Orthogonal Drawings with the Minimum Number of Bends *

(Extended Abstract)

Giuseppe Liotta[†] Francesco Vargiu[‡] Giuseppe Di Battista[§]

Abstract

We deal with the classical problem of constructing an orthogonal drawing of a 4-planar graph with the minimum number of bends along the edges. Recently, Garg and Tamassia proved the NP-completeness of the problem. The main result of the present paper is an algorithm that solves the problem for a biconnected 4-planar graph with n vertices in $O(n^k \cdot p(n))$ time, where k is the number of vertices of degree 4 and $p(n)$ is a polynomial.

1 Introduction and Overview

An *orthogonal drawing* of a graph is a planar drawing such that all the edges are polygonal chains of horizontal and vertical segments.

We deal with the classical problem of constructing orthogonal drawings with the minimum number of bends along the edges [12]. Tamassia [8] proposed an elegant algorithm that solves the problem in polynomial time for graphs with a fixed planar embedding. The algorithm is based on a combinatorial characterization that allows to map the problem into a min-cost flow one. Linear time heuristics have been proposed by Tamassia and Tollis in [9]. Such heuristics guarantee at most $2n + 4$ bends for a biconnected graph with n vertices. Recently, Kant [6] has proposed efficient heuristics with better bounds for triconnected 4-planar graphs and general 3-planar graphs (a graph is k-planar if it is planar and each vertex has degree at most k). Observe that orthogonal drawings make sense only for 4-planar graphs. Tamassia, Tollis and Vitter [10, 11] have given lower bounds and the first parallel algorithm. An annotated bibliography on graph drawing can be found in [3].

However, all the above papers work within a fixed planar embedding and the solutions obtained within a fixed embedding can be far from the optimum [2]. Hence, recently the variable-embedding version of the problem has been intensively investigated. The problem of finding the planar embedding that leads to the orthogonal drawing with the minimum number of bends has been shown to be NP-complete by Garg and Tamassia [5]. In the same paper it is shown that it is NP-hard to approximate the minimum number of bends in an orthogonal drawing of an n-vertex planar graph with an $O(n^{1-\epsilon})$ error for any positive ϵ. In [2] a polynomial time algorithm for non-fixed embedding 4-planar series-parallel graphs and biconnected 3-planar graphs is presented.

The contributions of the present paper are the following:

- We present an algorithm that receives as input an n-vertex biconnected 4-planar graph with k vertices of degree 4 and constructs an orthogonal representation of the input graph with the minimum number of bends in $O(n^k \cdot p(n))$ time, where $p(n)$ is a polynomial.
- We improve the time complexity of the algorithm presented in [2] for the special case that the input graph is a 3-planar series-parallel graph.
- We investigate the variation of the number of bends of an orthogonal drawing when it is "rolled up".

The paper is organized as follows. Preliminaries are in Section 2. The variation of the number of bends of an orthogonal drawing obtained when "rolling-up" an orthogonal drawing is studied in Section 3. The algorithms are given in Section 4. Omitted proofs can be found in the forthcoming full paper.

2 Preliminaries

We assume familiarity with connectivity and planarity [4]. An *embedded graph* is a planar graph with a given *embedding*, i.e. an ordering for the edges incident on the vertices such that there exists a planar drawing of the graph that respects the ordering. A *k-planar* graph is a planar graph whose vertices have degree at most k.

An acyclic digraph G with exactly one source s, exactly one sink t and with the edge (s,t) is an *st-digraph.* A *split pair* of G is either a separation pair of G or a pair of adjacent vertices. A *split component* of a split pair $\{u, w\}$ is either an edge (u, w) or a maximal subgraph G_{uw} of G such that $\{u, w\}$ is not a split pair of G_{uw}; u and w are the *poles* of the split component. An *st-orientation* of a graph G is

*Work partially supported by Progetto Finalizzato Sistemi Informatici e Calcolo Parallelo of the Italian National Research Council (CNR) and by Esprit BRA of the EC Under Contract 7141 Alcom II. All the authors are with the Dipartimento di Informatica e Sistemistica, Università di Roma "La Sapienza", via Salaria 113, I-00198 Roma, Italia.

[†]liotta@dis.uniroma1.it

[‡]vargiu@dis.uniroma1.it

[§]dibattista@iasi.rm.cnr.it

an orientation of the edges of G into an st-digraph (each biconnected graph can be oriented into an st-digraph [4]).

A *SPQ^*R tree* [1] T of G is a rooted ordered tree describing a recursive decomposition of G with respect to its split pairs, and it will be used to represent all the embeddings of G with s and t on the external face. Nodes of T are of 4 types: S, P, Q^* and R. Each node μ has an associated acyclic digraph called *skeleton* of μ denoted by $skeleton(\mu)$. Non-root nodes of T are called *internal nodes*. Tree T is recursively defined as follows:

Chain case: If G consists of a simple path from s to t then T is a single Q^*-node μ whose skeleton is G itself.

Series case: If G is 1-connected, let $c_1,\ldots, c_{k-1}$ ($k \geq 2$) be the cutvertices of G such that no cutvertex has degree less than three; $c_1,\ldots, c_{k-1}$ partition G into its blocks $G_1,\ldots, G_k$ in this order from s to t. The root of T is an S-node μ. Graph $skeleton(\mu)$ is the chain $e_1,\ldots,e_k$, where edge e_i goes from c_{i-1} to c_i, $c_0 = s$ and $c_k = t$.

Parallel case: If s and t are a split pair for G with split components $G_1,\ldots, G_k$ ($k \geq 2$), encountered in this order when going around s in clockwise order, the root of T is a P-node μ. Graph $skeleton(\mu)$ consists of k parallel edges from s to t, denoted $e_1, \ldots, e_k$.

Rigid case: If none of the above cases applies, let $\{s_1,t_1\}$,...,$\{s_k,t_k\}$ be the maximal split pairs of G ($k \geq 1$), and for $i = 1..k$, let G_i be the union of all the split components of $\{s_i,t_i\}$. The root of T is an R-node μ. Graph $skeleton(\mu)$ is obtained from G by replacing each subgraph G_i with edge e_i from s_i to t_i.

We call *pertinent graph* of μ the graph whose decomposition tree is the subtree rooted at μ. Observe that the skeleton of an R-node has only two possible embeddings with the poles on the external face, corresponding to the two possible flippings of the component around the poles. We refer to such possible embeddings as *embedding A* and *embedding B*.

Property 1 *Let T be the SPQ^*R tree of a 4-planar graph and let μ be a node of T with poles u and w. All vertices of $skeleton(\mu)$, except eventually u and w, have degree at least* 3.

Let G be an st-digraph, it is always possible to define an SPQ^*R tree T of G such that: (1) The root of T is a P-node with two children; one of them is a $Q*$-node representing edge (s,t); (2) Each internal P-node has children R, S, or Q^*-nodes; (3) Each S-node has two children; (4) Each Q^*-node has children edges of G. Such a tree is called *decomposition tree* of G.

Let G be an st-digraph and $G_{uw} \subset G$ be a split component of G. The edges of G that are incident on a pole v ($v = u, w$) of G_{uw} and that belong (do not belong) to G_{uw} are called *internal edges* (*external edges*) of v with respect to G_{uw}; the number of such edges is called *internal degree* (*external degree*) of v with respect to G_{uw}. An *orthogonal st-digraph* is a digraph whose edges are chains of horizontal and vertical segments. Observe that an orthogonal st-digraph is a 4-planar graph. An *optimal orthogonal representation* of an st-digraph G is an orthogonal st-digraph H that is isomorphic to G and has the minimum number of bends along the edges; H is called *optimal orthogonal st-digraph.* In the following we consider only orthogonal st-digraphs and orthogonal representations of st-digraphs such that the edge (s,t) is on the external face.

Let H be an orthogonal st-digraph and $H_{uw} \subset H$ be a split component of H. A pole v ($v = u, w$) of of H_{uw} is called *bridge pole* when its internal degree with respect to H_{uw} is one; v is called *nonbridge pole* when its internal degree with respect to H_{uw} is greater than one. The *alias vertex* of a pole v ($v = u, w$) of H_{uw} is a dummy vertex v' placed on an external edge of v with respect to of H_{uw} and such that the dummy edge (v, v') has no bends. A pole admits one or two alias vertices depending on its external degree. Let u' (w') be an alias vertex of u (w). Let P_{uw} be any undirected simple path in H_{uw} from u to w. A *spine* $S_{u'w'}$ of H_{uw} is the simple path obtained by concatenating edge (u', u), path P_{uw}, and edge (w, w'). In [2] it is shown that the number $n(S_{u'w'})$of right turns minus the number of left turns encountered along $S_{u'w'}$ when going from u' to w' does not depend on the choice of path P_{uw}.

The *spirality* $\sigma_{H_{uw}}$ of H_{uw} is defined as follows; three cases are possible, depending on the number of alias vertices of u and w. (1) Both u and w have just one alias vertex, say u' and w', respectively. Let $S_{u'w'}$ be a spine of H_{uw}; $\sigma_{H_{uw}} = n(S_{u'w'})$. (2) Pole u has just one alias vertex, say u'; w has two alias vertices, say w' and w''. Let $S_{u'w'}$ and $S_{u'w''}$ be two spines of H_{uw}; $\sigma_{H_{uw}} = (n(S_{u'w'}) + n(S_{u'w''}))/2$. (3) Pole u has two alias vertices, say u' and u''; w has two alias vertices, say w' and w''. Suppose u' and w' are on the same face of H. Let $S_{u'w'}$ and $S_{u''w''}$ be two spines of H_{uw}; $\sigma_{H_{uw}} = (n(S_{u'w'}) + n(S_{u''w''}))/2$.

Property 2 *[2] If the poles of H_{uw} satisfy condition of Case 2 of the above definition, then $2\sigma_{H_{uw}}$ is an odd integer number; else (the poles of H_{uw} satisfy either condition of Case 1 or condition of Case 3 of the above definition), then $\sigma_{H_{uw}}$ is an integer number.*

The component H_{uw} is *optimal within the spirality* $\sigma_{H'_{uw}}$ if it doesn't exist any H''_{uw} such that $\sigma_{H'_{uw}} = \sigma_{H''_{uw}}$ and such that H'_{uw} has more bends than H''_{uw}.

Theorem 1 *[2] Let H be an optimal orthogonal st-digraph with n vertices and let H_{uw} be a split component of H. Then (i) H_{uw} is optimal within its spirality; (ii) $|\sigma_{H_{uw}}| \leq 3n - 2$.*

The following lemma descends from [2].

Lemma 1 *Let μ be a node of T and let H_{uw} be its pertinent graph. If μ is an S-node, the spirality of*

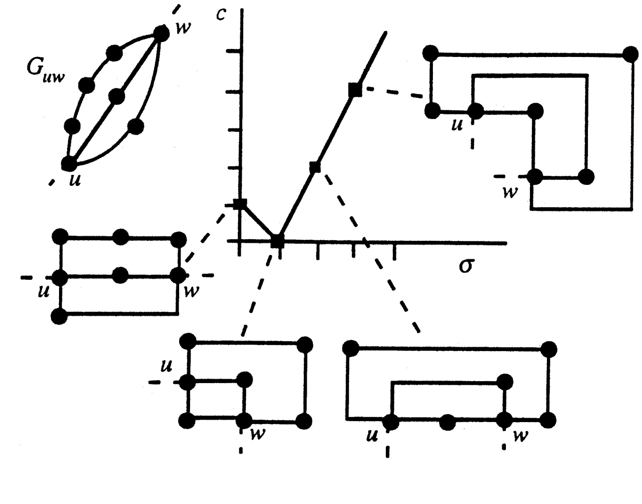

Figure 1: An example of non monotonic cost function.

H_{uw} is the sum of the spiralities of the pertinent graphs of the children of μ. If μ is a P-node, let $H'_{u'w'}$ be any of the pertinent graphs of the children of μ; $|\sigma_{H_{uw}} - \sigma_{H'_{u'w'}}| \leq 2$.

3 Spirality and Cost Functions

The *cost function* of a split component associates to each value σ of spirality the cost of an orthogonal representation of the component optimal within spirality σ. In this section we study the interplay between spirality and cost.

The following property can be trivially proved.

Property 3 *The cost function of a split component is piecewise linear and it is symmetric with respect to the cost axis.*

Intuitivelly, one would expect that the number of bends of a split component monotonically increases when the spirality of such component is augmented. In [2] the following result is proved.

Lemma 2 *[2] The cost function of a split component of a 3-planar graph is nondecreasing, piecewise linear, and convex.*

Surprisingly, a similar lemma does not hold for general 4-planar graphs. To give an example, in Fig. 1 we show the cost function of a parallel component that has a minimum for $\sigma = 1$. The following lemma shows that the behavior of a cost function can be even worse.

Lemma 3 *There exists a series-parallel split component of a 4-planar graph whose cost function is neither convex nor monotonic.*

Sketch of proof: Consider the split component of Fig. 2 □

However, the cost function of the split component of Fig. 2 has only one non-convexity. Now we show an infinite family of series-parallel split components

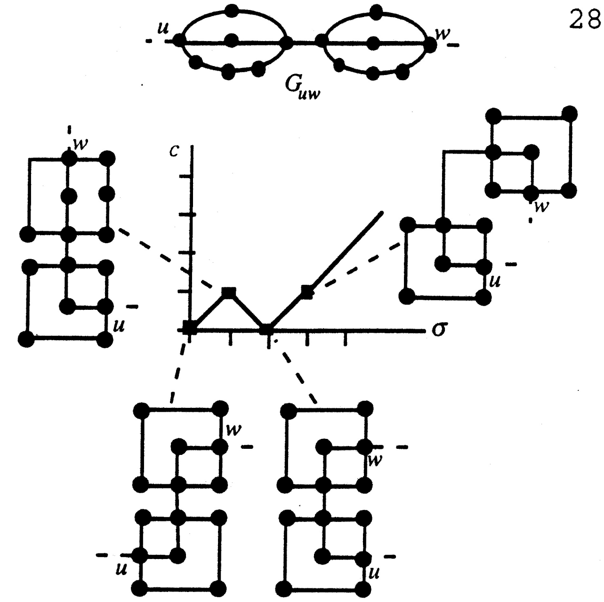

Figure 2: An example of non-convex cost function.

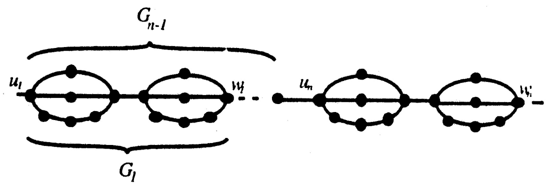

Figure 3: The series composition G_n.

whose cost functions ripple a linear number of times. Let G_n be the split component recursively defined as follows: G_1 is the split-component of Lemma 3; G_n ($n > 1$) is the series composition of G_1, one edge, and G_{n-1} (see Fig. 3). Observe that G_n has $14n$ vertices.

Lemma 4 *The cost function of G_n has value zero for $\sigma = \pm 2i$ ($i = 0, \ldots, n$) and has value different from zero for all the remaining values of σ.*

Sketch of proof: For the symmetry of the cost functions (see Property 3) we can restrict our attention to nonnegative values of spirality. The proof is by induction on n. We prove first that the cost function of G_1 has value 0 only for spirality 0 and 2.

Let G_0 be the split-component of Fig. 1. The split component G_1 is the series composition of two copies of G_0 separated by one edge e_0. The cost function of e_0 is a linear function with slope 1. The cost fuction of G_0 has value 0 only for spirality 1. Namely, for every value of spirality less than 4, the cost fuction has the behavior depicted if Fig. 1. Since any orthogonal representation of G_0 has a spine S with three vertices, for values of spirality greater or equal to 4, such orthogonal representation has at least one bend on S. From Lemma 1, the spirality of an orthogonal representation of G_1 is the sum of the spiralities of its components. It follows that an orthogonal representation of G_1 has at least one bend except when the spirality of e_0 is 0 and the spiralities of the orthogonal representations of G_0 are either 1 or -1. Thus, the cost function of G_1 has value 0 only for spirality 0 and 2.

Suppose now the lemma holds for G_{n-1}. We prove the lemma for G_n. G_n is the series of G_{n-1}, one edge e and G_1. Again, from Lemma 1, the spirality of an orthogonal representation of G_n is the sum of the spiralities of its components. It follows that an orthogonal representation of G_n has at least one bend except when the spirality of e is 0 and the spiralities of the orthogonal representations of G_{n-1} and G_1 are such that the corresponding cost functions have value 0. Thus the cost function of G_n has value 0 only for $\sigma = 2i$ $(i = 0, \ldots, n)$. □

The following theorem summarizes the results of this section.

Theorem 2 *There exists an infinite family of 4-planar split components whose cost functions are neither convex nor monotonic.*

4 Optimal Orthogonal Drawings of 4-planar Biconnected Graphs

Aim of this section is to describe a drawing algorithm that receives as input a biconnected 4-planar graph (called simply graph in the rest) G and produces as output an optimal orthogonal representation of G.

4.1 High-Level Description of the Algorithm

The basic idea of the algorithm is to incrementally construct an optimal orthogonal representation of G by composing orthogonal representations of its split components, that are optimal within given values of spirality. To do that we orient the edges of G into an st-digraph and compute a decomposition tree T of G. Also, the nodes of T are equipped with a data structure, called *optimal set*, devised to describe optimal orthogonal representations of the split components of G. Let μ be a node of T and let G_μ be the pertinent graph of μ. The optimal set of μ is a set of distinct orthogonal representations of G_μ, each one optimal within a distinct value of spirality. Since we are interested in considering only the orthogonal representations of G_μ that may appear in an optimal orthogonal representation of G, the cardinality of the optimal set of μ, according to Theorem 1, does not exceed $6n - 3$.

A high-level description of the algorithm follows.

Algorithm Optimal Orthogonal Drawing
input: graph G.
output: optimal orthogonal representation of G.
Step1: For each edge (u, v) of G:

Compute an st-orientation G' of G such that u is the source, v is the sink.

Compute a decomposition tree T of G'.

Compute the optimal sets of the nodes of T.

Define an optimal orthogonal representation of G' by composing the orthogonal representations contained in the optimal sets.

Step 2: Choose the orthogonal representation that, among the computed ones, has the minimum number of bends.
end Algorithm.

The computation of the optimal sets of Q^*-, S- and P-nodes can be performed in polynomial time exploiting Lemma 1. Furthermore, in [2] it is described how to compute in polynomial time the optimal set of an R-node whose pertinent graph has vertices with degree at most three. In this paper the problem of computing the optimal set of an R-node whose pertinent graph G_μ has vertices of degree 4 is mapped to a min-cost flow problem on a network where the cost functions for the arcs of the network are the cost functions of the split components of G_μ.

The flow network is described in Subsection 4.2. Clearly, if all the cost functions of the arcs in the network were convex, the min-cost flow could be computed in polynomial time [7]. In Subsection 4.3 we show that the min-cost flow problem on the network associated to G_μ can be solved in $O(n^k \cdot p(n))$, where k is the number of vertices of degree 4 and $p(n)$ is a polynomial of the number n of vertices of G. In the same section we discuss the overall time complexity of the algorithm.

4.2 The flow network model

Let T be a decomposition tree of an st-digraph, let μ be an R-node of T with pertinent graph G_μ and poles u, w and let *embedding* A and *embedding* B for *skeleton*(μ) be given.

Let *skeleton*$_A(\mu)$ (*skeleton*$_B(\mu)$) be the embedded graph obtained by adding (u, w) to *skeleton*(μ) with *embedding* A (*embedding* B) and such that: (1) (u, w) is on the external face, and (2) when going along (u, w) from u to w, the external face is left to the right.

We associate to μ, *embedding* A, and a given value of spirality σ a network $N_\mu(\sigma, A)$ defined as follows (the definition of a network $N_\mu(\sigma, B)$ associated to μ, σ and *embedding* B is analogous).

- Nodes: there is a node (*vertex-node*) for each vertex of *skeleton*$_A(\mu)$; there is a node (*face-node*) for each face of *skeleton*$_A(\mu)$; and there are two extra nodes: a source node s and a sink node t.
- Arcs:
  - There are two arcs (f', f'') and (f'', f') for each pair of faces sharing an edge e. The arcs have capacity, lower bound, and cost assigned with the following rule. Arcs

(f', f'') and (f'', f') have infinite capacity, lower bound zero, and cost defined by the cost function of the corresponding split component of G_μ with poles the endpoints of e.

- There is one arc (v, f) where v is a vertex that belongs to face f. Arc (v, f) has infinite capacity and zero lower bound and cost.
- There is one arc (s, f) for each internal face f composed by less than 4 edges. Arc (s, f) has capacity 4 minus the number of edges belonging to the face and zero lower bound and cost.
- There is one arc (s, v) for each vertex v. Arc (s, v) has capacity equal to $4 - deg(v)$ and zero lower bound and cost.
- There is one arc (f, t) for each internal face f composed by more than 4 edges. Arc (f, t) has capacity equal to the number of edges belonging to the face minus 4 and zero lower bound and cost.
- There is one arc (f_e, t) for the external face f_e. Arc (f_e, t) has capacity equal to 4 plus the number of edges belonging to the face and zero lower bound and cost.

The flow value z in $N_\mu(\sigma, A)$ is constrained equal to the sum of the capacities of the arcs outgoing node s.

The intuitive interpretation of network $N_\mu(\sigma, A)$ is the following. Each unit of flow in the network represents an angle of $\pi/2$, or, from another point of view, one unit of spirality. For each pair $(f', f''), (f'', f')$ linking two face-nodes, their difference of flow represents the spirality of the split components whose virtual edge separates f' and f'' in $skeleton_A(\mu)$; the cost of the flow represents the number of bends of the orthogonal representation of G_μ.

Observe that all the components whose vertices have degree at most 3 have integer spirality. So, in this case, the problem of computing the optimal orthogonal representation for a given value of spirality σ can be solved as an integer min-cost flow problem by adding constraints that force the component to have spirality σ. To do that, let f' and f'' be the faces sharing (u, w) and let f'' be the external face of $skeleton_A(\mu)$. We give zero cost both to (f', f'') and to (f'', f'); if $\sigma \leq 4$ then we give lower bound and capacity $4 - \sigma$ to (f', f'') and zero capacity to (f'', f'); if $\sigma > 4$ then we give lower bound and capacity $\sigma - 4$ to (f'', f') and zero capacity to (f', f'').

When solving the problem for components that contain vertices of degree 4 we have several additional problems:

1. Some virtual edges of $skeleton_A(\mu)$ might represent components whose spirality is not an integer. Furthermore, the component G_μ itself might have noninteger spiralities (see Property 2). Since the cost functions of such components have meaningful values only for noninteger values of spirality and since the flow has integer values, the network problem is not sound in this case.
2. It might exist a degree-4 vertex v of G_μ corresponding to a degree-3 vertex in $skeleton_A(\mu)$. This causes an ambiguity in the correspondence between the values of the angles between pairs of virtual edges incident on v in $skeleton_A(\mu)$ and the values of the angles between pairs of edges incident on v in the orthogonal representation of G_μ.

To solve the above problems we add several constraints to the flow network as follows. Suppose $e = (u', w')$ is a virtual edge (shared by faces f' and f'') of $skeleton_A(\mu)$, corresponding to the pertinent graph G_ν of a node ν of T. We distinguish two cases.

- Both u' and w' have internal degree 2 in G_ν. Observe that u' and w' have degree 4 in G (otherwise they would have degree 2 in $skeleton_A(\mu)$, which is impossible for Property 1) and have degree 3 in $skeleton_A(\mu)$. In this case the spirality of G_ν has integer values, but Problem 2 occurs. To solve it we constrain both the angle at u' on face f' and the angle at w' on face f'' to be flat. To do that capacity and lower bound of both arcs (u', f') and (w', f'') are set to one.
- Exactly one pole, say u' of G_ν has internal degree 2. In this case both Problems 1 and 2 occur: the spirality of G_ν is not integer since w' is a bridge-pole while u' is a nonbridge pole with external degree 2; and there is ambiguity for the angles around u'. Suppose f' (f'') is to the left (right) of e. We constrain the angle at u' on face f' to be flat by setting to one capacity and lower bound of arc (u', f'). Also, the cost functions of arcs (f', f'') and (f'', f') are shifted of 0.5 to the left to give appropriate costs to integer values of flow.

Theorem 3 *Let T be a decomposition tree of an st-digraph and let μ be an R-node of T, whose pertinent graph is G_μ. Let H_μ be an orthogonal representation of G_μ such that (1)H_μ is optimal within spirality σ, and (2)skeleton(μ) has embedding A (embedding B). Then it corresponds to H_μ a minimum cost integer flow in $N_\mu(\sigma, A)$ ($N_\mu(\sigma, B)$). Furthermore, an orthogonal representation H_μ of G_μ such that (1)H_μ is optimal within spirality σ, and (2)skeleton(μ) has embedding A (embedding B) can be computed from the minimum cost integer flow in $N_\mu(\sigma, A)$ ($N_\mu(\sigma, B)$).*

4.3 Computational Complexity of the Algorithm

Let T be a decomposition tree of an n-vertex *st*-digraph with k vertices of degree 4; let μ be an R-node of T whose pertinent graph is G_μ. In order to compute the optimal set of μ, for each value σ of spirality in the optimal set, we solve the min-cost flow problem on both network $N_\mu(\sigma, A)$ and $N_\mu(\sigma, B)$ and we choose the cheapest solution; such solution, according to Theorem 3, corresponds to an orthogonal representation of G_μ optimal within spirality σ.

Lemma 5 *The optimal set of* μ *can be computed in* $O(n^k \cdot (n^3 \log n))$.

Sketch of proof: Suppose first all the cost functions of the components of G_μ are convex. This implies that for any given σ, the cost functions of the arcs of both $N_\mu(\sigma, A)$ and $N_\mu(\sigma, B)$ are convex. Thus, for each value σ of spirality in the optimal set of μ, the min-cost flow problem is solvable in $O(n^2 \log n)$ time (observe that the number of vertices in the network is $O(n)$). Since the cardinality of the optimal set of μ is $O(n)$ (see Theorem 1) the overall time complexity is $O(n^3 \log n)$. Suppose now G_μ has h components with non-convex cost functions; for Theorem 1, each cost function is defined on at most $3n - 6$ different values of spirality . The corresponding networks $N_\mu(\sigma, A)$ and $N_\mu(\sigma, B)$ have $2 \cdot h$ arcs whose cost functions are non-convex. On each of the two networks, a flow of minimum cost can be computed in $O(n^h \cdot (n^3 \log n))$ time by considering all the combinations for the possible values of the flows in the arcs; the number of such combinations is $O(n^h)$, because there are $O(n)$ possible values for the flow on each arc and for each pair $(f', f''), (f'', f')$ the optimality of the solution implies one of the two arcs with zero flow. The proof is completed by observing that $h = O(k)$, because only the components of G_μ containing vertices of degree 4 can have non convex cost functions (see also Lemma 2. □

We are now ready to discuss the time complexity of Algorithm *Optimal Orthogonal Drawing.*

Theorem 4 *Algorithm* Optimal Orthogonal Drawing *computes an optimal orthogonal representation of an n-vertex graph with k vertices of degree* 4 *in* $O(n^k \cdot p(n))$ *where* $p(n)$ *is a polynomial. Furthermore, it computes an optimal orthogonal representation of a* 3*-planar series-parallel graph with n vertices in* $O(n^3)$ *time.*

Sketch of proof: The computation of an *st*-orientation of G, the construction of the decomposition tree and the definition of an optimal orthogonal drawing [2] can be performed in $O(n)$ time ([4], [1], [2]). The computation of the optimal sets requires polynomial time for the P-, S-, and Q^*-nodes and $O(n^k \cdot (n^3 \log n))$ time for the R-nodes. Thus, $O(n^k \cdot p(n))$ is the time required to compute the optimal orthogonal representation of a 4-planar graph. Suppose now the input graph ia a 3-planar series-parallel graph; this means that the nodes of the decomposition tree of T are only of the P, S and Q^* type and that no vertex of the input graph has degree 4. In this case the optimal sets of the nodes of T can be computed by visiting T from the leaves to the root as follows. We compute first the optimal sets of the Q^*-nodes, which requires $O(n)$ time. The optimal set of an P-node μ with children μ_1 and μ_2 can be easily constructed in $O(n)$ time by exploiting Lemma 1 and using ordered structures for the optimal sets of μ_1 and μ_2. The optimal set of an S-node μ with children μ_1 and μ_2 is constructed by considering, for each possible value σ of spirality, the optimal sets of μ_1 and of μ_2 and finding two representations such that the sum of their spiralities equals σ (see Lemma 1) and the sum of their costs is minimum. Such task can be performed in $O(n)$ time by using ordered structures for the optimal sets of μ_1 and μ_2 and exploiting the convexity of the cost functions of the pertinet graphs of μ, μ_1 and μ_2. □

References

[1] G. Di Battista and R. Tamassia, "Incremental Planarity Testing," *Proc. 30th FOCS*, pp. 436-441, 1989.

[2] G. Di Battista, G. Liotta, and F. Vargiu, " Spirality of Orthogonal Representations and Optimal Drawings of Series-Parallel Graphs and 3-Planar Graphs," *Proc. 3rd WADS*, pp. 151-162, 1993.

[3] G. Di Battista, P. Eades, R. Tamassia, and I.G. Tollis, "Algorithms for Automatic Graph Drawing: An Annotated Bibliography," to appear in *Computational Geometry: Theory and Applications.*

[4] S. Even, *Graph Algoritms*, Computer Science Press, Potomac, MD, 1979.

[5] A. Garg and R. Tamassia, "On the Computational Complexity of Upward and Rectilinear Planarity Testing," Tech. Rep. CS-94-10, Brown Univ. 1994.

[6] G. Kant, "A New Method for Planar Graph Drawings on a Grid," *Proc. 33rd FOCS*, pp. 101-110, 1992.

[7] E. L. Lawler, *Combinatorial Optimization: Networks and Matroids*, Holt, Rinehart and Winston, New York, 1976.

[8] R. Tamassia, "On Embedding a Graph in the Grid with the Minimum Number of Bends," *SIAM J. on Computing*, vol. 16, no. 3, pp. 421-444, 1987.

[9] R. Tamassia and I.G. Tollis, "Planar Grid Embedding in Linear Time," *IEEE Trans. on Circuits and Systems*, vol. CAS-36, no. 9, pp. 1230-1234, 1989.

[10] R. Tamassia, I. G. Tollis, and J. S. Vitter, "Lower Bounds and Parallel Algorithms for Planar Orthogonal Grid Drawings," *Proc. IEEE Symp. on Parallel and Distributed Processing*, 1991.

[11] R. Tamassia, I. G. Tollis, and J. S. Vitter, "Lower Bounds for Planar Orthogonal Drawings of Graphs," *Information Processing Letters*, vol. 39, pp. 35-40, 1991.

[12] L. Valiant, "Universality Considerations in VLSI Circuits," *IEEE Trans. on Computers*, vol. c-30, no. 2, pp. 135-140, 1981.