_b or some vertex in-between). This again would split face f, a contradiction. So there are no vertices in this path, meaning that the right ends of e_b and e_t are connected by a single vertical edge as desired. □

As a consequence, fixing the topmost edge of a face and its orientation fixes the orientation for all edges of that face. We can propagate this to all faces of a slab, which gives the crucial insight for our algorithm.

Lemma 9 *Let D, D' be two sIUBVRs of a graph G. Assume that some outer-face edge e has the same orientation $\alpha \in \{\text{NE},\text{SE}\}$ in D and D', and is the topmost edge in its slab in both D and D'. Then the slab of e in D and the slab of e in D' contain exactly the same faces and edges, in exactly the same order from top to bottom.*

Proof. Set $e_1{=}e$ and let f_1 be the unique inner face adjacent to e. Since e_1 is topmost and has orientation α, we know exactly the trapezoidal shape that f_1 must take, and therefore, the unique other edge e_2 that is diagonal and on f_1. Further, e_2 has the opposite orientation of e_1. Now repeat with e_2. Generally, once e_i is fixed, let f_i be the face incident to e_i that is not f_{i-1}. If f_i is the outer-face then stop. Else the orientation of e_i determines the trapezoidal shape of f_i and hence the unique other diagonal edge e_{i+1} on f_i and its orientation. Repeating the process determines all edges and faces that intersect the slab. □

We use $D(e, \alpha)$ to denote the subgraph formed by the inner faces $f_1, f_2, \ldots$ in the above proof, and equip the edges of $D(e, \alpha)$ with the orientation as they are determined in the process. From the proof of Lemma 9, it follows that we can determine $D(e, \alpha)$ from the planar embedding of G alone, without needing to know an sIUBVR. If there exists some sIUBVR with e in orientation α as topmost edge of a slab, then $D(e, \alpha)$ expresses the part of it within that slab. Furthermore, in this case the edges $C(e, \alpha) := \{e_1, e_2, \ldots\}$ (see proof of Lemma 9) span the slab and $D(e, \alpha)-C(e, \alpha)$ is the union of two induced paths (along the sides of the slab). Let $L(e, \alpha)$ be the path that contains the left end of e and let $R(e, \alpha)$ be the other one. If anything goes wrong while determining $L(e, \alpha)$ and $R(e, \alpha)$ (e.g. if some edge obtains two contradictory directions for $D(e, \alpha)$, or if $D(e, \alpha)-C(e, \alpha)$ is not the union of two paths) then we discard the node $v(e, \alpha)$ since it cannot lead to an sIUBVR.

Now we add an arc $v(e_\ell, \alpha_\ell) \to v(e_r, \alpha_r)$ if (e_ℓ, α_ℓ) is *compatible* with (e_r, α_r). The latter means that $D(e_\ell, \alpha_\ell)$ and $D(e_r, \alpha_r)$ could occur on two consecutive slabs of an sIUBVR of G. It is not hard to test this in linear time: The two partial representations fix all the (downward) directions and orientations of all the involved edges. If this results in contradicting direction for edges, then no such sIUBVR of G can exist. Otherwise we can uniquely determine the relative position of bars for all involved vertices and simply test that these bars created no unwanted lines-of-sights. Finally the vertices in $R(e_\ell, \alpha_\ell) \cup L(e_r, \alpha_r)$ must induce a path in G, else putting the two partial representations would skip some inner faces or represent some vertices twice. See Fig. 8b and the appendix for details.

To finalize H, we add an arc $s \to v(e, \alpha)$ if $D(e, \alpha)$ could be the leftmost slab; this can be read directly from the planar embedding since then $L(e, \alpha)$ must consist of outer-face edges. Similarly we add an arc $v(e, \alpha) \to t$ if $D(e, \alpha)$ could be rightmost slab. This finishes the construction of H.

One can now easily show that G has an sIUBVR D if and only if H has a directed path s to t. Namely, given D we can find the vertices $v(e, \alpha)$ for which e is the topmost edge in some slab and has orientation α and argue that these form a path in H. Vice versa, if there is such a path, then each node $v(e, \alpha)$ on it defines a partial drawing $D(e, \alpha)$, and we can glue these partial drawings together since the arcs ensure compatibility. One can then argue that the result exactly represents G. Details are in the appendix.

Clearly, our approach gives a polynomial-time algorithm to test whether a 2-connected graph has an sIUBVR. We argue in the appendix how with a bit more care we can achieve a run-time of $O(n^2)$ for 2-connected graphs. We also discuss how to handle cutvertices in the appendix by splitting the graph into its 2-connected components. Overall, we achieve:

Theorem 10 *Let G be a plane graph with n vertices. Then we can test in $O(n^2)$ time whether G has an sIUBVR.*

Uniqueness? Our testing algorithm relies strongly on the fact that once the topmost edge and its orientation are determined, the representation within one slab is unique (up to moving bars up or down). Once one such slab is fixed, often the adjacent slab is fixed as well. In light of this, it may come as a surprise that an sIUBVR is not always unique. Indeed, we can construct an example where for one slab we have fixed the topmost edge and its orientation, and still at an adjacent slab we can have choices as to which edges and faces cross the slab. See Fig. 8d.

6 Conclusion and open problems

In this paper, we studied IUBVRs. We showed that recognizing whether a graph has a weak IUBVR is NP-hard, but in contrast testing whether it has a strong one is polynomial. We also showed that trees and 2-connected outer-planar graphs with maximum degree 4 have a weak IUBVR. We leave some open problems:

- In macramé, it is possible to knot more than two strings together, but typically no more than a small constant. What graphs are possible if vertex bars have the form $[i_v - k, i_v] \times y$ for, say, $k \leq 4$?
- For some graphs, the existence of an sIUBVR depends on the embedding, e.g. see Fig. 9. Can we test whether a planar graph (without fixed embedding) has an sIUBVR?

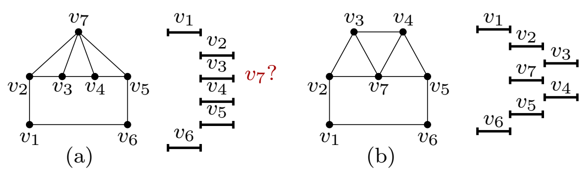


Figure 9: (a) Embedding for G that has no sIUBVR (b) A different embedding of the same G and its sIUBVR

References

[1] T. Andreae. Some results on visibility graphs. *Discrete Applied Mathematics*, 40(1):5–17, 1992.

[2] P. Bose, A. Dean, J. Hutchinson, and T. Shermer. On rectangle visibility graphs. In *International Symposium on Graph Drawing*, pages 25–44. Springer, 1996.

[3] A. M. Dean and N. Veytsel. Unit bar-visibility graphs. *Congressus Numerantium*, 160:161–176, 2003.

[4] P. Duchet, Y. Hamidoune, M. Las Vergnas, and H. Meyniel. Representing a planar graph by vertical lines joining different levels. *Discrete Mathematics*, 46(3):319–321, 1983.

[5] V. Dujmovic, M. Fellows, M. Kitching, G. Liotta, C. McCartin, N. Nishimura, P. Ragde, F. Rosamond, S. Whitesides, and D. Wood. On the parameterized complexity of layered graph drawing. *Algorithmica*, 52:267–292, 2008.

[6] L. S. Heath and A. L. Rosenberg. Laying out graphs using queues. *SIAM J. Comput.*, 21(5):927–958, 1992.

[7] L. A. Melnikov. Problem at the sixth Hungarian Colloquium on Combinatorics, 1981.

[8] R. H. J. M. Otten and J. van Wijk. Graph representations in interactive layout design. In *IEEE Internat. Symp. on Circuits and Systems, 1978*, pages 914–918, 1978.

[9] P. Rosenstiehl and R. E. Tarjan. Rectilinear planar layouts and bipolar orientations of planar graphs. *Discrete & Computational Geometry*, 1(4):343–353, 1986.

[10] M. Suderman. Pathwidth and layered drawings of trees. *International Journal of Computational Geometry & Applications*, 14(3):203–225, 2004.

[11] R. Tamassia and I. G. Tollis. A unified approach to visibility representations of planar graphs. *Discrete & Computational Geometry*, 1(4):321–341, 1986.

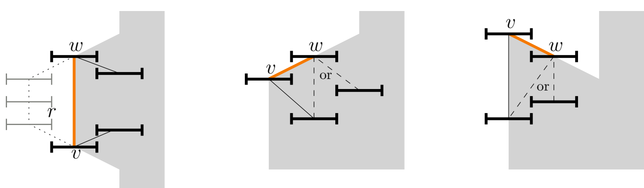
[12] C. Thomassen. Plane representations of graphs. *Progress in graph theory*, pages 43–69, 1984.

[13] L. W. Wiglesworth. *A study of unit bar-visibility graphs*. PhD thesis, University of Louisville, 2008.

[14] S. K. Wismath. Characterizing bar line-of-sight graphs. In *Symposium on Computational Geometry*, pages 147–152. ACM, 1985.

Appendix A wIUBVR for outerplanar graphs

We now explain how to create a wIUBVR of a 2-connected outerplanar graph with maximum degree 4. As before, we root the weak dual G^* (which is a tree) at a leaf r^*. For any face $f^* \neq r^*$ corresponding to face f, the *parent-face* is the face p corresponding to the parent p^* of f^*. Let (v, w) be the edge that f shares with its parent-face; it will be convenient to direct (v, w) so that f is to its right. The *subgraph* $G_{v,w}$ *attached at* (v, w) is the graph formed by the faces in the subtree rooted at f^*. We create up to 5 possible drawings of $G_{v,w}$, where the β-drawing, for $\beta \in \{\mathrm{NW}, \mathrm{N}, \mathrm{NE}, \mathrm{SE}, \mathrm{SW}\}$, satisfies the following (see also Fig. 10):

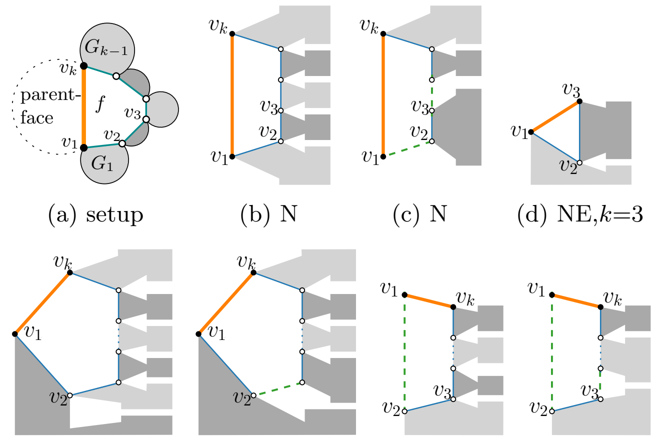


(a) N (b) NE (c) SE

Figure 10: Drawing types. The bold orange edge is the edge shared with the parent-face, and the entire drawing must reside within the gray region (extended infinitely rightward). (a) also illustrates how to add root r at the end.

- (v, w) is the β-edge at v.
- One of v, w is in layer 1, i.e., the leftmost layer.
- If $\beta = \mathrm{N}$, then v has a NE-edge, w has a SE-edge, and layer 1 contains no other vertices.
- If $\beta = \mathrm{NE}$, then v has a SE-edge and w has a S-edge or a SE-edge (or both). Layers 1 and 2 are empty above v, w.
- If $\beta = \mathrm{SE}$, then w has a S-edge or a SW-edge or both, and it has no NE-edge. Layers 1 and 2 are empty above v, w. (This in particular implies that SE-drawings can exist only if $\deg(v) = 2$; we will ensure that this holds.)
- $\beta = \mathrm{NW}$ is symmetric with $\beta = \mathrm{NE}$ and $\beta = \mathrm{SW}$ is symmetric with $\beta = \mathrm{SE}$; flip the corresponding drawings in Fig. 10 upside-down.

We create such drawings by going bottom-up in tree G^*. So let $f^* \neq r^*$ be a node of G^*, corresponding to face f. Enumerate the vertices of f as $v_1, \ldots, v_k$ in counter-clockwise order such that $v_1 \to v_k$ is the edge that f shares with its parent-face. We want to draw the subgraph G_{v_1,v_k} attached at (v_1, v_k). For $s = 1, \ldots, k-1$, denote by G_s the subgraph $G_{v_s,v_{s+1}}$ attached at (v_s, v_{s+1}) (it is empty if (v_s, v_{s+1}) is on the outer-face). Since v_1 has degree at most 4 and it is adjacent to $v_k \notin G_1$ and has one further neighbour in the parent-face of f, we can conclude that v_1 has degree at most 2 in G_1.



(a) setup (b) N (c) N (d) NE,k=3

(e) NE,k>3 (f) NE,k>3 (g) SE (h) SE

Figure 11: Drawing face f and merging subgraphs. We only show the corresponding layered drawing. Dashed green edges are outer-face edges. We show some of the cases where G_s is empty and therefore more ports are available for G_{s-1} and G_{s+1}.

We explain how to create a β-drawing of G_{v_1,v_k} only for $\beta = \mathrm{N}, \mathrm{NE}, \mathrm{SE}$, the cases $\beta = \mathrm{NW}, \mathrm{SW}$ are symmetric (flip the drawings for NE and SE upside-down); see Fig. 11.

Case 1: $\beta = \mathrm{N}$. We draw f as a trapezoid with (v_1, v_k) as long vertical edge in layer 1. Directing the other edges as $v_1 \to v_2 \to \cdots \to v_k$, their orientations (in order) are $\mathrm{NE}, \mathrm{N}, \ldots, \mathrm{N}, \mathrm{NW}$. With this we have the required NE-edge for v_1 and SE-edge for v_k.

For each $1 \leq s \leq k-1$, recursively find the β'-drawing of G_s where β' is the orientation that we just assigned to $v_s \to v_{s+1}$. The goal is to merge these drawings; for this we need to argue that no port at a vertex v_s is used repeatedly (it could be used by f or by G_{s-1} or by G_s).

One easily verifies that there is no conflict between f and G_s due to the restrictions on the drawing type for G_s; for example; if G_s is a N-drawing, then its leftmost layer contains only (v_s, v_{s+1}) and so it uses no port at v_s that was used by f. But it is less obvious that G_{s-1} and G_s, presuming they are both non-empty, could not both use a port of v_s. Recall that G_{s-1} uses a NE-drawing or a N-drawing, so v_s has a S-edge or a SE-edge in G_{s-1} that is not (v_{s-1}, v_s). Likewise G_s uses a NW-drawing or a N-drawing, and v_s has a N-edge or a NE-edge in G_s that is not (v_s, v_{s+1}). This gives four distinct edges at v_s, so by $\deg(v_s) \leq 4$ there are no others. So all edges of v_s in G_{s-1} go southward while all its edges in G_s go northward and there is no conflict among the ports.

With this, we can merge the drawings of the attached subgraphs into the drawing of the face. Because no ports at any v_s can be used by two subgraphs, this does not lead to overlap as long as we compact the drawing of G_s vertically so that it occupies only minimal space below v_s and above v_{s+1}. Also, if G_1 is non-empty then it uses a NE-drawing and so v_1 has a SE-edge other than (v_1, v_2). Since v_1 has at no other neighbours in G_1 as argued earlier, it has no S-edge and layer 1 is empty below v_1. Similarly layer 1 is empty above v_2. So we have created the desired N-drawing of G_{v_1,v_k}.

Case 2: β = NE and $k = 3$. We draw f as a triangle with $v_1 \to v_3$ as NE-edge of v_1; thus $v_1 \to v_2$ is a SE-edge and $v_2 \to v_3$ is a N-edge, giving the edges required for a NE-drawing. As before, for each attached subgraph find the drawing that respects these orientations (this is feasible for G_1 since v_1 has at most two neighbours in G_1). If both G_1 and G_2 are non-empty, then v_2 has a NE-edge in G_2 and a S-edge or SW-edge in G_1, and so there is no port-conflict at v_2. Therefore, we can merge the drawings of the sub-graphs. One easily verifies that layer 1 and 2 remain empty above v_1 and v_k, so we obtain a NE-drawing.

Case 3: β = NE and $k > 3$. We draw f as a pentagon with $v_1 \to v_3$ as NE-edge of v_1; $v_1 \to \cdots \to v_k$ receive orientations (in order) SE, NE, N, ..., N, NW. As before, for each attached subgraph G_s find the drawing that respects these orientations. The argument that there is no conflict among ports is the same as for Case 1, except at vertex v_2. Here (presuming both subgraphs G_1 and G_2 are non-empty) v_2 receives a S-edge or SW-edge in G_1 and a SE-edge in G_2, and by maximum degree 4 it has no other edges and there are no port-conflicts. Therefore we can merge the drawings of the sub-graphs. One easily verifies that layer 1 and 2 remain empty above v_1 and v_k, so we obtain a NE-drawing.

Case 4: β = SE. We know that this happens only if $\deg(v_1) = 2$. We draw f as a trapezoid with $v_1 \to v_k$ as SE-edge, so the edges $v_1 \to \cdots \to v_k$ receive orientation S, NE, N, ..., N. Since $\deg(v_1) = 2$ there is no subgraph attached at (v_1, v_2). For $2 \le s < k-1$, find the drawing of G_s that respects the assigned orientations. Verify as for Case 1 that this cannot lead to conflict among the ports. Therefore we can merge the drawings of the subgraphs. One easily verifies that layer 1 and 2 remain empty above v_1 and v_k.

We must argue that v_k has no NE-edge. We know that v_k has at most three neighbours in G_{v_1,v_k} (because it has one more in the parent-face of f). If it has three neighbours, then the neighbour x other than v_1 and v_k must be in subgraph G_{k-1}. But G_{k-1} uses a N-drawing or a NE-drawing; either way it has an edge different from (v_{k-1}, v_k) that is a S-edge or SE-edge of v_k. So (v_k, x) has southerly orientation, as do (v_k, v_1) and (v_k, v_{k-1}). Hence v_k has no NE-edge and we obtain a SE-drawing.

With this, we can draw any subgraph that corresponds to a strict subtree of G^*. To finish off, as before let r^* be the root of G^* and let (v, w) be the unique chord of r (recall that r^* is a leaf). Find a N-drawing of the graph $G_{v,w}$ attached at (v, w), which places (v, w) as vertical edge in the leftmost layer. We can now add r as a trapezoid with (v, w) as long edge on the right and all other vertices in one layer further left. This gives the desired wIUBVR of G.

Appendix B Details of the NP-hardness

We start by describing the components used later to form a vertex gadget. We define a *k-zigzag* to be the graph that consists of a $(2k-1)$-cycle $v_1, \ldots, v_k, u_{k-1}, \ldots, u_1$ with chords (v_i, u_j) for $i = 2, \ldots, k-1$ and $j \in \{i-1, i\}$; see Fig. 12a. We call vertices $v_1, \ldots, v_k$ *squared*, vertices $u_1, \ldots, u_{k-1}$ *circular*, and the vertices v_1 and v_k *end vertices*. Since every interior face of a k-zigzag T is a triangle, choosing the orientation of a single edge of T fixes the orientation of all of its edges. Note that the edges on the path $v_1, \ldots, v_k$ are all drawn with the same orientation β; we say that T *is drawn with orientation* β or that it is a *β-k-zigzag* or just *β-zigzag*.

We construct a *node block* L as follows; see Fig. 12b. Let T be a 3-zigzag, and let T' and T'' be two 4-zigzags. Identify vertex v_4 of T with vertex v_1' of T' and vertex v_5' of T' with vertex v_1'' of T''. Finally, add a path P=(v_3', x, y, v_3''), (called a *fixating path*). Let a be vertex v_1 of T and let b be vertex v_5'' of T''. A node block $L_i(v)$ consists of a 3-zigzag T_i, two 4-zigzags T_i' and T_i'' and a fixating path P_i.

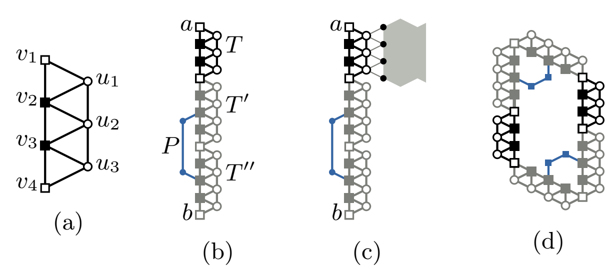


Figure 12: (a) A 3-zigzag. (b) A node block (c) connected to a tube and (d) the node gadget of a degree-2 vertex.

Let $n = |V_G|$ and set $M = 24n$. For any vertex $v \in V_G$ with $\deg(v)$=d, the *node gadget* N_v of v consists of a cycle of d node blocks $L_1(v), \ldots, L_d(v)$ in H obtained by identifying vertex b_i of $L_i(v)$ with vertex a_{i+1} of $L_{i+1}(v)$ for i=$1, \ldots, d$ (where L_{d+1}=L_1). See Fig. 12d. Let $u_1, \ldots, u_d$ be the neighbours of v in G

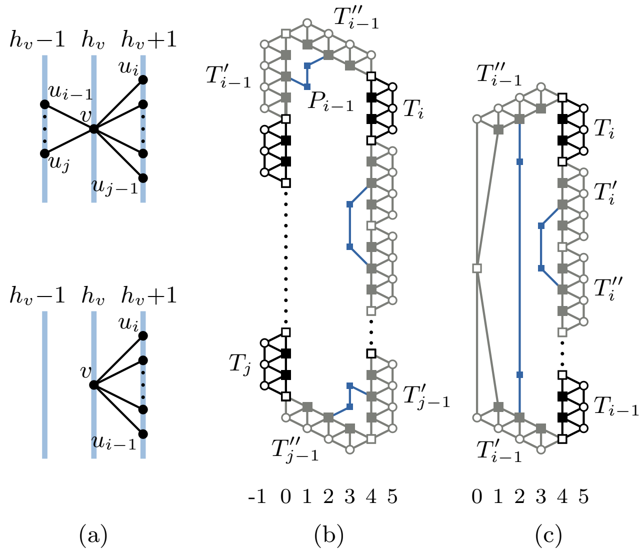


Figure 13: (a) A node v in G; (b)–(c) the node gadget of v in H when v has neighbours on (b) both sides and (c) only to the right. In this and the following figures, we omit the (v) identifier of the zigzags and we label the layers without the addition of $h_v \cdot (2M+8)$ to reduce clutter.

in clockwise order as defined by the embedding. Assign edge (v, u_i) in G to the 3-zigzag $T_i(v)$ in H, $1 \le i \le d$.

Now every edge (v, u) of E_G is assigned to exactly two 3-trapezoids in H, say $T_i(v)$ and $T_j(u)$. Attach an M-tube M_e to $T_i(v)$ and $T_j(u)$ as depicted in Fig. 12c. This completes the construction of H.

From G to H. We now show that we can construct a wIUBVR D of H from a leveled-planar drawing Γ of G. Enumerate the levels of Γ from left to right as $0, 1, \ldots, m$. Let v be a node of G with $\deg(v)=d$ that is drawn in level h_v. We draw N_v in the layers $h_v \cdot (2M+8)-1$ to $h_v \cdot (2M+8)+5$ as follows. Let (v, u_i) be the edge assigned to $T_i(v)$, $1 \le i \le d$.

First, assume that v has at least one neighbour in both level h_v+1 and level h_v-1. Let u_i be the top-most neighbour of v in level h_v+1 and let u_j be the bottom-most neighbour of v in level h_v-1; see Fig. 13a (top). We draw the trapezoids $T_i(v), T'_i(v) \ldots, T_{j-1}(v), T'_{j-1}(v)$ with S-orientation such that all their squared vertices lie in layer $h_v \cdot (2M+8)+4$ and all their circular vertices lie in layer $h_v \cdot (2M+8)+5$; see Fig. 13b. The interior vertices of the fixating paths $P_i, \ldots, P_{j-2}$ are placed one layer left of their endpoints, that is, in the layer $h_v \cdot (2M+8)+3$. Symmetrically, we place the trapezoids $T_j(v), T'_j(v) \ldots, T_{i-1}(v), T'_{i-1}(v)$ with N-orientation such that all their squared vertices lie in layer $h_v \cdot (2M+8)$ and all their circular vertices lie in layer $h_v \cdot (2M+8)-1$. The interior vertices of $P_j, \ldots, P_{i-2}$ are symmetrically placed in layer $h_v \cdot (2M+8)+1$. The trapezoid $T''_{i-1}(v)$ is drawn with SE-orientation in the layers $h_v \cdot (2M+8)$ to $h_v \cdot (2M+8)+4$. By this construction, the endpoints of P_{i-1} lie in the layers $h_v \cdot (2M+8)$ and $h_v \cdot (2M+8)+2$, so we can place its interior vertices in layer $h_v \cdot (2M+8)$. Symmetrically, the trapezoid $T''_{j-1}(v)$ is drawn with NW-orientation in the layers $h_v \cdot (2M+8)$ to $h_v \cdot (2M+8)+4$ and the interior vertices of P_{j-1} are placed in layer $h_v \cdot (2M+8)+3$.

Second, assume that all neighbours of v lie in level h_v+1 as shown in Fig. 13a (bottom). Again let u_i be the top-most neighbour of v. We place the trapezoids $T_i(v), T'_i(v) \ldots, T_{i-1}(v)$ with S-orientation such that all their squared vertices lie in layer $h_v \cdot (2M+8)+4$ all their circular vertices lie in layer $h_v \cdot (2M+8)+5$; see Fig. 13c. As in the previous case, the interior vertices of $P_i, \ldots, P_{i-2}$ are placed in layer $h_v \cdot (2M+8)+3$. The trapezoid $T'_{i-1}(v)$ is placed with NW-orientation and $T''_{i-1}(v)$ is placed with NE-orientation in the layers $h_v \cdot (2M+8)+4$ to $h_v \cdot (2M+8)$ This way, the endpoints of P_{i-1} both lie in layer $h_v \cdot (2M+8)+2$, so we can draw the path vertically in this layer.

Finally, if all neighbours of v lie in level h_v-1, then we place the vertices symmetrical to the previous case by choosing u_i as the bottom-most neighbour of v such that the circular vertices of the trapezoids $T_i(v), T'_i(v) \ldots, T_{i-1}(v)$ lie in layer $h_v \cdot (2M+8)-1$.

By this construction, if any edge between a node v in level h_v and a node u in level h_v+1 is assigned to v_i and u_j in H, then the circular vertices of $T_i(v)$ are drawn in layer $h_v \cdot (2M+8)+5$ and the circular vertices of $T_j(u)$ are drawn in layer $(h_v+1) \cdot (2M+8)-1$, so there are exactly $2M+1$ layers between them that we use to place the M-tube that is connected to $T_i(v)$ and $T_j(u)$. This completes the construction; see Fig. 14 for an example.

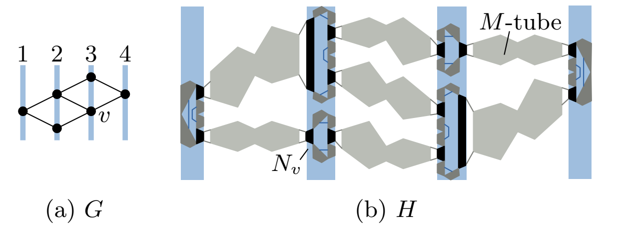


Figure 14: An example of a drawing of H obtained from a drawing of G. The node gadgets lie completely inside the shaded rectangles and the M-tubes lie in between.

From H to G. Next, we show how to construct a leveled-planar drawing Γ of G from a wIUBVR D of H. We enumerate the layers of D from left to right as $0, 1, \ldots, m$. For any vertex $u \in V_H$, let $\ell(u)$ be the layer of u in D.

Let $v \in V_G$ with $\deg(v)=d$ and let $e=(v, u) \in E_G$ be assigned to the trapezoids $T_i(v)$ and $T_j(u)$. Since $T_i(v)$

and $T_j(v)$ are connected to an M-tube M_e, both have to be drawn with S-orientation or with N-orientation. Furthermore, $T_i(v)$ is a S-zigzag if and only if $T_j(u)$ is a N-zigzag. Assume w.l.o.g. that $T_i(v)$ is a S-zigzag. We aim to show that N_u lies completely to the right of N_v, that is, $\ell(u^*) > \ell(v^*)$ for every $u^* \in N_u$ and $v^* \in N_v$. The node gadget N_v consists of d node blocks, so it contains $24d \leq 24n$ vertices. Analogously, N_u consists of at most $24n$ vertices. Hence, both N_v and N_u lie in at most $24n$ consecutive layers. Let $v' \in T_i(v)$ and $u' \in T_j(u)$. Since $T_i(v)$ and $T_j(u)$ are connected to the same M-tube M_e, we have $\ell(u') \geq \ell(v')+2M+2$. Furthermore, for any vertex $v^* \in N_v$ we have $\ell(v^*) \leq \ell(v') + 24n$ and for any vertex $u^* \in N_u$ we have $\ell(u^*) \geq \ell(u') - 24n$. Hence,

$$\begin{aligned}\ell(u^*) &\geq \ell(u') - 24n \geq \ell(v') + 2M + 2 - 24n \\ &\geq \ell(v^*) + 2M + 2 - 48n \\ &= \ell(v^*) + 48n + 2 - 48n > \ell(v^*).\end{aligned}$$

Let $v \in G$ and let $M_i(v)$ be the M-tube attached to the 3-zigzag $T_i(v)$ in N_v, $i = 1, \ldots, d = \deg(v)$. Since D is planar, the M-tubes $M_1(v), \ldots, M_d(v)$ either all leave N_v to the right, all leave N_v to the left, or there is some $1 \leq i, j \leq d$ such that $M_i(v), \ldots, M_{j-1}(v)$ leave N_v to the right and $M_j(v), \ldots, M_{i-1}(v)$ leave N_v to the left. It follows that N_v has one of the following properties.

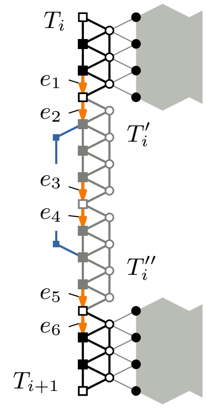
(P1) Every $T_i(v)$, $1 \leq i \leq d$, is a S-zigzag. There is some $1 \leq j \leq d$ such that $M_j(v), \ldots, M_d(v), M_1(v), \ldots, M_{j-1}(v)$ are ordered from top to bottom in this order.

(P2) Every $T_i(v)$, $1 \leq i \leq d$, is a N-zigzag. There is some $1 \leq j \leq d$ such that $M_j(v), \ldots, M_d(v), M_1(v), \ldots, M_{j-1}(v)$ are ordered from bottom to top in this order.

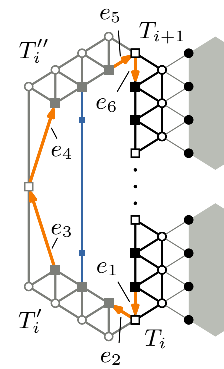
(P3) There is some i, $1 \leq i \leq d$, and some $j \neq i$, $1 \leq j \leq d$, such that $T_i(v), T_{i+1}(v), \ldots, T_{j-1}(v)$ are S-zigzags, $M_i(v), M_{i+1}(v), \ldots, M_{j-1}(v)$ are ordered from top to bottom in this order, $T_j(v), T_{j+1}(v), \ldots, T_{i-1}(v)$ are N-zigzags, and $M_j(v), M_{j+1}(v), \ldots, M_{i-1}(v)$ are ordered from bottom to top in this order.

Lemma 11 *Let N_v be a node gadget. The circular vertices of all S-3-zigzags of N_v lie in the same layer $\overrightarrow{\ell_v}$ and the circular vertices of all N-3-zigzags of N_v lie in the same layer $\overleftarrow{\ell_v}$ with $\Delta\ell{=}\overrightarrow{\ell_v}{-}\overleftarrow{\ell_v}{=}6$. Furthermore, the drawing of N_v spans at least 6 layers between $\overleftarrow{\ell_v}$ and $\overrightarrow{\ell_v}$.*

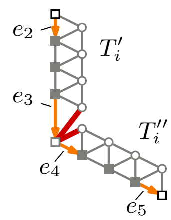
Proof. Consider first the case that D is drawn with property (P1), that is, Every $T_i(v)$, $1 \leq i \leq d$, is a S-zigzag and there is some $1 \leq j \leq d$ such that $M_j(v), \ldots, M_d(v), M_1(v), \ldots, M_{j-1}(v)$ are ordered from top to bottom in this order. Let $1 \leq i \leq n, i \neq j{-}1$. Then, T_i and T_{i+1} are S-zigzags and $M_{i+1}(v)$ lies below $M_i(v)$. We will argue that $T_i'(v)$ and $T_i''(v)$ are also S-zigzags. Intuitively, between two zigzags in counterclockwise order, there can never be a "rightwards bend" because of the order of the edges around each squared vertex, as otherwise at least one port has to be used twice.



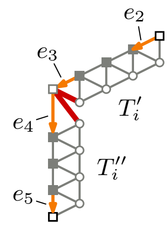
(a) $T_i'(v)$ and $T_i''(v)$ are S-zigzags



(b) $T_i'(v)$ is a NW-zigzag and $T_i''(v)$ is a NE-zigzag



(c) $T_i'(v)$ is a S-zigzag and $T_i''(v)$ is a SE-zigzags

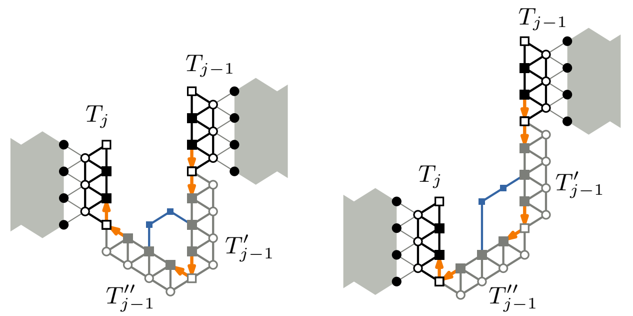


(d) $T_i'(v)$ is a SW-zigzag and $T_i''(v)$ is a S-zigzag

Figure 15: $T_i(v)$ and $T_{i+1}(v)$ are S-zigzags.

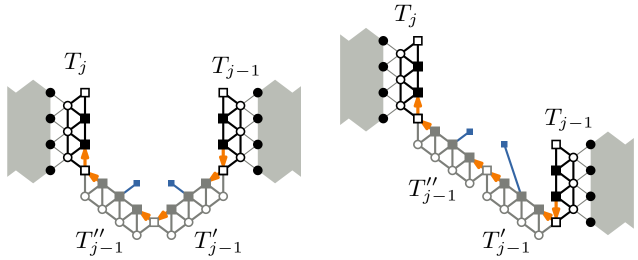
Consider the (directed) edges $e_1, \ldots, e_6$ in Fig. 15. By construction, e_1 and e_6 are S-edges. Since there are two edges between e_1 and e_2 around their common vertex in clockwise order, e_2 and thus e_3 has to be a S-, SW-, or NW-edge. Symmetrically, e_5 and thus e_4 has to be a S-, SE-, or NE-edge. However, since there are two edges between e_3 and e_4 around their common vertex in clockwise order, there are only two compatible configurations: either both are a S-edge (see Fig. 15a), or e_3 is a NW-edge and e_4 is a NE-edge (see Fig. 15b); otherwise, at least one port between e_3 and e_4 has to be used twice (see Figures 15c and 15d. However, the second case is a contradiction to the face that $M_i(v)$ lies completely above $M_{i+1}(v)$. Hence, $e_1, \ldots, e_6$ are S-edges and thus $T_i(v)$, $T_i'(v)$, $T_i''(v)$, and T_{i+1} are S-zigzags.

Now, let $i{=}j{-}1$. In this case, the argument is exactly the same; with the only difference that now $M_i(v){=}M_{j-1}(v)$ is the bottom-most one M-tube, so



(a) $T'_{j-1}(v)$ is a S-zigzag and $T''_{j-1}(v)$ a NW-zigzag

(b) $T'_{j-1}(v)$ a S-zigzag and $T''_{j-1}(v)$ is a SW-zigzag



(c) $T'_{j-1}(v)$ is a SW-zigzag and $T''_{j-1}(v)$ is a NW-zigzag

(d) $T'_{j-1}(v)$ and $T''_{j-1}(v)$ are NW-zigzags

Figure 16: $T_{j-1}(v)$ is a S-zigzag and $T_j(v)$ is a N-zigzag.

$M_{i+1}(v){=}M_j(v)$ lies above it. Hence, the case that e_3 and e_4 are S-edges is a contradiction to this property. Thus, $T'_i(v)$ is a NW-zigzag and $T''_i(v)$ is a NE-zigzag; see Fig. 15b. Hence, N_v spans exactly the 6 layers between $\overrightarrow{\ell_v} - 5$ and $\overrightarrow{\ell_v}$.

This shows that $T_j(v), T'_j(v), T''_j(v), \ldots, T_{j-1}(v)$ are all S-zigzags. Hence, the circular vertices of the 3-zigzags $T_1(v), \ldots, T_d(v)$, which are all S-3-zigzags, all lie in the same layer $\overrightarrow{\ell_v}$.

The case that D is drawn with property (P2) is completely symmetric.

Consider now the case that D is drawn with property (P3), so there is some $1 \leq i, j \leq d$, $i \neq j$, such that $T_i(v), \ldots, T_{j-1}(v)$ are S-zigzags and $T_j(v), \ldots, T_{i-1}(v)$ are N-zigzags. With the same argument as above, we show that the circular vertices of $T_i(v), \ldots, T_{j-1}(v)$ all lie in the same layer $\overrightarrow{\ell_v}$ and that the circular vertices of $T_j(v), \ldots, T_{i-1}(v)$ all lie in the same layer $\overleftarrow{\ell_v}$. It remains to show that $\Delta\ell{=}6$.

Consider the S-zigzag $T_{j-1}(v)$, the N-zigzag $T_j(v)$, and the (directed) edges $e_1, \ldots, e_6$ in Fig. 16. Following the same arguments as above, $T'_{j-1}(v)$ has to be a S-, SW-, or NW-zigzag and $T''_{j-1}(v)$ has to be a N-, SW-, or NW-zigzag. However, the fixating path $P_{j-1}(v)$ forces either $T'_{j-1}(v)$ to be a S-zigzag or $T''_{j-1}(v)$ to be a N-zigzag; see Fig. 16a and Fig. 16b for the former case, the latter case is symmetric. In particular, assume that $T'_{j-1}(v)$ and $T''_{j-1}(v)$ are both drawn SW- or NW-zigzags; see Fig. 16c and Fig. 16d, the remaining cases are symmetric. Then, there are three layers between the endpoints of $P_{j-1}(v)$, but $P_{j-1}(v)$ only has two interior vertices. Hence, it is impossible to draw $P_{j-1}(v)$. Thus, $T'_{j-1}(v)$ and $T''_{j-1}(v)$ have to be drawn as in Fig. 16a and Fig. 16b (up to symmetry) and one can easily see that this implies $\Delta\ell{=}6$ and N_v spans exactly the 7 layers between $\overleftarrow{\ell_v}$ and $\overrightarrow{\ell_v}$. □

We now show that every node gadget N_v lies completely inside the layers $h_v{\cdot}(2M{+}8){-}1$ to $h_v{\cdot}(2M{+}8){+}5$ for some $0 \leq h_v \leq n$.

First, we show that all vertices in layer 0 belong to a node gadget. Assume to the contrary that there is some vertex u in layer 0 that belongs to an M-tube. By construction, every M-tube is connected to a vertex of a node gadget on both sides; hence, there has to be at least one vertex that completely lies to the left of the M-tube. This is a contradiction to layer 0 being the leftmost layer that contains a vertex.

Let $s \in V_G$ be a node such that some vertex of N_s lies on layer 0 in D. For any node $v \in V_G$, let d_v be the length of the shortest path between s and v in G. We now analyze the layers that contain the vertices of node gadgets.

Lemma 12 *For any node $v \in V_G$, $\overleftarrow{\ell_v}{=}h_v \cdot (2M+8)-1$ and $\overrightarrow{\ell_v}{=}h_v \cdot (2M+8)+5$ for some $0 \leq h_v \leq d_v$.*

Proof. We prove the lemma by induction over d_v.

If $d_v{=}0$, then $v{=}s$. Let $h_s{=}0$. By choice of s, the leftmost vertex of N_s lies in layer $0 > 0 \cdot (2M+8) - 1$. Since there are no node gadgets that lie to the left of N_s, it is drawn as depicted in Fig. 13c. Hence, the N_s is drawn with property (P1) and by Lemma 11 all circular vertices of its S-3-zigzags lie in layer $5{=}h_v{\cdot}(2M{+}8){+}5$.

Now, assume that the lemma holds for all vertices $w \in V_G$ with $d_w \leq k \geq 0$. Let $v \in V_G$ with $d_v{=}k{+}1$ and let (u, v) be the last edge on the shortest path from s to v in G. Let u_i and v_j be the vertices the edge (u, v) is assigned to.

Assume first that $T_i(u)$ is a S-3-zigzag and $T_j(v)$ is a N-3-zigzag. By induction, u_i lies in a layer $\overrightarrow{\ell_u}{=}h_u \cdot (2M{+}8){+}5$ for some $0 \leq h \leq d_u$. The vertices u_i and v_j are connected to the same M-tube. Since an M-tube spans exactly $2M{+}1$ layers, it follows that $\ell(v_j){=}\overrightarrow{\ell_u} + (2M{+}2){=}h_v \cdot (2M+8) - 1$ for $h_v{=}h_u{+}1$. Hence, by Lemma 11, $\overleftarrow{\ell_v}{=}h_v{\cdot}(2M{+}8){-}1$ and $\overrightarrow{\ell_v}{=}h_v{\cdot}(2M{+}8){+}5$.

If that $T_i(u)$ is a N-3-zigzag and $T_j(v)$ is a S-3-zigzag, then analogously $\ell(v_j){=}\overleftarrow{\ell_u}{-}(2M+2){=}h_v \cdot (2M+8)+5$ for $h_v{=}h_u{+}1$ and thus $\overleftarrow{\ell_v}{=}h_v \cdot (2M+8)-1$ and $\overrightarrow{\ell_v}{=}h_v \cdot (2M+8)+5$. □

We are now ready to create the drawing Γ of G. We draw every node $v \in V_G$ in level h_v in Γ. Let $v_1, \ldots, v_k$

be the vertices in level h, $0 \le h \le n$. By Lemma 11 and 12, each node gadget $N_{v_1}, \dots, N_{v_k}$ contains at least one vertex in layer $h_v \cdot (2M+8)$. Since each of these node gadgets is connected to at least one M-tube, there is some order, say $v_1, \dots, v_k$, such that the vertices in layer belong to $N_{v_1}, \dots, N_{v_k}$ from bottom to top. We draw each node v_i, $1 \le i \le k$ at coordinate (h, i). Since D is planar, the obtained drawing Γ of G is also planar. Let $(v, u) \in E_G$ be assigned to vertices v_i of N_v and u_j of N_v. Assume w.l.o.g. that $T_i(v)$ is a S-zigzag and $T_j(u)$ is a N-zigzag. Since v_i and u_j are connected to a common M-tube, we have $\ell(v_i) = \overrightarrow{\ell_u} = h_u \cdot (2M+8) + 5$ and

$$\begin{aligned} h_u \cdot (2M+8) - 1 = \overleftarrow{\ell_u} &= \ell(u_j) \\ &= h_v \cdot (2M+8) + 5 + 2M + 2 \\ &= (h_v{+}1) \cdot (2M+8) - 1, \end{aligned}$$

so $h_u = h_v{+}1$. Hence, v is placed in level h_v and u is placed in level $h_v{+}1$, so Γ is a leveled-planar drawing.

This completes the proof of Theorem 5.

Appendix C Details of testing for an sIUBVR

C.1 Compatibility of configurations

We first explain in more detail how to test whether two configurations (e_ℓ, α_ℓ) and (e_r, α_r) are compatible, i.e., what properties must hold if some sIUBVR has $D(e_\ell, \alpha_\ell)$ and $D(e_r, \alpha_r)$ in adjacent slabs, say in slabs $i{-}1$ and i corresponding to layers $i{-}1, i$ and $i{+}1$.

In the following, we consider paths $R(e_\ell, \alpha)$ and $L(e_r, \alpha_r)$ to be directed downward, i.e., from the end in e_ℓ / e_r to the other end. We claim that the following conditions are necessary for compatibility:

- The vertices U of $R(e_\ell, \alpha_\ell) \cup L(e_r, \alpha_r)$ form an induced path P of G that can be directed such that it is consistent with the directions of $R(e_\ell, \alpha_\ell)$ and $L(e_r, \alpha_r)$. This holds because the union of the two paths is drawn on level i as a vertical path.
- In particular, the vertices I in $R(e_\ell, \alpha_\ell) \cap L(e_r, \alpha_r)$ must form a subpath of P and the edges among them must be directed the same in $R(e_\ell, \alpha_\ell)$ and $L(e_r, \alpha_r)$. Also, the vertices in $U - I = R(e_\ell, \alpha_\ell) \oplus L(e_r, \alpha_r)$ must occur at the beginning or end of P.
- If α_ℓ=SE, then $R(e_\ell, \alpha_\ell)$ is a prefix of P, i.e., no vertex of P comes before the first vertex of $R(e_\ell, \alpha_\ell)$. This holds because the left end v_ℓ of e_ℓ is above the right end w_ℓ, and w_ℓ is the topmost vertex of $R(e_\ell, \alpha_\ell)$. If any vertices of $L(e_r, \alpha_r)$ were above w_ℓ, then there would be an edge from them to v_ℓ (or higher up), and hence e_ℓ would not be on the outer-face.
- Similarly we have three more requirements at the bottom/top ends to avoid unwanted edges.
  - if α_r=NE, then no vertex in P comes before $L(e_r, \alpha_r)$.
  - Let $\overline{e_\ell}$ be the bottommost diagonal edge of $D(e_\ell, \alpha_\ell)$. If its direction (in $D(e_\ell, \alpha_\ell)$) is NE, then no vertex in P comes after $R(e_\ell, \alpha_\ell)$.
  - Let $\overline{e_r}$ be the bottommost diagonal edge of $D(e_r, \alpha_r)$. If its direction (in $D(e_\ell, \alpha_\ell)$) is SE, then no vertex in P comes after $R(e_r, \alpha_r)$.

It is not hard to see that these conditions are also sufficient. If they are satisfied, then draw P, in order, in level i from top to bottom. In level $i{-}1$ place the vertices of $L(e_\ell, \alpha_\ell)$ so that their position relative to the vertices in $R(e_\ell, \alpha_\ell)$ is the same as it was in $D(e_\ell, \alpha_\ell)$. Observe that this results in exactly the same set of edges as the set that spans slab $i{-}1$. Additional edges could only come from vertices above/below $R(e_\ell, \alpha_\ell)$ in P but such vertices either require e_ℓ to be directed NE or $\overline{e_\ell}$ to be directed SE, neither of which results in a line-of-sight. Similarly, we place the vertices of $R(e_r, \alpha_r)$ in level $i{+}1$ to obtain the desired representation of the two slabs.

For later use, we note one more property:

Observation 2 *Assume that (e_ℓ, α_ℓ) is compatible with (e_r, α_r). Then the edges in $R(e_\ell, \alpha_\ell) \oplus L(e_r, \alpha_r)$ are on the outer-face.*

Proof. Consider just the edges of $R(e_\ell, \alpha_\ell) - L(e_r, \alpha_r)$, the others are symmetric. Continuing in the notations introduced above, we saw that all these edges are drawn vertically in layer i, either above or below $D(e_r, \alpha_r)$. Say they are above. Then no diagonal edge that spans slab i is above them, thus they can reach the outer-face through slab i. □

C.2 Correctness of the construction

Recall that we combined all $D(e_i, \alpha_i)$ for some path $s \to v(e_1, \alpha_1) \to \cdots \to v(e_k, \alpha_k) \to t$. Let D be the sIUBVR that is induced by the resulting bars. For $i{=}2, \dots, k$, set $I_i := R(e_{i-1}, \alpha_{i-1}) \cap L(e_i, \alpha_i)$. By Observation 2, I_i is a path that connects two outer-face vertices. We can therefore split the graph G into subgraphs by splitting at all paths $I_2, \dots, I_k$. More precisely, using $I_1 = L(e_1, \alpha_1)$ and $I_{k+1} = R(e_k, \alpha_k)$, we set G_i to be the graph formed by all faces that can reach the inner face at e_i along a path of inner faces without crossing I_i or I_{i+1}.

It is now straightforward to show by induction that the first i slabs of D (i.e., what we obtain when putting together $D(e_1, \alpha_1) \cup \cdots \cup D(e_i, \alpha_i)$) is an sIUBVR of $G_1 \cup \cdots \cup G_i$. This is straightforward for $i{=}1$ since $D(e_1, \alpha_1)$ represents exactly G_1. When adding in $D(e_{i+1}, \alpha_{i+1})$, we add exactly the faces of G_{i+1} since $D(e_{i+1}, \alpha_{i+1})$ covers them, and we do not add extra edges since the compatibility-condition ensures that vertices in $L(e_{i+1}, \alpha_{i+1})$ (if any) that are added in layer $i+1$ do not add edges to layer i.

C.3 Run-time

We now turn towards the time-complexity of testing whether a 2-connected plane graph G has a sIUBVR. There are $O(n)$ edges on the outer-face of G, hence H has $O(n)$ vertices. As we will argue below, it also has $O(n)$ arcs. Computing the directed path in H (if any) hence takes $O(n)$ time, and we can extract the sIUBVR from it in $O(n)$ time as well.

However, computing the arcs of H is non-trivial and takes cubic time if done in a straightforward way, and quadratic time if we are careful. As a first step, compute sets $L(e, \alpha)$ and $R(e, \alpha)$ for all configurations (e, α) where e is on the outer-face and $\alpha \in \{\mathrm{NE}, \mathrm{SE}\}$; this can be done in $O(n)$ time per configuration and hence overall quadratic time. While doing this we can easily check whether (e, α) is compatible with the left/right boundary and hence find all arcs incident to s and t.

The remaining arcs all connect $v(e, \alpha)$ to $v(e', \beta)$ for some outer-face edges e, e' and directions $\alpha, \beta \in \{\mathrm{NE}, \mathrm{SE}\}$. To find such an arc, we do four tests for each configuration (e, α):

- Walk clockwise along the outer-face starting at e until you encounter the first edge e' that does not belong to $R(e, \alpha)$. Test for both β=NE and β=SE whether (e, α) is compatible with (e', β).

- Walk counter-clockwise along the outer-face starting at e until you encounter the first edge e'' that does not belong to $L(e, \alpha)$. Test for both γ=NE and γ=SE whether (e'', γ) is compatible with (e, α).

To see that this suffices, observe that if (e, α) it compatible with (e', β), then the clockwise path Q from e to e' on the outer-face belongs to either $R(e, \alpha)$ or $L(e', \beta)$. (This holds by Observation 2: The edges in $R(e, \alpha) \oplus L(e', \beta)$ are on the outer-face, and because G is 2-connected, path Q cannot include any edges not in them.) If Q belongs to $R(e, \alpha)$, then our first test will use exactly this e' and hence detect the compatibility. If Q belongs to $L(e, \alpha)$, then at the time of performing the test for configuration (e', β), the second test will use e as e'' and hence detect compatibility. So this determines all arcs of H as needed, and there are $O(n)$ of them. Notice that one test of compatibility can be done in $O(n)$, and so the overall run-time is quadratic.

C.4 Dealing with cutvertices

So far, we assumed that G is a 2-connected plane graph. If G has a cutvertex, then we will argue that we can process each 2-connected component (*blocks*) separately. For this, we need to argue some restrictions on the structure near cutvertices. We assume in the following that G is not a path, else it trivially has an sIUBVR. We need two definitions. First, recall that the edges at a vertex can be classified as NE-edge etc. by their relative directions; we say that two edges incident to a vertex v *use consecutive ports* if their directions are consecutive in the cyclic order $\{\mathrm{N}, \mathrm{NE}, \mathrm{SE}, \mathrm{S}, \mathrm{SW}, \mathrm{NW}\}$. Second, define an *subdivided leg* of graph G to be a maximal induced path for which one end has degree 1 in G. The other end (which necessarily has degree at least 3 in G) is called the *attachment point* of the subdivided leg.

Lemma 13 *Let G be a plane graph that has an sIUBVR D. Let v be a cutvertex of G that is in layer i.*

1. *v is on the outer-face of G.*
2. *If w_1, w_2 are two vertices in the same layer but in different cut-components of v, then they are both in layer i with v between them.*
3. *If e_1, e_2 are two incident edges of v in different cut-components of v, then they do not use consecutive ports at v.*
4. *v has at most three cut-components.*
5. *If v has exactly three cut-components, then one of them is a subdivided leg.*
6. *For any block B containing v the IUBVR D_B of B induced by D is a strong IUBVR and contains v as topmost or bottommost vertex in layer i.*

Proof. 1. Recall that an inner face forms a trapezoid, hence is drawn convex in the associated layered drawing. But at least one face at v contains v repeatedly and so cannot be convex. So v must be adjacent to the outer-face.

2. The vertices within one layer induce a path. So if w_1, w_2 are in the same layer, they are connected by the path of the vertices that are between them on the layer. Any such path must contain v since w_1, w_2 are in different cut-components.

3. Inspection of all cases shows that if two such edges use consecutive ports, then their other endpoints are connected by an edge, contradicting that they are in different cut-components.

4. v has only 6 ports, and we must skip one port whenever we switch from one cut-component to the next in the order of edges around v.

5. The three cut-components must use 3 ports at v without using consecutive ones; up to symmetry these are the N-,SE-, and SW-port. Then the cut-component that uses the N-port must entirely lie within layer i to avoid having an edge to the other two components. So it forms an induced path and its topmost vertex has degree 1, hence it is a subdivided leg.

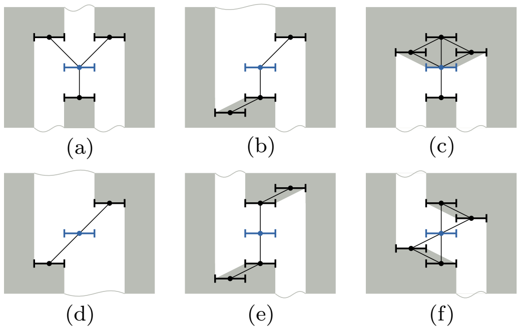


Figure 17: Possible cutvertex configurations

6. Let x, y be two vertices in B and assume that they have a line-of-sight in D_B, but not in D. The vertex z that blocked the line-of-sight in D must share a level with at least one of them, say z and x are in the same level. Since z is in a different 2-connected component, this must be level i and vertex v must be between z and x. But then vertex v would block the line-of-sight in D_B. So D_B is a strong IUBVR. If there were vertices of B both above and below v in layer i, then the N-edge and S-edge at v belong to B. All other edges at v hence would then use consecutive ports with an edge in B, contradicting that v is a cutvertex.

□

We assume in the following that G satisfies conditions (1),(4) and (5), i.e., the conditions that do not depend on the choice of the sIUBVR.

Lemma 14 *Let G be a plane graph that has an sIUBVR. Let G' be the graph obtained from G by removing all subdivided legs. Then the blocktree of G' is a simple path.*

Furthermore, the blocktree can be enumerated as $B_0 - w_1 - B_1 - \cdots - B_{k-1} - w_k - B_k$ for blocks $B_0, \ldots, B_k$ and cutvertices $w_1, \ldots, w_k$ such that $\ell(w_1) \leq \cdots \leq \ell(w_k)$ (where $\ell(w_i)$ denotes the layer of w_i), and all vertices of B_i lie within $[\ell(w_i), \ell(w_{i+1})]$ (where $\ell(w_0) := 1$ and $\ell(w_{k+1}) := \infty$).

Proof. Call a cutvertex of G *non-trivial* if it has at least two cut-components that contain cycles; these are the same as the cutvertices of G'. Notice that a non-trivial cutvertex has exactly two cut-components with cycles (which correspond to cut-components of G') by Lemma 13.

Fix a non-trivial cutvertex v and the two cut-components C_1, C_2 of v that have cycles. We claim that, up to renaming, all vertices in C_1 must be in the level of v and farther left while all vertices in C_2 must be in the level of v or farther right. For otherwise, since both cut-components have cycles, they would both use the same adjacent layer of v, leading to an edge from $C_1 - v$ to $C_2 - v$, a contradiction. We say that v *separates* its cut-components that have cycles.

Now consider a block B that is not a bridge and hence has cycles and occupies at least two layers. Assume two non-trivial cutvertices v_1, v_2 belong to B and are in the same layer. Since v_1 separates B from some other cut-component at v_1, it must be in the leftmost or rightmost layer of B. Say v_1 and v_2 are in the leftmost layer of B. Then the other cut-components C_1 and C_2 at v_1 and v_2 must be to the left of v_1 and v_2. We may choose C_1 and C_2 to contain cycles, so they must use layers strictly to the left of v_1 and v_2. But then there is an edge from $C_1 - v_1$ to $C_2 - v_2$, an impossibility. So for any non-bridge block B, no two non-trivial cutvertices can be in the same layer, and if there are two, they must be in the leftmost and rightmost layer of B. We say that B is *between* its non-trivial cutvertices. In particular, this implies that any non-bridge block has at most two non-trivial cutvertices. Putting things together, therefore every block and cutvertex of G' has at most two incident cutvertices/blocks, which means that the blocktree of G' is as desired.

The second claim follows almost immediately. Let ℓ be the leftmost and rightmost level that contain vertices of G'. If all of G' is drawn within layer ℓ, then the second claim holds trivially. So assume G' uses some layers further right, and let B be a block of G' that spans slab ℓ. Observe that there cannot be two such blocks B, B', because otherwise we could find a cycle that contains edges of both blocks by using the paths within layers ℓ and $\ell{+}1$ and the diagonal edges in the two blocks that span the slab. Observe further that B cannot have a non-trivial cutvertex in ℓ. For otherwise both of its incident cut-components in G' would have to use layer $\ell{+}1$, leading to an edge between them, a contradiction. Since B lies between its non-trivial cutvertices, therefore B has only one non-trivial cutvertex, in its rightmost layer. Thus B is a leaf of the blocktree of G'; call it B_0 and enumerate the rest of the blocktree correspondingly. In particular w_1 is the (unique) non-trivial cutvertex of B_0 and lies in its rightmost column. Block B_1 cannot use layers to the left of w_1 since w_1 separates B_0 and B_1. So B_1 is either drawn entirely within $\ell(w_1)$ (then it is necessarily a bridge) or it is drawn in $\ell(w_1)$ and further right, and its unique other non-trivial cutvertex lies in its rightmost level. Either way we obtain $\ell(w_1) \leq \ell(w_2)$ and B_1 lies only within these layers. Repeating the argument for the remaining blocks and cutvertices of G' gives the claim. □

We note here that this lemma mirrors nicely the characterization of T that have an sIUBVR: We know that this exists if and only if T is a subdivided caterpillar, which means that it, after removing subdivided legs, is a path (and hence its blocktree is also a path).

We need one last characterization of how subdivided legs can be drawn.

Observation 3 *Let G be a plane graph that has an sIUBVR D. Let G' be the graph obtained from G by removing subdivided legs and let D' be its induced sIUBVR. Let P be a subdivided leg whose attachment point v is not in the leftmost or rightmost level of D'. Then P is drawn vertically in the level of v, and either immediately above v or immediately below v.*

Proof. Let i be the level of v, and assume for contradiction that P contains a diagonal edge, say vertex w_P of P is in level $i{+}1$. By assumption some vertex w' of G' also resides in level $i{+}1$. This contradicts Lemma 13 since w_P and w' are in different cut-components of v. So P must reside within level i, and be immediately above or below v to create the edge between v and its neighbour in P. □

Now we can explain the full algorithm. First, detect all subdivided legs (this can be done in linear time by extending paths from vertices of degree 1) and remove them while marking their attachment point. So we have G' and compute its blocktree of G'. This must split into a path $B_0 - w_1 - B_1 - \cdots - w_k - B_k$, else G has no sIUBVR. Note that if G has an sIUBVR, then we may without loss of generality assume that none of the vertices of B_0 are farther right than the vertices of B_k, for otherwise we can rotate the representation by 180°. We hence can require the levels of the cutvertices to satisfy $\ell(w_1) \leq \ell(w_2) \leq \cdots \leq \ell(w_k)$.

For each block B_i of G' that is not a bridge, let H_i be the auxiliary graph computed as before, with super-source s_i and super-sink t_i. We modify H_i slightly to remove some arcs that clearly cannot lead to a solution. Namely, assume that H_i has an arc $a{=}v(e_\ell, \alpha_\ell) \rightarrow v(e_r, \alpha_r)$. If arc a is used in a solution, then the resulting sIUBVR contains $D(e_\ell, \alpha_\ell)$ and $D(e_r, \alpha_r)$ in consecutive slabs, and in particular, fixes exactly the vertices $U{=}R(e_\ell, \alpha_\ell) \cup L(e_r, \alpha_R)$ that are in the common layer of the slabs (say layer j). It also fixes the direction of incident edges of U. We remove arc a from H_i if this placement of U contradicts restrictions from Lemma 13 or Observation 3. In particular we remove a if

- U contains cutvertex w_i or w_{i+1}. (This would contradict that these two cutvertices are the leftmost or rightmost within their 2-connected component, while arc a implies that there are vertices both left and right of layer j.)
- U contains a cutvertex $w \neq w_i, w_{i+1}$ of G, and w is not the topmost or bottommost vertex of U. (Note that we are studying here cutvertices of G, not G', so such a cutvertex w can exist if it is the attachment point for some subdivided leg.)
- U contains a cutvertex $w \neq w_i, w_{i+1}$ of G, but the edges to U use ports such that we cannot attach the subdivided leg vertically at w without using consecutive ports and while respecting the planar embedding.

With this, any path from s_i to t_i in H_i leads to an sIUBVR of B_i to which we can add all subdivided legs whose attachment point is not in the leftmost or rightmost layer.

Now we want to create an auxiliary graph for $B_i \cup B_{i+1}$. Assume first that neither B_i nor B_{i+1} is a bridge. We then combine the two auxiliary graphs H_i and H_{i+1}, by eliminating vertices t_i and s_{i+1} and adding arcs between some of their neighbours. Consider any (e, α) and (e', β) such that we had arcs $(e, \alpha) \rightarrow t_i$ and $s_{i+1} \rightarrow (e', \beta)$ in H_i, i.e., we could have had these configurations in the rightmost/leftmost slab in representations of B_i/B_{i+1}. We can eliminate any such vertices if $w_{i+1} \notin R(e, \alpha)$ or $w_{i+1} \notin L(e', \beta)$, since we know that this is required in an sIUBVR of G'. If w_{i+1} occurs in both sets, then this determines a unique way of merging $D(e, \alpha)$ and $D(e', \beta)$, and with it, the direction of all edges incident to $R(e, \alpha) \cup L(e' \beta)$. We add an arc $a{=}v(e, \alpha) \rightarrow v(e', \beta)$ if this gives a feasible representations that allows adding subdivided legs. More precisely, we add arc a only if

- no port at w_i is used by edges in both partial drawings,
- no two consecutive ports at w_i are used by edges to B_i and B_{i+1},
- for any $w \in R(e, \alpha) \cup L(e', \beta)$ that is an attachment point of a subdivided leg, the incident edges in B_i and/or B_{i+1} use ports such that it is possible to add the subdivided leg vertically at w without using consecutive ports or violating the planar embedding.

(It may sound as if we could create $\Omega(n^2)$ arcs here, but similarly as in the 2-connected case we need not test all combinations of (e, α) and (e', β); we can read from the planar embedding and the outer-face path of G' a constant-size set of edges e' that need to be tested for each configuration (e, α).)

Thus if neither B_i nor B_{i+1} is a bridge, then we can combine their auxiliary graphs. If B_i is a bridge (w_i, w_{i+1}), then we create an auxiliary graph for $B_i \cup B_{i+1}$ similarly. Namely, let in this case H_i consist of vertices s_i and t_i and four more vertices $v(e, \mathrm{N}), v(e, \mathrm{NE}), v(e, \mathrm{SE})$ and $v(e, \mathrm{S})$, representing the possibility of drawing (w_i, w_{i+1}) while respecting $\ell(w_i) \leq \ell(w_{i+1})$. Each vertex determines the representation of B_i in its entirety and so we can combine this graph with H_{i+1} as above, adding arcs only if this allows for merging of subdivided legs. Similarly we deal with the case where B_{i+1} is a bridge.

With this, any directed path from s_i to t_{i+1} in the combined graph leads to an sIUBVR of $B_i \cup B_{i+1}$ where we can add all subdivided legs whose attachment point is not in the leftmost and rightmost layer. Repeatedly merging, we obtain one graph H where any path from s_0 to t_k corresponds to an sIUBVR of G' where we can add all subdivided legs whose attachment point is not in the leftmost or rightmost layer.

As a final step, we must modify H to deal with subdivided legs whose attachment points are in the leftmost layer (the rightmost layer is dealt with similarly). This is slightly different from before since Observation 3 does not apply. In fact, a subdivided leg that attaches at (say) the topmost vertex v in the leftmost layer may well use the NW-port at v, allowing the NE-port of v to be used by G'. More precisely, let $a{=}s_0 \to v(e,\alpha)$ be an arc in H_0 (and hence H). If arc a is used for a solution, then this determines the layout of $L(e,\alpha)$ in the leftmost layer. Let v_t and v_b be the topmost and bottommost vertex of $L(e,\alpha)$. We must remove a from H if one of the following happens:

- $L(e,\alpha)$ contains a cutvertex w of G that is neither v_t nor v_b.
- $v_t \neq v_b$, both v_t and v_b are cutvertices of G, v_t uses the NE-port and v_b uses the SE-port for edges in B_0. (In this case, neither of the attached subdivided legs at v_t and v_b can be drawn vertically, so they both must go to the left, but this would create an edge between them which is not allowed.)

It should be clear from the construction that if G has an sIUBVR, then there exists a path from s_0 to t_k in the final constructed auxiliary graph. For the other direction, assume we have such a path. This implies a path from s_i to t_i in each subgraph H_i and so an sIUBVR D_i for each block B_i. Furthermore, we eliminated sufficiently many arcs such that $D_0 \cup \ldots D_k$ can be combined into one, and all subdivided legs can be attached without creating unwanted edges. So we obtain an sIUBVR of G as desired. Therefore testing for an sIUBVR equals finding a path from s_0 to t_k in a directed graph, which takes linear time. Building H can be done in $O(n^2)$, since for each of the $O(n)$ arcs the conditions for eliminating or adding it can be tested in $O(n)$. Theorem 10 follows.