Monotone Drawings of Graphs with Fixed Embedding ⋆

Patrizio Angelini[1], Walter Didimo[2], Stephen Kobourov[3], Tamara Mchedlidze[4], Vincenzo Roselli[1], Antonios Symvonis[4], and Stephen Wismath[5]

[1] Università Roma Tre, Italy
[2] Università degli Studi di Perugia, Italy
[3] University of Arizona, USA
[4] National Technical University of Athens, Greece
[5] University of Lethbridge, Canada

Abstract. A drawing of a graph is a *monotone drawing* if for every pair of vertices u and v, there is a path drawn from u to v that is monotone in some direction. In this paper we investigate planar monotone drawings in the *fixed embedding setting*, i.e., a planar embedding of the graph is given as part of the input that must be preserved by the drawing algorithm. In this setting we prove that every planar graph on n vertices admits a planar monotone drawing with at most two bends per edge and with at most $4n - 10$ bends in total; such a drawing can be computed in linear time and requires polynomial area. We also show that two bends per edge are sometimes necessary on a linear number of edges of the graph. Furthermore, we investigate subclasses of planar graphs that can be realized as embedding-preserving monotone drawings with straight-line edges, and we show that biconnected embedded planar graphs and outerplane graphs always admit such drawings, which can be computed in linear time.

1 Introduction

A drawing of a graph is a *monotone drawing* if for every pair of vertices u and v, there is a path drawn from u to v that is monotone in some direction. In other words, a drawing is monotone if, for any given direction d (e.g., from left to right) and for each pair of vertices u and v, there exists a suitable rotation of the drawing for which a path from u to v becomes monotone in the direction d.

Monotone drawings have been recently introduced [1] as a new visualization paradigm, which is well motivated by human subject experiments by Huang and Eades [8] who showed that the "geodesic tendency" (paths follow a given direction) is important in comprehending the underlying graph. Monotone drawings are related to well-studied drawing conventions, such as upward drawings [5,7], greedy drawings [2,9,10], and the

⋆ Research partially supported by the MIUR project AlgoDEEP prot. 2008TFBWL4, by the ESF project 10-EuroGIGA-OP-003 GraDR "Graph Drawings and Representations", by NSERC, and by the European Union (European Social Fund - ESF) and Greek national funds through the Operational Program "Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF) - Research Funding Program: Heracleitus II. Investing in knowledge society through the European Social Fund. Work on these results began at the 6th Bertinoro Workshop on Graph drawing. Discussion with other participants is gratefully acknowledged.

M. van Kreveld and B. Speckmann (Eds.): GD 2011, LNCS 7034, pp. 379–390, 2012.

geometric problem of finding monotone trajectories between two given points in the plane avoiding convex obstacles [3].

Planar monotone drawings with straight-line edges form a natural setting and it is known that biconnected planar graphs and trees always admit such drawings, for some combinatorial embedding of the graph [1]. However, the question whether a simply connected planar graph always admits a planar monotone drawing or not is still open.

On the other hand, in the *fixed embedding setting* (i.e., the planar embedding of the graph is given as part of the input and the drawing algorithm is not allowed to alter it) it is known [1] that there exist simply connected planar embedded graphs that admit no straight-line monotone drawings.

In this paper we study planar monotone drawings of graphs in the fixed embedding setting, answering the natural question whether monotone drawings with a given constant number of bends per edge can always be computed, and identifying some subclasses of planar graphs that always admit planar monotone drawings with straight-line edges. Our contributions are summarized below:

- We prove that every n-vertex planar embedded graph has an embedding-preserving monotone drawing with *curve complexity* 2, that is, the maximum number of bends along an edge is 2, and with at most $4n - 10$ bends in total. Such a drawing can be computed in linear time and has polynomial area.
- We show that our bound on the curve complexity is tight, by describing an infinite family of embedded planar graphs that require two bends on a linear number of edges in any embedding-preserving monotone drawing.
- We investigate what subfamilies of embedded planar graphs can be realized as embedding-preserving monotone drawings with straight-line edges. We prove that biconnected embedded planar graphs and outerplane graphs always admit such a drawing, which can be computed in linear time.

The paper is structured as follows. Basic definitions and results are given in Section 2. An algorithm for computing embedding-preserving monotone drawings of general embedded planar graphs with at most two bends per edge is described in Section 3. Algorithms for computing straight-line monotone drawings of meaningful subfamilies of embedded planar graphs are given in Section 4. Concluding remarks and open questions are presented in Section 5. For space reasons some proofs are sketched or omitted.

2 Preliminaries

We assume familiarity with basic concepts of graph drawing (see, e.g., [5]). Let G be a planar graph and let ϕ be a planar embedding of G. The embedding ϕ defines the set of internal faces and the outer face of G. For every vertex v of G, the embedding ϕ also defines the circular clockwise order of the edges incident to v. Graph G along with an embedding ϕ is called an *embedded planar graph*, and is denoted by G_ϕ. Any subgraph of G_ϕ obtained by removing some edges from G_ϕ is a subgraph that *preserves* the planar embedding ϕ. A *drawing of* G_ϕ is a planar drawing of G with embedding ϕ.

A *subdivision* of a graph G is obtained by replacing each edge of G with a path. A k*-subdivision* of G is such that any path replacing an edge of G has at most k internal vertices. A graph G is *connected* if every pair of vertices is connected by a path and is *biconnected* (resp. *triconnected*) if removing any vertex (resp. any two vertices) leaves G connected. In order to handle the decomposition of a biconnected graph into its triconnected components, we use the well-known *SPQR-tree* data structure [6].

A *monotone drawing* Γ of a planar graph G (of an embedded planar graph G_ϕ) is a drawing of G (of G_ϕ) such that for every pair of vertices u and v there exists a path from u to v in Γ that is monotone in some direction.

A monotone drawing of any tree T can be constructed in polynomial area by using *Algorithm DFS-based* [1], which relies on the concept of the *Stern-Brocot tree* [11,4] $\mathcal{SB}$, an infinite tree whose nodes are in bijective mapping with the irreducible positive rational numbers. Algorithm DFS-based assigns to the edges of the tree T slopes $\frac{1}{1}, \frac{2}{1}, \ldots, \frac{n-1}{1}$ (which are the first $n-1$ elements of the rightmost path of $\mathcal{SB}$) according to a DFS-visit of T. Polynomial area is ensured by the following property of $\mathcal{SB}$.

Property 1. [4,11] The sum of the numerators of the elements of the i-th level of $\mathcal{SB}$ is 3^{i-1} and the sum of the denominators of the elements of the i-th level of $\mathcal{SB}$ is 3^{i-1}.

The following property is also satisfied by any monotone drawing Γ of a tree T.

Property 2. [1] Any drawing Γ' of T such that the slopes of each edge $e \in T$ in Γ' is the same as the slope of e in Γ is monotone. Also, the slopes of any two leaf-edges e' and e'' of T in Γ are such that e' and e'' diverge, that is, the elongations of e' and e'' do not cross each other.

3 Monotone Drawings with Bends of Embedded Planar Graphs

In this section we study monotone drawings of embedded planar graphs. We remark that it is still unknown whether every planar graph admits a straight-line monotone drawing in the variable embedding setting, while it is known that straight-line monotone drawings do not always exist if the embedding of the graph is fixed [1]. We therefore investigate monotone drawings with bends along some edges, and we show that two bends per edge are always sufficient and sometimes necessary for the existence of a monotone drawing in the fixed embedding setting.

We need some preliminary definitions. An *upright spanning tree* T of an embedded planar graph G_ϕ is a rooted ordered spanning tree of G_ϕ such that: (i) T preserves the planar embedding of G_ϕ; (ii) the root of T is a vertex r of the outer face of G_ϕ; (iii) there exists a planar drawing of G_ϕ that contains an upward drawing of T such that no edge goes below r. Fig. 1(b) and (c) show two different ordered spanning trees of the embedded planar graph of Fig. 1(a): The first one is an upright spanning tree, while the second is not. Given an embedded planar graph G_ϕ, an upright spanning tree T of G_ϕ can be computed as follows. Construct any planar straight-line drawing Γ of G_ϕ. Orient the edges of G_ϕ in Γ according to the upward direction. Let r be a vertex on the outer face of G_ϕ with the smallest y-coordinate in Γ. Then, compute any spanning tree T of G_ϕ rooted at r such that the left-to-right order of the children of r in T is consistent with the left-to-right order of the neighbors of r in Γ and the left-to-right order of the children of each vertex w in T is consistent with the clockwise order of the neighbors of w in G_ϕ, computed starting from the edge connecting w to its parent in T.

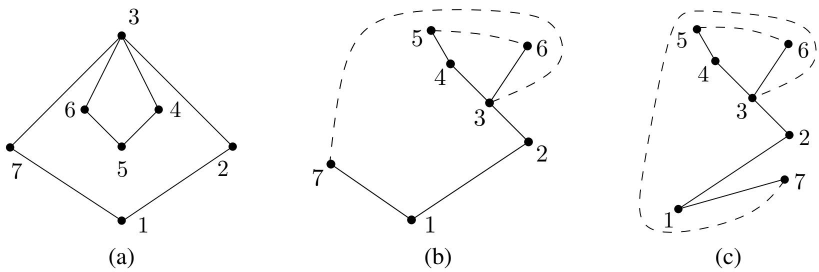


Fig. 1. (a) A drawing Γ of an embedded planar graph G_ϕ. (b) An upright spanning tree of G_ϕ. (c) A spanning tree of G_ϕ that is not upright.

Let T be an upright spanning tree of G_ϕ. The *rgbb-coloring of G_ϕ with respect to T* is a coloring of the edges of G_ϕ with four colors (red, green, blue, and black) such that: An edge is colored black if it belongs to T; an edge is colored green if it connects two leaves of T; an edge is colored red if it connects a leaf to an internal vertex of T; an edge is colored blue if it connects two internal vertices of T.

We denote by $C(G_\phi, T)$ the rgbb-coloring of G_ϕ with respect to T. We prove the following lemma.

Lemma 1. *Let G_ϕ be an embedded planar graph with n vertices, let T be an upright spanning tree of G_ϕ, and let $C(G_\phi, T)$ be the rgbb-coloring of G_ϕ with respect to T. Then we can compute a monotone drawing Γ of G_ϕ such that each black or green edge of $C(G_\phi, T)$ is drawn as a straight-line segment, each red edge has 1 bend, and each blue edge has 2 bends. The running time of the algorithm is $O(n)$ and the drawing Γ has $O(n) \times O(n^2)$ area.*

Proof. First, starting from G_ϕ and T, construct a graph G'_ϕ and an upright spanning tree T' of G'_ϕ such that: (i) G'_ϕ is a 2-subdivision of G_ϕ, (ii) T is a subtree of T', and (iii) all the edges of G'_ϕ that are not in T' connect two leaves of T'. Fig. 2(a) and (b) show a graph G_ϕ with an upright spanning tree T and the corresponding graph G'_ϕ with its upright spanning tree T' satisfying (i)–(iii). Then, the monotone drawing of G_ϕ with curve complexity 2 is constructed by first computing a straight-line monotone drawing of G'_ϕ and then replacing each subdivision vertex with a bend; see Fig. 2(c).

Graphs G'_ϕ and T' are constructed as follows. Initialize $G'_\phi = G_\phi$ and $T' = T$. Subdivide each red edge (s, t) of G'_ϕ with a vertex k and add edge (t, k) to T', where t is the internal vertex of T'. Subdivide each blue edge (s, t) of G'_ϕ twice, with two vertices k and z, and add edges (s, k) and (t, z) to T'.

The straight-line monotone drawing of G'_ϕ is computed in two steps. First, with *Algorithm DFS-based* [1], we construct a straight-line monotone drawing of T', and then we add the remaining (non-tree) edges as straight-line segments, which results in using two segments for red edges and three segments for blue edges.

To argue the monotonicity for non-tree edges, recall that, by Property 2, it is possible to elongate the edges of T' without affecting monotonicity and planarity.

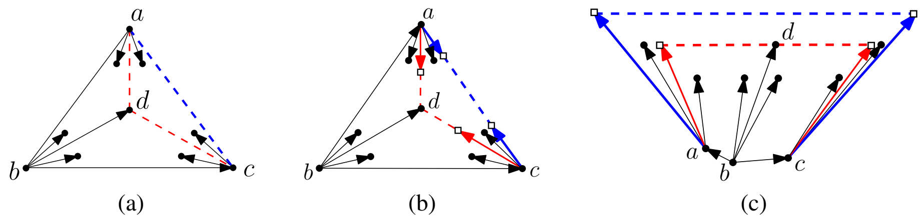


Fig. 2. (a) A graph G_ϕ with an upright spanning tree T rooted at vertex b. Solid edges belong to T, while dashed edges do not. Blue edges are thicker than red edges, which are thicker than black edges. (b) The corresponding graph G'_ϕ with its upright spanning tree T'. Solid edges belong to T', while dashed edges do not. Subdivision vertices are drawn as squares. (c) A straight-line monotone drawing of G'_ϕ that corresponds to a monotone drawing of G_ϕ with bent edges.

Further, as *Algorithm DFS-based* assigns slopes $\frac{1}{1}, \frac{2}{1}, \ldots, \frac{n-1}{1}$ to the edges of T', the elongation of each leaf-edge (u, v) intersects each vertical line $x = k$, where k is any integer value greater than the x-coordinate of u, at an integer grid point. Moreover, as by Property 2 the leaf-edge elongations diverge, such intersections appear in the same order on each vertical line $x = k'$, where k' is any integer value greater than the x-coordinate of every internal vertex of T'; see Fig. 3(a).

Another key observation is that the graph G_L induced by the leaves of T' is outerplanar and can be augmented, by adding dummy edges, to a biconnected outerplanar graph in which each internal face is a 3-cycle in such a way that the order of the vertices on the outer face is the same as the left-to-right order of the leaves of T'; see Fig. 3(b).

The vertices of G_L are assigned to levels in such a way that the end-vertices of each edge of G_L are either on the same level or on adjacent levels, as follows. The first and the last vertex in the left-to-right order of the leaves of T' have level 1. Note that, these two vertices are adjacent, as G_L is a biconnected outerplanar graph and the order of the vertices on its outer face is the same as the left-to-right order of the leaves of T'. Then, starting from this edge, consider any edge (u, v) on the outer face of the graph induced by the vertices whose level has been already assigned and consider the unique vertex w that is connected to both u and v, and whose level has not been assigned yet, if any. Note that, either u and v have the same level i or one of them has level i and the other has level $i + 1$. In both cases, assign level $i + 1$ to w, as shown in Fig. 3(b) and (c).

Let l be the number of levels of G_L. Then, place all the vertices at level i, with $i = 1, \ldots, l$, on a vertical line $x = k + l - i + 1$, where k is the x-coordinate of the rightmost internal vertex of T'. This placement, together with the fact that each such vertical line intersects the elongations of all the leaf-edges in the same order, ensures the planarity of the straight-line drawing of G_L. Further, as the order of the vertices on the outer face of G_L is the same as the left-to-right order of the leaves of T', the edges of T' do not cross any edge of G_L, hence ensuring the planarity of G'_ϕ; see Fig. 3(c).

The drawing of G'_ϕ is monotone because between any two vertices there exists a monotone path composed only of edges of T', while edges not in T' do not affect the

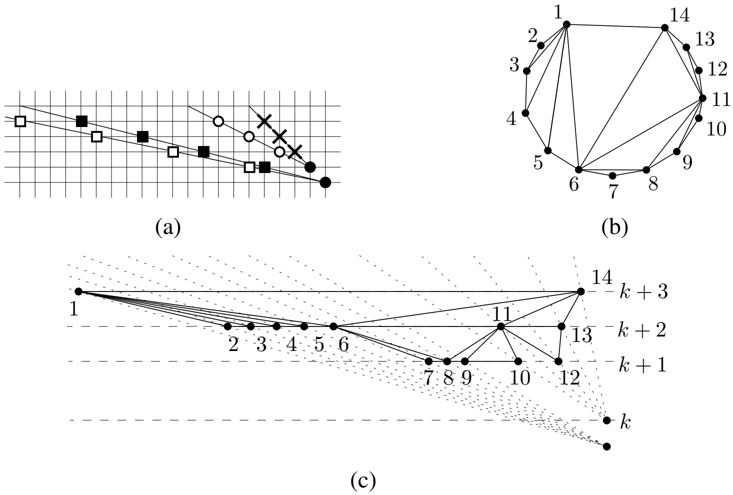


Fig. 3. For readability, the drawings in (a) and (c) are rotated to 90° and the grid unit distances in (c) are not uniform. (a) Leaf-edge elongations have integer intersections with all the vertical lines in the same order. (b) An augmented graph G_L. (c) The drawing of G_L (where $l = 3$).

monotonicity. Hence, monotonicity is maintained when dummy edges are removed. Note that, any monotone path traversing a leaf-edge of T' has the corresponding leaf as an end-vertex. If the leaf is a subdivision vertex of any non-black edge, it does not belong to G_ϕ. Hence, all the monotone paths in G_ϕ are composed only of edges of T, whose drawing is monotone since it is a subtree of T'. Therefore, the drawing of G_ϕ is monotone, each red edge has one bend, and each blue edge has two bends.

In order to compute the area of the obtained drawing, recall that *Algorithm DFS-based* [1] produces a drawing of T' in $O(n) \times O(n^2)$ area. Since the number of vertical lines added to host the drawing of G_L is equal to the number l of levels assigned to the vertices of G_L, and since l is bounded by the number of leaves, which is $O(n)$, the area of the whole drawing is still $O(n) \times O(n^2)$.

It is easy to see that the drawing can be computed in $O(n)$ time, by considering the individual steps. The computation of the three necessary graphs, T, G'_ϕ and T', can be performed in linear time. Also, the slopes of the edges of T' can be computed in linear time with *Algorithm DFS-based* [1] by constructing the Stern-Brocot tree and by performing a rightmost DFS visit of it. Further, graph G_L can be augmented in linear time. Finally, the assignment of levels to the vertices of G_L is also performed in linear time, as each vertex is considered just once and its level is assigned only based on the levels of its two neighbors. This concludes the proof of Lemma 1. $\square$

Note that, according to Lemma 1 there always exists a monotone drawing Γ of G_ϕ with curve complexity 2 and at most $4n - 10$ bends in total, as G_ϕ has at most $3n - 6$ edges and every spanning tree of G_ϕ has $n - 1$ edges. Using the algorithm described in Lemma 1, Γ has at most $2(3n-6-n+1) = 4n-10$ edges in total, and this upper bound

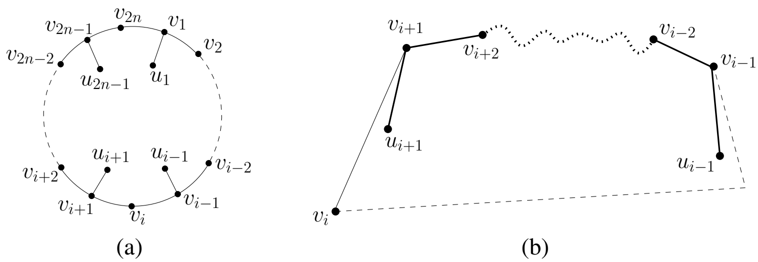


Fig. 4. (a) A graph G_ϕ with $3n$ vertices that does not admit any embedding-preserving straight-line monotone drawing. (b) Edges (v_{i-1}, v_i) and (v_i, v_{i+1}) can not be drawn as straight-line segments.

is asymptotically tight, as there exist embedded planar graphs that require a linear total number of bends in any monotone drawing. Namely, we first prove in Lemma 2 that there exist embedded planar graphs requiring at least one bend on some edges. Then, based on this lemma, we prove in Lemma 3 that there exist infinitely many embedded planar graphs whose monotone drawings require two bends on a linear number of edges.

Lemma 2. *For every $n \geq 3$ there exists an embedded planar graph G_ϕ with $3n$ vertices and $3n$ edges that does not admit any straight-line monotone drawing.*

Sketch of Proof: We describe an embedded planar graph G_ϕ that does not admit any straight-line monotone drawing (refer to Fig. 4(a)). G_ϕ consists of a simple cycle $C = v_1, \ldots, v_{2n}$ of length $2n$ and of n vertices $u_1, u_3, \ldots, u_{2n-1}$ of degree 1, called *legs*, incident to the vertices $v_1, v_3, \ldots, v_{2n-1}$ of C with odd indices, respectively. The embedding of G_ϕ is such that all the legs are inside C, that is, they are inside the unique internal face of C. As by Property 2 any two consecutive legs (v_{i-1}, u_{i-1}) and (v_{i+1}, u_{i+1}) diverge in any straight-line monotone drawing, it is not possible to connect vertices v_{i-1} and v_{i+1} by drawing edges (v_{i-1}, v_i) and (v_i, v_{i+1}) as straight-line segments. Refer to Fig. 4(b). □

The next lemma shows that there are infinitely many embedded planar graphs that require two bends per edge on a linear number of edges in any embedding-preserving monotone drawing.

Lemma 3. *For every odd $n \geq 9$ there exists an embedded planar graph G_ϕ with n vertices and $\frac{3}{2}(n-1)$ edges such that every monotone drawing of G_ϕ has at least $\frac{n-3}{6}$ edges with at least two bends and thus at least $\frac{n-3}{3}$ bends in total.*

Sketch of Proof: Refer to Fig. 5. Consider an odd integer $n \geq 9$. We construct G_ϕ iteratively. Let G^1_ϕ be a triangle graph. Graph G^i_ϕ is constructed from G^{i-1}_ϕ as follows. Initialize $G^i_\phi = G^{i-1}_\phi$. Let (u, v, w) be a triangular internal face of G^i_ϕ. Add 6 new vertices $u_1, u_2, v_1, v_2, w_1, w_2$ and 9 new edges (u, u_1), (u, u_2), (u_1, u_2), (v, v_1), (v, v_2), (v_1, v_2), (w, w_1), (w, w_2), (w_1, w_2) to G^i_ϕ in such a way that all the new vertices are inside (u, v, w). Note that the n-vertex graph G^i_ϕ is planar and has $\frac{3}{2}(n-1)$ edges. Any monotone drawing of G_ϕ has at least $\frac{n-3}{6}$ edges with at least two bends. □

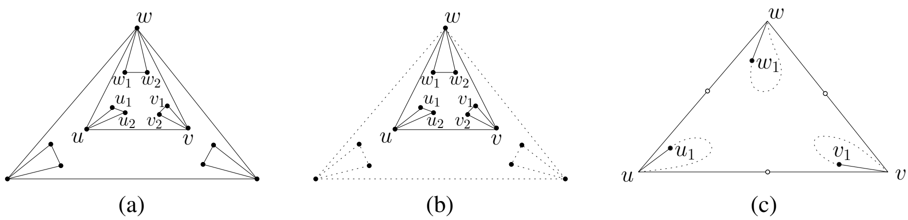


Fig. 5. (a) An example of a graph G_ϕ with $n = 15$ vertices, that coincides with a graph G^3_ϕ constructed from G^2_ϕ by adding vertices $u_1, u_2, v_1, v_2, w_1, w_2$ inside triangular face u, v, w. (b) A subgraph G^t_ϕ of G_ϕ induced by a triangle (u, v, w) and all the vertices inside it. (c) A subdivision (white circles) of the subgraph G^h_ϕ (solid edges) of G^t_ϕ induced by u, v, w, u_1, v_1, w_1. By Lemma 2, this subdivision does not admit any straight-line monotone drawing.

Lemma 1 and Lemma 3 together provide a tight bound on the curve complexity of monotone drawings in the fixed embedding setting. The next theorem summarizes the main contribution of this section.

Theorem 1. *Every embedded planar graph with n vertices admits a monotone drawing with curve complexity 2, at most $4n - 10$ bends in total, and $O(n) \times O(n^2)$ area; such a drawing can be computed in $O(n)$ time. Also, there exist infinitely many embedded planar graphs any monotone drawing of which requires two bends on $\Omega(n)$ edges.*

4 Monotone Drawings with Straight-Line Edges

In this section we prove that there exist meaningful subfamilies of embedded planar graphs that can be realized as straight-line monotone drawings. In particular, we prove that both the class of outerplane graphs and the class of embedded planar biconnected graphs have this property.

4.1 Outerplane Graphs

An embedded planar graph G_ϕ is an *outerplane graph* if all its vertices are on the outer face. We prove the following result.

Theorem 2. *Every outerplane graph admits a straight-line monotone drawing. Also, there exists an algorithm that computes such a drawing in $O(n)$ time and $O(n) \times O(n^2)$ area.*

Proof. Let T be an upright spanning tree of G_ϕ obtained by performing a "rightmost DFS" visit of G_ϕ; see Fig. 6(a). Consider a decomposition of G_ϕ into its maximal biconnected components. Observe that, for each maximal biconnected component B that is connected to the root of T through a cut-vertex v, T contains all the edges of B except for the internal chords (dashed edges in Fig. 6(a)) and for the leftmost edge incident to v (dotted edges in Fig. 6(a)).

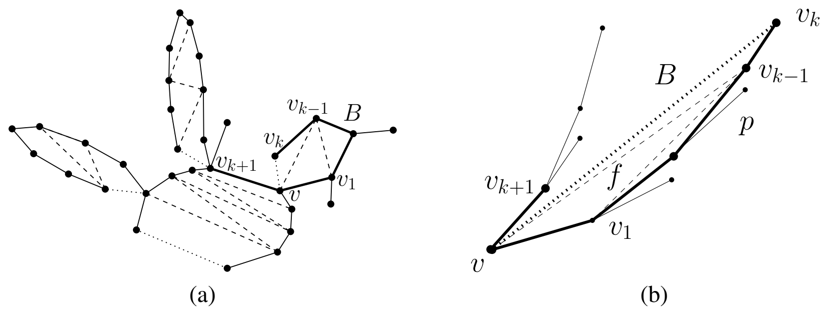


Fig. 6. (a) An outerplane graph G_ϕ and the upright spanning tree T of G_ϕ obtained by performing a "rightmost DFS" visit. Edges of T are represented as solid segments. (b) A strictly convex drawing of a maximal biconnected component B of G_ϕ.

A straight-line monotone drawing of G_ϕ is constructed by first computing a straight-line monotone drawing of T, with *Algorithm DFS-based* [1], and then reinserting the edges not in T as straight-line segments. In order to reinsert such edges, for each maximal biconnected component B, consider the path $p = (v, v_1, \ldots, v_k)$ that is composed of the edges belonging both to B and to T.

According to *Algorithm DFS-based* [1] the slopes of the edges of p are all positive and increasing with respect to the distance from v in p. Hence, path p is drawn in T as a polygonal line that is convex on the left side, that is, the straight-line segment connecting any two non-consecutive vertices of p completely lies to the left of p; see Fig. 6(b). Thus, reinserting edge (v, v_k) as the straight-line segment between v and v_k determines that (v, v_k) is the leftmost edge of B incident to v in the drawing and that the boundary of B, that is, the cycle composed of the edges of p plus (v, v_k), delimits a strictly-convex region f.

We show that f does not contain any other vertex of T. Namely, the vertex v_{k+1} such that edge (v, v_{k+1}) follows (v, v_1) in the counter-clockwise order of the edges around v in T lies outside f. This is due to the fact that, according to *Algorithm DFS-based*, the slope of (v, v_{k+1}) is greater than the slope of (v_{k-1}, v_k) which in turn is greater than the slope of (v, v_k); see Fig. 6(b).

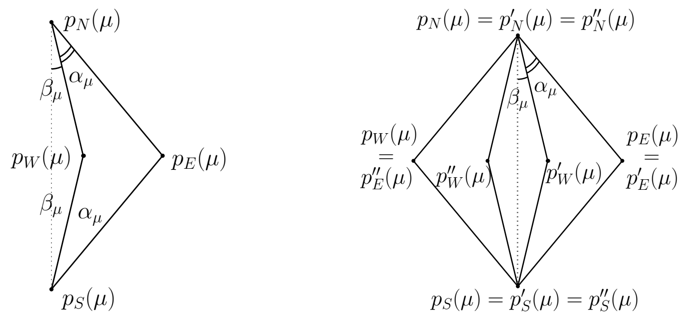
Hence, f is an empty strictly-convex region, and the chords of B can be reinserted as straight-line segments while maintaining planarity.

The area of the drawing is the same as the area of T computed by *Algorithm DFS-based*, namely $O(n) \times O(n^2)$. The drawing can be computed in $O(n)$ time. Namely, drawing T by using *Algorithm DFS-based* takes $O(n)$ time [1], and the same holds for reinserting missing edges. □

4.2 Biconnected Graphs

It is known [1] that straight-line monotone drawings of biconnected planar graphs in the variable embedding setting can always be computed. This result is obtained by means of an algorithm that exploits SPQR-trees and that preserves any given embedding, as long as the graph contains no parallel component whose poles are connected by an edge. However, this algorithm can be easily modified in order to compute monotone drawings with curve complexity 1 of every embedded biconnected planar graph, as the edges connecting the poles of a parallel component could be placed in their correct position by adding a bend, when necessary.

In this section we prove that in fact we can compute a monotone drawing of every embedded biconnected planar graph with no bends at all.



(a) (b)

Fig. 7. (a) A boomerang. (b) A diamond

As for the variable-embedding setting case [1], our algorithm relies on a bottom-up visit of the SPQR-tree of the biconnected graph G in which at each step a drawing of the pertinent graph of the currently considered node μ is constructed inside a *boomerang* $boom(\mu)$, that is, a quadrilateral composed of points $p_N(\mu)$, $p_E(\mu)$, $p_S(\mu)$, and $p_W(\mu)$ such that $p_W(\mu)$ is inside triangle $\triangle(p_N(\mu), p_S(\mu), p_E(\mu))$ and $2\alpha_\mu + \beta_\mu < \frac{\pi}{2}$, where $\alpha_\mu = p_W(\mu)\widehat{p_S(\mu)}p_E(\mu) = p_W(\mu)\widehat{p_N(\mu)}p_E(\mu)$ and $\beta_\mu = p_W(\mu)\widehat{p_S(\mu)}p_N(\mu) = p_W(\mu)\widehat{p_N(\mu)}p_S(\mu)$; see Fig. 7(a).

In order to cope with the fixed-embedding setting, we introduce a new shape, called *diamond* and denoted by $diam(\mu)$, that is a convex quadrilateral $(p_N(\mu), p_E(\mu), p_S(\mu), p_W(\mu))$ composed of two boomerangs $boom'(\mu) = (p'_N(\mu), p'_E(\mu), p'_S(\mu), p'_W(\mu))$ and $boom''(\mu) = (p''_N(\mu), p''_E(\mu), p''_S(\mu), p''_W(\mu))$ such that $p_N(\mu) = p'_N(\mu) = p''_N(\mu)$, $p_S(\mu) = p'_S(\mu) = p''_S(\mu)$, $p_E(\mu) = p'_E(\mu)$ and $p_W(\mu) = p''_E(\mu)$; see Fig. 7(b).

A diamond is used for any P-node μ having an edge e between its poles. Namely, one of the two boomerangs composing the diamond contains the child components of μ that come before e in the ordering of the components around the poles, while the other boomerang contains the other components. Note that, since P-nodes might be contained into diamonds, the algorithm for drawing S- and R-nodes inside their own boomerangs has to be adapted to deal with this case. We have the following.

Theorem 3. *Every biconnected embedded planar graph admits a straight-line monotone drawing, which can be computed in linear time.*

5 Conclusions and Open Problems

In this paper we studied monotone drawings of graphs in the fixed embedding setting. Since not all embedded planar graphs admit an embedding-preserving monotone drawing with straight-line edges, we focused on computing embedding-preserving monotone drawings with low curve complexity. We proved that curve complexity 2 always suffices and that this bound is worst-case optimal. Furthermore, we described algorithms for computing straight-line monotone drawings for meaningful subfamilies of embedded planar graphs. All the algorithms presented in this paper can be performed in linear time and most of them produce drawings which require polynomial area.

The results in this paper naturally give rise to several interesting open problems; some of them are listed below.

Existential Questions

Problem 1. Finding meaningful subfamilies of embedded planar graphs (other than outerplane graphs and embedded biconnected graphs) that admit monotone drawings with curve complexity smaller than 2.

Problem 2. Is it possible to characterize the embedded planar graphs that admit monotone drawings with curve complexity smaller than 2?

Complexity Questions

Problem 3. Given an embedded planar graph G_ϕ and an integer $k \in \{0,1\}$, what is the complexity of deciding whether G_ϕ admits a monotone drawing with curve complexity k?

Problem 4. Given a graph G and an integer $k \in \{0,1\}$, what is the complexity of deciding whether there exists an embedding ϕ such that G_ϕ admits a monotone drawing with curve complexity k?

Problem 5. Given a graph G and an integer $k \in \{0,1\}$, what is the complexity of deciding whether there exists an embedding ϕ such that G_ϕ does not admit any monotone drawing with curve complexity k?

Notice that, although Problems 3-5 are related, there is no evidence that answering one of them implies an answer for any other.

Algorithmic Questions

Problem 6. Is there any algorithm that computes monotone drawings of embedded biconnected planar graphs in polynomial area?

Problem 7. Is there any algorithm that computes monotone drawings of outerplane graphs in subcubic area?

References

1. Angelini, P., Colasante, E., Di Battista, G., Frati, F., Patrignani, M.: Monotone Drawings of Graphs. In: Brandes, U., Cornelsen, S. (eds.) GD 2010. LNCS, vol. 6502, pp. 13–24. Springer, Heidelberg (2011)
2. Angelini, P., Frati, F., Grilli, L.: An algorithm to construct greedy drawings of triangulations. J. Graph Algorithms Appl. 14(1), 19–51 (2010)
3. Arkin, E.M., Connelly, R., Mitchell, J.S.B.: On monotone paths among obstacles with applications to planning assemblies. In: Symposium on Computational Geometry, pp. 334–343 (1989)
4. Brocot, A.: Calcul des rouages par approximation, nouvelle methode. Revue Chronometrique 6, 186–194 (1860)
5. Di Battista, G., Eades, P., Tamassia, R., Tollis, I.G.: Graph Drawing. Prentice Hall, Upper Saddle River (1999)
6. Di Battista, G., Tamassia, R.: On-line planarity testing. SIAM J. Comput. 25, 956–997 (1996)
7. Garg, A., Tamassia, R.: Upward planarity testing. Order 12, 109–133 (1995)
8. Huang, W., Eades, P., Hong, S.-H.: A graph reading behavior: Geodesic-path tendency. In: PacificVis, pp. 137–144 (2009)
9. Leighton, T., Moitra, A.: Some results on greedy embeddings in metric spaces. Discrete & Computational Geometry 44(3), 686–705 (2010)
10. Papadimitriou, C.H., Ratajczak, D.: On a conjecture related to geometric routing. Theor. Comput. Sci. 344(1), 3–14 (2005)
11. Stern, M.A.: Ueber eine zahlentheoretische funktion. Journal fur die reine und angewandte Mathematik 55, 193–220 (1858)