DIEI Dipartimento di Ingegneria Elettronica e dell'informazione

RT 006-05

Upward Spirality and Upward Planarity Testing

W. Didimo, F. Giordano, and G. Liotta

Università di Perugia, ITALIA

8 September 2005

Via G. Duranti 93, 06125 Perugia, Tel.: 0755853653, Fax: 0755853654,
Web: www.diei.unipg.it, E-mail: segreteria@diei.unipg.it

Upward Spirality and Upward Planarity Testing *

Walter Didimo †
didimo@diei.unipg.it

Francesco Giordano †
giordano@diei.unipg.it

Giuseppe Liotta †
liotta@diei.unipg.it

Abstract

Let G be a digraph whose $SPQR$-tree has no R-node. The main result of this paper is a polynomial-time algorithm that tests whether G is upward planar and, if so, returns an upward planar representation of G. As an application of this result, a new FPT algorithm is presented that solves the upward planarity testing problem for general digraphs. Our results use the new notion of upward spirality that, informally speaking, is a measure of the "level of winding" that a triconnected component of G can have in an upward planar representation of G.

1 Introduction

An *upward planar drawing* of a planar digraph G is a crossing-free drawing of G such that the vertices of G are mapped to points of the plane and the edges of G are drawn as simple curves that are monotone in the upward direction. The upward planarity testing problem asks whether a planar digraph G has an upward planar drawing. A digraph that admits an upward planar drawing is an *upward planar digraph*.

Previous work and new results

The upward planarity testing problem is a classical subject of investigation in the graph drawing literature, and many papers have been devoted to this subject during the last decade. While we refer the interested reader to books chapters which extensively cover the subject [5, 14], we briefly recall here some of the most relevant research directions on upward planarity. The problem can be studied by either assuming that G is given together with a planar embedding (i.e. a circular ordering of the edges incident around each vertex) and the testing algorithm is not allowed to change this embedding, or by assuming that the planar embedding is not given. For example, Figure 1(a) depicts a digraph that does not have an upward drawing unless one can change its embedding as shown in Figure 1(b). Unfortunately, not all planar digraphs are upward planar. The digraph G of Figure 1(c) is not upward planar independent of the choice of its planar embedding; note that the undirected underlying graph of G is a series-parallel graph (i.e. the $SPQR$-tree of G does not have any R-node).

Bertolazzi et al. [1] present an $O(n^2)$-time algorithm that tests whether a digraph with a given planar embedding is upward planar. On the other hand, Garg and Tamassia [9] show that the problem is in general NP-complete if a planar embedding is not given and hence the testing algorithm

*This work is partially supported by the MIUR Project ALGO-NEXT: Algorithms for the Next Generation Internet and Web: Methodologies, Design and Applications

†Dipartimento di Ingegneria Elettronica e dell'Informazione, Università degli Studi di Perugia, Italy

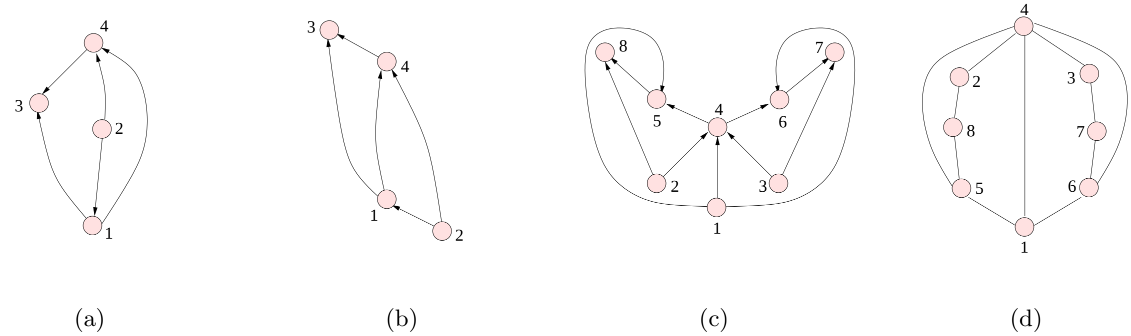


Figure 1: (a) A planar embedded digraph that does not admit an upward planar drawing within the given embedding.,(b) An upward planar drawing of the digraph in (c), obtained by changing the embedding. (c) A digraph G that is not upward planar. (d) The underlying undirected graph of G is a series-parallel graph.

is required to give a negative answer only if there is no planar embedding of the input graph that is upward planar. The result by Garg and Tamassia has motivated a considerable research effort both on testing upward planarity in polynomial time for meaningful families of planar digraphs and on devising exponential techniques that can perform well in practice.

Papakostas [15] presents an $O(n^2)$-time algorithm for testing the upward planarity of outerplanar digraphs. Hutton and Lubiw describe [13] an $O(n^2)$-time testing algorithm for digraphs that have a single source. Bertolazzi et al. [3] improve this last result by showing an optimal $O(n)$ testing algorithm for the same class of digraphs studied by Hutton and Lubiw. Bertolazzi et al. [2] describe a branch-and-bound testing algorithm for biconnected planar digraphs, and experimentally prove that it performs rather fast in practice for digraphs with up to 200 vertices and relatively low density. Very recently, fixed parameter tractable algorithms have also been designed: Chan [4] presents an $O(t! \cdot 8^t \cdot n^3 + (2 \cdot t)^{3 \cdot 2^c} t! \cdot 8^t \cdot n)$-time algorithm where c and t are the number of cut-vertices and the number of triconnected components of G, respectively. Healy and Lynch [12] improve Chan's result by giving an $O(2^t \cdot t! \cdot n2)$-time algorithm; in the same paper, Healy and Lynch describe a second upward planarity testing algorithm whose time complexity is $O(n^2 + k^4(2k+1)!)$, with $k = |E| - |V|$. Further, Healy and Lynch [10] show that a planar digraph is upward planar if and only if its blocks satisfy a specific set of conditions.

Based on the observation that the proof on NP-completeness by Garg and Tamassia [9] uses some special graphs that are triconnected (the so-called "tendrils"), we consider in this paper the question about whether the upward planarity testing problem can be solved in polynomial time for those digraphs whose $SPQR$-tree does not have any R-node (see, e.g., the digraph of Figure 1(c)). We then extend the study to testing upward planarity for general digraphs. More precisely, the main contributions of this paper are as follows.

- We introduce and study the concept of *upward spirality*. Informally speaking, upward spirality is a measure of how much a component of a digraph is "rolled-up" in an upward planar drawing. A similar concept was introduced in the literature in the context of orthogonal drawings [6].

- Let G be a series-parallel digraph with n vertices, that is G is a digraph whose $SPQR$-tree does not have any R-node. We use upward spirality to show that there exists an $O(n^4)$-time algorithm to test whether G is upward planar and, if so, to compute an upward planar representation of G.

- We exploit the above described results to design an FPT algorithm for upward planarity testing of general digraphs whose time complexity is $O(d^t \cdot n^3 + d \cdot t^2 \cdot n + d^2 \cdot n)$, where d is the maximum diameter of any split component of G and t is the number of non-leaf nodes in the $SPQR$-tree of G (i.e., t is the number of non-trivial triconnected components). To the best of our knowledge, this is the first FPT algorithm for upward planarity testing whose time complexity does not have any factorial multiplicative factors. We also remark that, our algorithm still runs in polynomial time even if the digraph is not biconnected and any block is a series-parallel digraph.

Overview of the Approach and Structure of the Paper

An overview of the main ideas behind the polynomial-time algorithm for upward planarity testing of a series-parallel digraph G is as follows.

- We prove that the upward spirality of a series or parallel component of G is at most $O(n)$ in any upward drawing of G.

- We study upward spirality both for series and for parallel compositions of subgraphs of G. We show the relationships between the value of upward spirality of the subgraph resulting by the (series or parallel) composition and the upward spiralities of its components. We also show that two different upward representations of the same component are "interchangeable" if they have the same upward spirality and other minor additional properties.

- We exploit the above two ingredients to design a polynomial time algorithm that visits the decomposition tree of G starting from the leaves and proceeding towards the root. At each visited node μ of the decomposition tree, the algorithm performs $O(n)$ series or parallel compositions to (possibly) construct $O(n)$ upward representations of the subgraph $G_\mu \subset G$ represented by μ. Each upward representation has one of the $O(n)$ possible values of upward spirality that G_μ can assume in an upward drawing of G. We call the set of upward representations of G_μ computed at this step the *feasible set* of μ.

- If the feasible set of G_μ is empty (which means that there is no upward representation of G_μ for any of the possible $O(n)$ values of upward spirality), then the testing algorithm stops and returns that G is not upward planar. Else the algorithm continues the visit of the decomposition tree of G. If the root is visited, the algorithm performs a final step to check if it is possible to construct an upward planar representation of G from the feasible set of the root and the feasible set of the child of the root.

It has to be recalled that an algorithmic approach similar to the one described in this paper is adopted in [6] to compute orthogonal drawings of 4-planar series-parallel graphs with the minimum number of bends. We remark however that orthogonal drawings have vertex degree that is bounded by a constant, which easily translates into a constant number of possible permutations to be considered when arranging the components of a parallel composition during the computation of the feasible set of a P-node. Unfortunately this is not the case for the upward planarity testing problem, where the vertex degree of the input graph G is in general $O(n)$. Hence, a brute force approach that tries all possible permutations of the components to construct the feasible set of a parallel composition would immediately give rise to an exponential time algorithm. As it will be shown in the next sections, this difficulty can be handled by modeling the computation of the feasible set of a P-node as a suitable set of network flow problems.

The remainder of the paper is structured as follows. Preliminaries can be found in Section 2. Section 3 is devoted to the concept of upward spirality. The $O(n^4)$-time upward planarity testing algorithm for series-parallel digraphs is described in Section 4. FPT algorithms for biconnected digraphs and general digraphs are given in Section 5 and Section 6. Conclusions and open problems are discussed in Section 7.

2 Preliminaries

We assume familiarity with basic concepts of graph drawing and graph planarity [5]. In Subsection 2.1 we first recall basic definitions and results on upward planarity. In Subsection 2.2 we recall the notion of $SPQR$-tree and, at the end of the session, we explain what we mean for series-parallel digraph.

2.1 Upward Planar Drawings

Let G be a planar digraph. A drawing Γ of G is an *upward planar drawing* if it has no edge-crossing and all the edges of G are drawn as curves monotonically increasing in the vertical direction. If G admits an upward planar drawing, it is called an *upward planar digraph.*

Suppose that G has a given planar embedding. A vertex of G is *bimodal* if the circular list of its incident edges can be partitioned into two (possibly empty) lists, one consisting of incoming edges and the other consisting of outgoing edges. If all vertices of G are bimodal then G and its embedding are called *bimodal.* Acyclicity and bimodality are necessary conditions for the upward planar drawability of an embedded planar digraph [1]. However, they are not sufficient conditions.

Let f be a face of an embedded planar bimodal digraph G and suppose that the boundary of f is visited clockwise if f is internal, and counterclockwise if f is external. Let $a = (e_1, v, e_2)$ be a triplet such that v is a vertex of the boundary of f and e_1, e_2 are incident edges of v that are consecutive on the boundary of f. Triplet a is called an *angle of f*. Also, a is a *switch angle of f* if the direction of e_1 is opposite to the direction of e_2 (note that e_1 and e_2 may coincide if G is not biconnected). If e_1 and e_2 are both incoming in v, then a is a *sink-switch of f*; if they are both outgoing, a is a *source-switch of f*. Observe that the number of source-switches of f is equal to the number of sink-switches of f. Let $2n_f$ be the total number of switches of f (both source- and sink-switches); the *capacity of f* is defined as $c_f = n_f - 1$ if f is an internal face, and $c_f = n_f + 1$ if f is the external face.

Let v be a source or a sink vertex of G. We observe that any two edges incident on v that are consecutive around v, always determine a switch angle in every face incident on v. For this motivation a source or a sink of G is also called a *switch vertex* of G; a vertex that is not a switch vertex is called an *internal vertex* of G.

An assignment of the switch vertices of G to its faces is *upward consistent* if the following properties hold: (a) a switch vertex is assigned to exactly one of its incident faces; (b) for each face

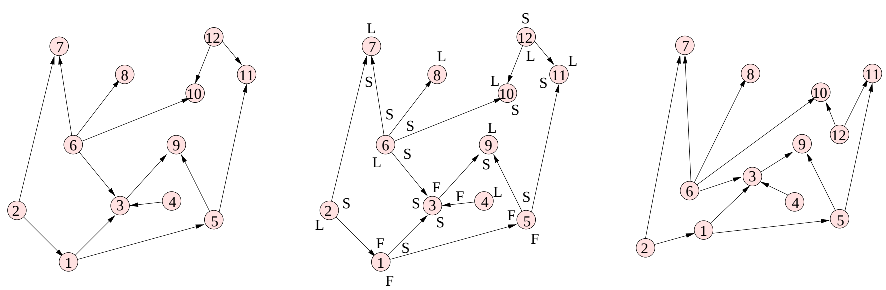
f, the number of switch vertices assigned to f is equal to c_f.

The following theorem gives a characterization of the class of embedded digraphs that are upward planar.

Theorem 1 [1] *Let G be an acyclic planar digraph with a given planar bimodal embedding. G is upward planar if and only if it admits an upward consistent assignment.*

If G has an upward consistent assignment, then the *upward planar representation of G* corresponding to that assignment is a labeled planar embedding of G such that for each face f and for each angle $a = (e_1, v, e_2)$ of f we have that: (a) a is labeled L if it is a switch of f and v is a source or a sink assigned to f; (b) a is labeled F if it is not a switch of f; (c) a is labeled S otherwise.

From an upward planar representation of G it is always possible to construct an upward planar drawing of G such that each edge is drawn as a straight-line segment (or as a polyline) and geometric angles corresponding to angles labeled L are greater than π, while geometric angles corresponding to angles labeled S are less than π. Figure 2 shows an embedded planar digraph G, an upward planar representation U_G of G, and an upward planar drawing of G within U_G.



(a) (b) (c)

Figure 2: (a) A planar embedded bimodal digraph G. (b) An upward planar representation U_G of G. (c) An upward planar drawing of G within U_G.

Now, consider an embedded planar digraph G and a labeling of its angles with labels L, S, and F. If v is a vertex of G, we denote by $L(v)$, $S(v)$, and $F(v)$ the number of angles at v that are labeled L, S, and F, respectively. The *degree* of v is defined as the number of angles at v, and is denoted as $deg(v)$. Also, if f is a face of G, $L(f)$, $S(f)$, and $F(f)$ denote the number of angles of f that are labeled L, S, and F, respectively.

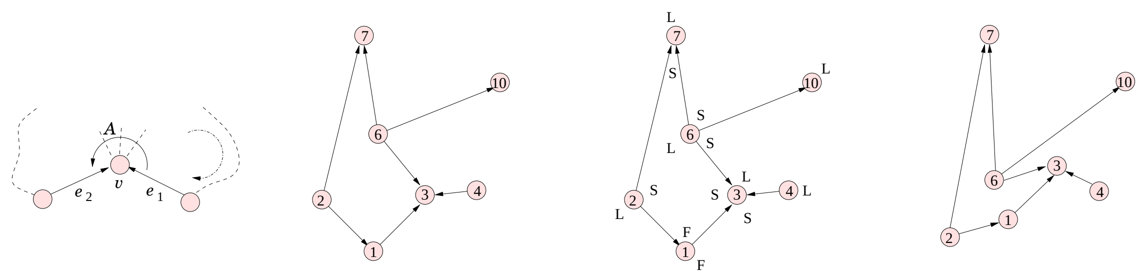
The following result is a restatement of the results in [1, 8], and gives a characterization of the classes of labeled planar embedded digraphs that define upward planar representations.

Lemma 1 *Let G be a planar embedded digraph with a given labeling of its angles with labels L, S, and F. G and its labeling define an upward planar representation if and only if the following properties hold:*

- `UP1`: *If v is a switch vertex of G then: $L(v) = 1$, $S(v) = deg(v) - 1$, $F(v) = 0$;*
- `UP2`: *If v is not a switch vertex of G then: $L(v) = 0$, $S(v) = deg(v) - 2$, $F(v) = 2$;*

- UP3: *If f is a face of G then: $L(f) = S(f) - 2$ if f is internal and $L(f) = S(f) + 2$ if f is external.*

Given an upward planar representation U_G of an embedded planar digraph G, the angles labeled L, S, and F are called *large*, *small*, and *flat* angles, respectively. Let G' be a subgraph of G. Digraph G' has an *upward planar representation $U_{G'}$ induced by U_G*, which is defined as follows. Let $a = (e_1, v, e_2)$ be an angle of G', and let A be the counterclockwise sequence of angles of U_G between e_1 and e_2 (see Figure 3(a)). Angle a in $U_{G'}$ is labeled: L if A either contains one large angle or two flat angles; F if A contains only one flat angle; S otherwise. Figure 3 shows a subgraph of the digraph G in Figure 2(a), the upward planar representation induced by the one in Figure 2(b), and a corresponding upward planar drawing.



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

Figure 3: (a) The sequence A of angles between two edges e_1, e_2 incident on the same vertex v. (b) A sub-graph G' of the digraph G in Figure 2(a). (c) The upward planar representation U'_G of G' induced by U_G in Figure 2(a). (d) An upward planar drawing of G within $U_{G'}$.

2.2 $SPQR$-trees and Series-Parallel Digraphs

Let G be a biconnected undirected graph. A *split pair* of G is either a separation pair or a pair of adjacent vertices of G. A *split component* of G with respect to (or determined by) a split pair $\{u, v\}$ is either an edge (u, v) or a maximal subgraph C of G such that C contains u and v, and $\{u, v\}$ is not a split pair of C. Let $e = (s, t)$ be an edge of G, referred to as the *reference edge* of G in the following. A *maximal split pair* $\{u, v\}$ of G with respect to $\{s, t\}$ is a split pair of G distinct from $\{s, t\}$ such that every other split pair $\{u', v'\}$ of G determines a split component containing s, t, u, v.

An $SPQR$-tree T of G with respect to edge e, describes a decomposition of G in terms of its triconnected components [7]. The decomposition is done recursively on the split pairs of G, by starting with the split pair $\{s, t\}$. Namely T is a rooted tree whose nodes are of four types: S, P, Q, and R. Q-nodes represent single edges, i.e., trivial triconnected components; S-nodes and P-nodes represent series and parallel components, respectively; R-nodes represent components that are triconnected graphs. Each node μ of T has an associated biconnected multigraph, called the *skeleton* of μ, and denoted as $skel(\mu)$. Tree T is recursively defined as follows:

- **Trivial case** If G consists of exactly two parallel edges between s and t, then T is a Q-node, whose skeleton is G itself. Q represents the edge that is parallel to the reference edge.

- **Parallel case** If the split pair $\{s, t\}$ has at least three split components $G_0, \ldots, G_{k-1}$ $(k \geq 3)$, the root of T is a P-node μ. $skel(\mu)$ consists of k parallel edges between s and t, denoted as $e_0, \ldots, e_{k-1}$, where each e_i represents component G_i $(i = 1, \ldots, k)$, and e_0 is the reference edge.

- **Series case** If the split pair $\{s, t\}$ has exactly two maximal split components, one of them is the reference edge e, and we denote by G' the other. If G' is a chain of biconnected components $G_1, \ldots, G_k$ and cutvertices $c_1, \ldots, c_{k-1}$ $(k \geq 2)$, then the root of T is an S-node μ. $skel(\mu)$ is the cycle $e_0, e_1, \ldots, e_k$, where $e_0 = e, c_0 = s, c_k = t$, and $e_i = (c_{i-1}, c_i)$ $(i = 1, \ldots, k)$.

- **Rigid case** If none of the above cases applies, let $\{s_1, t_1\}, \ldots, \{s_k, t_k\}$ be the maximal split pairs of G with respect to $\{s, t\}$ $(k \geq 1)$, and for $i = 1, \ldots, k$, let G_i be the union of all split components of $\{s_i, t_i\}$ except the one containing the reference edge e. The root of T is an R-node, and $skel(\mu)$ is obtained from G by replacing each subgraph G_i with the edge $e_i = (s_i, t_i)$.

Except for the trivial case, μ has children $\mu_1, \ldots, \mu_k$ in this order, such that μ_i is the root of the $SPQR$-tree of graph $G_i \cup e_i$ with respect to reference edge e_i. Edge e_i is said to be the *virtual edge* of μ_i in $skel(\mu)$, and of node μ in $skel(\mu_i)$. The end-vertices s, t of e are called the *poles* of $skel(\mu)$. Subgraph $G_i \subseteq G$ is called the *pertinent graph* of μ_i, and is denoted as G_{μ_i}. The *poles* of G_{μ_i} coincide with the poles of $skel(\mu_i)$. The tree T so obtained has a Q-node associated with each edge of G, except for the reference edge e. The construction of T is completed by adding another Q-node representing e, and making it as the root of T. The skeleton of the root node consists of two parallel edges: the reference edge and a virtual edge representing the rest of the graph. For each Q-node μ (included the root), the pertinent graph of μ is its corresponding edge of G.

Tree T implicitly describes all possible planar embeddings of G with edge e on the external face, obtained combining the planar embeddings of the skeletons of its nodes. Indeed, while the skeletons of Q- and S-nodes have only one planar embedding, the skeleton of a P-node has a number of planar embeddings that is equal to the number of possible permutations of its non-reference edges, and the skeleton of an R-node has two different planar embeddings, obtained by flipping the component with respect to its reference edge. If G has n vertices, T has $O(n)$ nodes. Figure 8 shows an example of $SPQR$-tree of a graph.

Suppose now that G is given with an *st*-numbering of its vertices, such that the source and the sink of this numbering are the end-vertices s, t of the reference edge of G. If T is the $SPQR$-tree of G with respect to e, given any node μ of T, let u and v be the two poles of μ, so that u precedes v in the *st*-numbering. We call u and v the *first pole* and the *second pole* of G_μ, respectively. If G has a fixed planar embedding with reference edge e on the external face, the *right face* of G_μ is the face to the right of G_μ in G, while moving from u to v. The *left face* of G_μ is the face to the left of G_μ in G, while moving from u to v. The path on the right face of G_μ, going from u to v, is called the *right path* of G_μ. The path on the left face of G_μ, going from u to v, is called the *left path* of G_μ.

In the remainder of the paper, we consider $SPQR$-trees of directed graphs (digraphs) G. In this case, the computation of the decomposition tree is done exactly as defined above, by ignoring the orientation of the edges of G. Notice that there is no connection between the orientation of the edges of G and the definition of first and second poles of the pertinent digraphs.

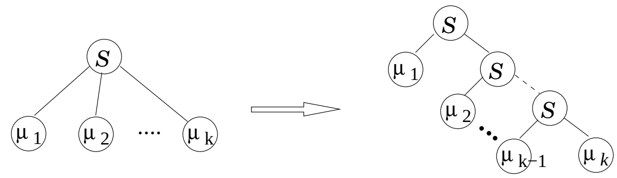


Figure 4: Transformation of an $SPQR$-tree into its canonical form.

Also, in order to simplify the description of our upward planarity testing algorithm (see Section 4), we use *canonical $SPQR$-trees*. Namely, in a canonical $SPQR$-tree, each S-node has always two children. A canonical $SPQR$-tree T of G can be constructed from an $SPQR$-tree of G by applying on every S-node the transformation illustrated in Figure 4. Of course, a canonical $SPQR$-tree of G has a number of nodes that is still linear in the number of vertices of G.

We say that a biconnected digraph G is a *series-parallel digraph* if its $SPQR$-tree only consists of Q-, S-, and P-nodes. In Sections 3 and 4 we shall assume that our digraphs are always biconnected.

3 Upward Spirality

In this section we introduce and study the notion of upward spirality, which will be extensively used in the next sections to design upward planarity testing algorithms. In the following, we assume that G is a biconnected digraph, T an $SPQR$-tree of G, U_G an upward representation of G, and G_μ the pertinent digraph of a node μ of T, with first pole u and second pole v.

In Subsection 3.1 we classify the poles of G_μ in U_G and define two suitable paths for G_μ. In Subsection 3.2 the definition of upward spirality is given together with lemmas that show how to compute the upward spirality of series and parallel compositions, given the upward spiralities of their components.

3.1 Types of Poles and Spines

Let $P =< v_1, e_1, v_2, \dots, v_i, e_i, \dots, e_{k-1}, v_k >$ be any simple (undirected) path (possibly a simple cycle) in G, and let U_P be the upward planar representation of P induced by U_G. Consider a vertex v_i ($i \in \{2, \dots, k-1\}$) that is a switch of P, and denote by $a = (e_{i-1}, v_i, e_i), a' = (e_i, v_i, e_{i-1})$ the two angles at v_i in U_P (see Figure 5). Walking on P from v_1 to v_k, we say that v_i is a *left turn* (resp. *right turn*) of U_P if a (resp. a') is large (see Figure 5). In the following, we denote by $n(U_P)$ the number of right turns minus the number of left turns of U_P, and we call $n(U_P)$ the *turn number* of P in U_G, or simpler, the *turn number* of U_P. Similarly, if P is a simple cycle, i.e. $v_1 = v_k$, and we walk clockwise on P, we say that we encounter a left turn (resp. right turn) of U_P on any switches of P that has a large angle (resp. small angle) inside the cycle; we denote by $n(U_P)$ the number of right turns minus the number of left turns of U_P, walking clockwise, and we still call it the turn number of U_P. The following property is an immediate consequence of Lemma 1.

Property 1 *If P is a simple cycle of U_G, then its turn number is $n(U_P) = 2$.*

Figure 5: (a) The two angles at a switch vertex of a path. (b) An upward representation U_P of a path P, with 2 left turns and 1 right turn; $n(U_P) = -1$.

Denoted by $w \in \{u, v\}$ any of the two poles of G_μ, we want to classify w on the basis of the labeling of the angles at w in U_G. The label of the angle at w in the right face (resp. in the left face) of G_μ is called the *right inter-label* (resp. the *left inter-label*) of w. An *intra-label* of w is any label of an angle at w internal at G_μ. We assign to each angle label an integer weight, in such a way that labels S, F, and L have weight 0, 1, and 2, respectively. The *intra-labeling weight* of w is the sum of the weights of all intra-labels of w. Observe that, from properties `UP1` and `UP2` of Lemma 1, the intra-labeling weight of w ranges from 0 to 2. More precisely, if this weight is 0, w has only S intra-labels; if it is 1, w has exactly one F intra-label and the others are S intra-labels; finally, if the intra-labeling weight is 2, w has either two F intra-labels or one L intra-label, and the others are S intra-labels.

In U_G, we describe the angles labeling of the pole w of G_μ, by using a string $t_w = XY\lambda$, such that X is the left inter-label of w, Y is the right inter-label of w, and λ is the intra-labeling weight of w. We say that t_w is the *pole category* of w. We remark that, since U_G is an upward planar representation, not all categories $XY\lambda$ ($X, Y \in \{S, F, L\}, \lambda \in \{0, 1, 2\}$) are possible for a pole w of a pertinent digraph of G. Indeed, as also observed above, the sum of all angle labels at w must verify `UP1` and `UP2`, and w must be bimodal. Hence, the following lemma immediately follows (see also Figure 6):

Lemma 2 *The possible pole categories of any pole of G_μ in U_G are: $SS0$, $SS1$, $SS2$, $SF0$, $SF1$, $FS0$, $FS1$, $FF0$, $SL0$, $LS0$.*

In order to introduce the notion of upward spirality we need to identify two suitable vertices that we call the *left external vertex of w*, denoted as w_l, and the *right external vertex of w*, denoted as w_r, where w is still any of the two poles of G_μ. The right and the left external vertices of w are defined based on the pole category t_w of w, with respect to G_μ in U_G. More precisely, let e_l be the edge incident on w, that is on the left path of G_μ and that does not belong to G_μ; let e_r be the edge incident on w, that is on the right path of G_μ and that does not belong to G_μ. Also, let x be the end-vertex of e_l other than w and let y be the end-vertex of e_r other than w. The external vertices w_l and w_r of w are defined as follows:

- **Case 1:** One of the following three subcases is verified: (i) $t_w \in \{SS0, SF0, FS0, FF0\}$; (ii) $t_w = SL0$ and w is the first pole of G_μ; (iii) $t_w = LS0$ and w is the second pole of G_μ. In this case $w_l = w_r = w$. See Figure $7(a) - (e)$.

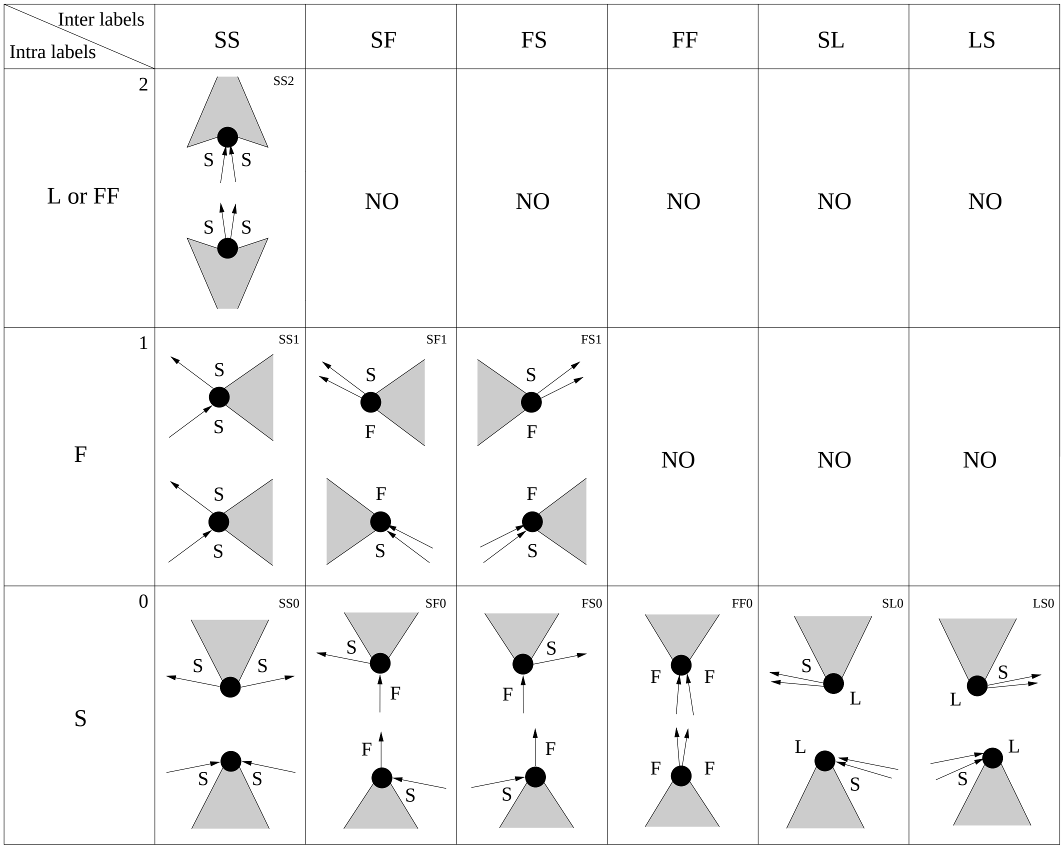


Figure 6: A schematic illustration of all possible pole categories for the first pole of a pertinent digraph within an upward planar representation. The grey portion denotes the internal part of the pertinent digraph. The two labels around the pole represent the inter-labels of the pole. The illustration for the second pole can be constructed in symmetric way.

- **Case 2:** One of the following two subcases is verified: (i) $t_w \in \{FS1, SF1, SS1, SS2\}$; (ii) $t_w = SL0$ and w is the second pole of G_μ; (iii) $t_w = LS0$ and w is the first pole of G_μ. In this case $w_l = x$ and $w_r = y$. See Figure $7(f) - (j)$.

Let u_l, u_r be the left and the right external vertices of the first pole u of G_μ and let v_l, v_r be the left and the right external vertices of the second pole v of G_μ. Let P_{uv} be an (undirected) path from u to v in G_μ. The undirected path $P_l = (u_l, u) \cup P_{uv} \cup (v, v_l)$ is called a *left spine of* G_μ. The path $P_r = (u_r, u) \cup P_{uv} \cup (v, v_r)$ is called a *right spine of* G_μ. For example, the left spine and the right spine of a pertinent digraph are highlighted in Figure 8.

In general G_μ can have more than one left spine and more than one right spine, depending on the choice of P_{uv}. However, the following lemma shows that the turn number of a spine of a pertinent digraph of an upward representation is an invariant property of the upward representation itself.

Lemma 3 *let P'_r, P''_r be two distinct right spines of G_μ and let P'_l, P''_l be two distinct left spines of G_μ. Then $n(U_{P'_r}) = n(U_{P''_r})$ and $n(U_{P'_l}) = n(U_{P''_l})$.*

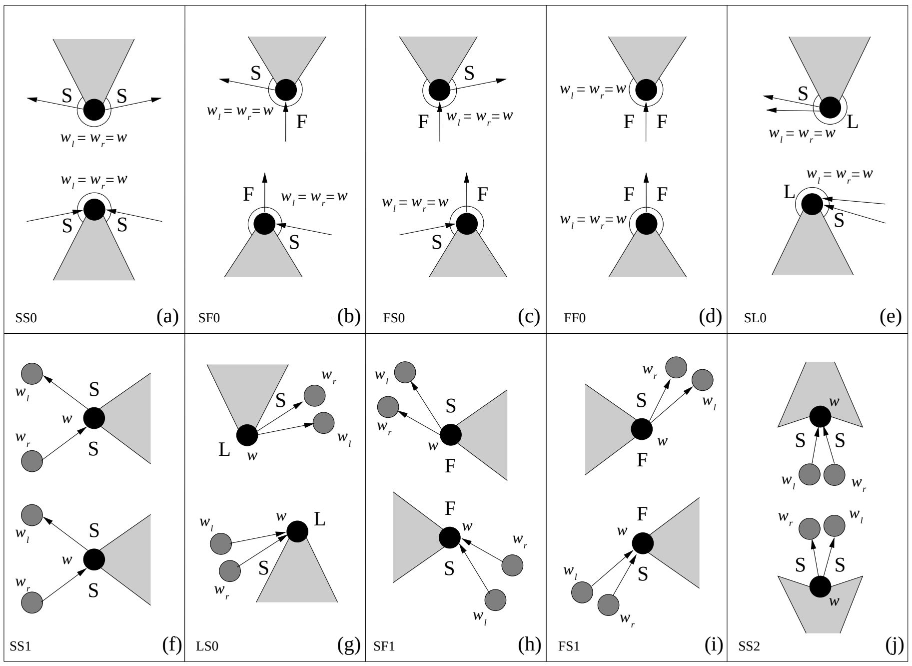


Figure 7: Illustration of the external vertices for the first pole of a pertinent digraph.

Proof: We prove the first equality; the proof of the second one is symmetric. Let $P_{v_r u_r}$ be a simple (undirected) path in $G - G_\mu$, from v_r to u_r. This path exists since G is biconnected. Let C' and C'' be the following simple cycles: $C' = P'_r \cup P_{v_r u_r}$ and $C'' = P''_r \cup P_{v_r u_r}$. From Property 1, we have that $n(U_{C'}) = n(U_{C''}) = 2$. Denote by a'_{u_r}, a'_{v_r} the contributions of the turns at vertices u_r and v_r inside C', i.e., $a'_{u_r} = -1$ if u_r is a left turn of C', $a'_{u_r} = 1$ if u_r is a right turn of C', and $a'_{u_r} = 0$ if u_r is not a turn; the same definition applies for a'_{v_r}. Analogously, denote by a''_{u_r}, a''_{v_r} the contributions of the turns at vertices u_r and v_r inside C''. We have that $n(U_{C'}) = n(U_{P'_r}) + n(U_{P_{v_r u_r}}) + a'_{u_r} + a'_{v_r}$, and $n(U_{C''}) = n(U_{P''_r}) + n(U_{P_{v_r u_r}}) + a''_{u_r} + a''_{v_r}$, which implies that $n(U_{P'_r}) + a'_{u_r} + a'_{v_r} = n(U_{P''_r}) + a''_{u_r} + a''_{v_r}$. If u_r and v_r are both distinct from u and v, then $a'_{u_r} = a''_{u_r}$ and $a'_{v_r} = a''_{v_r}$. Therefore, in this case, $n(U_{P'_r}) = n(U_{P''_r})$. On the other side, if u_r coincides with u, by the definition of the external vertices, the intra-labeling weight of u is 0, and then we still have $a'_{u_r} = a''_{u_r}$. The same reasoning applies if v_r coincides with v. This proves the thesis. □

For example, in Figure 8, $G_{\mu'}$ has only two left spines, that also coincide with the right spines. The turn number of these spines is -1. Based on Lemma 3, we can denote by $n_l(U_{G_\mu})$ the turn number of any left spine of G_μ in U_G, without ambiguity; similarly, $n_r(U_{G_\mu})$ denotes the turn number of any right spine of G_μ.

3.2 Upward Spirality and its Properties

Based on the notion of left spine and right spine we can now define the concept of upward spirality of the pertinent digraph G_μ within U_G. The *upward spirality* of G_μ within U_G (or simpler, the *upward spirality* of U_{G_μ}), is denoted as $\sigma(U_{G_\mu})$ and is defined as follows:

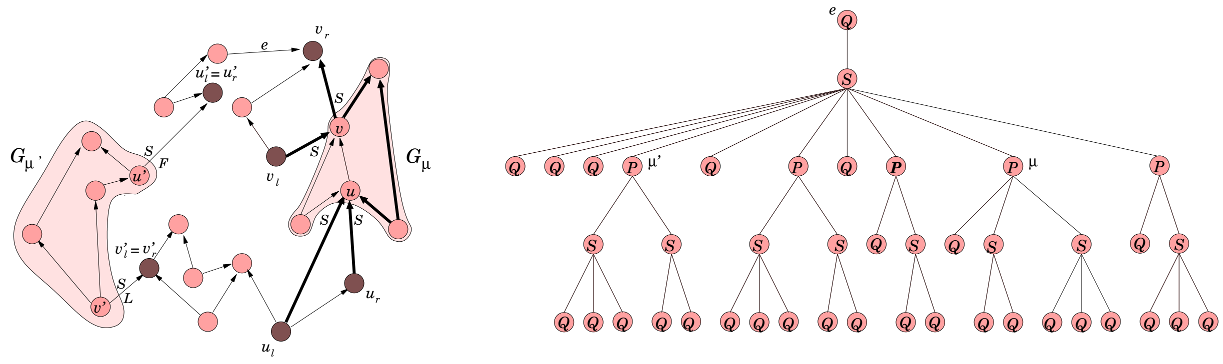


Figure 8: An upward planar representation of a series-parallel digraph G, and an $SPQR$-tree T of G rooted at edge e. G_μ and $G_{\mu'}$ are the pertinent digraphs of nodes μ and μ' of T, with poles u, v, u', v', respectively. The pole categories of u and v are $SS2$ and $SS1$, respectively. The ones of u' and v' are $FS1$ and $SL0$, respectively. The left and the right spines of G_μ constructed on the right path of G_μ are highlighted.

$$\sigma(U_{G_\mu}) = \frac{n_l(U_{G_\mu}) + n_r(U_{G_\mu})}{2}$$

For example, in Figure 8, $\sigma(U_{G_\mu}) = -1/2$, and $\sigma(U_{G_{\mu'}}) = -1$. Suppose now that P_l and P_r are a left spine and a right spine of G_μ, constructed using the same path $P_{uv} =< u, w_1, w_2, \ldots, w_k, v >$ between the poles u, v of G_μ. As already done in the proof of Lemma 3, we can rewrite the turn number of the spines as follows:

$$n(U_{P_l}) = n(U_{P_{uv}}) + a_{u_l} + a_{v_l}, \quad n(U_{P_r}) = n(U_{P_{uv}}) + a_{u_r} + a_{v_r},$$

where $a_{u_l} = n(U_{P_{u_l}})$, $a_{u_r} = n(U_{P_{u_r}})$, $a_{v_l} = n(U_{P_{v_l}})$, $a_{v_r} = n(U_{P_{v_r}})$, and $P_{u_l} =< u_l, u, w_1 >$, $P_{u_r} =< u_r, u, w_1 >$, $P_{v_l} =< w_k, v, v_l >$, $P_{v_r} =< w_k, v, v_r >$. Of course, $a_{u_l}, a_{u_r}, a_{v_l}, a_{v_r} \in \{-1, 0, 1\}$. From the invariant property of Lemma 3, the upward spirality of U_{G_μ}, can be rewritten as follows:

$$\sigma(U_{G_\mu}) = n(U_{P_{uv}}) + \frac{(a_{u_l} + a_{u_r})}{2} + \frac{(a_{v_l} + a_{v_r})}{2} \qquad (1)$$

Observe that $\sigma(U_{G_\mu})$ can be rewritten according to Formula (1), using any path P_{uv} and, of course, $a_{u_l}, a_{u_r}, a_{v_l}, a_{v_r}$ change accordingly. In order to uniquely refer to the values $a_{u_l}, a_{u_r}, a_{v_l}, a_{v_r}$ for the upward spirality of U_{G_μ}, we aim at writing $\sigma(U_{G_\mu})$ in a kind of canonical form, choosing always a "special" path P_{uv}. To decide how to choose such a path, we first emphasize some properties of the different paths of G_μ. We define the following equivalence relationship between any two paths P'_{uv}, P''_{uv} of G_μ, within a given upward representation U_G of G. We say that P'_{uv}, P''_{uv} are *turn equivalent* if $n(U_{P'_{uv}}) = n(U_{P''_{uv}})$, i.e, if they have the same turn number. Since $\sigma(U_{G_\mu})$ assumes the same value if we use P'_{uv} or P''_{uv} in Formula (1), and since $a_{u_l}, a_{u_r}, a_{v_l}, a_{v_r} \in \{-1, 0, 1\}$, the following property immediately holds.

Property 2 *The turn-equivalence relationship partitions the set of the undirected paths of U_{G_μ}, from the first to the second pole, into a finite set of equivalence classes.*

The following lemma gives a relationship among the right path, the left path, and the other paths of G_μ.

Lemma 4 *Let P^r_{uv} be a path of G_μ that is turn-equivalent to the right path of G_μ, and let P^l_{uv} be a path of G_μ that is turn-equivalent to the left path of G_μ. If P_{uv} is any path of G_μ between u and v, then $n(U_{P^l_{uv}}) \geq n(U_{P_{uv}}) \geq n(U_{P^r_{uv}})$.*

Proof: Consider first the inequality involving P^l_{uv}. Denote by $a^{min}_{u_l}, a^{min}_{u_r}, a^{min}_{v_l}, a^{min}_{v_r}$ the minimum values that $a_{u_l}, a_{u_r}, a_{v_l}, a_{v_r}$ can take in G_μ, respectively, and consider again Formula (1). What we are going to demonstrate is that $a_{u_l} = a^{min}_{u_l}, a_{u_r} = a^{min}_{u_r}, a_{v_l} = a^{min}_{v_l}, a_{v_r} = a^{min}_{v_r}$, if we choose in Formula (1) as path P_{uv}, the path P^l_{uv}. If P^l_{uv} is such that $a_{v_l} = -1$, then it is trivially true that $a_{v_l} = a^{min}_{v_l}$. Suppose that $a_{v_l} = 0$. In this case there is no turn traversing v, while moving on path $P^l_{uv} \cup (v, v_l)$. Since P^l_{uv} is the leftmost path of G_μ, this implies that cannot exist any other path P_{uv} of G_μ such that $P_{uv} \cup (v, v_l)$ causes a left turn traversing v, because otherwise the planarity of the upward representation would be broken. If $a_{v_l} = 1$, path $P^l_{uv} \cup (v, v_l)$ has a right turn on v. Again, since P^l_{uv} is the leftmost path of G_μ, any other path $P_{uv} \cup (v, v_l)$ must have a right turn on v, due to the planarity of the upward representation. The same reasoning can be applied to prove that a_{u_r}, a_{u_l}, and a_{v_r} are minimum for P^l_{uv}. Therefore, since $\sigma(U_{G_\mu})$ is an invariant property, from Formula (1) we have that $n(U_{P^l_{uv}}) \geq n(U_{P_{uv}})$. The validity of the inequality $n(U_{P_{uv}}) \geq n(U_{P^r_{uv}})$ follows a symmetric proof. $\square$

In Formula (1) we now choose as path P_{uv} any path P^r_{uv} that is turn-equivalent to the right path of G_μ, and we consider the corresponding values $(a_{u_l} + a_{u_r})/2$ and $(a_{u_l} + a_{u_r})/2$. Denote $n(U_{P^r_{uv}})$ by $\alpha(U_{G_\mu})$, and denote $(a_{u_l} + a_{u_r})/2$, $(a_{u_l} + a_{u_r})/2$ by $\alpha_u(U_{G_\mu})$ and $\alpha_v(U_{G_\mu})$, respectively. The upward spirality of U_{G_μ} can be rewritten as follows:

$$\sigma(U_{G_\mu}) = \alpha(U_{G_\mu}) + \alpha_u(U_{G_\mu}) + \alpha_v(U_{G_\mu}) \tag{2}$$

We call $\alpha(U_{G_\mu})$ the *internal spirality* of U_{G_μ}, and $\alpha_u(U_{G_\mu})$, $\alpha_v(U_{G_\mu})$ the *first-pole spirality* and the *second-pole spirality*, respectively. Notice that, from the proof of Lemma 4, each of the terms $a_{u_l}, a_{u_r}, a_{v_l}, a_{v_r}$ in Formula (1) takes the maximum possible value when $P_{uv} = P^r_{uv}$. This also implies that, for any choice of P_{uv}, $(a_{u_l} + a_{u_r})/2 \leq \alpha_u(U_{G_\mu})$ and $(a_{v_l} + a_{v_r})/2 \leq \alpha_v(U_{G_\mu})$. Therefore, for each pole category, it is possible to determine the exact value of the two pole spiralities, since we know that they take the maximum possible value and since we know what are the two external vertices. Figure 9 shows the values of the first- and the second-pole spiralities for each pole category.

Pole categories	*First–pole spirality*	*Second–pole spirality*
SS2	*1*	*1*
SS1	*1/2*	*1/2*
SF1	*0*	*0*
FS1	*1*	*1*
SS0	*0*	*0*

Pole categories	*First–pole spirality*	*Second–pole spirality*
SF0	*0*	*0*
FS0	*0*	*0*
FF0	*0*	*0*
SL0	*0*	*–1*
LS0	*1*	*0*

Figure 9: The values of the first- and the second-pole spiralities for each pole category.

The next results provide lower and upper bounds to the value of the upward spirality of a pertinent digraph within a given upward representation, and prove that the upward spirality can take only a number of values that is linear in the number of the vertices of the digraph.

Lemma 5 *Let $\overline{n}$ be the minimum number of switches on any path between the poles u and v of G_μ. Then, $-\overline{n} - 2 \leq \sigma(U_{G_\mu}) \leq \overline{n} + 2$.*

Proof: Consider a path P_{uv} of G_μ that has $\overline{n}$ switches. Obviously, for this path we have $-\overline{n} \leq n(U_{P_{uv}}) \leq \overline{n}$. On the other hand, from Lemma 4, we have that: $-\overline{n} \leq n(U_{P_{uv}}) \leq n(U_{P^l_{uv}})$ and $n(U_{P^r_{uv}}) \leq n(U_{P_{uv}}) \leq \overline{n}$. Since $a_{u_l}, a_{v_l}, a_{u_r}, a_{v_r} \in \{-1, 0, +1\}$, from Formula (1) it follows that: $-\overline{n} - 2 \leq n(U_{P^l_{uv}}) - 2 \leq \sigma(U_{G_\mu}) \leq n(U_{P^r_{uv}}) + 2 \leq \overline{n} + 2$. □

Lemma 6 *$\alpha_u(U_{G_\mu}) + \alpha_v(U_{G_\mu}) \in \{-1, -1/2, 0, 1/2, 1, 3/2, 2\}$.*

Proof: It is immediate by combining all possible values for $\alpha_u(U_{G_\mu})$ and $\alpha_v(U_{G_\mu})$ (see Figure 9). □

From Lemma 5 and Lemma 6 the following result holds.

Theorem 2 *Let G be a digraph with n vertices, let T be an $SPQR$-tree of G, and let G_μ be the pertinent digraph of a node μ of T. There are at most $O(n)$ values for the upward spirality of G_μ within any upward planar representation of G.*

In the following we describe the relationships between the upward spirality of series and parallel compositions, and the upward spiralities of their components. Without loss of generality, we assume that T is a canonical $SPQR$-tree of G.

Lemma 7 *Let μ be an S-node of T with children μ_1 and μ_2. Let G_μ be the pertinent digraph of μ, with poles u and v, and let G_{μ_1}, G_{μ_2} be the pertinent digraphs of μ_1, μ_2, with poles $u_1 = u, v_1$, and $u_2 = v_1, v_2 = v$, respectively. The following relationship holds:*

$$\sigma(U_{G_\mu}) = \sigma(U_{G_{\mu_1}}) + \sigma(U_{G_{\mu_2}}) \tag{3}$$

Proof: According to Formula (2), we need to prove that: $\alpha(U_{G_\mu}) + \alpha_u(U_{G_\mu}) + \alpha_v(U_{G_\mu}) = \alpha(U_{G_{\mu_1}}) + \alpha_u(U_{G_{\mu_1}}) + \alpha_v(U_{G_{\mu_1}}) + \alpha(U_{G_{\mu_2}}) + \alpha_u(U_{G_{\mu_2}}) + \alpha_v(U_{G_{\mu_2}})$. This equality can be simplified. First notice that $\alpha_u(U_{G_\mu}) = \alpha_u(U_{G_{\mu_1}})$ and $\alpha_v(U_{G_\mu}) = \alpha_v(U_{G_{\mu_2}})$. Also, let e_1, e_2 be the ordered pair of edges incident on $v_1 = u_2$ that we encounter going from u to v on the right path of G_μ. Denote by a the angle $(e_1, v_1 = u_2, e_2)$. We have that $\alpha(U_{G_\mu}) - a = \alpha(U_{G_{\mu_1}}) + \alpha(U_{G_{\mu_2}})$. Thus, we only need to prove that $a = \alpha_v(U_{G_{\mu_1}}) + \alpha_u(U_{G_{\mu_2}})$. Only a few cases of series compositions are possible, according to the different pole categories of $U_{G_{\mu_1}}$ and $U_{G_{\mu_2}}$ (refer to Figure 10). The validity of the equality for each case can be proved using the table in Figure 9. For example, if $t_{v_1} = SS1$, $t_{u_2} = SS1$, we have $a = 1$, and indeed, from the table in Figure 9, we have that $\alpha_v(U_{G_{\mu_1}}) = \alpha_u(U_{G_{\mu_2}}) = 1/2$. The analysis for the other cases is done analogously. □

Lemma 8 *Let μ be a P node of T with children $\mu_1, \ldots, \mu_k$, ordered from left to right. Let G_μ be the pertinent digraph of μ and let $G_{\mu_1}, \ldots, G_{\mu_k}$ be the pertinent digraphs of $\mu_1, \ldots, \mu_k$, respectively. For each $i = 1, \ldots, k$, the following relationships hold:*

$$\alpha(U_{G_\mu}) = \alpha(U_{G_{\mu_i}}) + \delta^{(i)}(U_{G_\mu}), \qquad \delta^{(i)}(U_{G_\mu}) \in \{0, 1, 2, 3, 4\} \tag{4}$$

$$\alpha(U_{G_{\mu_1}}) \geq \alpha(U_{G_{\mu_2}}) \geq \cdots \geq \alpha(U_{G_{\mu_k}}) = \alpha(U_{G_\mu}) \tag{5}$$

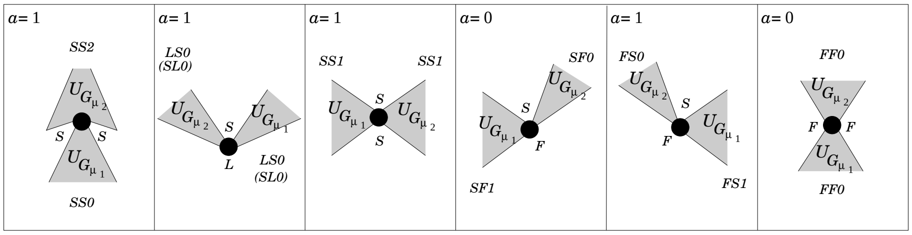


Figure 10: Different types of series compositions, according to the different pole categories of $U_{G_{\mu_1}}$ and $U_{G_{\mu_2}}$.

Proof: We first prove Equality (4). Let $P^r_{uv} =< u = v_1, e_1, v_2, \ldots, e_{k-1}, v_k = v >$ be the right path of U_{G_μ} and let $P^r_{uv_i} =< u = v^i_1, e^i_1, v^i_2, \ldots, e^i_{k_i-1}, v^i_{k_i} = v >$ be the right path of $U_{G_{\mu_i}}$. From the definition of internal spirality, $\alpha(U_{G_\mu}) = n(U_{P^r_{uv}})$ and $\alpha(U_{G_{\mu_i}}) = n(U_{P^r_{uv_i}})$, with $n(U_{P^r_{uv}}) \neq n(U_{P^r_{uv_i}})$ in general. Denote by λ_{u,u^i} and λ_{v,v^i} the sum of the weights of all intra-labels inside the angle (e_1, u, e^i_1) and $(e_{k-1}, v, e^i_{k_i-1})$, respectively. It is easy to see that $n(U_{P^r_{uv}}) = n(U_{P^r_{uv_i}}) + \lambda_{u,u^i} + \lambda_{v,v^i}$. Therefore, assuming $\delta^{(i)}(U_{G_\mu}) = \lambda_{u,u^i} + \lambda_{v,v^i}$, Equality (4) holds. In order to prove Inequality (5), it is sufficient to apply iteratively the relationship of Lemma 4, to the right and the left paths of the children of μ. $\square$

Let U_G be an upward representation of G, G_μ the pertinent digraph of a node μ of G, with first pole u and second pole v, and U_{G_μ} the upward representation of G_μ induced by U_G. Consider the subgraph G' of G consisting of G_μ plus the edges incident on u and v that are external to G_μ, and let $U'_{G'}$ be any upward planar representation of G' such that the planar embedding of the external edges of G_μ and the angle labels between these edges in $U'_{G'}$ are the same as in U_G. Notice that, the planar embedding of G_μ in $U'_{G'}$ can be different from the one in U_G. Although G' is not biconnected, the definitions of pole categories and upward spirality of U'_{G_μ} apply the same. Indeed, the left and right paths of G_μ in $U'_{G'}$ are still defined. Denote by $t'_u = X'_u Y'_u \lambda_u$ and $t'_v = X'_v Y'_v \lambda_v$ the pole categories of u and v for U'_{G_μ}.

The operation of *substitution* of U_{G_μ} with U'_{G_μ} in U_G defines a new planar embedded digraph $S(U'_{G_\mu}, U_G)$ with angle labels S, F, and L such that:

- The planar embedding and the labels of the angles of subgraph $G - G_\mu$ are the same as in U_G.
- The planar embedding and the labels of the angles of subgraph G_μ are the same as in U'_{G_μ}.
- The inter-labels of G_μ at u and at v are X'_u, Y'_u, X'_v, Y'_v, respectively.

We say that U_{G_μ} is *substitutable* with U'_{G_μ} in U_G if $S(U'_{G_\mu}, U_G)$ is still an upward planar representation of G. The following theorem is the main result of this section (see also Figure 11).

Theorem 3 *Let G be a biconnected planar digraph, T an SPQR-tree of G, G_μ the pertinent digraph of a node μ of T, and U_G an upward planar representation of G. Let G' be the subgraph of G consisting of G_μ plus the edges incident on u and v that are external to G_μ, and let $U'_{G'}$ be any upward planar representation of G' such that the planar embedding of the external edges of G_μ and* *the angle labels between these edges in $U'_{G'}$ are the same as in U_G. If U'_{G_μ} and U_{G_μ} have the same upward spirality and the same pole categories (i.e. $t'_u = t_u$, $t'_v = t_v$), then U_{G_μ} is substitutable with U'_{G_μ} in U_G.*

Proof: Denote by U''_G the planar embedded digraph $S(U'_{G_\mu}, U_G)$. In order to prove that U_{G_μ} is substitutable with U'_{G_μ} in U_G, we have to prove that conditions of Lemma 1 are satisfied for U''_G. Conditions `UP1` and `UP2` are true for each vertex of U''_G by the definition of the operation of substitution and by the hypothesis that U_G and $U'_{G'}$ are upward planar representations. Concerning Condition `UP3` it is clearly verified for each face of U''_G that is internal to U'_{G_μ} and for each face of U''_G that is neither the right face f'_r nor the left face f'_l of U'_{G_μ} in U''_G. We need to prove that `UP3` holds for f'_r and f'_l. Face f'_r is formed by the right path P'^{r}_{uv} of U'_{G_μ} and a path P'_{vu} from v to u. Denoted by f_r the right face of U_{G_μ}, f_r is formed by the right path P^r_{uv} of U_{G_μ} and a path P_{vu} from v to u. From the operation of substitution, $P'_{vu} = P_{vu}$, and then the S and L labels for the angles on these two paths are the same in f_r and f'_r. On the other hand, U'_{G_μ} and U_{G_μ} have the same pole categories, and then the same pole spiralities; since they also have the same upward spirality, from Formula 2, they have the same internal spirality. This implies that P'_r and P_r are turn equivalent, and then the difference of S and L labels on the angles on P'_r is the same as for the angles on P_r. Since f_r verifies `UP3`, then f'_r verifies `UP3`, too. A symmetric reasoning applies for f'_l, observing that also the left paths of U'_{G_μ} and U_{G_μ} are turn equivalent. This proves the thesis. □

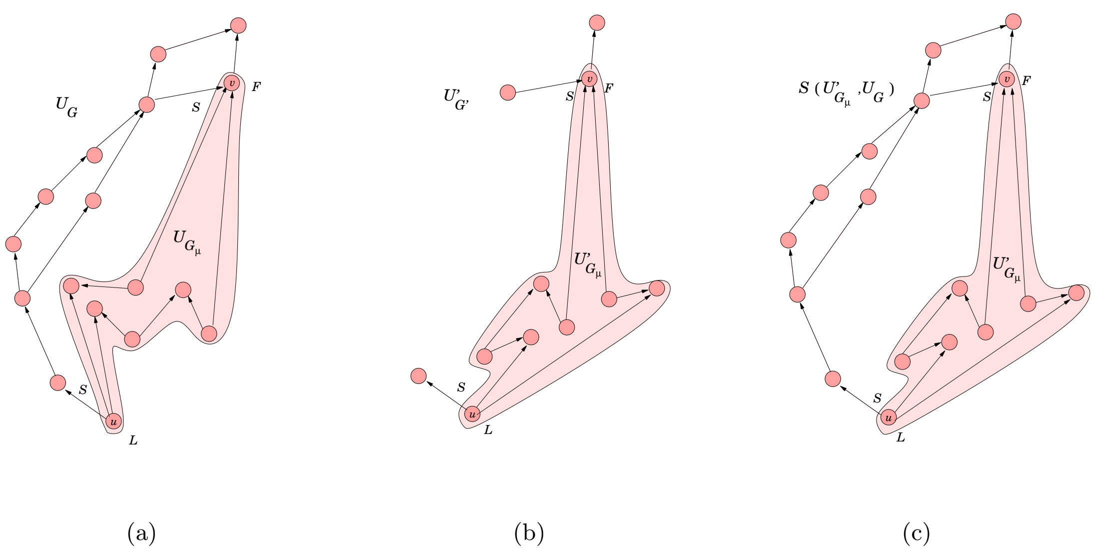


Figure 11: (a) An upward planar representation U_G with an induced planar representation U_{G_μ} highlighted; The upward spirality of U_{G_μ} is 0. (b) An upward planar representation $U'_{G'}$ of the subgraph G' formed by G_μ and the external edges incident on the poles of G_μ; U'_{G_μ} has upward spirality 0. (c) The upward planar representation $S(U'_{G_\mu}, U_G)$ obtained by substituting U_{G_μ} with U'_{G_μ} in U_G.

4 Upward Planarity Testing of Series-Parallel Digraphs

In this section we describe a polynomial-time algorithm for the upward planarity testing of a series-parallel digraph G. In the affirmative case, the algorithm also constructs an upward planar representation of G in polynomial time.

The outline of the algorithm is as follows. For each possible choice of an edge e of G, the algorithm computes the $SPQR$-tree T of G with reference edge e. Then, the algorithm visits T from bottom to top, in post-order. Each time a node μ of T is visited, μ is equipped with a set of upward planar representations of G_μ (which we call *feasible set of* μ), such that each upward planar representation is constrained to have assigned pole categories and an assigned value of upward spirality. Using the result of Theorem 3, for each possible combination of pole categories and upward spirality value, the algorithm stores only one constrained upward planar representation, if there exists one. The feasible set of each S-node and P-node of T is computed by considering the feasible sets of its children. In this way, the algorithm incrementally tries to construct an upward planar representation of G with edge e on the external face, from the leaves to the root, while exploring a subset of upward planar representations that is "representative" of the whole set of upward planar representations of G. The algorithm ends if the feasible set of a node is empty or if the feasible sets of all nodes have been successfully computed.

In Subsection 4.1, we formalize the definition of feasible sets. In Subsections 4.2 and 4.3 we describe how the algorithm computes the feasible sets of each node of T. Finally, in Subsection 4.4 we explain how the algorithm tries to construct a final upward planar representation of G by using the feasible set of the root of T (which is the Q-node associated with the reference edge e) and the feasible set of the child of the root.

4.1 Feasible Sets

Let G be a biconnected planar digraph with a given $SPQR$-tree T, computed with respect to a reference edge e. Let G_μ be the pertinent digraph of a node μ of T, with first pole u and second pole v. We want to consider upward planar representations of G_μ that are constrained to have prescribed pole categories and prescribed upward spirality when embedded in an upward planar representation of G, which does not necessarily exist.

A *feasible tuple* of μ is defined as follows:

$$\tau_\mu =< U_{G_\mu}, \sigma(U_{G_\mu}), t_u, t_v >$$

where U_{G_μ} is an upward planar representation of G_μ with pole categories t_u, t_v and upward spirality $\sigma(U_{G_\mu})$. Let $\tau'_\mu =< U'_{G_\mu}, \sigma(U'_{G_\mu}), t'_u, t'_v >$ and $\tau''_\mu =< U''_{G_\mu}, \sigma(U''_{G_\mu}), t''_u, t''_v >$ be two feasible tuples of μ. We say that U'_{G_μ} and U''_{G_μ} are *spirality equivalent* if $\sigma(U'_{G_\mu}) = \sigma(U''_{G_\mu})$, $t'_u = t''_u$, and $t'_v = t''_v$. In this case, we also say that τ'_μ and τ''_μ are *spirality equivalent*.

A *feasible set* $\mathcal{F}_\mu$ of μ is a set of feasible tuples of μ such that there is exactly one representative tuple for each class of spirality equivalent feasible tuples of μ.

Suppose now that G is upward planar with edge e on the external face. The next lemma proves that there exists an upward planar representation U_G of G, with e on the external face, such that, for each node μ of T, the upward planar representation U_{G_μ} induced by U_G is in the feasible set of μ. The lemma guarantees that our algorithm is able to find an upward planar representation of G with e on the external face, if there exists one.

Lemma 9 *Let G be an upward planar digraph with edge e on the external face, and let T be the SPQR-tree of G with respect to e. There exists an upward planar representation U_G of G such that: (i) e belongs to the external face of U_G; (ii) for each node μ of T, there exists a feasible tuple $\tau_\mu =< U_{G_\mu}, \sigma(U_{G_\mu}), t_u, t_v >$ in the feasible set of μ, where U_{G_μ} is the upward representation of G_μ induced by U_G.*

Proof: From the hypothesis, there exists an upward planar representation U'_G of G with e on the external face. Then, for each node μ of T there exists a feasible tuple $\tau'_\mu =< U'_{G_\mu}, \sigma(U'_{G_\mu}), t'_u, t'_v >$, where U'_{G_μ} is the upward planar representation induced by U'_G and t'_u, t'_v are the pole categories of U'_{G_μ}. From the definition of feasible sets there exists a tuple $\tau_\mu =< U_{G_\mu}, \sigma(U_{G_\mu}), t_u, t_v >$ in $\mathcal{F}_\mu$ that is spirality equivalent to τ'_μ. Notice that τ_μ and τ'_μ may be coincident. Therefore, from Theorem 3, we can substitute U'_{G_μ} with U_{G_μ} in U'_G, obtaining a new upward planar representation U''_G of G. We can iterate this reasoning for all nodes of T. □

The feasible set of a node μ is computed based on the typology of μ. Hence, we distinguish between three cases, i.e., μ is a Q-node, μ is an S-node, or μ is a P-node. The feasible sets of Q- and S-nodes can be computed in a relatively simple way, since the skeletons of these nodes have only one planar embedding. The computation of the feasible set of a P-node is a more complicated task, since the skeleton of a P-node with k children has $O(k!)$ possible planar embeddings, and we want to keep the computation polynomial in the number of vertices of the graph.

4.2 Constructing the Feasible Sets of Q-nodes and S-nodes

All the Q-nodes have the same feasible set, which can be computed with a pre-processing step. Namely, if μ is a Q-node, both the internal spirality and the internal-labeling weight of any upward planar representation U_{G_μ} of G_μ are equal to 0. From the table in Figure 9, it follows that we can only have three upward spirality values for U_{G_μ}: 0, 1, and -1. More precisely, if (u, v) is the (undirected) edge represented by μ, the algorithm inserts in $\mathcal{F}_\mu$ a tuple for each possible combination of upward spirality and pole categories as listed below:

- $\sigma(U_{G_\mu}) = 0$, $t_u \in \{SS0, SF0, FS0, FF0, SL0\}$, $t_v \in \{SS0, SF0, FS0, FF0, LS0\}$.
- $\sigma(U_{G_\mu}) = 0$, $t_u = LS0$ and $t_v = SL0$.
- $\sigma(U_{G_\mu}) = 1$, $t_u = LS0$, $t_v \in \{SS0, SF0, FS0, FF0, LS0\}$.
- $\sigma(U_{G_\mu}) = -1$, $t_u \in \{SS0, SF0, FS0, FF0, SL0\}$, $t_v = SL0$.

In all these tuples, U_{G_μ} is the edge (u, v) oriented upward. Of course, the feasible set of each Q-node can be computed in $O(1)$ time.

Let μ be an S-node of T, and let u and v be the first pole and the second pole of G_μ, respectively. Let μ_1, μ_2 be the two children of μ; denote by $u_1 = u$, v_1 the first pole and the second pole of G_{μ_1}; also denote by $u_2 = v_1$, $v_2 = v$ the first pole and the second pole of G_{μ_2}. The feasible set of μ is computed using the relationship of Lemma 7. More precisely, for each pair of tuples $\tau_1 =< U_{G_{\mu_1}}, \sigma(U_{G_{\mu_1}}), t_{u_1}, t_{v_1} >\in \mathcal{F}_{\mu_1}$, $\tau_2 =< U_{G_{\mu_2}}, \sigma(U_{G_{\mu_2}}), t_{u_2}, t_{v_2} >\in \mathcal{F}_{\mu_2}$, the algorithm checks if they are in one of the cases listed in Figure 10; in other words, it checks if the inter-labels of t_{v_1} and t_{u_2} are the same, and if the orientations of the edges incident on $u_2 = v_1$ in $U_{G_{\mu_1}}$ and $U_{G_{\mu_2}}$ are compatible. In the affirmative case, it constructs a new tuple $\tau =< U_{G_\mu}, \sigma(U_{G_\mu}), t_u, t_v >$, which will be inserted in $\mathcal{F}_\mu$, only if $\mathcal{F}_\mu$ does not already contain a spirality equivalent tuple; τ is defined as follows:

- $\sigma(U_{G_\mu}) = \sigma(U_{G_{\mu_1}}) + \sigma(U_{G_{\mu_2}})$;
- $t_u = t_{u_1}$, $t_v = t_{v_2}$;
- U_{G_μ} is the series composition of $U_{G_{\mu_1}}$ and $U_{G_{\mu_2}}$ on the common vertex $u_2 = v_1$.

Since each feasible set has $O(n)$ tuples, the feasible set of an S-node can be computed in $O(n^2)$ time.

4.3 Constructing the Feasible Sets of P-nodes

Let μ be a P-node of T, with first pole u and second pole v. Let $\mu_1, \ldots, \mu_k$ be the children of μ. We remark that each G_{μ_i} ($i = 1, \ldots, k$) has $u_i = u$ and $v_i = v$ as the first pole and the second pole, respectively. In order to construct the feasible set of μ, we have to evaluate the possibility of constructing an upward planar representation U_{G_μ}, for each possible way of fixing $\sigma(U_{G_\mu}), t_u$, and t_v. Namely, for each choice of $\sigma(U_{G_\mu}), t_u, t_v$, the algorithm must verify if it is possible to select from the feasible sets of $\mu_1, \ldots, \mu_k$, a subset of upward planar representations $U_{G_{\mu_1}}, \ldots, U_{G_{\mu_k}}$ that can assume a "parallel configuration" compatible with $\sigma(U_{G_\mu}), t_u, t_v$. Indeed, the conditions of Lemma 8 allow us to limit the number of these configurations, so that it is not needed to consider all permutations of the children of μ in the $skel(\mu)$.

We shall define the set of possible configurations on the basis of t_u and t_v; each configuration will consist of a sequence of groups, in such a way that each group can host a certain number of upward planar representations, all having the same pole categories and the same internal spirality (which also implies the same upward spirality). The groups in the sequence will be ordered according to their values of internal spirality. In this way, on the basis of $\sigma(U_{G_\mu})$ and for each configuration above defined, the algorithm tries to select a set of upward representations $U_{G_{\mu_1}}, \ldots, U_{G_{\mu_k}}$ from the feasible sets of $\mu_1, \ldots, \mu_k$ and to assign each of them to a group in the configuration.

Subsection 4.3.1 describes in details how the configurations are defined, and the algorithm for constructing the feasible sets is given in Subsection 4.3.2.

4.3.1 Defining Configurations

We first define local configurations with respect to t_u and with respect to t_v, independently. Then, we describe how it is possible to merge the local configurations to get the final configurations associated with t_u and t_v. We concentrate on the first pole u of U_{G_μ}; symmetric reasonings apply for the second pole v.

A *local configuration* $\mathcal{A}_{t_u}$ is a sequence of *groups*, where each group g has the following properties:

- All the upward representations assigned to g will have the same pole category t_g^u for u.
- All the upward representations $U_{G_{\mu_i}}$ assigned to g will have the same value of "angle off-set" δ_g, where δ_g is the sum of the weights of all intra-labels between the edges incident on u and that are on the right path of U_{G_μ} and $U_{G_{\mu_i}}$, respectively. Observe that, $\delta_g \in \{0, 1, 2\}$.

From the possible pole categories introduced so far and from the above properties, all groups with pole category $t_g^u \neq SS0$, must host exactly one upward representation, while groups with pole category $t_g^u = SS0$ can host zero or many upward representations. Indeed, a group with pole category $SS0$ does not give any contribution to the off-set δ_g and it allows many distinct upward representations to stay together inside it in any ordering. The groups of any local configuration are ordered according to increasing values of δ_g. The upward planar representations $U_{G_{\mu_1}}, \ldots, U_{G_{\mu_k}}$ will be incident on u in counterclockwise order, according to the ordering of the groups they are assigned to. For example, Figure 12, shows the two possible local configurations for the case $t_u = SS1$. From the fact that $\delta_g \in \{0, 1, 2\}$ and that the pole categories are a finite set, it follows that the maximum number of groups of a local configuration is a constant number with respect to the vertices of G.

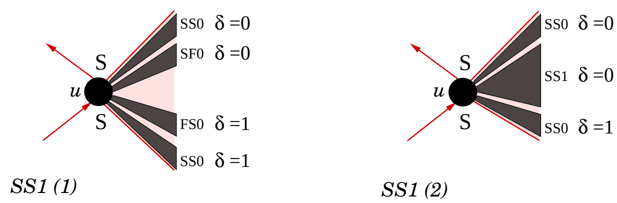


Figure 12: The two local configurations for the case $t_u = SS1$. The groups in each local configuration are the dark sectors. Each group is labeled with a string that denotes the associated pole category. The angle off-set of each group is also reported.

The following properties are immediate from a case analysis, completely illustrated in Figure 13.

Property 3 *There are at most 5 local configurations for each pole category.*

Property 4 *There are at most 7 groups in any local configuration.*

Given the local configurations for t_u and t_v, we define all possible *configurations*. Namely, for each pair of local configurations $\mathcal{A}_{t_u}$ and $\mathcal{A}_{t_v}$ we define a new set of configurations $\{\mathcal{A}^1_{t_u,t_v}, \ldots, \mathcal{A}^h_{t_u,t_v}\}$, such that each $\mathcal{A}^i_{t_u,t_v}$ still consists of an ordered sequence of *groups*. Each group $g \in \mathcal{A}^i_{t_u,t_v}$ originates by selecting and merging two groups $g_u \in \mathcal{A}_{t_u}$ and $g_v \in \mathcal{A}_{t_v}$, according to the following rules:

- All upward representations in g will have the same pole categories $t_g^u = t_{g_u}^u$, $t_g^v = t_{g_v}^v$.
- All upward representations in g will have the same internal spirality $\alpha(U_{G_\mu}) + \delta_g$, where $\alpha(U_{G_\mu})$ is the internal spirality of U_{G_μ} and $\delta_g = \delta_{g_u} + \delta_{g_v}$.

The groups in $\mathcal{A}^i_{t_u,t_v}$ are ordered according to increasing values of internal spirality. Every group that has at least one of the two poles in a category distinct from $SS0$ will host exactly one upward representation, while groups whose both poles are in the category $SS0$ can host any number of upward representations (possibly zero).

Notice that, given two local configurations $\mathcal{A}_{t_u}$ and $\mathcal{A}_{t_v}$, not all possible combinations of groups g_u and g_v from $\mathcal{A}_{t_u}$ and $\mathcal{A}_{t_v}$ can be considered to construct a valid configuration $\mathcal{A}^i_{t_u,t_v}$. Indeed, there are some subsets of combinations of g_u and g_v that cannot give rise to any valid upward planar representation for G_μ. In the following we describe how to define all (and only) the valid configurations.

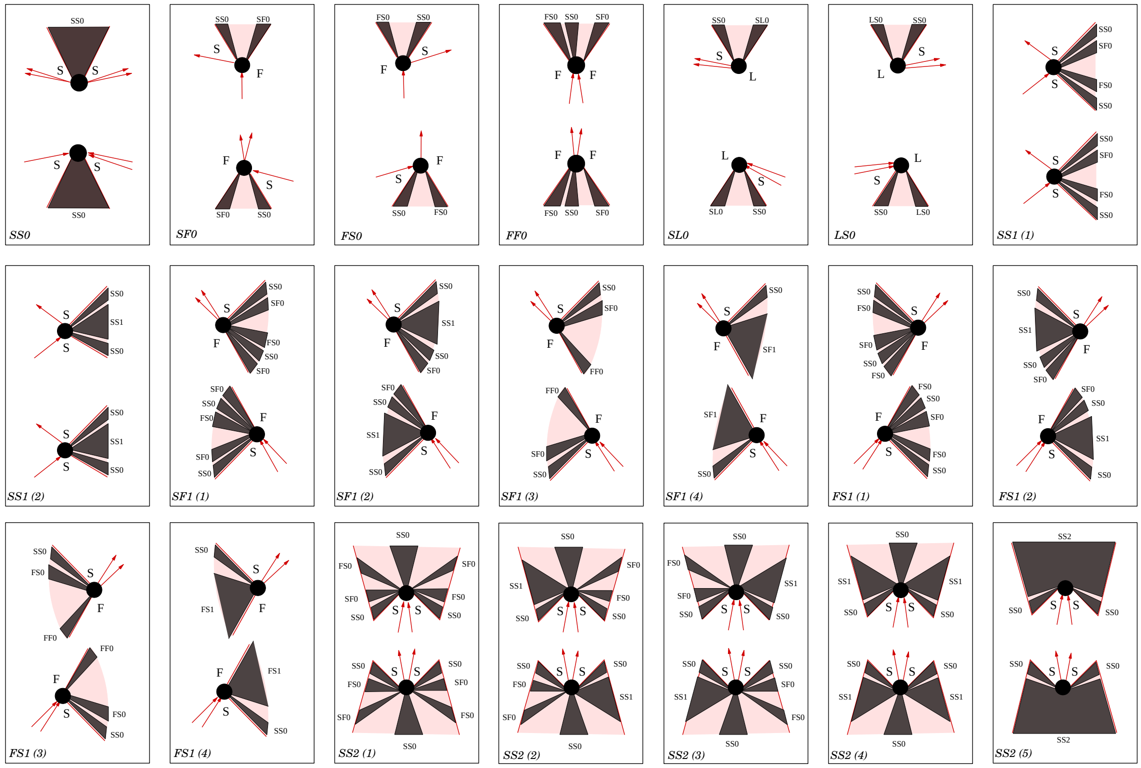


Figure 13: Illustration of all possible local configurations associated with the first pole, and their groups (dark sub-regions).

Let $\mathcal{A}_{t_u} = \{g1_u, \ldots, g_u^p\}$ and $\mathcal{A}_{t_v} = \{g1_v, \ldots, g_v^q\}$. We construct a 2-layered (bipartite) graph $H = (U \cup V, E)$ such that: (i) The vertices of U are in correspondence with the groups $g1_u, \ldots, g_u^p$, and are assigned to the bottom level from right to left in this order. (ii) The vertices of V are in correspondence with the groups $g_v^1, \ldots, g_v^q$, and are assigned to the top level from right to left in this order; (iii) There is a straight-line edge $e = (g_u^j, g_v^k) \in E$ for each $j \in \{1, \ldots, p\}$ and $k \in \{1, \ldots, q\}$; this edge represents a group g that is obtained by merging g_u^j and g_v^k according to the above definitions. Group g may occur in zero or more configurations $\mathcal{A}^i_{t_u,t_v}$.

From Property 4, $H = K_{p,q} \subseteq K_{7,7}$. Figures 14(a) and 14(b) show an example of H for two specific local configurations. We represent all possible configurations $\{\mathcal{A}^i_{t_u,t_v}\}$ as all maximal subgraphs H' of H that have the following properties:

1. H' is a 2-layered planar subgraph of H.

2. Each vertex x of H' whose corresponding group must contain one upward representation has degree one.

In particular, the first condition for H' guarantees that the upward representation associated with the configuration is planar. The second condition avoids that in the same configuration there are two groups with the same first pole category or the same second pole category that cannot contain at least one upward representation at the same time. The maximality of the graph

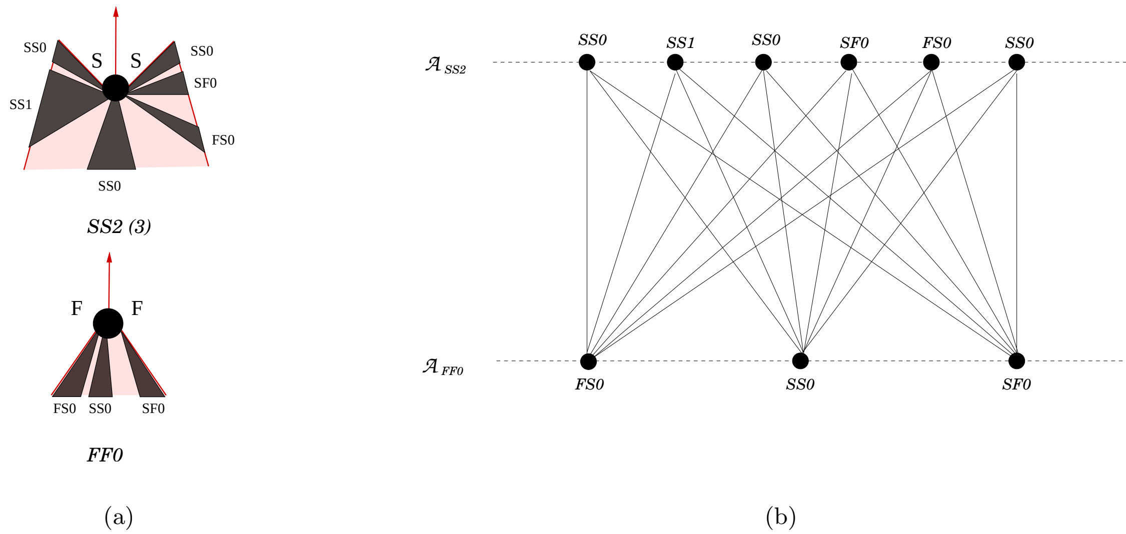


Figure 14: (a) Two local configurations; (b) Layered graph representing all the configurations that can originate from the local configurations in (a).

guarantees that every possible group is considered in the configuration; possibly, some groups will be empty, that is, we shall maintain the possibility of not assigning any upward representation to some groups of a configuration. Since $H' \subseteq H \subseteq K_{7,7}$, the following lemma holds:

Lemma 10 *Given two poles categories t_u, t_v for an upward planar representation U_{G_μ}, the number of configurations $\mathcal{A}^1_{t_u,t_v}, \ldots, \mathcal{A}^h_{t_u,t_v}$ is a constant number with respect to the number of vertices of G.*

The whole set of configurations associated with each pair of pole categories can be generated and stored in constant time. Remark this set does not depend on the specific input digraph of the testing algorithm.

4.3.2 Computing the Feasible Sets

The algorithm constructs the feasible set of μ as follows. For each configuration $\mathcal{A}^i_{t_u,t_v}$ and for each value of spirality $\sigma(U_{G_\mu})$, it checks if it is possible to select a set $U_{G_{\mu_1}}, \ldots, U_{G_{\mu_k}}$ of upward planar representations from the feasible sets of $\mu_1, \ldots, \mu_k$, and to assign each of these representations to a group of $\mathcal{A}^i_{t_u,t_v}$. To solve this assignment problem, the algorithm searches a feasible flow in a suitable network $\mathcal{N}$; each arc e of $\mathcal{N}$ has lower capacity $l(e)$ and upper capacity $u(e)$. Network $\mathcal{N}$ is defined as follows:

- $\mathcal{N}$ has the following nodes:
  - A node x_{μ_i} for each child μ_i of μ.
  - A node y_g for each group g of $\mathcal{A}^i_{t_u,t_v}$
  - A dummy source s, and a dummy sink t.

- $\mathcal{N}$ has the following arcs:
  - An arc $e = (s, x_{\mu_i})$ for each node x_{μ_i}; $l(e) = 1$ and $u(e) = 1$.
  - An arc $e = (y_g, t)$ for each node y_g; $l(e) = 1$ and $u(e) = 1$ if $t_g^u \neq SS0$ or $t_g^v \neq SS0$, while $l(e) = 0$ and $u(e) = k$, where k is the number of children of μ, otherwise.
  - An arc $e = (x_{\mu_i}, y_g)$ for each pair $\{\mu_i, g\}$ such that $\mathcal{F}_{\mu_i}$ contains a tuple τ_{μ_i} with internal spirality $\alpha(U_{G_{\mu_i}}) = \alpha(U_{G_\mu}) + \delta_g$ and pole categories $t_{u_i} = t_g^u, t_{v_i} = t_g^v$; $l(e) = 0$ and $u(e) = 1$. A flow of value 1 on such an arc e, means that the representation $U_{G_{\mu_i}}$ of τ_{μ_i} is assigned to group g.

If a feasible flow is found on $\mathcal{N}$, then a new tuple $\tau =< U_{G_\mu}, \sigma(U_{G_\mu}), t_u, t_v >$ is added to the feasible set of μ. More precisely, U_{G_μ} is constructed from the feasible flow by parallel composition of all $U_{G_{\mu_i}}$, according to the ordering of the groups in $\mathcal{A}^i_{t_u,t_v}$, and to the assignment of the upward representations to these groups.

Concerning the running time required to compute a feasible flow in $\mathcal{N}$, we observe that, $\mathcal{N}$ has $O(k)$ nodes and $O(k)$ arcs, where k is the number of children of μ. Also, the value of a feasible flow is k. By applying the Ford-Fulkerson algorithm, which performs k successive flow augmentations, the feasible flow can be computed in $O(k^2)$ time. Since we have to try a flow network for each possible configuration and for each choice of upward spirality value, the feasible set $\mathcal{F}_\mu$ of μ can be computed in $O(nk^2)$ time. Indeed, the number of configurations are $O(1)$ and the spirality values are $O(n)$. The following lemma summarizes the time complexity of computing the feasible sets for all nodes of T.

Lemma 11 *The feasible sets of all nodes of T can be computed in $O(n^3)$ time, where n is the number of vertices of the input graph G.*

Proof: The cost of computing the feasible sets of all S-nodes is $O(n^3)$. Denoted by k_μ the number of children of a P-node μ, and denoted by $\mathcal{P}$ the set of all P-nodes, the cost of computing the feasible sets of the nodes in $\mathcal{P}$ is $\sum_{\mu\in\mathcal{P}} nk_\mu^2 \leq n(\sum_{\mu\in\mathcal{P}} k_\mu)^2 \leq n^3$. □

4.4 Test and Construction of the Final Upward Representation

Once all feasible sets have been computed for the nodes of T, the algorithm performs a final step to verify if it is possible to construct an upward planar representation from the feasible set of the root of T (which is a Q-node) and the one of its child. Namely, let μ be the root and let ν be its child. The following lemma holds.

Lemma 12 *G admits an upward planar representation U_G if and only if there exist two tuples $\tau_\mu =< U_{G_\mu}, \sigma(U_{G_\mu}), t_{u_\mu}, t_{v_\mu} >\in \mathcal{F}_\mu$, $\tau_\nu =< U_{G_\nu}, \sigma(U_{G_\nu}), t_{u_\nu}, t_{v_\nu} >\in \mathcal{F}_\nu$ such that:*

1. $\sigma(U_{G_\mu}) - \sigma(U_{G_\nu}) = 2$
2. $Y_{u_\mu} = X_{u_\nu}$, $Y_{v_\mu} = X_{v_\nu}$, *where* $t_w = X_w Y_w \lambda_w$ *and* $w \in \{u_\mu, u_\nu, v_\mu, v_\nu\}$.

Proof: Suppose given two tuples $\tau_\mu =< U_{G_\mu}, \sigma(U_{G_\mu}), t_{u_\mu}, t_{v_\mu} >\in \mathcal{F}_\mu$ and $\tau_\nu =< U_{G_\nu}, \sigma(U_{G_\nu}), t_{u_\nu}, t_{v_\nu} >\in \mathcal{F}_\nu$, satisfying Conditions 1 and 2. We construct a labeled embedded digraph U_G by attaching U_{G_μ} and U_{G_ν} on poles u and v. The labeling of the angles around the poles u and v of U_G is the common labeling of U_{G_μ} and U_{G_ν} (see Condition 2). We prove that U_G is upward planar. Properties `UP1` and `UP2` of Lemma 1 are immediately satisfied by each vertex of U_G distinct from u and v, since these vertices have the same labeling that the one in U_{G_μ} and U_{G_ν}. Properties `UP1` and `UP2` for u and v are implied by Condition 2 and by the fact that U_{G_μ} and U_{G_ν} are upward representations. Moreover, property `UP3` of Lemma 1 immediately holds for each face of U_{G_ν} distinct from the right face f_r and the left face f_l of U_{G_ν} (f_r coincides with the external face of U_G and f_l coincides with the internal face of U_G that contains edge (u, v)). In order to prove that Property `UP3` also holds for f_r, we need some further considerations. Denote by $G_{\mu'} = (v_{\mu'}, u_{\mu'})$, with $v_{\mu'} = u_\mu$ and $u_{\mu'} = v_\mu$, the pertinent digraph obtained by reversing the orientation of the poles of G_μ. Notice that, from the cases illustrated in Figure 9, $\sigma(U_{G_\mu}) = -\sigma(U_{G_{\mu'}})$. By computing the sum of the upward spiralities of $U_{G_{\mu'}}$ and U_{G_ν} according by Formula 2, we obtain that $\sigma(U_{G_\mu})+\sigma(U_{G_\nu}) = \alpha_u(U_{G_{\mu'}})+\alpha_v(U_{G_{\mu'}})+\alpha(U_{G_\nu})+\alpha_u(U_{G_\nu})+\alpha_v(U_{G_\nu})$ (in which we have omitted $\alpha(U_{G_{\mu'}})$ since it is zero). Further, observe that, regardless of their actual composition, locally at each of the poles u and v, $G_{\mu'}$ and G_ν can be regarded as a series composition. Hence, from proof of Lemma 7 and with analogous significance of terms, it follows that $\alpha_u(U_{G_{\mu'}}) + \alpha_v(U_{G_\nu}) = a_u$ and $\alpha_u(U_{G_\nu})+\alpha_v(U_{G_{\mu'}}) = a_v$. Therefore, $\sigma(U_{G_{\mu'}})+\sigma(U_{G_\nu}) = \alpha(U_{G_\nu})+a_u+a_v$, that is, $\sigma(U_{G_{\mu'}})+\sigma(U_{G_\nu})$ is equal to the turn number obtained by walking on f_r counterclockwise. From Condition 1, this turn number is equal to -2, that is, Property `UP3` holds for the external face f_r. The proof of Property `UP3` for f_l is done with a symmetric reasoning.

Conversely, let U_G be an upward planar representation of G with edge (u, v) on the external face. From Lemma 9, there exist two tuples $\tau_\mu =< U_{G_\mu}, \sigma(U_{G_\mu}), t_{u_\mu}, t_{v_\mu} >\in \mathcal{F}_\mu$ and $\tau_\nu =< U_{G_\nu}, \sigma(U_{G_\nu}), t_{u_\nu}, t_{v_\nu} >\in \mathcal{F}_\nu$, where U_{G_μ} and U_{G_ν} are the upward representations of G_μ and G_ν, respectively, induced by U_G. So, Condition 2 immediately follows by applying properties `UP1` and `UP2` of Lemma 1 to the poles u and v, while Condition 1 is implied by Property `UP3` on the external face of U_G. □

According to Lemma 12, the algorithm looks for two tuples that verifies the conditions 1 and 2 in the statement. If these tuples are found, the final upward planar representation is returned, otherwise the upward planarity testing fails.

The next theorem summarizes the main result of this section. The final time complexity of the testing algorithm follows from the above discussion, iterating over all possible $SPQR$-trees of G (one for each possible choice of the reference edge).

Theorem 4 *Let G be a biconnected series-parallel digraph with n vertices. There exists an $O(n^4)$-time algorithm that tests if G is upward planar; in positive case, the algorithm constructs an upward planar representation of G in the same running time.*

4.5 Speed-up of the Testing Algorithm and Running Example

In this section we wish to develop and discuss a complete running example of the described testing algorithm, by showing how actually applying the procedures given so far, for the determination of the feasible set of each pertinent digraph and the construction of the whole upward planar representation. We consider the input digraph shown in Figure 15.

Before getting started with it, however, we wish to provide some further considerations that allow us to speed-up the algorithm complexity by a constant factor. Indeed, the practical complexity of the algorithm can be simplified by applying some observations that make it possible to reduce

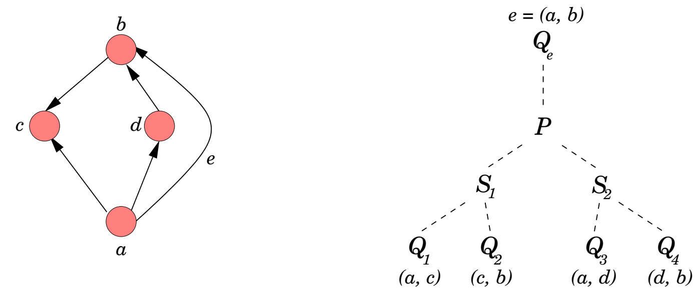


Figure 15: The input digraph of the running example and its $SPQR$-tree with e as reference edge.

the size of the feasible sets in many cases. To this aim, we use three different types of arguments, which are described in the next subsections.

4.5.1 Edges incident on the poles

We can easily reduce the size of the feasible set of a node μ by taking into account the orientation and the number of the edges incident on the poles of μ. Indeed, every pole category is compatible only with certain orientations of the edges incident on the poles of μ. This implies that, looking at these edges, we can avoid to consider some pole categories. For example, consider the feasible set of the Q-node Q_1 in Figure 15. Looking at the orientation of the edges incident on a and c, and looking at the number of such edges, we conclude that pole a can only belong to the categories $SS0, SL0$ and $LS0$, while pole c can only belong to the categories $SL0$ and $LS0$.

4.5.2 Minimum Number of Children per Configuration

The second argument is used to simplify the feasible sets of the P-nodes, just looking at their number of children. In fact, some configurations for a P-node μ require a minimum number of children for μ; if μ has not such a number of children, these configurations can be discarded. For example, consider the P-node in Figure 15. Using the argument of Subsection 4.5.1, we can only consider the pole categories $t_a \in \{LS0, SL0\}$ and $t_b =\in \{SF1, FS1\}$. As showed in Figure 13, for the second pole b, we should consider four local configurations for each of its pole categories. However, the P-node has only two children and then, from Figure 13, the only local configurations that we need to consider for it are $SF1(3), SF1(4), FS1(3)$ and $FS1(4)$. All the others, indeed, require at least two children. Moreover, $SF1(4)$ and $FS1(4)$ are not compatible with the orientations of the edges for the children S_1 and S_2 of P, so we can exclude them as well. Finally, we can also observe that, since $SF1(3)$ and $FS1(3)$ have three groups each, we can directly omit the group $SS0$ for both, because no children will be ever associated with it; this allows a simplification of the flow problems to solve.

4.5.3 Odd and Even Internal Spiralities

The third argument is based on an additional relationship between the pole categories and the internal spirality values of a pertinent digraph G_μ with poles u and v. Let $P_{uv} = < u \equiv w_1, e_1, w_2, e_2, \ldots, w_{k-1}, e_{k-1}, w_k \equiv v >$ be any (undirected) simple path from u to v in G_μ. The following property is straightforward.

Property 5 *If both e_1 and e_{k-1} enter u and v, or both e_1 and e_{k-1} leave u and v, then P_{uv} has an odd number of switches. Otherwise it has an even number of switches.*

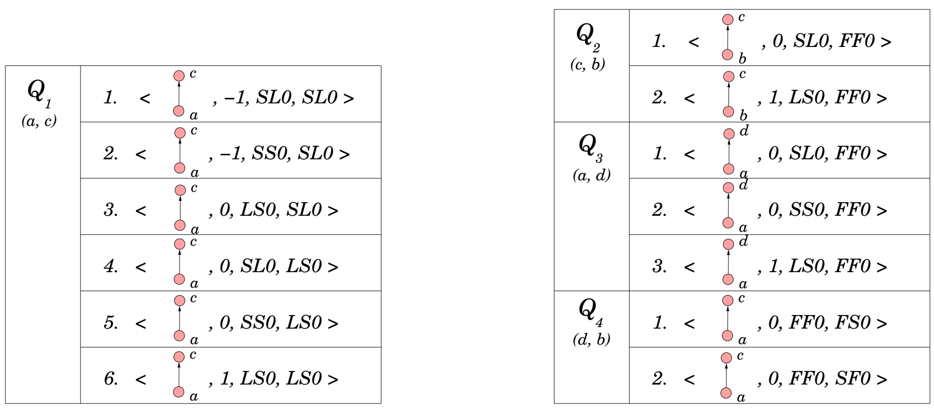
By applying the above simple property specifically to the right path P^r_{uv} of U_{G_μ}, i.e. if we assume that $P_{uv} = P^r_{uv}$, the following property immediately follows:

Property 6 *If the extremal edges of the right path of U_{G_μ} are both incoming edges or both outgoing edges of the poles of G_μ, then the internal spirality of U_{G_μ} is odd, otherwise the internal spirality of U_{G_μ} is even.*

The above property allows us to immediately discard specific combinations of pole categories and internal spirality values in the computation of the feasible sets of P-nodes.

4.5.4 Running Example

With the aid of the considerations given above, we are now ready to discuss the complete running example. In particular, in the following we shall make use of the rules for the computation of the feasible sets given in Section 4.2 and 4.3. consider the input digraph in Figure 15 and its $SPQR$-tree T. As outlined before, the testing algorithm visits T from bottom to top, and equips each T's node with its feasible set.



(a) (b)

Figure 16: Feasible set of the Q-nodes of Figure 15, distinct from the root.

The feasible sets of the Q-nodes Q_1, Q_2, Q_3 and Q_4 are computed using the speed-up given in Subsection 4.5.1; the result is shown in Figure 16. The feasible sets for the S-nodes S_1 and S_2 is straightforward; Figure 17 shows the result. The computation of the feasible sets for the P-node can be done by also taking into account the speed-up of Subsection 4.5.3, concerning the internal spirality values which can be actually supported by the P-node. Figure 18 illustrates the configurations we need to consider and a corresponding flow network for each allowed value of internal spirality; Figure 19 shows the final feasible set of the P-node.

(a) (b)

Figure 17: Feasible set of the S-nodes of Figure 15.

Finally, according to the rules given in Section 4.4, a final upward representation is found, by combining the feasible set of the P-node and the feasible set of the reference edge e (showed in Figure 20). The result is depicted in Figure 21 and is obtained, in particular, by combining the sixth feasible tuple in the feasible set of the Q_e-node and the first feasible tuple in the feasible set of the P-node.

5 Extensions to General Biconnected Digraphs

In this section we extend the testing algorithm described in Section 4 in order to design a Fixed Parameter Tractable upward planarity testing algorithm for general biconnected planar digraphs. To do that we describe how to compute the feasible set of an R-node μ. Since we cannot rely on any relationship between the upward spirality of U_{G_μ} and the upward spirality of its children, in order to test if U_{G_μ} can be constructed, we consider all possible combinations of tuples for the virtual edges of $skel(\mu)$. More precisely, for each virtual edge e_i of $skel(\mu)$ we substitute to e_i the upward planar representation $U_{G_{\mu_i}}$ of a tuple in the feasible set of μ_i, where μ_i is the child of μ represented by e_i. In this way, a "partial candidate" representation U'_{G_μ} of G_μ is obtained, which is not necessarily upward planar. To test if U'_{G_μ} is upward planar, we apply on U'_{G_μ} the flow-based upward planarity testing algorithm proposed by Bertolazzi et al. [1], where the assignment of the switches to the faces is constrained for the part of the representation that is already fixed. In order to construct the feasible set of μ, we need to run the testing algorithm over all possible combinations of upward spirality and pole categories of U_{G_μ}. For each given value of upward spirality σ and for each choice of pole categories t_u, t_v, we enrich the partial representation U'_{G_μ} with an *external gadget*, that will force U_{G_μ} (in case it exists) to have upward spirality σ and pole categories t_u, t_v. This external gadget will have a fixed upward planar representation, which is still translated in a set of constraints on the flow network. To have a better intuition of the role of the external gadget, consider the component of a digraph shown in Figure 22(a) and suppose that we want to test if this component admits an upward planar representation with upward spirality 1 and pole categories $t_u = LS0, t_v = SS0$. We can attach to it a gadget like that shown in Figure 22(b) and we

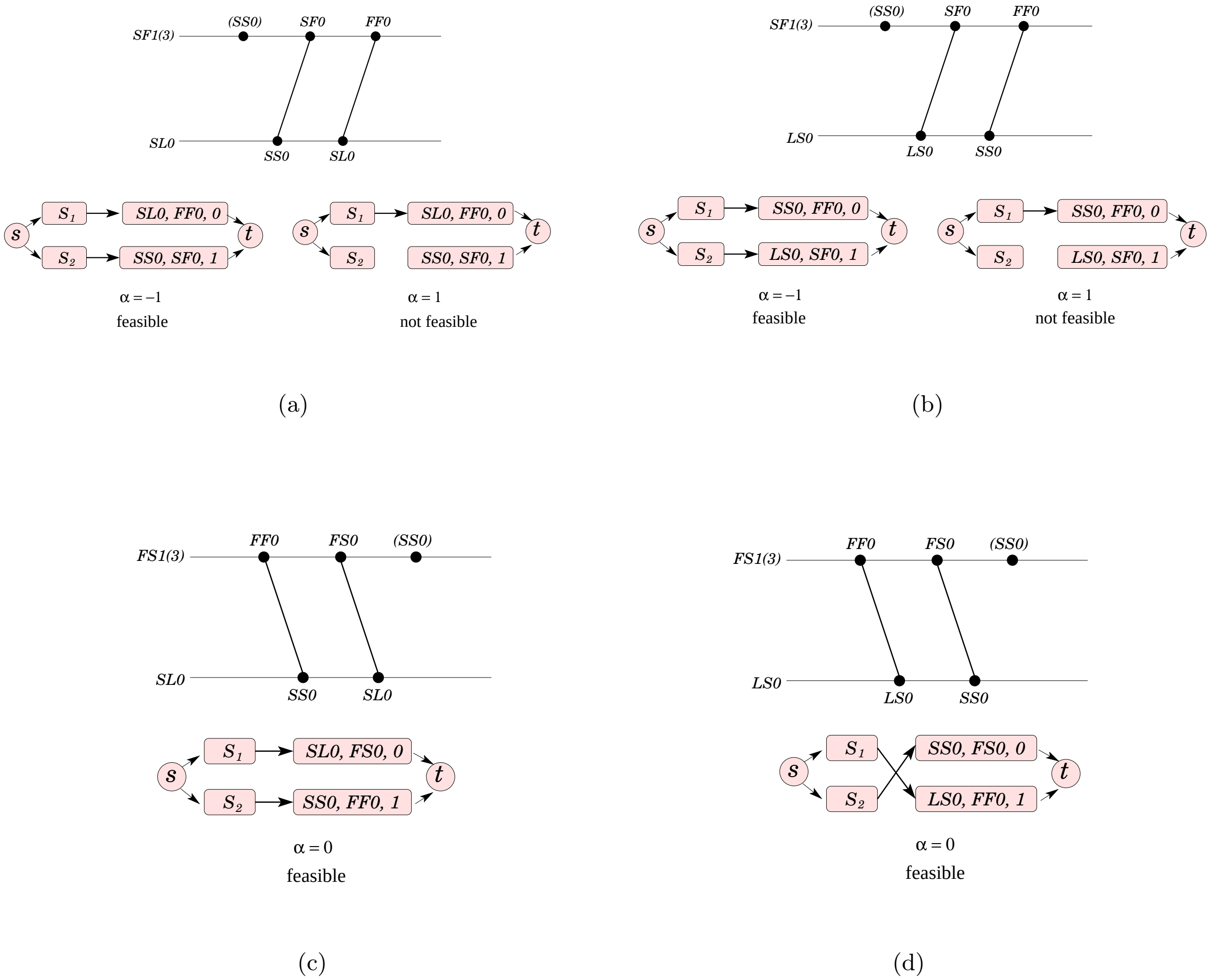


Figure 18: Computing the feasible set for the P-node of Figure 15.

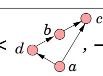
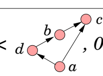
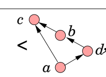
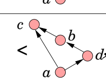
P	1.	<  , −1, SL0, SF1 >
	2.	<  , 0, LS0, SF1 >
	3.	<  , 1, SL0, FS1 >
	4.	<  , 2, LS0, FS1 >

Figure 19: The feasible set for the P-node of Figure 15.

Figure 20: The feasible set of the root Q_e.

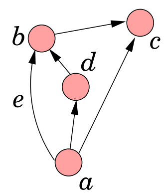
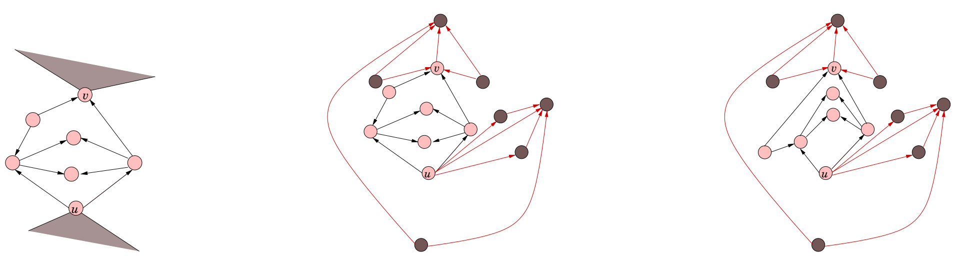


Figure 21: A final upward planar representation for the digraph showed in Figure 15.

can perform the flow-based upward planarity testing in such a way that the upward representation of the gadget is fixed. Figure 22(c) shows an example of an upward planar representation of the component within the given constraints.



(a) (b) (c)

Figure 22: Example of external gadget.

In the next subsection we explain in details how to define the external gadget of an R-node and its upward planar representation, depending on the choice of σ, t_u, and t_v.

5.1 Computing External Gadgets for R-nodes

Every external gadget for an R-node μ has the same undirected general structure, depicted in Figure 23(a). It is formed by two *pole connectors*, which will be directly attached to the poles of G_μ, and by a chain of vertices called *spirality backbone.* The number of vertices of the spirality backbone and the orientation of the edges of the gadget are defined depending on the upward spirality σ and on the pole categories t_u, t_v that we want for U_{G_μ}.

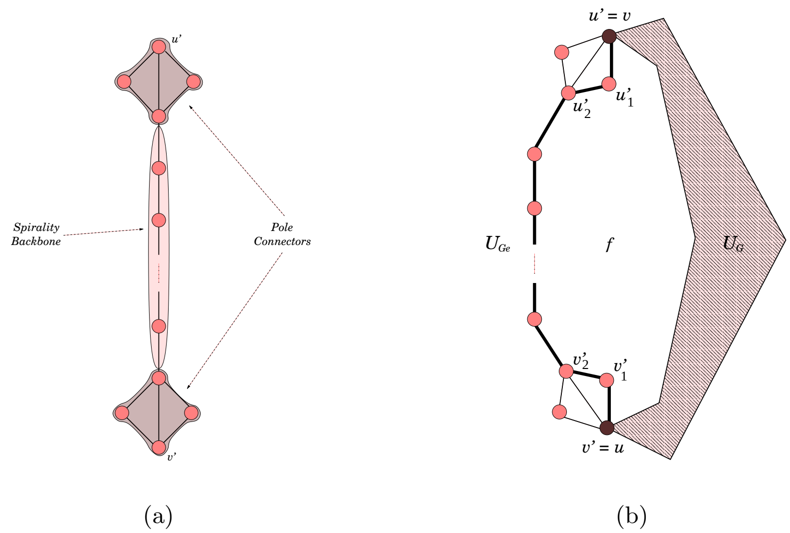


Figure 23: (a) General structure of an external gadget for a rigid component. (b) How the gadget is attached to the rigid component.

To understand how to complete the definition of the external gadget and how to fix an upward representation for it on the basis of σ, t_u, and t_v, we need some further considerations (refer to Figure 23(b)). Denote by G_e the final external gadget of G_μ and suppose that U_{G_e} is an upward planar representation of G_e that forces U_{G_μ} to have upward spirality σ and pole categories t_u, t_v. The pole connectors of G_e are oriented and embedded upward in U_{G_e} on the basis of t_u, t_v, choosing one suitable representation among those shown in Figure 24. Indeed, these representations are enough to cover all the cases for t_u, t_v.

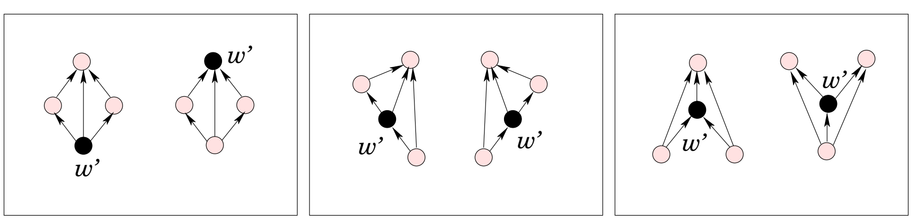


Figure 24: Upward planar representations for a pole connector; the dark vertex w' denotes either u' or v'.

Also, denote by v', u' the first pole and the second pole of G_e, respectively; u' will coincide with the second pole v of G_μ and v' will coincide with the first pole u of G_μ. Using Formula 2, we have: $\sigma(U_G)+\sigma(U_{G_e}) = \alpha(U_G)+\alpha_u(U_G)+\alpha_v(U_G)+\alpha(U_{G_e})+\alpha_u(U_{G_e})+\alpha_v(U_{G_e})$. With the same technique used in the proof of Lemma 12, we can rewrite $\sigma(U_G) + \sigma(U_{G_e}) = \alpha(U_G) + \alpha(U_{G_e}) + a_u + a_v = -2$, and then the internal spirality of U_{G_e} is $\alpha(U_{G_e}) = -(\alpha(U_G) + a_u + a_v + 2)$.

Consider the path P from u' to v' within face f shown in Figure 23(b). In order to construct G_e and U_{G_e} in such a way that U_{G_e} has the internal spirality $\alpha(U_{G_e})$ obtained above, we define $d = |\alpha(U_{G_e})|$ switch angles along P that will be either all left or all right turns in U_{G_e}, depending on the sign of $\alpha(U_{G_e})$. In particular, we want that every vertex of the spirality backbone and also the vertices u'_2, v'_2 determine switch angles inside f. From the upward representation previously decided for the pole connectors and from this last goal, the number of vertices of the spirality backbone and the upward planar representation of the whole gadget remains completely fixed.

We remark that, the construction of U_{G_e} above described can fail when Property 6 cannot be satisfied. If this happens, it means that an upward planar representation U_{G_μ} with the desired spirality σ and pole categories t_u, t_v cannot be constructed.

We conclude this subsection with an example that illustrates the construction of an external gadget and of its upward planar representation. Suppose that for the rigid component G_μ of Figure 25(a) we want to test if there is an upward planar representation having pole categories $t_u = SS2$ and $t_v = LS0$. According to Property 6, U_{G_μ} can only assume even internal spirality values. Suppose we want for it an internal spirality 2. We have that: (i) $t_u \Rightarrow t_{v'} = SS0 \Rightarrow a_u = 1$, and $t_v \Rightarrow t_{u'} = LS0 \Rightarrow a_v = -1$; (ii) $d = |2 + 1 - 1 + 2| = 4$; (iii) $\alpha(U_{G_e}) = -4$. The resulting upward representations U_{G_e} and U_{G_μ} are shown in Figure 25(b).

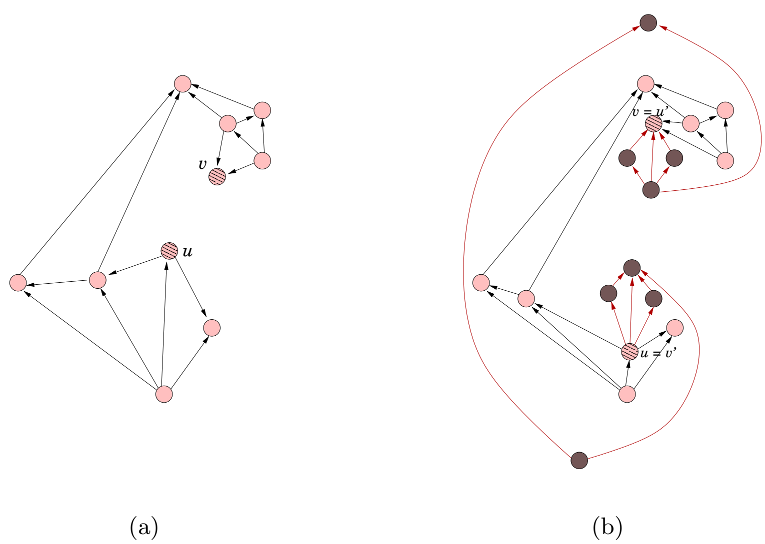


Figure 25: (a) A pertinent digraph G_μ of an R-node; (b) an upward planar representation U_{G_μ} of G_μ, along with the upward planar representation of an external gadget. The representation of the gadget is defined so to force U_{G_μ} to have pole categories $t_u = SS2$ and $t_v = LS0$, and internal spirality 2.

6 Extensions to General Digraphs

Our FPT algorithm can be extended to general planar digraphs, using a recent result of Healy and Lynch [10, 11]. Without loss of generality, assume that G is connected, otherwise the upward planarity testing can be performed independently on every connected component. Healy and Lynch describe a decomposition strategy that allows the blocks of G to be tested separately with certain extra conditions applied to them, and they show how the conditions for each block can be deduced in $O(n^2)$ time.

Denote by B a block of G. According to our notation, each condition on B can take one of the following forms:

1. B is upward planar;
2. B is upward planar with a given vertex on the external face;
3. B is upward planar with a given vertex v on the external face and external large angle at v.
4. B is upward planar with a given vertex v on the external face and external flat angle at v.
5. B is upward planar with a given vertex v on the external face and with an external edge that enters v.
6. B is upward planar with a given vertex v on the external face and with an external edge that leaves v.

Healy and Lynch [12] describe an $O(2^t \cdot t! \cdot n2)$-time FPT upward planarity testing algorithm that relies on their decomposition strategy and on a tailored version of Bertolazzi et al.'s algorithm for testing the above conditions on each block. We follow the same approach, but we use a constrained version of our FPT algorithm for testing the upward planarity conditions on each block, instead of relying on Bertolazzi et al.'s algorithm. In this way we can avoid to explore all the planar embeddings of each block.

Our upward planarity testing algorithm can be slightly adapted to take into account the constraints in each of the above conditions. More precisely, concerning the second constraint, we have just to restrict our investigation to those $SPQR$-tree T that are rooted at an edge incident on v. This is also sufficient for the last two constraints. Concerning the third constraint, the modification we introduce on the algorithm only concerns the final step, that is the construction of the final upward planar representation (see Section 4.4). More precisely, let μ_1 be the root of T, and let μ_2 be its child. Let u_1, v_1 be the two poles of G_{μ_1} and let $u_2 = u_1, v_2 = v_1$ be the two poles of G_{μ_2}. Suppose that v_1 is the vertex around which we wish to have an external L label. The algorithms looks for two tuples $\tau_{\mu_1} =< U_{G_{\mu_1}}, \sigma(U_{G_{\mu_1}}), t_{u_1}, t_{v_1} >\in \mathcal{F}_{\mu_1}$, $\tau_{\mu_2} =< U_{G_{\mu_2}}, \sigma(U_{G_{\mu_2}}), t_{u_2}, t_{v_2} >\in \mathcal{F}_{\mu_2}$, such that: (i) $\sigma(U_{G_{\mu_1}}) - \sigma(U_{G_{\mu_2}}) = 2$, (ii) $Y_{u_1} = X_{u_2}, Y_{v_1} = X_{v_2}$, where $t_{u_i} = X_{u_i} Y_{u_i} \lambda_{u_i}$ and $t_{v_i} = X_{v_i} Y_{v_i} \lambda_{v_i}$ $(i = \{1, 2\})$, and (iii) $t_{u_1} \in \{FF0, SL0, LS0\}, t_{v_1} \in \{SL0, LS0\}$. An analogous modification applies for the fourth constraint.

6.1 Time Complexity

The feasible set of an R-node μ, computed with the above procedure, requires to consider all possible combinations of tuples in the feasible set of the children of μ, and, for each of these combinations, we need to consider all possible values of upward spirality and pole categories. The procedure must be also applied to the two possible planar embeddings of $skel(\mu)$. Denote by t the number of non-trivial triconnected components of G and denote by d the maximum diameter of a split component of G. The feasible set of an R-node of μ can be then computed in $O(d^{t_\mu} \cdot n^2)$ time, where n is the number of vertices of G, and $t_\mu \leq t$ is the number of virtual edges (distinct from the reference edge) of μ. Indeed, the minimum number of switches in any path between the poles of G_μ is at most d, and therefore, from Lemma 5, the upward spirality of U_{G_μ} can take $O(d)$ possible values and the feasible set of any node of T has $O(d)$ tuples. Also, $O(n^2)$ is the complexity of the upward planarity testing of Bertolazzi et al. Hence, the feasible set of all R-nodes can be computed in $O(d^t \cdot n^2)$ time.

Concerning the extension to general digraphs G, since the sum of the number of vertices of all blocks of G is $O(n)$, where n is the number of vertices of G, the time complexity of our FPT testing algorithm does not change. We want to remark that, if each block of G is a series-parallel digraph, we are still able to test G for upward planarity in polynomial time. Therefore, the following two theorems hold.

Theorem 5 *Let G be a connected planar digraph with n vertices. Suppose that each block of G has at most t (non-trivial) triconnected components, and that each split component of a block has a diameter at most d. There exists an $O(d^t \cdot n^3 + d \cdot t^2 \cdot n + d^2 \cdot n^2)$-time algorithm that tests if G is upward planar and, if so, that constructs an upward planar drawing of G.*

Theorem 6 *Let G be a connected planar digraph with n vertices and such that each block is a series-parallel digraph. There exists an $O(n^4)$-time algorithm that tests if G is upward planar and, if so, that constructs an upward planar drawing of G.*

7 Open Problems

Several questions are naturally raised by the results in this paper. We report here three of those that in our opinion are among the most interesting.

- Use upward spirality to design a polynomial-time algorithm to compute a quasi-upward planar drawing of a series-parallel digraph with the minimum number of bends. For a definition of quasi-upward planar drawings see [2].
- Find an FPT algorithm whose only parameter is the number of R-nodes in the decomposition tree of the input digraph.
- Find a combinatorial characterization of upward planarity of (series-parallel) digraphs.

Acknowledgements

We thank Giuseppe Di Battista for the useful discussions on the topic of this paper.

References

[1] P. Bertolazzi, G. D. Battista, G. Liotta, and C. Mannino. Upward drawings of triconnected digraphs. *Algorithmica*, 6(12):476–497, 1994.

[2] P. Bertolazzi, G. Di Battista, and W. Didimo. Quasi-upward planarity. *Algorithmica*, 32(3):474–506, 2002.

[3] P. Bertolazzi, G. Di Battista, C. Mannino, and R. Tamassia. Optimal upward planarity testing of single-source digraphs. *SIAM Journal on Computing*, 27:132–169, 1998.

[4] H. Chan. A parameterized algorithm for upward planarity testing. In *Annual European Symposium on Algorithms (Proc. ESA '04)*, volume 3221 of *Lecture Notes Comput. Sci.*, pages 157–168, 2004.

[5] G. Di Battista, P. Eades, R. Tamassia, and I. G. Tollis. *Graph Drawing*. Prentice Hall, Upper Saddle River, NJ, 1999.

[6] G. Di Battista, G. Liotta, and F. Vargiu. Spirality and optimal orthogonal drawings. *SIAM Journal on Computing*, 27(6):275–298, 1998.

[7] G. Di Battista and R. Tamassia. On-line planarity testing. *SIAM J. Comput.*, 25:956–997, 1996.

[8] W. Didimo and M. Pizzonia. Upward embeddings and orientations of undirected planar graphs. *Journal of Graph Algorithms and Applications*, 7(2):221–241, 2003.

[9] A. Garg and R. Tamassia. On the computational complexity of upward and rectilinear planarity testing. *SIAM Journal on Computing*, 31(2):601–625, 2001.

[10] P. Healy and K. Lynch. Building blocks of upward planar digraphs. In *Graph Drawing (Proc. GD '04)*, volume 3383 of *Lecture Notes Comput. Sci.*, pages 296–306, 2004.

[11] P. Healy and K. Lynch. Building blocks of upward planar digraphs. Technical report, 2005. TR UL-CSIS-05-2.

[12] P. Healy and K. Lynch. Fixed-parameter tractable algorithms for testing upward planarity. In *Conference on Current Trends in Theory and Practice of Computer Science (Proc. SOFSEM '05)*, volume 3381 of *Lecture Notes Comput. Sci.*, pages 199–208, 2005.

[13] M. D. Hutton and A. Lubiw. Upward planarity testing of single-source acyclic digraphs. *SIAM Journal on Computing*, 25(2):291–311, 1996.

[14] M. Kaufmann and D. Wagner. *Drawing Graphs*. Springer Verlag, 2001.

[15] A. Papakostas. Upward planarity testing of outerplanar dags. In *Graph Drawing (Proc. GD '95)*, volume 894 of *Lecture Notes Comput. Sci.*, pages 7298–306, 1995.