Karlsruhe Institute of Technology

Orthogonal Graph Drawing with Flexibility Constraints

Study Thesis of

Thomas Bläsius

At the faculty of Computer Science
Institute of Theoretical Informatics (ITI)

Reviewer: Prof. Dr. Dorothea Wagner
Advisors: Dipl.-Inform. Marcus Krug,
Dipl.-Inform. Ignaz Rutter

14. April 2010 – 3. August 2010

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Karlsruhe, den 03. August 2010

Thomas Bläsius

Contents

1. Introduction

Since many kinds of data can be expressed as graphs it is an important field of research to visualize graphs in a human-readable way. A popular convention is the orthogonal graph drawing convention where edges consist only of vertical and horizontal segments. This restriction yields potentially very clear drawings since the human eye may easily adapt to the flow of an edge if it does not have too many bends or crossings with other edges. Hence, we are interested in orthogonal drawings with few bends that are planar, i.e., two different edges do not intersect except, possibly, in a common endpoint. Since only planar graphs with maximum degree 4 admit planar orthogonal drawings we restrict our considerations to such graphs and call them *4-planar*. Unfortunately, not all 4-planar graphs admit planar orthogonal drawings without bends, for example the K_4 and the octahedron, depicted in Figure 1, require edges with two and three bends, respectively. Hence, one could allow edges to have bends and try to minimize the number bends. Previous approaches focus on the minimization of the maximum number of bends per edge or the total number of bends in the drawing. In typical applications some edges could have varying importance for the readability depending on their semantic and importance for the application, for example Figure 2 shows a weighted graph where the number of allowed bends of every edge depends on its weight. Thus, it is convenient to allow some edges to have more bends than others. This yields the problem FlexDraw, which asks whether a given 4-planar graph G with a function $\text{flex} : E \longrightarrow \mathbb{N}_0$ admits a planar orthogonal drawing such that each edge e has at most $\text{flex}(e)$ bends. We call the number of allowed bends expressed by $\text{flex}(\cdot)$ the *flexibility* of an edge. In this work we give a polynomial-time algorithm for FlexDraw for graphs with *positive flexibility*, i.e., for graphs where each edge has positive flexibility.

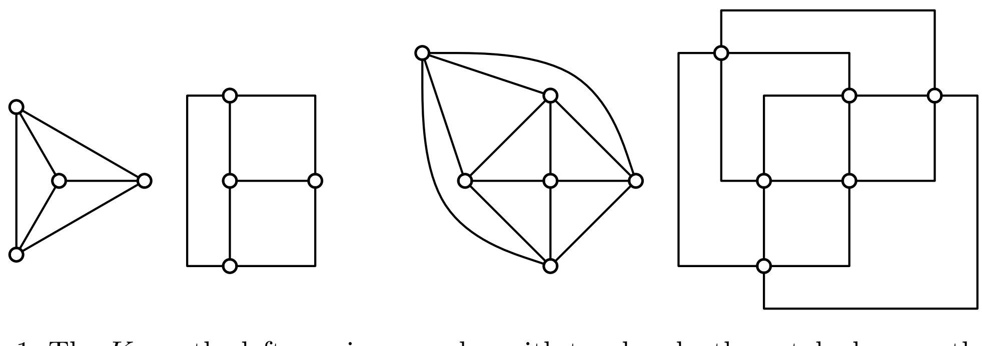


Figure 1: The K_4 on the left requires an edge with two bends, the octahedron on the right an edge with three bends.

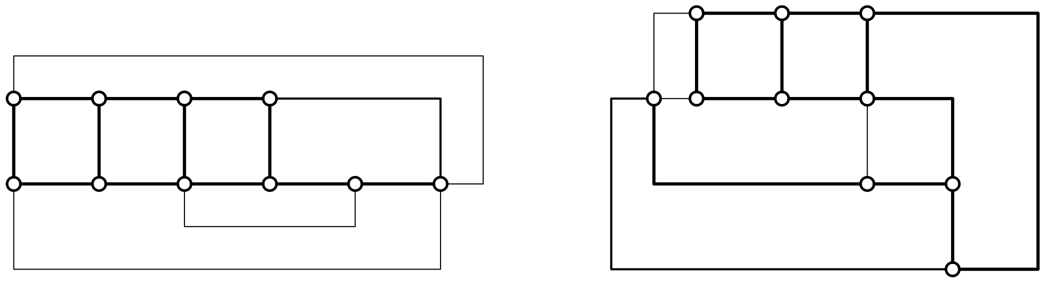


Figure 2: Drawings of a weighted graph, where the weights are indicated by the thickness of the edges. The total number of bends is less in the right drawing, but the left drawing is clearer since the heavy edges have less bends.

The problem FlexDraw generalizes the well-studied problem to decide whether a given graph is β-embeddable for some non-negative integer β. A 4-planar graph is *β-embeddable* if it admits an orthogonal drawing with at most β bends per edge. Garg and Tamassia [GT01] show that it is $\mathcal{NP}$-hard to decide 0-embeddability for arbitrary 4-planar graphs, hence it is also $\mathcal{NP}$-hard to minimize the total number of bends. Bertolazzi et al. [BDD00] give a branch-and-bound algorithm with exponential running time minimizing the total number of bends for biconnected graphs. Di Battista et al. [DLV98] show that 0-embeddability can be solved in polynomial time for planar graphs with maximum degree 3 and for series-parallel graphs. On the other hand, Biedl and Kant [BK94] show that every 4-planar graph admits a drawing with at most two bends per edge with the only exception of the octahedron, which requires an edge with three bends. Similar results are obtained by Liu et al. [LMS98]. Liu et al. [LMPS92] also claim to have found a characterization of the planar graphs with minimum degree 3 and maximum degree 4 that admit an orthogonal drawing with at most one bend per edge. They also claim that this characterization can be tested in polynomial time. Unfortunately, their paper does not include any proofs and to the best of our knowledge a proof of these results did not appear. Katz et al. [KKRW10] introduced the Geodesic Embeddability problem that considers whether a 4-planar graph admits a planar orthogonal drawing such that all edges are monotone. They show that this problem is equivalent to 1-embeddability.

While the above mentioned papers consider planar graphs, there are also several results for *plane graphs*, i.e., planar graphs with fixed planar embedding. For instance Tamassia's flow network can be used to minimize the total number of bends in polynomial time [Tam87]. Note that this solves 0-embeddability for graphs with fixed embeddings whereas the same problem is $\mathcal{NP}$-hard for graphs with arbitrary embeddings. Tamassia's flow network is quite flexible and can for example be modified in a straightforward manner to solve FlexDraw for graphs with fixed embedding. Morgana et al. [MdMS04] characterize the class of plane graphs that admit an orthogonal 1-bend drawing by forbidden configurations. They also present a quadratic algorithm that either detects a forbidden configuration or computes a 1-bend drawing.

In this work we describe an efficient algorithm solving FlexDraw for 4-planar graphs with positive flexibility. Since FlexDraw contains the problem of 1-embeddability as a special case this closes the complexity gap between the $\mathcal{NP}$-hardness result for 0-embeddability by Garg and Tamassia [GT01] and the efficient algorithm for computing 2-embeddings by Biedl and Kant [BK94].

1.1 Outline

In Section 1.2, we fix some notation used in this work. In Section 2, we consider graphs with fixed embeddings and show that it is impossible to construct rigid graphs, where rigid

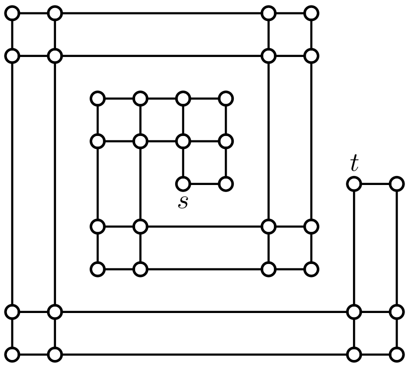


Figure 3: A graph with flexibility 0 for every edge. The angle between s and t is fixed to 720°.

means that every drawing must be "wound up" as the spiral in Figure 3. Note that the $\mathcal{NP}$-hardness proof of 0-embeddability by Garg and Tamassia [GT01] crucially relies on the possibility to construct rigid graphs. In Section 2.1, we show that the construction of rigid graphs is impossible if we allow at least one bend per edge. In fact we show that it is sufficient to know the largest possible angle between two designated vertices on the outer face to know all possible values for this angle. In Section 2.2, we show how to compute this maximum angle by simply using a variant of Tamassia's flow network [Tam87].

In Section 3, we drop the planar embedding which was fixed before. In Section 3.1, we first generalize our results concerning the angle between two designated vertices to graphs without fixed planar embedding. Again we obtain that computing the maximum angle would be sufficient to know all possible angles. This shows that the behavior of whole graphs is similar to the behavior of single edges, where their largest possible angle is determined by the flexibility. We will see that computing this angle (or deciding that no valid drawing exists) solves FlexDraw for positive flexibility. In Section 3.2 we show how to replace subgraphs by simpler graphs. In Section 3.3, we describe the well known SPQR-tree introduced by Di Battista and Tamassia [DT96a, DT96b]. It decomposes the graph in smaller, easy to handle subgraphs and can be used to enumerate all planar embeddings. Putting that together with the replacement of subgraphs yields an efficient algorithm that solves FlexDraw for 4-planar biconnected graphs with positive flexibility. Finally, in Section 4, we extend this result to arbitrary 4-planar graphs with positive flexibility.

1.2 Notation

Connectivity and st-Graphs

A graph is *connected* if there exists a path between any pair of vertices. A *separating k-set* is a set of k vertices whose removal disconnects the graph. Separating 1-sets and 2-sets are *cutvertices* and *separation pairs*. A graph is *biconnected* if it does not have a cut vertex and it is *triconnected* if it does not have a separation pair. Note that a triconnected planar graph has a unique embedding up to reflection and the choice of the outer face [Whi32, Kel80]. The maximal biconnected components of a graph are called *blocks*. A *bridge* is an edge whose removal disconnects the graph.

A *weak st-graph* is a graph G with two designated vertices s and t such that s and t are not cutvertices and the graph $G + st$ is 4-planar, G is an *st-graph* if $G + st$ is additionally biconnected. An orthogonal drawing $\mathcal{R}$ of a (weak) st-graph with positive flexibility is *valid* if each edge e has at most flex(e) bends and s and t are embedded on the outer face. A valid orthogonal drawing of a (weak) st-graph is *tight* if all angles at s and t in inner faces are 90°.

Orthogonal Representation

The *orthogonal representation* introduced by Tamassia [Tam87] describes orthogonal drawings of plane graphs (i.e., planar graphs with fixed embedding), by listing the faces as sequences of bends. The advantage of the orthogonal representation is that it neglects the lengths of the segments. Thus, it is possible to apply different operations on the drawing, without the need to worry about the exact geometry.

The orthogonal representation $\mathcal{R}$ of a plane graph G contains one lists $\mathcal{R}(f_i)$ for each face f_i of G, where $\mathcal{R}(f_i)$ is a circular list of *edge descriptions* containing the edges on the boundary of f_i in clockwise order (counter-clockwise order if f_i is the outer face). Every edge description $r \in \mathcal{R}(f_i)$ contains three information: edge(r) denotes the edge represented by r, bends(r) lists the bends of edge(r) where angles of 90° and 270° are denoted by 1 and -1 respectively and finally if r' is the successor of r in $\mathcal{R}(f_i)$ the angle α between edge(r) and edge(r') in f_i is represented by angle$(r) = -2, -1, 0, 1$ for $\alpha = 360°, 270°, 180°, 90°$, respectively. Every edge has exactly two edge descriptions, if r is one of them, the other is denoted by $\bar{r}$ and we obtain edge$(r) =$ edge$(\bar{r})$. The sum over all bends in bends(r) is denoted by $\Sigma\,$bends(r).

Figure 4a shows an example of an orthogonal drawing. If we enclose each edge description by $\langle\cdot\rangle$ and mark angle(r) bold we obtain the following orthogonal representation:

$$\begin{aligned}
\mathcal{R}(f_1) &= (\langle -1-1 \quad \mathbf{+1}\rangle, \langle \quad \mathbf{-2}\rangle, \langle \quad \mathbf{+1}\rangle, \langle -1-1 \quad \mathbf{\pm 0}\rangle) \\
\mathcal{R}(f_2) &= (\langle +1+1 \quad \mathbf{+1}\rangle, \langle -1+1 \quad \mathbf{+1}\rangle) \\
\mathcal{R}(f_3) &= (\langle -1+1 \quad \mathbf{+1}\rangle, \langle +1+1 \quad \mathbf{+1}\rangle)
\end{aligned}$$

Due to the fact, that every face f_i is a rectilinear polygon, it is clear that every orthogonal representation $\mathcal{R}$ of an orthogonal drawing has the following three properties.

(I) The edge description $\bar{r}$ is consistent with r, which means that bends$(\bar{r})$ is obtained by exchanging -1 with 1 in bends(r) and reversing it (an angle of 90° forms an angle of 270° in the adjacent face and the bends are traversed in the opposite direction).

(II) The interior bends of any face f_i sum up to 4 and the exterior bends to -4:

$$\sum_{r \in \mathcal{R}(f_i)} (\Sigma\,\text{bends}(r) + \text{angle}(r)) = \begin{cases} -4, & \text{if } f \text{ is the outer face,} \\ +4, & \text{if } f \text{ is an inner face.} \end{cases}$$

(III) The angles around every node sum up to 360°.

On the other hand, given a representation $\mathcal{R}$ that has Properties I - III we can use the algorithm given by Tamassia [Tam87] to compute an orthogonal drawing efficiently. Hence, we can consider orthogonal representations instead of drawings and if we make modifications in a given orthogonal representation, we only have to ensure that the three properties remain satisfied.

An orthogonal representation $\mathcal{R}$ is called *normalized* if every edge has only bends in one direction, i.e., either the number of 1s or -1s in bends(r) is zero. Every orthogonal representation $\mathcal{R}$ can be normalized by successively eliminating the sequences $-1+1$ and $+1-1$ in bends(r) and updating bends$(\bar{r})$ accordingly. It is clear that this operation does not harm any of the three properties. Hence, it is not a restriction to consider only normalized orthogonal representations and from now on all orthogonal representations we consider are implicitly assumed to be normalized.

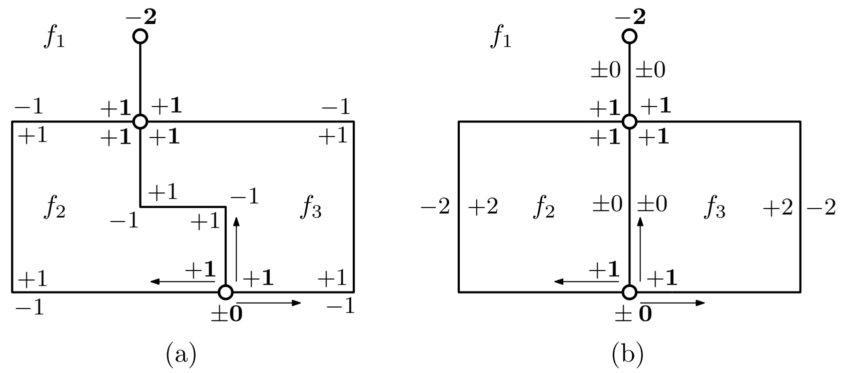


Figure 4: A simple orthogonal drawing (a) and its normalized version (b).

Rotation and the Path from s to t

For a normalized orthogonal representation $\mathcal{R}$ it is sufficient to know $\Sigma\,\text{bends}(r)$ instead of $\text{bends}(r)$. Therefore $\Sigma\,\text{bends}(r)$ is called the *rotation* of the edge description r and is denoted by $\text{rot}_{\mathcal{R}}(r)$. The value of $\text{rot}_{\mathcal{R}}(r)$ denotes the number of bends $\text{edge}(r)$ has and the sign determines the direction of the bends (positive to the right, negative to the left). Note that Property I simplifies to the equation $\text{rot}_{\mathcal{R}}(r) = -\text{rot}_{\mathcal{R}}(\bar{r})$ for normalized orthogonal representations.

Figure 4b shows the normalized drawing of the orthogonal representation from Figure 4a. We obtain the following normalized orthogonal representation where the sequence $\text{bends}(r)$ is replaced by the rotation $\text{rot}_{\mathcal{R}}(r)$:

$$\begin{aligned}
\mathcal{R}(f_1) &= (\langle -2 \quad \mathbf{+1}\rangle, \langle \pm 0 \quad \mathbf{-2}\rangle, \langle \pm 0 \quad \mathbf{+1}\rangle, \langle -2 \quad \pm\mathbf{0}\rangle)\\
\mathcal{R}(f_2) &= (\langle +2 \quad \mathbf{+1}\rangle, \langle \pm 0 \quad \mathbf{+1}\rangle)\\
\mathcal{R}(f_3) &= (\langle \pm 0 \quad \mathbf{+1}\rangle, \langle +2 \quad \mathbf{+1}\rangle)
\end{aligned}$$

Now we extend the term rotation to other objects than single edges. Let $\mathcal{R}(f_i)$ be an orthogonal representation of the face f_i and let $r, r' \in \mathcal{R}(f_i)$ be two edge descriptions such that r' is the successor of r. Then the rotation between them is defined as $\text{rot}_{\mathcal{R}}(r, r') = \text{angle}(r)$. If f_i is the external face and the vertex v incident to $\text{edge}(r)$ and $\text{edge}(r')$ has a unique angle in f_i we represent this angle by $\text{rot}_{\mathcal{R}}(v) = \text{rot}_{\mathcal{R}}(r, r')$. Note that this is the case if v is not a cutvertex. Hence, if G is a (weak) st-graph, we have that $\text{rot}_{\mathcal{R}}(s)$ and $\text{rot}_{\mathcal{R}}(t)$ represent the angles at s and t in the outer face, respectively. For a

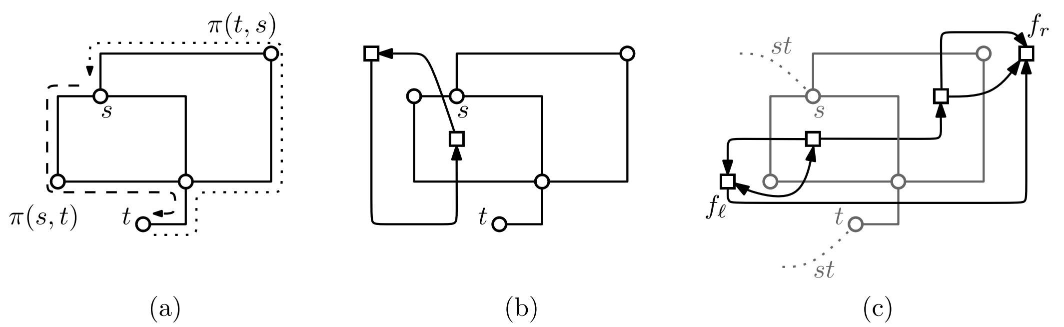


Figure 5: An st-graph G where the paths $\pi(s,t)$ and $\pi(t,s)$ are depicted dashed and dotted, respectively, with $\text{rot}(\pi(s,t)) = 0$, $\text{rot}(\pi(t,s)) = -3$, $\text{rot}(s) = 1$ and $\text{rot}(t) = -2$ (a). Bending the edges along a cycle in the dualgraph (b) and the flex graph $G^\times$ of G (c).

path $\pi = (r_1, \ldots, r_k)$ of consecutive elements in $\mathcal{R}(f_i)$ the rotation is defined as the total rotation along this path:

$$\mathrm{rot}_{\mathcal{R}}(\pi) = \mathrm{rot}_{\mathcal{R}}(r_1) + \sum_{i=2}^{k} (\mathrm{rot}_{\mathcal{R}}(r_{i-1}, r_i) + \mathrm{rot}_{\mathcal{R}}(r_i))$$

If it is clear from the context which representation $\mathcal{R}$ is meant, we simply write $\mathrm{rot}(\cdot)$ instead of $\mathrm{rot}_{\mathcal{R}}(\cdot)$.

The paths we normally consider are the paths from s to t and from t to s on the outer face of (weak) st-graphs. Note that these paths are unique since s and t are not cutvertices. For a given orthogonal representation $\mathcal{R}$ with s and t on the outer face f_1 we define the path $\pi(s,t)$ as the path of edge descriptions in $\mathcal{R}(f_1)$ connecting s and t in counter-clockwise direction. Note that the rotation $\mathrm{rot}_{\mathcal{R}}(\pi(s,t))$ describes the angle between the nodes s and t. Analogously we define $\pi(t,s)$ as the path from t to s. Note that Property II yields the equation $\mathrm{rot}(\pi(s,t)) + \mathrm{rot}(t) + \mathrm{rot}(\pi(t,s)) + \mathrm{rot}(s) = -4$. See Figure 5a for an example.

2. Orthogonal Drawings with Fixed Embedding

In this section we consider the situation where G is an st-graph with positive flexibility $\text{flex}(\cdot)$ and fixed planar embedding $\mathcal{E}$ such that the vertices s and t are on the outer face f_1. We wish to know for which integers ρ we can find a valid and tight orthogonal representation $\mathcal{R}$ such that $\text{rot}_{\mathcal{R}}(\pi(s,t)) = \rho$. Therefore, we define the *maximum rotation* of G with respect to the planar embedding $\mathcal{E}$ as $\text{maxrot}_{\mathcal{E}}(G) = \max_{\mathcal{R}\in\Omega} \text{rot}_{\mathcal{R}}(\pi(s,t))$ where Ω is the set of all valid orthogonal representations of G with planar embedding $\mathcal{E}$. In Section 2.1 we will show that at least for every integer $-1 \leq \rho \leq \text{maxrot}_{\mathcal{E}}(G)$ there exists a valid and tight orthogonal representation $\mathcal{R}$ such that $\text{rot}_{\mathcal{R}}(\pi(s,t)) = \rho$, which shows that rigid graphs do not exist for graphs with positive flexibility. In Section 2.2 we will show how to compute $\text{maxrot}_{\mathcal{E}}(G)$ efficiently.

2.1 Possible Rotation Values

To show that we can reach the claimed interval we wish to reduce the rotation on the path $\pi(s,t)$ for a given orthogonal representation $\mathcal{R}$, i.e., to construct an orthogonal representation $\mathcal{R}'$ such that $\text{rot}_{\mathcal{R}'}(\pi(s,t)) = \text{rot}_{\mathcal{R}}(\pi(s,t)) - 1$. We start with the following observation: Let $\mathcal{R}$ be an orthogonal representation of the graph G, let $f_1, \dots, f_k, f_1$ be a simple directed cycle in the dual graph G^* and for an edge $(f_i f_{i+1})$ in this cycle let e_i be the corresponding edge in G with the edge descriptions $r_i \in \mathcal{R}(f_i), \bar{r}_i \in \mathcal{R}(f_{i+1})$. Then we obtain a new orthogonal representation by decreasing $\text{rot}(r_i)$ and increasing $\text{rot}(\bar{r}_i)$ by 1 for $1 \leq i \leq k$; see Figure 5b for an example of such a cycle and the resulting graph. To prove that the result is again an orthogonal representation we have to check whether Properties I– III remain satisfied. Since r_i is changed consistently with $\bar{r}_i$ and the angles around vertices are not changed it is clear that Properties I and III are satisfied. Property II holds since each face of G has either none of its edge descriptions changed or exactly one of them is increased by 1 and exactly one of them is decreased by 1.

Since we are only interested in valid orthogonal representations we delete edges from G^* that would violate the flexibility of the corresponding edge in G. Additionally, we split the outer face f_1 into f_ℓ and f_r such that the edges in G^* belonging to edges on $\pi(s,t)$ are separated from the edges belonging to $\pi(t,s)$. For an example see Figure 5c depicting the flex graph of the graph from Figure 5a assuming that $\text{flex}(e) = 1$ for every edge e. More precisely we define the *flex graph* $G^\times$ as follows.

Let G be a (weak) st-graph with flexibility flex($\cdot$) and let $\mathcal{R}$ be a valid orthogonal representation of G. We start by adding st to G where st is embedded in the outer face f_1 such that f_1 is split into f_ℓ and f_r bounded by $\pi(s,t)$ and st and by $\pi(t,s)$ and st, respectively. We denote the resulting graph by $G_{s,t}$ and its dual graph by $G^*_{s,t}$. The flex graph $G^\times$ of G with valid orthogonal representation $\mathcal{R}$ has the same nodes as $G^*_{s,t}$. For an edge e in G let f_u and f_v be its incident faces in $G_{s,t}$ and let $r_u \in \mathcal{R}(f_u)$ and $r_v \in \mathcal{R}(f_v)$ be the corresponding edge descriptions. We add the edge $(f_u f_v)$ if $-\operatorname{flex}(e) < \operatorname{rot}_\mathcal{R}(r_u)$ and, analogously, we add $(f_v f_u)$ if $-\operatorname{flex}(e) < \operatorname{rot}_\mathcal{R}(r_v)$. With this definition an edge $(f_u f_v)$ in $G^\times$ implies that $\operatorname{rot}(r_u)$ can be decreased without harming the flexibility of edge(r_u). Additionally a simple directed path $f_\ell = f_1, \dots, f_k = f_r$ in $G^\times$ represents a cycle in the dualgraph of G, hence we can decrease $\operatorname{rot}(r_i)$ and increase $\operatorname{rot}(\bar{r}_i)$ by 1 as described above for cycles and obtain a valid orthogonal representation $\mathcal{R}'$ of G. Note that the path starts in f_ℓ and ends in f_r, hence we have that $\operatorname{rot}_{\mathcal{R}'}(\pi(s,t)) = \operatorname{rot}_\mathcal{R}(\pi(s,t)) - 1$. The following lemma shows that such a path always exists if $\operatorname{rot}_\mathcal{R}(\pi(s,t)) \geq 0$, its proof is illustrated in Figure 6.

Lemma 1. *Let G be a weak st-graph with positive flexibility and let $\mathcal{R}$ be a valid orthogonal representation of G with $\operatorname{rot}_\mathcal{R}(\pi(s,t)) \geq 0$. Then the flex graph $G^\times$ contains a directed path from f_ℓ to f_r.*

Proof. Assume that G together with the orthogonal representation $\mathcal{R}$ is a minimal counter example such that $G^\times$ does not contain a simple path from f_ℓ to f_r. We will use the fact, that all smaller graphs satisfying the conditions of this lemma cannot be counter examples to construct a path from f_ℓ to f_r in $G^\times$, which contradicts the assumption that G is a counter example.

First, we show that $G^\times$ contains at least one edge starting from f_ℓ. Let $\pi(s,t)$ be composed of the edge descriptions $r_1, \dots, r_k$ in $\mathcal{R}(f_1)$ where f_1 is the outer face of G. Then, by assumption we have

$$\operatorname{rot}(\pi(s,t)) = \sum_{i=1}^{k} \operatorname{rot}(r_i) + \sum_{i=2}^{k} \operatorname{rot}(r_{i-1}, r_i) \geq 0$$

Since the rotation $\operatorname{rot}(r_{i-1}, r_i)$ between two edges is at most 1, the second sum is strictly less than k. This yields the following lower bound for the first sum:

$$\sum_{i=1}^{k} \operatorname{rot}(r_i) > -k$$

Therefore, there must be at least one edge with edge description r_i on the path $\pi(s,t)$ such that $\operatorname{rot}(r_i) \geq 0$. Due to the fact that G has positive flexibility, the edge corresponding to edge(r_i) starting in f_ℓ is contained in the flex graph $G^\times$.

Now let $(f_\ell f_u)$ be an edge in $G^\times$. We distinguish three cases. If $\mathbf{f_u} = \mathbf{f_r}$ we are done, since the edge $(f_\ell f_u)$ is a directed path of length 1 from f_ℓ to f_r in $G^\times$.

If $\mathbf{f_u} = \mathbf{f_\ell}$ the corresponding edge e in G is a bridge whose removal does not disconnect s and t; see Figure 6a for an example. Let H be the connected component in $G-e$ containing s and t and let $\mathcal{S}$ be the restriction of the orthogonal representation $\mathcal{R}$ to H. Figure 6b shows the resulting graph with respect to the graph from Figure 6a. For the outer face of H we have that $\operatorname{rot}_\mathcal{S}(\pi(s,t)) + \operatorname{rot}_\mathcal{S}(t) + \operatorname{rot}_\mathcal{S}(\pi(t,s)) + \operatorname{rot}_\mathcal{S}(s) = -4$. Since the path $\pi(t,s)$ was not changed we have that $\operatorname{rot}_\mathcal{S}(\pi(t,s)) = \operatorname{rot}_\mathcal{R}(\pi(t,s))$. Moreover, since we only remove edges the angles at s and t do not decrease and thus we have $\operatorname{rot}_\mathcal{S}(t) \leq \operatorname{rot}_\mathcal{R}(t)$ and $\operatorname{rot}_\mathcal{S}(s) \leq \operatorname{rot}_\mathcal{R}(s)$. Hence, we have that $\operatorname{rot}_\mathcal{S}(\pi(s,t)) \geq -4 - \operatorname{rot}_\mathcal{R}(\pi(t,s)) - \operatorname{rot}_\mathcal{R}(s) -$

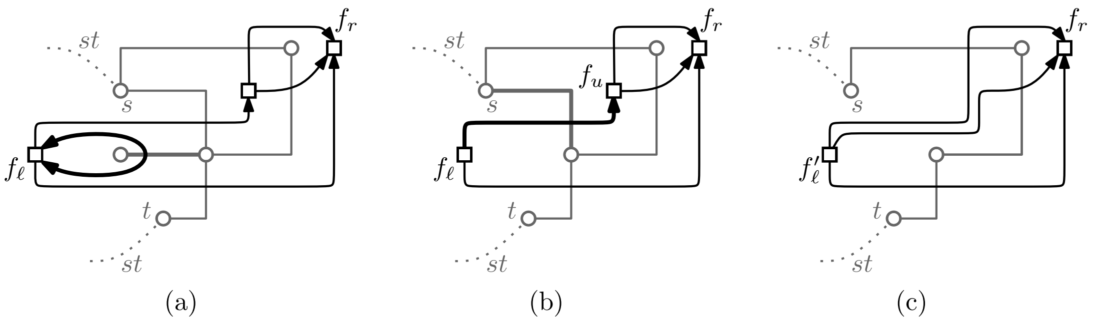


Figure 6: Removing the marked bridge in (a) yields (b), the new flex graph is obtained by removing the corresponding loop. Removing the marked edge in (b) yields (c), the faces f_ℓ and f_u are merged into f'_ℓ to obtain the new flex graph.

$\mathrm{rot}_\mathcal{R}(t) = \mathrm{rot}_\mathcal{R}(\pi(s,t)) \geq 0$. Since H satisfies the conditions of this lemma and has fewer edges than G it is not a counter example and its flex graph $H^\times$ contains a path from f_ℓ to f_r. Due to the fact that $H^\times$ is a subgraph of $G^\times$ this path was also contained in $G^\times$ and hence G cannot be a counter example.

Otherwise, $\mathbf{f_u}$ **is an internal face of G**. Let e be the edge in G corresponding to $(f_\ell f_u)$ in $G^\times$. We consider the graph H that is obtained from G by removing e. If s or t becomes a cutvertex due to this deletion (i.e., H is no longer a weak st-graph) we additionally remove the bridges incident to s or t whose removals do not disconnect s and t. Let $\mathcal{S}$ be the restriction of $\mathcal{R}$ to H. Note that the flex graph $H^\times$ is obtained from $G^\times$ by merging f_ℓ and f_u into a single node f'_ℓ (and deleting some edges if we had to remove bridges); compare with Figure 6b and the resulting graph in Figure 6c. As above we obtain that $\mathrm{rot}_\mathcal{S}(\pi(s,t)) \geq 0$ and hence in $H^\times$ there exists a path from f'_ℓ to f_r. The corresponding path in $G^\times$ (after undoing the contraction of f_ℓ and f_u) either starts at f_ℓ or at f_u and ends at f_r. In the former case we have found our path, in the latter case the path together with the edge $(f_\ell f_u)$ forms the desired path. This again contradicts the assumption that G is a counter example. □

With this lemma we can show that the possible values of $\mathrm{rot}(\pi(s,t))$ form an interval as mentioned before. However, we first show the existence of tight orthogonal representations that have nearly the same rotation. Recall that a valid orthogonal representation of an st-graph is called tight if all angles at s and t in inner faces are 90°.

Lemma 2. *Let G be a weak st-graph with positive flexibility and let $\mathcal{R}$ be a valid orthogonal representation. Then there exists a valid orthogonal representation $\mathcal{R}'$ of G with the same planar embedding such that $\mathcal{R}'$ is tight,* $\mathrm{rot}_{\mathcal{R}'}(\pi(s,t)) \geq \mathrm{rot}_\mathcal{R}(\pi(s,t))$ *and* $\mathrm{rot}_{\mathcal{R}'}(\pi(t,s)) \geq \mathrm{rot}_\mathcal{R}(\pi(t,s))$.

Proof. Let f_1 be the outer face of G and assume that f_2 is an inner face incident to s whose inner angle at s is larger than 90°. We show how to decrease this angle by 90° by only changing the number of bends on certain edges. Hence, by applying the described operation iteratively, we can reduce all internal angles at inner faces incident to s and t to 90°.

We will first split the vertex s into two vertices s_1 and s_2 such that f_2 becomes part of the outer face. Then we will reduce the rotation on the path from s_1 to s_2 and finally merge s_1 and s_2 back to s; see Figure 7 for an illustration of these steps.

Let e_1 and e_2 be the two edges incident to s such that e_1 occurs before e_2 when traversing the boundary of f_2 clockwise starting from s. Assume that e_1 is incident to f_1 (the case that

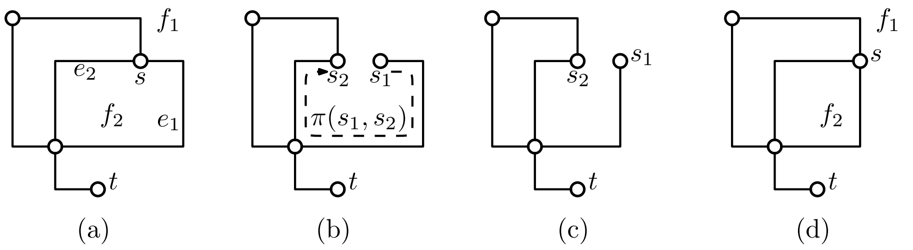


Figure 7: Orthogonal representation that is not tight since s has an angle of 180° in f_2 (a). Splitting s into s_1 and s_2 yields the path $\pi(s_1, s_2)$ with rotation at least 4 (b), hence the rotation can be reduced (c). Merging s_1 and s_2 back into s yields a tight orthogonal representation (d).

only e_2 is incident to f_1 works similarly). We split s into two vertices s_1 and s_2 and attach e_1 to s_1 and the remaining edges incident to s to s_2. Let H be the resulting graph and let $\mathcal{S}$ be the orthogonal representation of H induced by $\mathcal{R}$. The graph H with the designated vertices s_1 and s_2 is a weak st-graph with positive flexibility. Since f_2 is an internal face in G its total rotation in $\mathcal{R}$ is 4 and since the angle at s in f_2 was at least 180° we have that $\mathrm{rot}_{\mathcal{S}}(\pi(s_1, s_2)) \geq 4$. Hence H satisfies the conditions of Lemma 1 and the flex graph $H^\times$ of H contains a simple path that reduces the rotation along $\pi(s_1, s_2)$ by 1. This path either contains an edge stemming from $\pi(s_2, t)$ or an edge of $\pi(t, s_1)$ and hence either increases $\mathrm{rot}_{\mathcal{S}}(\pi(s_2, t))$ or $\mathrm{rot}_{\mathcal{S}}(\pi(t, s_1))$ by 1 where the other one remains unchanged. Denote the resulting orthogonal representation by $\mathcal{S}'$. We obtain $\mathcal{R}'$ by merging s_1 and s_2 back into s. Since $\mathrm{rot}_{\mathcal{S}'}(\pi(s_1, s_2)) = \mathrm{rot}_{\mathcal{S}}(\pi(s_1, s_2)) - 1$ we increase the rotation at s in f_2 by 1 (i.e., decrease the angle by 90°). It is clear that $\mathcal{R}'$ satisfies Properties I–III and we obtain the claimed inequalities $\mathrm{rot}_{\mathcal{R}}(\pi(s, t)) = \mathrm{rot}_{\mathcal{S}}(\pi(s_2, t)) \leq \mathrm{rot}_{\mathcal{S}'}(\pi(s_2, t)) = \mathrm{rot}_{\mathcal{R}'}(\pi(s, t))$ and $\mathrm{rot}_{\mathcal{R}}(\pi(t, s)) = \mathrm{rot}_{\mathcal{S}}(\pi(t, s_1)) \leq \mathrm{rot}_{\mathcal{S}'}(\pi(t, s_1)) = \mathrm{rot}_{\mathcal{R}'}(\pi(t, s))$. Note that aside from changing the numbers of bends on certain edges we did only change angles incident to s. $\square$

Given a valid orthogonal representation of a graph with positive flexibility with a non-negative rotation along the path from s to t Lemma 1 states the existence of another valid orthogonal representation with the same angles around vertices but reduced rotation along the path from s to t. On the other hand Lemma 2 states that a valid orthogonal representation can be made tight such that neither $\mathrm{rot}(\pi(s, t))$ nor $\mathrm{rot}(\pi(t, s))$ is decreased and hence both of them remain nearly unchanged. In the following theorem both lemmas are combined to show the existence of valid and tight orthogonal representations for all rotation values between $\mathrm{maxrot}_{\mathcal{E}}(G)$ and -1.

Theorem 1. *Let G be a weak st-graph with positive flexibility and fixed planar embedding $\mathcal{E}$. Then for each $\rho \in \{-1, \dots, \mathrm{maxrot}_{\mathcal{E}}(G)\}$ there exists a valid and tight orthogonal representation $\mathcal{R}$ of G with planar embedding $\mathcal{E}$ such that* $\mathrm{rot}_{\mathcal{R}}(\pi(s, t)) = \rho$.

Proof. Let $\rho \in \{-1, \dots, \mathrm{maxrot}_{\mathcal{E}}(G)\}$. We construct an orthogonal representation $\mathcal{R}$ with $\mathrm{rot}_{\mathcal{R}}(\pi(s, t)) = \rho$ as follows. Let $\mathcal{R}'$ be a valid orthogonal representation of G with planar embedding $\mathcal{E}$ such that $\mathrm{rot}_{\mathcal{R}'}(\pi(s, t)) = \mathrm{maxrot}_{\mathcal{E}}(G)$. By Lemma 2 we can make $\mathcal{R}'$ tight without decreasing $\mathrm{rot}(\pi(s, t))$. We obtain $\mathcal{R}$ by applying Lemma 1 successively until $\mathrm{rot}_{\mathcal{R}}(\pi(s, t)) = \rho$. $\square$

2.2 Computing the Maximum Rotation

In this section we will use a variant of Tamassia's flow network [Tam87] to compute $\mathrm{maxrot}_{\mathcal{E}}(G)$ for a given st-graph G with positive flexibility and fixed planar embedding $\mathcal{E}$. In Section 2.2.1 we describe the flow network in general and motivate why and how it works. In Section 2.2.2 we show how to use it for our purpose.

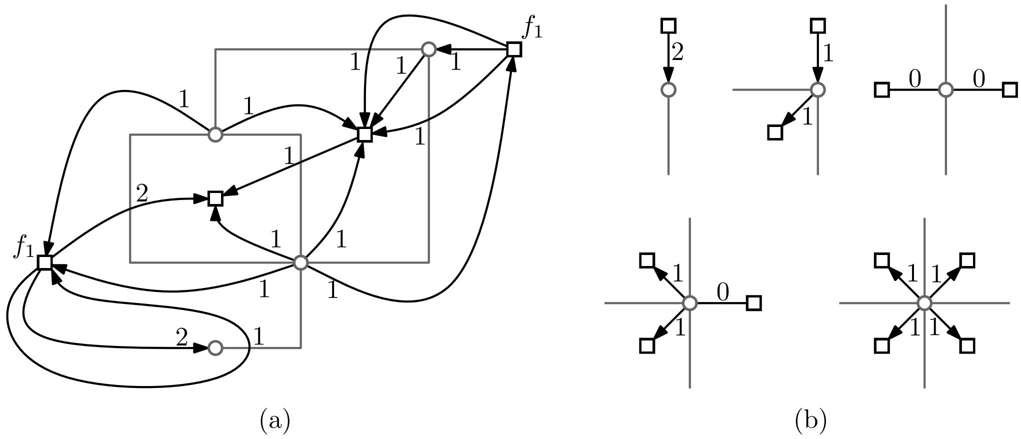


Figure 8: The example in (a) shows Tamassia's flow network where edges with flow 0 are omitted and the external face f_1 is split up to improve readability. In (b) all possible configurations for a single vertex are depicted.

2.2.1 Tamassia's Flow Network

Let G be a 4-planar graph with fixed planar embedding $\mathcal{E}$ and given normalized orthogonal representation $\mathcal{R}$. Before we give a precise definition of Tamassia's flow network we try to give an intuition on how to construct a flow network N together with a flow that represents the orthogonal representation $\mathcal{R}$. The flow network N contains one vertex for every face and one for every vertex in G, it contains directed edges between adjacent faces and for every face it contains edges to and from the vertices on its boundary (note that a vertex can occur multiple times on the boundary of the same face, hence there can be multiple edges). Let e be an edge in G incident to the faces f_u and f_v with its two edge descriptions $r \in \mathcal{R}(f_u)$ and $\bar{r} \in \mathcal{R}(f_v)$. If $\operatorname{rot}_{\mathcal{R}}(r)$ is positive we set the flow from f_v to f_u to $\operatorname{rot}_{\mathcal{R}}(r)$, if it is negative we set the flow from f_u to f_v to $-\operatorname{rot}_{\mathcal{R}}(r)$. This is consistent since $-\operatorname{rot}_{\mathcal{R}}(r) = \operatorname{rot}_{\mathcal{R}}(\bar{r})$ (Property I). Let $r, r' \in \mathcal{R}(f)$ be two edge descriptions on the boundary of the face f such that r' is the successor of r and v is the enclosed vertex. Then $\operatorname{rot}_{\mathcal{R}}(r, r')$ represents the angle at v in f and if it is positive we set the flow on the edge (v, f) to $\operatorname{rot}_{\mathcal{R}}(r, r')$, if it is negative we set the flow on (f, v) to $-\operatorname{rot}_{\mathcal{R}}(r, r')$. See Figure 8a for an example graph with an orthogonal representation and a corresponding flow.

Note that all information contained in the orthogonal representation can be extracted from the flow. If we construct the described flow from different orthogonal representations we maintain the following properties that are always satisfied. Since the rotation in every vertex is one of $-2, -1, 0, 1$ we can restrict the capacities of edges (f, v) to 2 and the capacities of edges (v, f) to 1. Since the total rotation around every inner face is 4 (Property II) we have a surplus of four units of flow for every inner face. Analogously, we have a lack of four units for the outer face. How much lack or surplus of flow a vertex has depends only on the degree of the vertex, since the sum of angles around every vertex is fixed (Property III); compare with Figure 8b. Hence, we get a feasible flow by setting sources and sinks with fixed out- and in-flow according to the surplus and lack of flow, which depends only on the graph and its planar embedding but not on the orthogonal representation. Finally, if we have two edges in N stemming from the same edge in G or from the same occurrence of a vertex in the boundary of a face, then the flow on at least one of them is 0. Note that such two edges have the same endpoints but they point in opposite directions. A flow in N that has this property is called *normalized.* As described above every normalized orthogonal representation yields a feasible normalized flow in N and two different representations yield two different flows since all information contained in the orthogonal representation is also contained in the flow. On the other hand it is easy to see that every feasible normalized

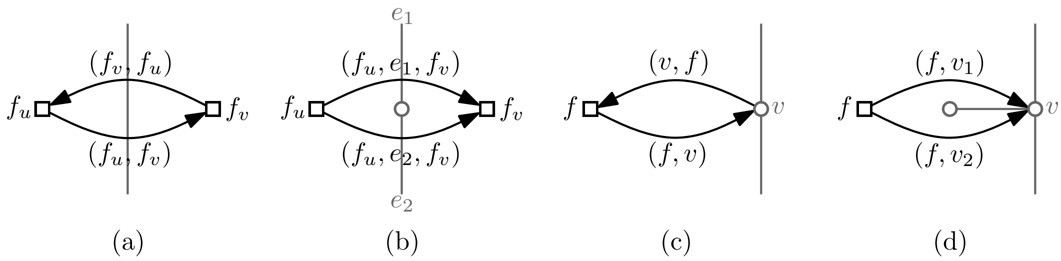


Figure 9: If the faces f_u and f_v are bounded by only one edge like in (a) the edges (f_u, f_v) and (f_v, f_u) are unique. If they are bounded by more edges we denote them like depicted in (b). If v occurs exactly once in the boundary of f we have unique edges (f, v) and (v, f) (c). Otherwise we denote them like in (d).

flow in N yields a normalized orthogonal representation, thus we get a bijection between all normalized orthogonal representations and the feasible normalized flows in N. Hence, the problem of finding an orthogonal representation satisfying several conditions reduces to the problem of finding a flow satisfying these conditions. For example the total number of bends can be minimized by simply solving a minimum cost flow problem. Note that the planar embedding in this setting is fixed, as the network N depends on the embedding.

To sum up we define the flow network $N = (U, D)$ of a 4-planar graph $G = (V, E)$ with fixed planar embedding $\mathcal{E}$ more precisely as follows. The vertices are defined as $U = V \cup F$ where F is the set of all faces in G. There are two types of edges $D = D_F \cup D_V$ where $D_F = E^*$ contains the edges from the dual graph of G, i.e., for every edge $e \in E$ with incident faces $f_u, f_v \in F$ we get the two directed edges (f_u, f_v) and (f_v, f_u). Since there can be multiple edges between faces we denote such an edge by (f_u, e, f_v) and (f_v, e, f_u) to make clear which edge is meant (see Figure 9a,b). For every face $f \in F$ and for every vertex $v \in V$ that lies on the boundary of f the set D_V contains the directed edges (v, f) and (f, v). Note that v could occur several times on the boundary of the face f, hence we denote the edges stemming from the ith occurrence of v in the boundary of f by (f, v_i) and (v_i, f) (see Figure 9c,d). We define the capacities $\text{cap}(\cdot)$ of edges in D as follows. For edges $(f_u, f_v) \in D_F$ we set $\text{cap}(f_u, f_v) = \infty$. For edges $(f, v) \in D_V$ we set $\text{cap}(f, v) = 2$ and for $(v, f) \in D_v$ we set $\text{cap}(v, f) = 1$. The network N has several sources and sinks with fixed out- and in-flow. Every inner face is a sink with in-flow 4. The outer face is a source with out-flow 4. Every vertex with degree 4 and 3 is a source with out-flow 4 and 2, respectively, and the vertices with degree 1 are sinks with in-flow 2. A flow in N is normalized if we have $\text{flow}(f_u, e, f_v) = 0$ or $\text{flow}(f_v, e, f_u) = 0$ for every two edges $(f_u, e, f_v), (f_v, e, f_u) \in D_F$ stemming from the same edge e and if $\text{flow}(f, v_i) = 0$ or $\text{flow}(v_i, f) = 0$ for every pair of edges $(f, v_i), (v_i, f) \in D_V$ stemming from the same occurrence of v in the boundary of f.

There is a bijection between all normalized orthogonal representations of G and all normalized flows in N and this bijection can be computed efficiently. Hence, we can simply add several restrictions or optimization criteria to the flow network and test whether there exists a normalized orthogonal representation with these restrictions by simply testing whether there exists a feasible flow in N or optimize the resulting drawing by solving a minimum cost flow problem. For example one could set the cost for every unit of flow over an edge in D_F to 1. The resulting flow with minimum cost would yield an orthogonal drawing that minimizes the total amount of bends on edges.

2.2.2 Using the Flow Network to Compute the Maximum Rotation

Now we use the flow network described in the previous section to compute $\text{maxrot}_{\mathcal{E}}(G)$ for a given st-graph G with fixed planar embedding $\mathcal{E}$. First we ensure that the constructed

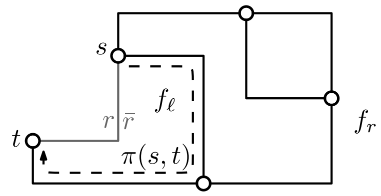


Figure 10: If $\mathrm{rot}(r)$ is maximized, $\mathrm{rot}(\pi(s,t))$ is also maximized and the angles at s and t in f_ℓ are both 90°.

orthogonal representations are valid, i.e., no edge has more bends than its flexibility allows. Thus we set the capacity of every edge $(f_u, e, f_v) \in D_F$ to $\mathrm{flex}(e)$. Since we wish to maximize the rotation on $\pi(s,t)$ we could add costs of -1 to edges (f_u, e, f_1) and costs of 1 to edges (f_1, e, f_u) where f_1 is the outer face and e is an edge on the path from s to t. By solving this minimum cost flow problem we either compute a valid orthogonal representation of G with planar embedding $\mathcal{E}$ that maximizes $\mathrm{rot}(\pi(s,t))$ or decide that such an orthogonal representation does not exist.

The same result can be achieved faster if we use only maximum flows instead of minimum cost flows and exploit the planarity of N to compute these maximum flows. The resulting running time is stated in the following theorem.

Theorem 2. *Given a weak st-graph G with fixed planar embedding $\mathcal{E}$ with s and t on the outer face we can compute* $\mathrm{maxrot}_{\mathcal{E}}(G)$ *in $\mathcal{O}(n^{3/2})$ time or decide that G does not admit a valid orthogonal representation with this embedding.*

Proof. We add to G the edge st and embed it into the outer face such that we split it into two parts f_ℓ and f_r where f_ℓ is bounded by $\pi(s,t)$ and st and f_r is the outer face of $G+st$. We claim that in a valid orthogonal representation of $G+st$ that maximizes $\mathrm{rot}(r)$ with its embedding we have that $\mathrm{maxrot}_{\mathcal{E}}(G) = \mathrm{rot}(r) + 2$, where r is the edge description of st in the outer face f_r. Figure 10 illustrates this claim and its proof.

The total rotation around the face f_ℓ is 4 where the rotation in the vertices s and t is at most 1. This yields the inequality $\mathrm{rot}(\bar{r}) + \mathrm{rot}(\pi(s,t)) \geq 2$. Since $\mathrm{rot}(r) = -\mathrm{rot}(\bar{r})$ we get $\mathrm{rot}(\pi(s,t)) \geq \mathrm{rot}(r) + 2$ and hence $\mathrm{maxrot}_{\mathcal{E}}(G) \geq \mathrm{rot}(r) + 2$. On the other hand, by Theorem 1, we can find a tight orthogonal representation of G with $\mathrm{rot}(\pi(s,t)) = \mathrm{maxrot}_{\mathcal{E}}(G)$. We can add the edge st such that the angles at s and t in f_ℓ are 90°. Thus, their rotation is 1 and we get for r a rotation of $\mathrm{maxrot}_{\mathcal{E}}(G) - 2$. Hence we obtain the inequality $\mathrm{maxrot}_{\mathcal{E}}(G) \leq \mathrm{rot}(r) + 2$ which yields the claim.

It remains to show that we can maximize $\mathrm{rot}(r)$ efficiently. First, we construct the flow network with capacity constraints as described above ensuring that the flow over every edge is not greater than its flexibility (where we get infinite capacity for the edges corresponding to st). Then we can compute a flow with multiple sinks and sources to compute a valid orthogonal representation of G or decide that there is no such representation. Since the flow network is planar and the in- and out-flow of the sinks and sources is fixed this can be done in $\mathcal{O}(n^{3/2})$ time [MN95].

Once we have a flow yielding a valid orthogonal representation we wish to maximize the flow over the edge (f_r, st, f_ℓ). We delete the edges (f_r, st, f_ℓ) and (f_ℓ, st, f_r) from the residual network. To maximize the flow on (f_r, st, f_ℓ) we wish to find as many augmenting paths as possible from f_ℓ to f_r. Hence we simply compute a maximum flow from f_ℓ to f_r in the residual network without the edges stemming from st. Since this network is planar and the source f_ℓ and the sink f_r lie on the same face a maximum flow can be computed in $\mathcal{O}(n)$ time [HKRS97]. □

Until now we considered the possible values for $\mathrm{rot}(\pi(s,t))$ we obtain for a given planar graph with positive flexibility and fixed planar embedding. We showed that at least all values between the maximum rotation and -1 can be achieved. Hence, the maximum rotation gives a good hint on how a plane graph behaves and we provided an algorithm computing it efficiently. In the following we extend these results to graphs with variable embeddings.

3. Biconnected Graphs with Variable Embedding

Until now the planar embedding of our input graph was fixed. Now, we assume that this embedding is variable. Following the approach of the previous section, we define the maximum rotation of an st-graph G as $\operatorname{maxrot}(G) = \max_{\mathcal{E} \in \Psi} \operatorname{maxrot}_{\mathcal{E}}(G)$ where Ψ contains all planar embeddings of G such hat s and t are embedded on the outer face.

In Section 3.1 we will show that the possible values for $\operatorname{rot}(\pi(s,t))$ form an interval depending only on $\operatorname{maxrot}(G)$, $\deg(s)$ and $\deg(t)$. In Section 3.2 we will show how to exploit this property to replace subgraphs with other subgraphs without changing $\operatorname{maxrot}(G)$. This leads to the definition of very simple gadgets that can model the behaviour of all possible weak st-graphs. In Section 3.4 we will use the SPQR-tree introduced in Section 3.3 to compute $\operatorname{maxrot}(G)$ for st-graphs or decide that G does not admit a valid orthogonal representation, where the replacement of subgraphs with our gadgets reduces the number of embeddings from exponentially to linearly many. This can be used to solve FlexDraw for biconnected graphs.

3.1 Possible Rotation Values

We restrict our considerations to st-graphs where $\deg(s), \deg(t) \leq 2$. We say that such an st-graph is of Type (1,1) if $\deg(s) = \deg(t) = 1$, it is of Type (1,2) if s or t has degree 1 and the other one has degree 2 and it is of Type (2,2) if $\deg(s) = \deg(t) = 2$. We get the following technical lemma.

Lemma 3. *Let G be an st-graph with $\deg(s), \deg(t) \leq 2$ and let $\mathcal{R}$ be a tight orthogonal representation of G. Then $\operatorname{rot}(\pi(s,t)) + \operatorname{rot}(\pi(t,s)) = -x$ where x is 0,1 and 2 for graphs of Type (1,1), (1,2) and (2,2), respectively.*

Proof. Since $\mathcal{R}$ satisfies Property II we have that $\operatorname{rot}(\pi(s,t)) + \operatorname{rot}(t) + \operatorname{rot}(\pi(t,s)) + \operatorname{rot}(s) = -4$. If s has degree 1 we have that $\operatorname{rot}(s) = -2$. If $\deg(s) = 2$ holds then s is incident to exactly one inner face and by assumption it has an angle of $90°$ in this face. Hence, in the outer face there is an angle of $270°$ and thus $\operatorname{rot}(s) = -1$. As the same analysis holds for t the claim follows. □

The following theorem shows that the possible rotation values of an st-graph form an interval that is nearly symmetrically to 0.

Theorem 3. *Let G be an st-graph with positive flexibility and let ρ be an integer. Then there exists a valid and tight orthogonal representation $\mathcal{R}$ of G with* $\mathrm{rot}(\pi(s,t)) = \rho$ *if and only if* $-\mathrm{maxrot}(G) - x \leq \rho \leq \mathrm{maxrot}(G)$ *where x depends on the Type of G and $x = 0, 1, 2$ for Types (1,1), (1,2) and (2,2), respectively.*

Proof. We first show the only if part. Let $\mathcal{R}$ be any orthogonal representation of G. By the definition of $\mathrm{maxrot}(G)$ we clearly have that $\mathrm{rot}_{\mathcal{R}}(\pi(s,t)) \leq \mathrm{maxrot}(G)$. By definition we also have that $\mathrm{rot}_{\mathcal{R}}(\pi(t,s)) \leq \mathrm{maxrot}(G)$ (otherwise by mirroring we could obtain an orthogonal representation $\mathcal{R}'$ with $\mathrm{rot}_{\mathcal{R}'}(\pi(s,t)) > \mathrm{maxrot}(G)$) and hence with Lemma 3 we obtain $-\mathrm{rot}(\pi(s,t)) - x \leq \mathrm{maxrot}(G)$.

It remains to show that for any given ρ in the range we can find a valid and tight orthogonal representation. By definition of $\mathrm{maxrot}(G)$ it is clear that we find a planar embedding $\mathcal{E}$ such that $\mathrm{maxrot}_{\mathcal{E}}(G) = \mathrm{maxrot}(G)$. Thus, if $-1 \leq \rho \leq \mathrm{maxrot}(G)$, by Theorem 1 we find a valid and tight orthogonal representation such that $\mathrm{rot}(\pi(s,t)) = \rho$. If $\rho \leq -2$, by Lemma 3 we need to find a valid orthogonal representation $\mathcal{R}$ with $\mathrm{rot}_{\mathcal{R}}(\pi(t,s)) = -\rho - x =: \rho'$. Note that by the definitions of ρ and x we have that $0 \leq \rho' \leq \mathrm{maxrot}(G)$. Thus by Theorem 1, we obtain a valid orthogonal representation $\mathcal{R}'$ of G with $\mathrm{rot}_{\mathcal{R}'}(\pi(s,t)) = \rho'$. We obtain the desired orthogonal representation $\mathcal{R}$ by mirroring $\mathcal{R}'$. □

Corollary 1. *Let G be an st-graph with positive flexibility. If G admits a valid orthogonal drawing then* $\mathrm{maxrot}(G) \geq 1$ *if G is of Type (1,1) or (1,2) and* $\mathrm{maxrot}(G) \geq -1$ *if G is of Type (2,2).*

Proof. If s or t has degree 1 and G admits a valid orthogonal representation, then the range of all possible rotations covers at least three integers, since the edge incident to s or t, respectively, can have the rotation -1, 0 and 1. If $\mathrm{maxrot}(G) < 1$ for graphs of Type (1,1) or (1,2) Theorem 3 yields a range that covers less then three different integers (or is completely degenerated), which contradicts the assumption that G admits a valid orthogonal representation. If $\mathrm{maxrot}(G) < -1$ and G is of Type (2,2) then Theorem 3 yields an empty range, which again contradicts the assumption that G admits a valid orthogonal representation. □

In this section we saw that the possible values for $\mathrm{rot}(\pi(s,t))$ form an interval that is nearly symmetric around 0. Note that the "nearly" symmetric stems only from the way how we measure the rotation: $\mathrm{rot}(\pi(s,t))$ is not exactly the negation of $\mathrm{rot}(\pi(t,s))$ if $\deg(s)$ and $\deg(t)$ are not both 1. Now, considering for example a graph of Type (1,1), we find that this graph behaves exactly like a single edge with flexibility $\mathrm{maxrot}(G)$.

3.2 Replacing Subgraphs

In Section 3.1, we have seen, that the possible values for the rotation on the path from s to t only depend on the maximum rotation and the degrees of s and t. Thus, two graphs of the same type with the same maximum rotation are in a sense equivalent. If we consider for example a graph of Type (1,1), i.e., s and t have degree 1, with maximum rotation ρ, then we have that the rotation can take all values between $-\rho$ and ρ. Hence the whole graph behaves similar to a single edge with flexibility ρ. We will now use this equivalence to show that we can replace subgraphs with other graphs of the same type with the same maximum rotation without changing the maximum rotation of the whole graph. Afterwards, we will give three very simple families of graphs, one for each Type of st-graph, called *gadgets*. These gadgets extend the idea of replacing graphs of Type (1,1) with single edges to graphs of Types (1,2) and (2,2).

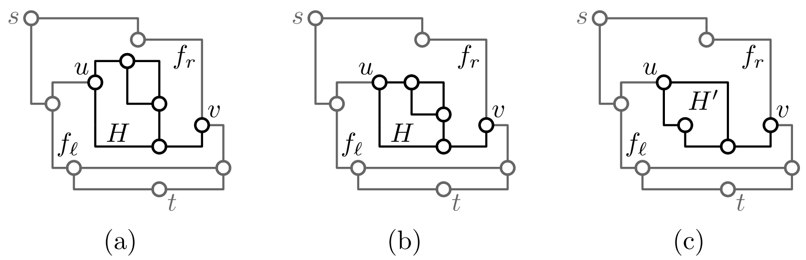


Figure 11: Illustration of Lemma 4, st-graph G with split pair $\{u, v\}$ splitting off H (a), replacement of H with a tight orthogonal representation (b) and replacement of H with a graph H' with $\operatorname{maxrot}(H) = \operatorname{maxrot}(H') = 3$ (c).

A pair of vertices $\{s, t\}$ is called a *split pair* if st is an edge in G or if $\{s, t\}$ is a separation pair. The *split components* with respect to the split pair $\{s, t\}$ are the maximal subgraphs H_i of G such that H_i contains s and t but $\{s, t\}$ is not a separation pair in H_i. For example $\{u, v\}$ in Figure 11a is a split pair and the black and gray subgraphs are the split components with respect to $\{u, v\}$.

Lemma 4. *Let G be an st-graph with positive flexibility and let $\{u, v\}$ be a split pair of G with a split component H such that the union of all other split components G^- contains s and t and H is an st-graph of Type (1,1), Type (1,2) or Type (2,2) (with respect to the vertices u and v). Let H' be a graph with designated vertices u', v' of the same type as H with* $\operatorname{maxrot}(H') = \operatorname{maxrot}(H)$.

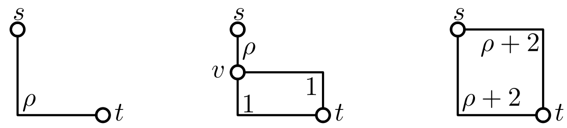
Then G admits a valid orthogonal representation $\mathcal{R}$ with $\operatorname{rot}_{\mathcal{R}}(\pi(s, t)) = \rho$ if and only if the graph G' that is obtained from G by replacing H with H' admits a valid orthogonal representation $\mathcal{R}'$ with $\operatorname{rot}_{\mathcal{R}'}(\pi(s, t)) = \rho$.

Proof. Given a valid orthogonal representation $\mathcal{R}$ of G we wish to find a valid orthogonal representation $\mathcal{R}'$ of G' such that $\operatorname{rot}_{\mathcal{R}'}(\pi(s, t)) = \operatorname{rot}_{\mathcal{R}}(\pi(s, t))$. The other direction is symmetric.

We first treat the case that H is of Type (1,1). Let $\mathcal{S}$ be the restriction of $\mathcal{R}$ to H. By Theorem 3 we have that $\operatorname{rot}_{\mathcal{S}}(\pi(u, v)) \in \{-\operatorname{maxrot}(H), \dots, \operatorname{maxrot}(H)\}$ and hence, again by Theorem 3, there exists a valid orthogonal representation $\mathcal{S}'$ of H' with $\operatorname{rot}(\pi(u', v')) = \operatorname{rot}(\pi(u, v))$. Since H is of Type (1,1) we have that $\operatorname{rot}_{\mathcal{S}'}(u') = \operatorname{rot}_{\mathcal{S}}(u)$, $\operatorname{rot}_{\mathcal{S}'}(v') = \operatorname{rot}_{\mathcal{S}}(v)$, $\operatorname{rot}_{\mathcal{S}'}(\pi(u', v')) = \operatorname{rot}_{\mathcal{S}}(\pi(u, v))$ and $\operatorname{rot}_{\mathcal{S}'}(\pi(v', u')) = \operatorname{rot}_{\mathcal{S}}(\pi(v, u))$. Hence by plugging $\mathcal{S}'$ into the restriction of the orthogonal representation $\mathcal{R}$ to G^- we obtain the desired representation $\mathcal{R}'$ of G'.

In the case where H is of Type (1,2) we can assume that u has degree 2 and $\deg(v) = 1$. Then the angle at u in f_i is 90° or 180° where f_i is the inner face of H incident to u. If this angle is 90°, i.e., $\mathcal{S}$ is tight, we replace it by a corresponding tight representation of H' with the same rotation, which exists by Theorem 3. For the case where we have an angle of 180° at u in f_i we show how to construct an orthogonal representation $\mathcal{R}^+$ of G having the same planar embedding as $\mathcal{R}$ such that $\operatorname{rot}_{\mathcal{R}^+}(\pi(s, t)) = \operatorname{rot}_{\mathcal{R}}(\pi(s, t))$ and the angle at u in f_i is 90°. Then $\mathcal{R}'$ can be constructed from $\mathcal{R}^+$ as above. These two steps are illustrated in Figure 11

By Lemma 2 there exists a valid and tight orthogonal representation $\mathcal{S}^+$ of H with either $\operatorname{rot}_{\mathcal{S}^+}(\pi(u, v)) = \operatorname{rot}_{\mathcal{S}}(\pi(u, v))$ or $\operatorname{rot}_{\mathcal{S}^+}(\pi(v, u)) = \operatorname{rot}_{\mathcal{S}}(\pi(v, u))$. Without loss of generality assume the former, the other case is symmetric. Since we have increased the outer angle at u we have that $\operatorname{rot}_{\mathcal{S}^+}(u) = \operatorname{rot}_{\mathcal{S}}(u) - 1$ and hence $\operatorname{rot}_{\mathcal{S}^+}(\pi(v, u)) = \operatorname{rot}_{\mathcal{S}}(\pi(v, u)) + 1$. Let f_ℓ and f_r be the faces in G whose boundaries contain $\pi(u, v)$ and $\pi(v, u)$, respectively.



(a) $G^{\rho}_{1,1}$ (b) $G^{\rho}_{1,2}$ (c) $G^{\rho}_{2,2}$

Figure 12: Gadgets for st-graphs with maximum rotation ρ depending on the Type.

Then we obtain $\mathcal{R}^+$ by plugging $\mathcal{S}^+$ into the restriction of $\mathcal{R}$ to G^- such that the angle at u in f_r is increased by 90° to 180°. Since the angle at u in f_i was decreased by 90° the sum of angles around u remains 360°. Additionally, by increasing the angle at u in f_r, its rotation is decreased by 1, which compensates the increased rotation along $\pi(v, u)$. Hence $\mathcal{R}^+$ is the claimed orthogonal representation. This finishes the treatment of graphs of Type (1,2). Graphs of Type (2,2) can be treated analogously. □

In Section 3.1, we characterized which values are possible for the rotation on the path from s to t. These possible values form an interval depending only on the maximum rotation and the type of the graph. With Lemma 4 we know that this rotation characterizes a graph, i.e., the replacement of subgraphs with other graphs having the same maximum rotation and type does not change the maximum rotation of the whole graph.

We now present the three gadgets, for st-graphs of Type (1,1), Type (1,2) and Type (2,2); see Figure 12. Let ρ be an integer. The graph $G^{\rho}_{1,1}$ is simply an edge st with $\mathrm{flex}(st) = \rho$. The graph $G^{\rho}_{1,2}$ has three vertices s, v, t and two edges between t and v, both with flexibility 1, and the edge vs with flexibility ρ. The gadget $G^{\rho}_{2,2}$ consists of two parallel edges between s and t, both with flexibility $\rho + 2$. Note that by Corollary 1 all edges of our gadgets have again positive flexibility and that $\mathrm{maxrot}(G^{\rho}_{1,1}) = \mathrm{maxrot}(G^{\rho}_{1,2}) = \mathrm{maxrot}(G^{\rho}_{2,2}) = \rho$. Moreover, each of these graphs has an essentially unique embedding with s and t on the outer face.

3.3 The SPQR-Tree

Let G be a biconnected planar graph. Recall that a pair of vertices $\{s, t\}$ is called a split pair if st is an edge in G or if $\{s, t\}$ is a separation pair. The split components with respect to the split pair $\{s, t\}$ are the maximal subgraphs H_i of G such that H_i contains s and t but $\{s, t\}$ is not a separation pair in H_i; see Figure 13 for an example. The idea behind the SPQR-Tree introduced by Di Battista and Tamassia [DT96a, DT96b] is the following. Given a biconnected planar graph G and a split pair $\{s, t\}$ one can replace every split component by a single edge called *virtual edge* associated with the split component, the resulting graph is called a *skeleton* and can be seen as a sketch of the graph. Then all

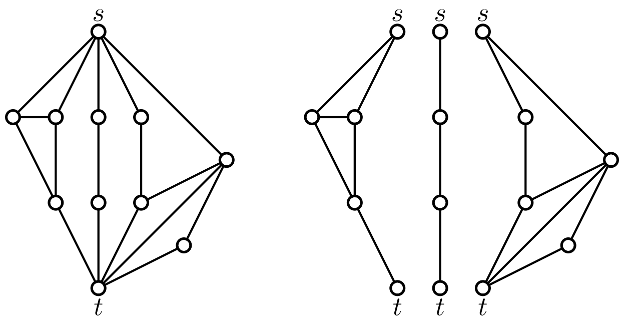


Figure 13: A graph with the split pair $\{s, t\}$ on the left and its corresponding split components on the right.

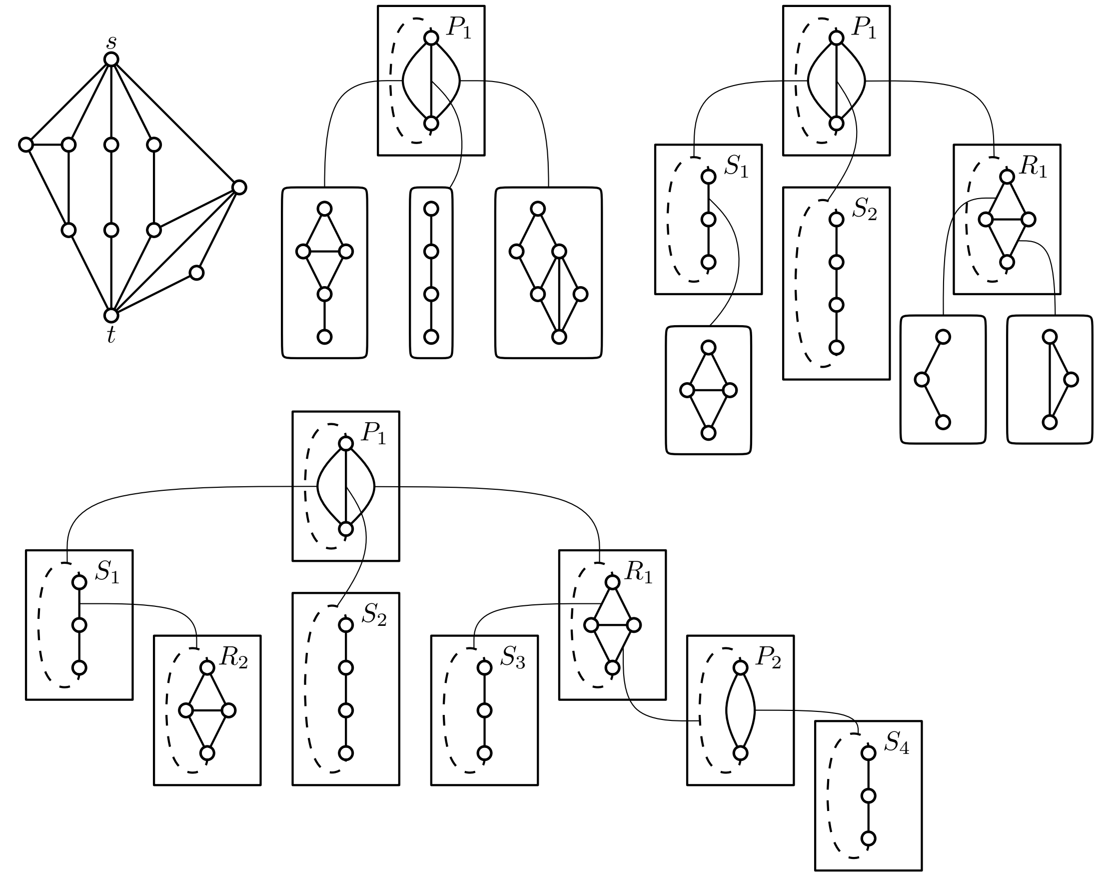


Figure 14: Since $\{s, t\}$ is a split pair of the graph in the top left we obtain the P-node P_1 with one subgraph associated with every edge in skel(P_1). Further decomposition of these subgraphs yields the S-nodes S_1 and S_2 and the R-node R_1. The resulting SPQR-tree is shown on the bottom. Note that the Q-nodes are omitted and the edges associated with the parent are depicted as dashed line.

planar embeddings of G having s and t on the outer face are obtained by combining all embeddings of the skeleton with the embeddings of the split components represented by the virtual edges such that every split component has s and t on the outer face.

For a graph G with two designated vertices s and t, called the *poles* of G, such that $G+st$ is biconnected, we obtain the SPQR-tree $\mathcal{T}$, representing all planar embeddings of G having s and t on the outer face, by computing the root μ of $\mathcal{T}$, i.e., computing the skeleton skel(μ) and graphs $H_1, \ldots, H_k$ associated with the virtual edges in skel(μ). The children of μ in $\mathcal{T}$ are obtained by computing the SPQR-tree for each subgraph H_i recursively. Four different cases can be distinguished; see Figure 14 for an example of the recursive decomposition of a graph, yielding its SPQR-tree.

Base Case: If G consists of a single edge from s to t then $\mathcal{T} = \mu$ is a single Q-node whose skeleton is G itself.
Series Case: If G is not biconnected μ is an S-node. Let $v_2, \ldots, v_k$ be the cutvertices ordered from s to t, set $v_1 = s$ and $v_{k+1} = t$ and let H_i be the block containing v_i and v_{i+1}. Then skel(μ) contains the k virtual edges $v_i v_{i+1}$ for $i = 1, \ldots, k$, each associated with the SPQR-Tree of H_i (with respect to its poles v_i, v_{i+1}) and additionally it contains the edge st associated with the parent of μ.
Parallel Case: If $\{s, t\}$ is a split pair in G with the split components $H_1, \ldots, H_k$ then μ is a P-node and skel(μ) consist of the two nodes s and t with k parallel edges each of them associated with one component H_i plus one additional edge from s to t associated with

the parent of μ.
Rigid Case: If none of the above cases applies μ is an R-node and we consider the *maximal split pairs* $\{u_1, v_1\}, \ldots, \{u_k, v_k\}$ with respect to s and t. A split pair $\{u, v\}$ is called maximal if for all other split pairs $\{u', v'\}$ the vertices u and v are in the same split component together with s or t. Let H_i be the union of all split components concerning $\{u_i, v_i\}$ that contain neither s nor t except from the case where s or t is part of the split pair. Then skel(μ) is obtained by replacing every H_i with the edge u_iv_i associated with the subgraph H_i and adding the edge st associated with the parent of μ. In the next recursion step H_i has the poles u_i and v_i. Note that skel(μ) is 3-connected.

If μ is an S-, P- or R-node, it has virtual edges $e_1, \ldots, e_k$ associated with its children $\mu_1, \ldots, \mu_k$ such that μ_i is the root of the SPQR-tree of the graph H_i. We call H_i the *pertinent graph* of μ_i and the *expansion graph* of e_i. We denote the pertinent graph of a node μ by pert(μ).

The resulting tree $\mathcal{T}$ is rooted and we obtain all embeddings with s and t on the outer face by combining all planar embeddings of skel(μ) having the virtual edge associated with the parent (depicted as dashed lines in Figure 14) on the outer face for every node μ in $\mathcal{T}$. By choosing an other pair of nodes s and t as initial split pair we obtain the same tree with an other root. Since there is exactly one Q-node for every edge e one could choose this Q-node as root to represent all planar embeddings with e on the outer face, thus we obtain all planar embeddings by rooting $\mathcal{T}$ in every Q-node once. Hence, to obtain all planar embeddings of the graph G it is sufficient to consider the embeddings of skel(μ) for the nodes μ in $\mathcal{T}$. If μ is an S-node, skel(μ) has only one embedding. If it is a P-node, it consists of k parallel edges and one additional edge fixed on the outer face representing the parent. Hence, we obtain all planar embeddings of skel(μ) by choosing a permutation for the k parallel edges. Thus, there are $k!$ different planar embeddings. If μ is an R-node its skeleton is 3-connected and the edge associated with the parent is fixed to the outer face. Hence, we have a unique embedding except for flipping around its poles.

Note that the SPQR-tree $\mathcal{T}$ of a planar graph G with n vertices has $\mathcal{O}(n)$ nodes and the total size of all skeletons is in $\mathcal{O}(n)$. Moreover, the SPQR-tree of G can be computed in linear time [GM01].

3.4 Solving FlexDraw for Biconnected Graphs

In Section 2.2 we have seen how to compute the maximum rotation for graphs with fixed embedding. We will now use the SPQR-tree to compute the maximum rotation for st-graphs with variable embedding. By identifying every edge in an arbitrary biconnected 4-planar graph with st we can decide whether the graph admits a valid orthogonal representation, i.e., we solve FlexDraw for biconnected graphs with positive flexibility.

We first describe an algorithm that computes maxrot(G) for a given 4-planar st-graph G with positive flexibility or decides that G does not admit a valid orthogonal representation. We use the SPQR-tree $\mathcal{T}$ of $G + st$, rooted at the Q-node corresponding to st to represent all planar embeddings of G with s and t on the outer face. Our algorithm processes the nodes of the SPQR-tree in a bottom-up fashion and computes the maximum rotation of each pertinent graph from the maximum rotations of the expansion graphs of its edges, i.e., the pertinent graphs of its children. For each node μ of the SPQR-tree we maintain a variable maxrot(μ). We will prove later that after processing a node we have that maxrot(μ) = maxrot(pert(μ)). For each Q-node μ we initialize maxrot(μ) to be the flexibility of the corresponding edge. We now show how to compute maxrot(μ) from the maximum rotations of its children. We make a case distinction based on the type of μ.

If μ **is an R-node** let $\mu_1, \dots, \mu_k$ be the children of μ corresponding to the virtual edges $e_1, \dots, e_k$ in skel(μ) and let $H_1, \dots, H_k$ be their pertinent graphs. Each virtual edge $e_i = (v_i, v_i')$ represents at least one incidence in G to v_i and v_i'. Since skel(μ) is 3-connected each node in skel(μ) has at least degree 3 and hence, since G is 4-planar, no virtual edge can represent more than two incidences, i.e., the poles v_i and v_i' have at most degree 2 in the pertinent graph H_i of μ_i. Hence, pert(μ_i) is of Type (1,1), (1,2) or (2,2). As we already know their maximum rotations we can simply replace each of the graphs by a corresponding gadget; we call the resulting graph G_μ. Since the embeddings of all gadgets are completely symmetric it is sufficient to compute the maximum rotations of G_μ for the only two embeddings $\mathcal{E}_1$ and $\mathcal{E}_2$ induced by the embeddings of skel(μ). We set $\text{maxrot}(\mu) = \max\{\text{maxrot}_{\mathcal{E}_1}(G_\mu), \text{maxrot}_{\mathcal{E}_2}(G_\mu)\}$ if one of them admits a valid representation. Otherwise we stop and return "infeasible".

If μ **is a P-node** we treat μ similar as in the case where μ is an R-node. Again, we have that each pole has degree at least 3 in skel(μ) and hence no virtual edge can represent more than two edge incidences. We replace each virtual edge with the corresponding gadget and try all possible embeddings of skel(μ), which are at most six, and store the maximum rotation or stop if none of the embeddings admits a valid representation.

If μ **is an S-node** let $\mu_1, \dots, \mu_k$ be the children of μ. We compute maxrot(μ) as

$$\text{maxrot}(\mu) = \sum_{i=1}^{k} \text{maxrot}(\mu_i) + k - 1$$

Theorem 4. *Given a 4-planar st-graph G with positive flexibility we can compute* maxrot(G) *in $O(n^{3/2})$ time or decide that G does not admit a valid orthogonal representation with s and t on the outer face.*

Proof. We prove the invariant that after processing the node μ we have maxrot(μ) = maxrot(pert(μ)). The proof is by induction on the height h of the SPQR-tree $\mathcal{T}$ of $G + st$, rooted at the Q-node corresponding to st. Let μ be the root of $\mathcal{T}$.

If $h = 1$ then G is a single edge e and μ its corresponding Q-node. Since maxrot(G) = flex(e) the claim holds. For $h > 1$ let $\mu_1, \dots, \mu_k$ be the children of μ. By induction we have that maxrot(μ_i) = maxrot(pert(μ_i)) for $i = 1, \dots, k$. We make a case distinction based on the type of μ.

If μ **is an R- or a P-node** then by Lemma 4 we have that maxrot(G_μ) = maxrot(pert(μ)) and since the gadgets have a unique embedding we consider all relevant embeddings of G_μ. Due to Corollary 1 all edges in G_μ have positive flexibility thus we can compute maxrot(G_μ) efficiently with Theorem 2. If none of the embeddings admits a valid orthogonal representation then obviously also pert(μ) and thus G do not admit valid orthogonal representations.

If μ **is an S-node** and the pertinent graphs of its children admit valid orthogonal representations then there always exists a valid orthogonal representation of pert(μ). Let $H_1, \dots, H_k$ be the pertinent graphs of the children of μ and let $v_1, \dots, v_{k+1}$ be the vertices in skel(μ) such that v_i and v_{i+1} are the poles of H_i. By the definition of maxrot and by Theorem 1 there exist tight orthogonal representations $\mathcal{R}_1, \dots, \mathcal{R}_k$ of $H_1, \dots, H_k$ with rot($\pi(v_i, v_{i+1})$) = maxrot(μ_i). We put these orthogonal representations together such that the angles at the nodes $v_2, \dots, v_k$ on $\pi(v_1, v_{k+1})$ are 90°. Hence, we get an orthogonal representation of pert(μ) with rot($\pi(v_1, v_{k+1})$) = $\sum_{i=1}^{k}$ maxrot(μ_i) + $k - 1$. On the other hand if we had an orthogonal representation of pert(μ) with a higher rotation, then at least one of its children μ_i would need to have a rotation that is bigger than maxrot(μ_i). This would contradict the assumption that maxrot(μ_i) was computed correctly for each child of μ.

This proves the correctness of the algorithm. For the running time note that the SPQR-tree can be computed in linear time [GM01]. We can compute $\operatorname{maxrot}(\mu)$ for a given node μ from the maximum rotations of its children in $O(|\operatorname{skel}(\mu)|^{3/2})$ time by Theorem 2 since each skeleton has only a constant number of embeddings. The total running-time follows from the fact that the total size of all skeletons is in $O(n)$. $\square$

This theorem can be used to solve FlexDraw for biconnected 4-planar graphs with positive flexibility. Such a graph G admits a valid orthogonal representation if and only if one of the graphs $G - e$, $e \in E(G)$, which is an st-graph with respect to the endpoints of e, admits a valid orthogonal representation such that e can be added to this representation. This can be done if and only if $\operatorname{maxrot}(G - e) + \operatorname{flex}(e) \geq 2$. This can be seen as follows. Let s and t be the endpoints of e. Adding e to $G - e$ creates a new interior face and the total rotation of this new face needs to be 4. We can have at most two $90°$ angles at s and t, hence $\operatorname{maxrot}(G - e) + \operatorname{flex}(e) \geq 2$ is a necessary condition. On the other hand, it is not hard to see that it is possible to add e to a tight orthogonal representation of $G - e$ with $\operatorname{rot}(\pi(s, t)) + \operatorname{flex}(e) \geq 2$. If $\operatorname{flex}(e) \geq 3$ then we can add e to a tight orthogonal representation of $G - e$ with $\operatorname{rot}(\pi(s, t)) = -1$, which exists since $\operatorname{maxrot}(G)$ is at least -1 by Corollary 1. Otherwise, we add e to a tight orthogonal representation of $G - e$ with $\operatorname{rot}(\pi(s, t)) = 2 - \operatorname{flex}(e)$, which is possible since $2 - \operatorname{flex}(e) \geq -1$ holds in this case. We obtain the following theorem; the running time is due to $O(n)$ applications of the algorithm for st-graphs.

Theorem 5. FlexDraw *can be solved in $O(n^{5/2})$ time for biconnected 4-planar graphs with positive flexibility.*

4. Generalization to Connected Graphs

In this section we generalize our results to connected 4-planar graphs that are not necessarily biconnected. To do that we first analyze for a single cutvertex which properties every cut component needs to have such that we can find a valid orthogonal representation of the whole graph. Afterwards we will use the *block-cutvertex tree* to decide whether a connected 4-planar graph admits a valid orthogonal drawing.

Lemma 5. *Let G be a connected 4-planar graph with cutvertex v and corresponding cut components $H_1, \ldots, H_k$. Then G admits a valid orthogonal representation if and only if all cut components H_i have valid orthogonal representations such that at most one of them does not have v on the outer face.*

Proof. The only if part is clear since a valid orthogonal representation of G induces valid orthogonal representations of all cut components H_i such that at most one of them does not have v on its outer face.

Now, for $i = 1, \ldots, k$ let $\mathcal{S}_i$ be valid orthogonal representations of the cut components H_i such that at most one of them does not have v on its outer face.

If all of them have v on their outer face then by Lemma 2 we can assume that these representations are tight. Then it is clear that the components $H_1, \ldots, H_k$ can be merged together in v maintaining their representations $\mathcal{S}_i$.

Otherwise, one of the representations, without loss of generality $\mathcal{S}_1$, does not have v on the outer face. If v has degree greater than 1 in at most one component we can simply merge the corresponding representations as bridges can always be added. The only problem that can arise is that there are exactly two components H_1 and H_2 such that v has degree 2 in both of them and both angles incident to v in H_1 are $180°$. We resolve this situation by either increasing or decreasing the number of bends of an incident edge and changing the angles at v appropriately. □

Recall that the maximal biconnected components of a graph are called blocks. The block-cutvertex tree of a connected graph is a tree whose nodes are the blocks and cutvertices of the graph. In the block-cutvertex tree a block B and a cutvertex v are joined by an edge if v belongs to B. Now let G be a connected 4-planar graph with positive flexibility and $\mathcal{B}$ its block-cutvertex tree. Let further B be a block of G that is a leaf in $\mathcal{B}$ and let v be the unique cutvertex of B.

If B is the whole graph G we return "true" if and only if G admits any valid orthogonal representation. This can be checked with the algorithm from the previous Section.

If B is not the whole graph G we check whether B admits a valid orthogonal representation having v on its outer face. This can be done with the algorithm from the previous section by rooting the SPQR-tree of B at all edges incident to v. If it does admit such an representation then by Lemma 5 G admits a valid orthogonal representation if and only if the graph G', which is obtained from G by removing the block B, admits a valid orthogonal representation. We check G' recursively. If B does not admit such an representation we mark B and proceed with another unmarked leaf. If we ever encounter another block B' that has to be marked we return "infeasible". This is correct as in this case B has to be embedded in the interior of B' and vice versa, which is obviously impossible. Checking a single block B can be done in $O(|B|^{5/2})$ time by Theorem 5. Since the total size of all blocks is in $O(n)$ the total running-time is $O(n^{5/2})$. This proves the following theorem.

Theorem 6. FlexDraw *can be solved in* $O(n^{5/2})$ *time for 4-planar graphs with positive flexibility.*

5. Conclusion

In this work we have considered the problem FLEXDRAW, which deals with the question if a 4-planar graph with a given flexibility function $\text{flex}(\cdot)$ admits an orthogonal drawing such that each edge e has at most $\text{flex}(e)$ bends.

We have shown that FLEXDRAW can be solved efficiently for graphs with positive flexibility by computing the maximum rotation for every pair of adjacent vertices. This computation relies on the fact that the possible rotation values for every graph form an interval around 0 and hence every graph behaves similar, depending only on the maximum rotation. This fact followed from the impossibility to construct rigid graphs. With this knowledge we were able to compute the maximum rotation considering all planar embeddings by traversing the SPQR-tree, where the replacement of processed subgraphs by simple gadgets with the same behavior reduced the number of embeddings we really had to consider to linearly many. For every embedding we considered we used a variant of Tamassia's flow network to compute the maximum rotation or decide that the graph does not admit a valid orthogonal representation with respect to the fixed planar embedding. Since the SPQR-tree can only be used for biconnected graphs we finally used the BC-tree to extend our result to not necessarily biconnected graphs. The resulting algorithm solves FLEXDRAW in $O(n^{5/2})$ time. This running time stems from linearly many flow computations.

A straightforward extension to the described algorithm would be the generalization to positive flexibility functions $\text{flex} : E \longrightarrow \mathbb{N} \cup \{\infty\}$, i.e., some edges may be bent arbitrarily often. Since FLEXDRAW can be solved in polynomial time for graphs with positive flexibility but is $\mathcal{NP}$-hard if $\text{flex}(e) = 0$ for all edges e [GT01], it is an interesting open question whether FLEXDRAW can still be solved in polynomial time if few edges are required to have no bends. For example one could try to use the results for 0-embeddability concerning graphs with maximum degree 3 and series-parallel graphs [DLV98] to solve FLEXDRAW if the subgraph induced by the edges with flexibility 0 has maximum degree 3 or is series-parallel. Or even simpler, one could require only a tree, forest, matching or constantly many edges to have 0 bends.

Bibliography

[BDD00] P. Bertolazzi, G. Di Battista, and W. Didimo, "Computing orthogonal drawings with the minimum number of bends," *IEEE Transactions on Computers*, vol. 49, no. 8, pp. 826–840, 2000.

[BK94] T. Biedl and G. Kant, "A better heuristic for orthogonal graph drawings," in *Algorithms – ESA '94*, ser. Lecture Notes in Computer Science, vol. 855. Springer, 1994, pp. 24–35.

[DLV98] G. Di Battista, G. Liotta, and F. Vargiu, "Spirality and optimal orthogonal drawings," *SIAM Journal on Computing*, vol. 27, no. 6, pp. 1764–1811, 1998.

[DT96a] G. Di Battista and R. Tamassia, "On-line maintenance of triconnected components with spqr-trees," *Algorithmica*, vol. 15, no. 4, pp. 302–318, 1996.

[DT96b] G. Di Battista and R. Tamassia, "On-line planarity testing," *SIAM Journal on Computing*, vol. 25, no. 5, pp. 956–997, 1996.

[GM01] C. Gutwenger and P. Mutzel, "A linear time implementation of spqr-trees," in *Proceedings of the 8th International Symposium on Graph Drawing (GD '00)*, ser. Lecture Notes in Computer Science, vol. 1984. Springer, 2001, pp. 77–90.

[GT01] A. Garg and R. Tamassia, "On the computational complexity of upward and rectilinear planarity testing," *SIAM Journal on Computing*, vol. 31, no. 2, pp. 601–625, 2001.

[HKRS97] M. R. Henzinger, P. Klein, S. Rao, and S. Subramanian, "Faster shortest-path algorithms for planar graphs," *Journal of Computer and System Sciences*, vol. 55, pp. 3–23, 1997.

[Kel80] A. K. Kelmans, "Concept of a vertex in a matroid and 3-connected graphs," *Journal of Graph Theory*, vol. 4, no. 1, pp. 13–19, 1980.

[KKRW10] B. Katz, M. Krug, I. Rutter, and A. Wolff, "Manhattan-geodesic embedding of planar graphs," in *Proceedings of the 17th International Symposium on Graph Drawing (GD '09)*, ser. Lecture Notes in Computer Science, D. Eppstein and E. Gansner, Eds., vol. 5849. Springer, 2010, pp. 207–218.

[LMPS92] Y. Liu, P. Marchioro, R. Petreschi, and B. Simeone, "Theoretical results on at most 1-bend embeddability of graphs," *Acta Mathematicae Applicatae Sinica (English Series)*, vol. 8, no. 2, pp. 188–192, 1992.

[LMS98] Y. Liu, A. Morgana, and B. Simeone, "A linear algorithm for 2-bend embeddings of planar graphs in the two-dimensional grid," *Discrete Applied Mathematics*, vol. 81, no. 1–3, pp. 69–91, 1998.

[MdMS04] A. Morgana, C. P. de Mello, and G. Sontacchi, "An algorithm for 1-bend embeddings of plane graphs in the two-dimensional grid," *Discrete Applied Mathematics*, vol. 141, no. 1–3, pp. 225–241, 2004.

[MN95] G. L. Miller and J. Naor, “Flow in planar graph with multiple sources and sinks,” *SIAM Journal on Computing*, vol. 24, no. 5, pp. 1002–1017, 1995.

[Tam87] R. Tamassia, “On embedding a graph in the grid with the minimum number of bends,” *SIAM Journal on Computing*, vol. 16, no. 3, pp. 421–444, 1987.

[Whi32] H. Whitney, “Congruent graphs and the connectivity of graphs,” *American Journal of Mathematics*, vol. 54, no. 1, pp. 150–168, 1932.