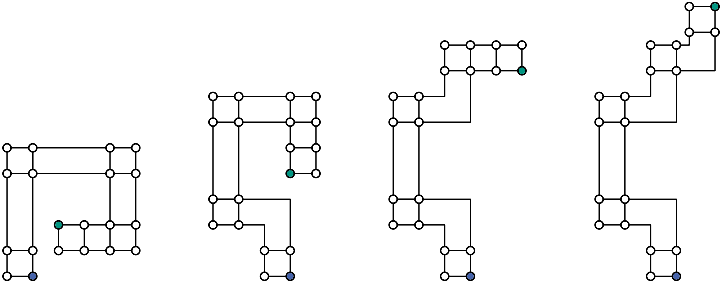
Orthogonal Graph Drawing with Flexibility Constraints

Thomas Bläsius, Marcus Krug, Ignaz Rutter, and Dorothea Wagner | February 4, 2013

INSTITUTE OF THEORETICAL INFORMATICS · ALGORITHMICS I · PROF. DR. DOROTHEA WAGNER



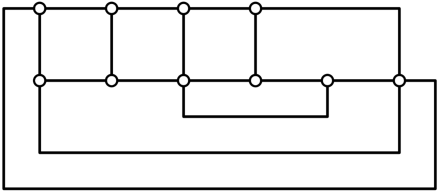
Motivation

Karlsruhe Institute of Technology

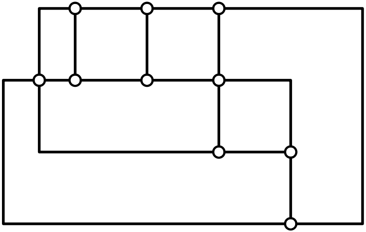
Classical Approach

Minimize total number of bends or maximum number of bends per edge.

Is minimization of bends always the best choice?



9 bends in total; at most 4 bends per edge



7 bends in total; at most 2 bends per edge

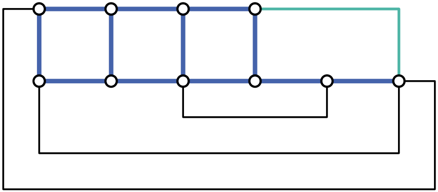
What should be done if different edges have different importance?

Motivation

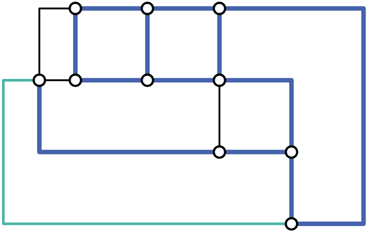
Classical Approach

Minimize total number of bends or maximum number of bends per edge.

Is minimization of bends always the best choice?



9 bends in total; at most 4 bends per edge



7 bends in total; at most 2 bends per edge

What should be done if different edges have different importance?

The FLEXDRAW Problem

Karlsruhe Institute of Technology

FLEXDRAW Problem

Given

- a 4-planar graph $G = (V, E)$
- and a function $\text{flex} : E \longrightarrow \mathbb{N}_0$ assigning a *flexibility* to every edge.

Find

- a planar orthogonal drawing of G
- such that every edge $e \in E$ has at most $\text{flex}(e)$ bends.

Related Work

Karlsruhe Institute of Technology

Fixed Planar Embedding

- Minimize the total number of bends. [**Tamassia** 1987]
- Modification solves FLEXDRAW with fixed embedding.

Variable planar Embedding

FLEXDRAW generalizes β-embeddability

- 0-embeddability is $\mathcal{NP}$-hard. [**Garg, Tamassia** 2001]
- 2-embeddability can be solved in polynomial time. [**Biedl, Kant** 1994] [**Liu, Morgana, Simeone** 1998]
- 1-embeddability $\subseteq$ this talk

Related Work

Fixed Planar Embedding

- Minimize the total number of bends. [**Tamassia** 1987]
- Modification solves FLEXDRAW with fixed embedding.

Variable planar Embedding

FLEXDRAW generalizes β-embeddability

- 0-embeddability is $\mathcal{NP}$-hard. [**Garg, Tamassia** 2001]
- 2-embeddability can be solved in polynomial time. [**Biedl, Kant** 1994] [**Liu, Morgana, Simeone** 1998]
- 1-embeddability $\subseteq$ this talk

Our Contribution

Theorem

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

- 1-embeddability can be solved in polynomial time.
- Means: Characterize all possible shapes of orthogonal drawings.

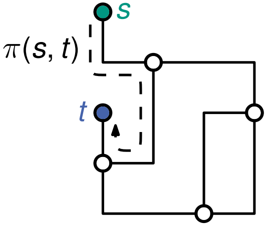
The Shape of a Drawing

The Path $\pi(s, t)$

Path from s to t on the outer face in counter-clockwise direction.

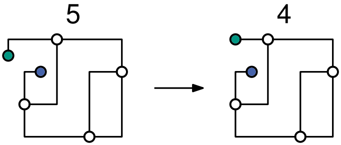
Rotation along $\pi(s, t)$

$\text{rot}(\pi(s, t)) = \#\{\text{bends to the right}\} - \#\{\text{bends to the left}\}$

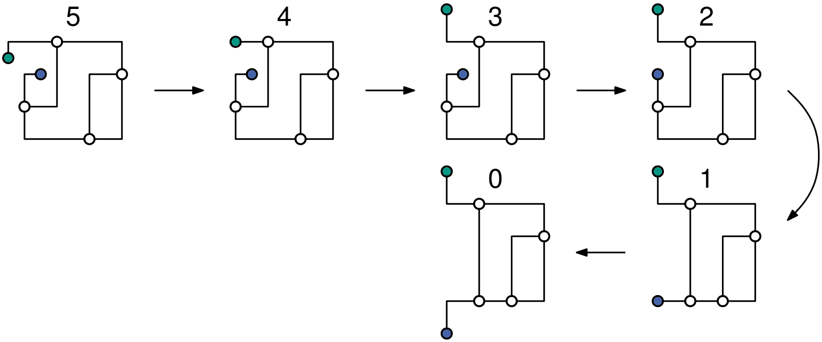


$\text{rot}(\pi(s, t)) = 2$

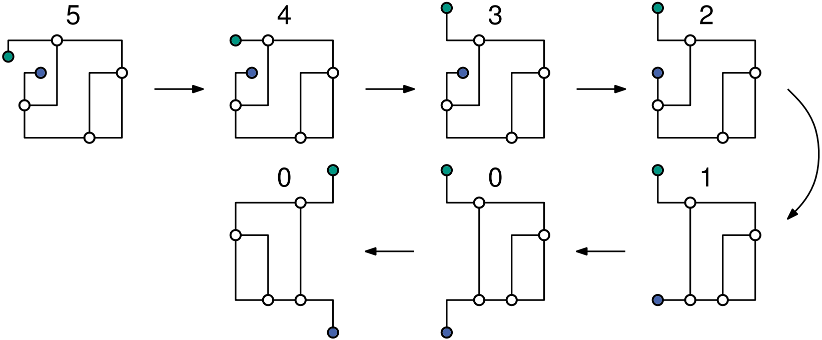
Graphs behave like Edges



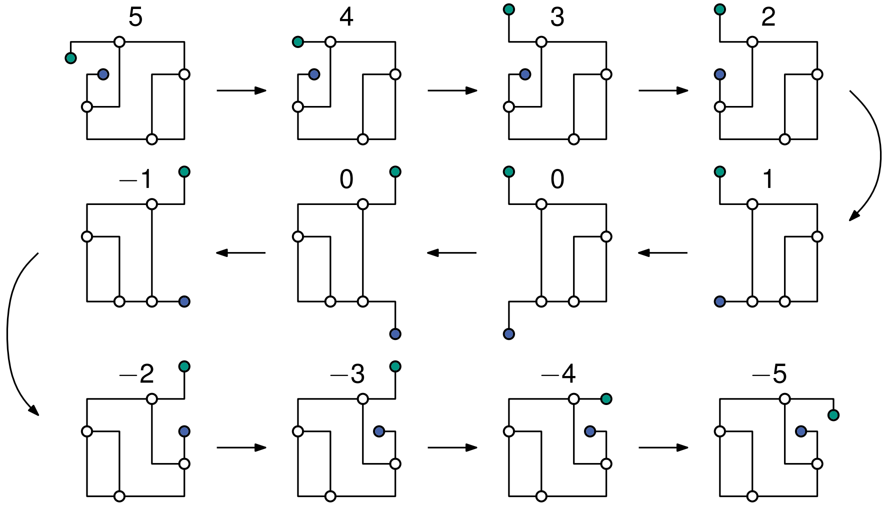
Graphs behave like Edges



Graphs behave like Edges



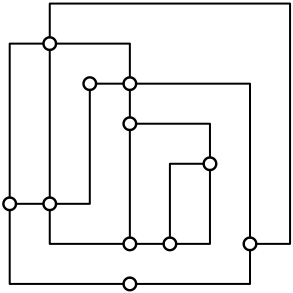
Graphs behave like Edges



Replacing Subgraphs

Check all Embeddings of a Graph for valid Drawings

- If we know the possible shapes of a subgraph,
- we can replace it by an edge.

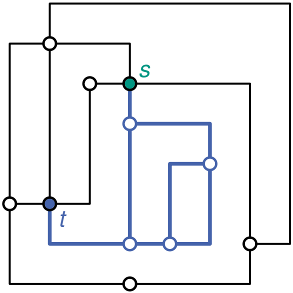


⇒ Number of embeddings is reduced.

Replacing Subgraphs

Check all Embeddings of a Graph for valid Drawings

- If we know the possible shapes of a subgraph,
- we can replace it by an edge.

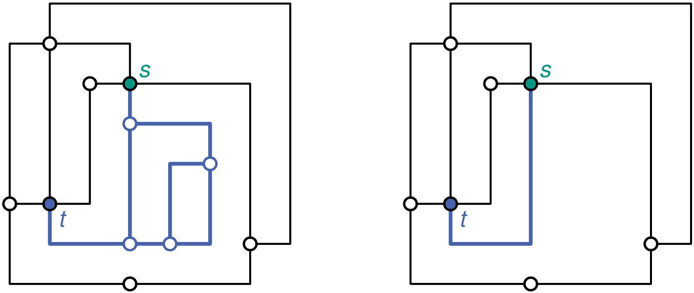


⇒ Number of embeddings is reduced.

Replacing Subgraphs

Check all Embeddings of a Graph for valid Drawings

- If we know the possible shapes of a subgraph,
- we can replace it by an edge.



$\Rightarrow$ Number of embeddings is reduced.

Outline

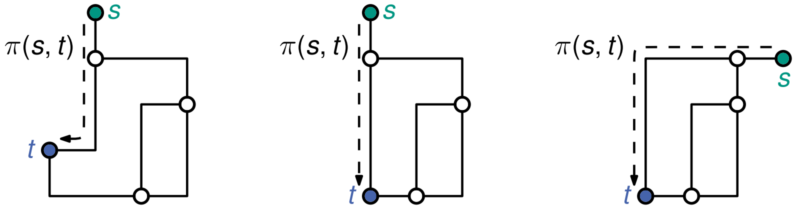
1. Fixed Embedding
   - Unwinding a given Drawing
   - Characterization of all Possible Shapes
2. Variable Embedding
   - Characterization of all Possible Shapes
   - Replacing Subgraphs by simple Gadgets
   - Considering all Planar Embeddings – SPQR-Tree

Reducing the Rotation

Karlsruhe Institute of Technology

Lemma

Given a graph with positive flexibility and a drawing with $\mathrm{rot}(\pi(s,t)) \geq 0$*.*
We can reduce the rotation by 1.



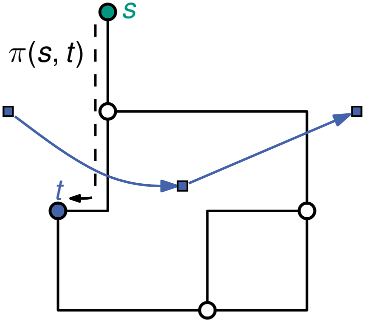
$\Rightarrow$ The possible values form an interval.

$\Rightarrow$ There are no “rigid” graphs.

Reducing the Rotation – Proof

The Flex Graph

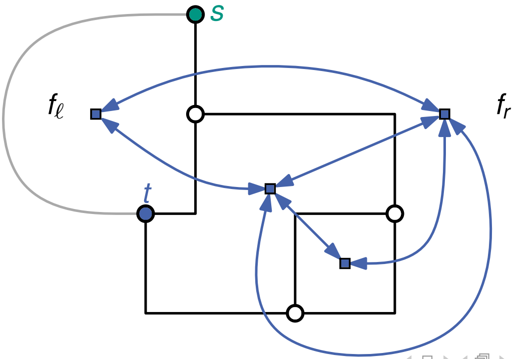
- Bend along a path crossing $\pi(s, t)$
- Split the outer face into f_ℓ and f_r. Consider the dual graph.
- Remove edges harming flexibility constraints.



Reducing the Rotation – Proof

The Flex Graph

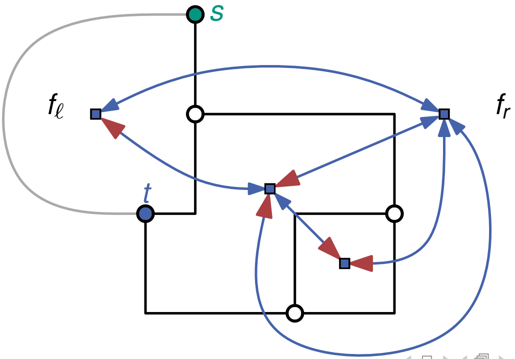
- Bend along a path crossing $\pi(s, t)$
- Split the outer face into f_ℓ and f_r. Consider the dual graph.
- Remove edges harming flexibility constraints.



Reducing the Rotation – Proof

The Flex Graph

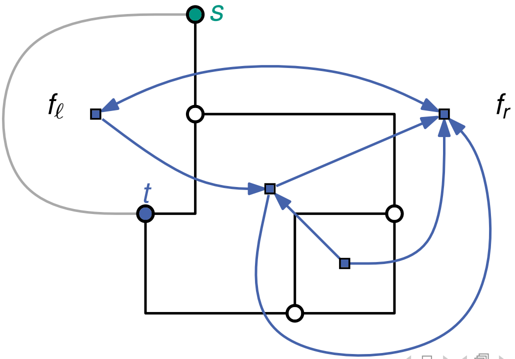
- Bend along a path crossing $\pi(s, t)$
- Split the outer face into f_ℓ and f_r. Consider the dual graph.
- Remove edges harming flexibility constraints.



Reducing the Rotation – Proof

The Flex Graph

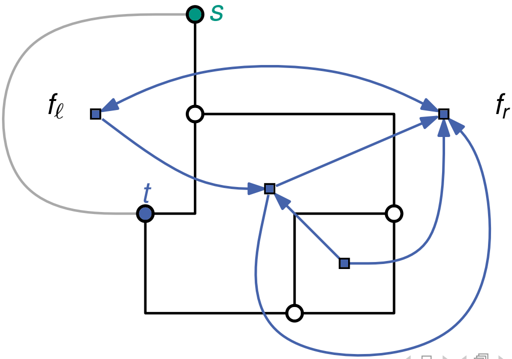
- Bend along a path crossing $\pi(s, t)$
- Split the outer face into f_ℓ and f_r. Consider the dual graph.
- Remove edges harming flexibility constraints.



Reducing the Rotation – Proof

Lemma

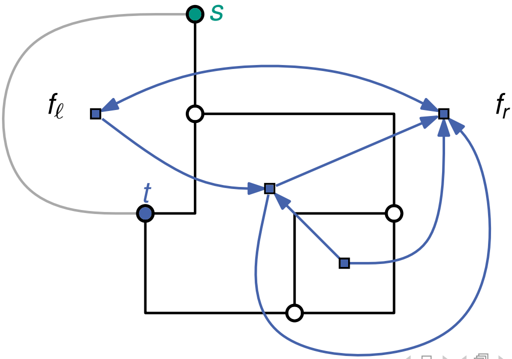
Positive flexibility ⇒ such a path exists.



Reducing the Rotation – Proof

The Flex Graph contains a Path from f_ℓ to f_r

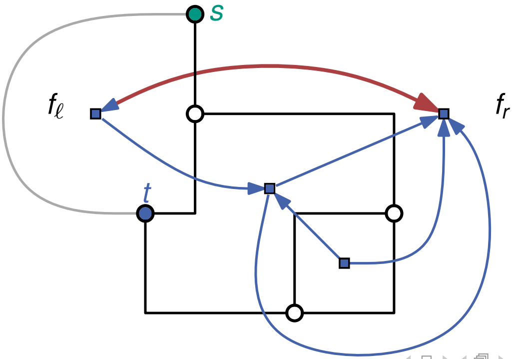
- Fact: positive flexibility and $\mathrm{rot}(\pi(s,t)) \geq 0 \Rightarrow$ edge (f_ℓ, f_u) exists
- If $f_u = f_r$ we are done.
- Else we remove the corresponding edge in G and continue with the remaining graph.



Reducing the Rotation – Proof

The Flex Graph contains a Path from f_ℓ to f_r

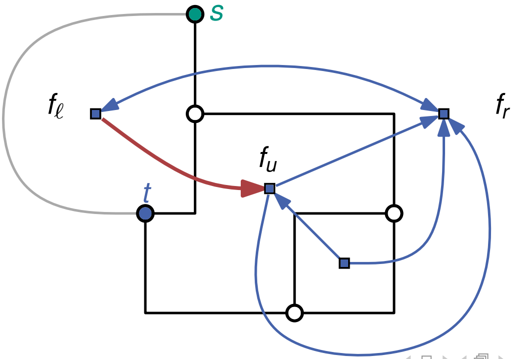
- Fact: positive flexibility and $\text{rot}(\pi(s,t)) \geq 0 \Rightarrow$ edge (f_ℓ, f_u) exists
- If $f_u = f_r$ we are done.
- Else we remove the corresponding edge in G and continue with the remaining graph.



Reducing the Rotation – Proof

The Flex Graph contains a Path from f_ℓ to f_r

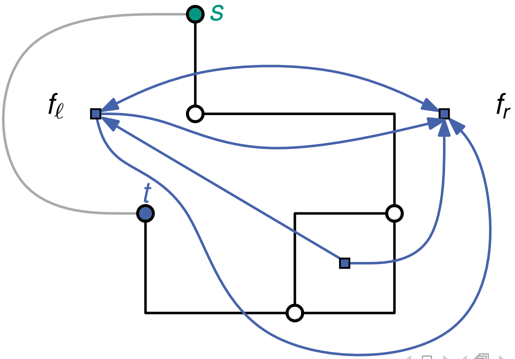
- Fact: positive flexibility and $\text{rot}(\pi(s,t)) \geq 0 \Rightarrow$ edge (f_ℓ, f_u) exists
- If $f_u = f_r$ we are done.
- Else we remove the corresponding edge in G and continue with the remaining graph.



Reducing the Rotation – Proof

The Flex Graph contains a Path from f_ℓ to f_r

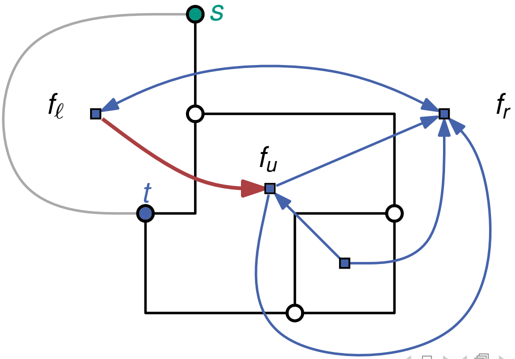
- Fact: positive flexibility and $\mathrm{rot}(\pi(s,t)) \geq 0 \Rightarrow$ edge (f_ℓ, f_u) exists
- If $f_u = f_r$ we are done.
- Else we remove the corresponding edge in G and continue with the remaining graph.



Reducing the Rotation – Proof

The Flex Graph contains a Path from f_ℓ to f_r

- Fact: positive flexibility and $\text{rot}(\pi(s,t)) \geq 0 \Rightarrow$ edge (f_ℓ, f_u) exists
- If $f_u = f_r$ we are done.
- Else we remove the corresponding edge in G and continue with the remaining graph.



Shape of a Graph

For the set Ω of all valid drawings with respect to the embedding $\mathcal{E}$ define

$$\text{maxrot}_{\mathcal{E}}(G) = \max_{\mathcal{R} \in \Omega} \text{rot}_{\mathcal{R}}(\pi(s,t))$$

Theorem

For $\rho \in [-1, \text{maxrot}_{\mathcal{E}}(G)]$ a valid drawing with $\text{rot}(\pi(s,t)) = \rho$ exists.

In $O(n^{3/2})$ time we can

- check if G admits a valid orthogonal drawing with respect to $\mathcal{E}$.
- compute $\text{maxrot}_{\mathcal{E}}(G)$.

Shape of a Graph

For the set Ω of all valid drawings with respect to the embedding $\mathcal{E}$ define

$$\mathrm{maxrot}_{\mathcal{E}}(G) = \max_{\mathcal{R}\in\Omega} \mathrm{rot}_{\mathcal{R}}(\pi(s,t))$$

Theorem

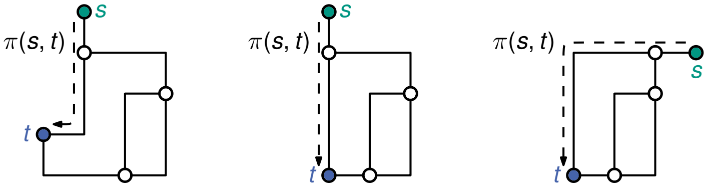
For $\rho \in [-1, \mathrm{maxrot}_{\mathcal{E}}(G)]$ a valid drawing with $\mathrm{rot}(\pi(s,t)) = \rho$ exists.

In $O(n^{3/2})$ time we can

- check if G admits a valid orthogonal drawing with respect to $\mathcal{E}$.
- compute $\mathrm{maxrot}_{\mathcal{E}}(G)$.

Summary – Fixed Planar Embedding

No “rigid” drawings: the rotation can always be reduced.



Theorem

For $\rho \in [-1, \mathrm{maxrot}_{\mathcal{E}}(G)]$ *a valid drawing with* $\mathrm{rot}(\pi(s, t)) = \rho$ *exists.*

Now we drop the fixed planar embedding.

Shape of a Graph

For the set Ψ of all embeddings with s and t on the outer face define

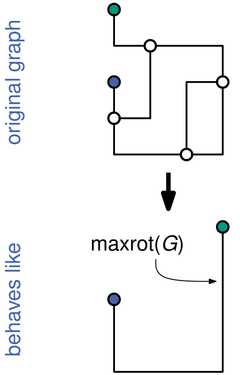
$$\operatorname{maxrot}(G) = \max_{\mathcal{E} \in \Psi} \operatorname{maxrot}_{\mathcal{E}}(G)$$

Theorem

If $\deg(s) = \deg(t) = 1$ *then* G *admits a valid drawing with* $\operatorname{rot}(\pi(s,t)) = \rho$ *if and only if* $\rho \in [-\operatorname{maxrot}(G), \operatorname{maxrot}(G)]$.

Replacing Subgraphs by Gadgets

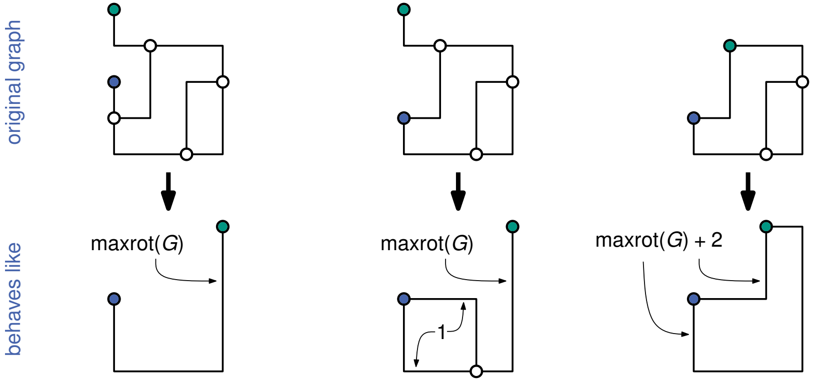
$\deg(s) = 1, \deg(t) = 1$



$[-\operatorname{maxrot}(G), \operatorname{maxrot}(G)]$

Replacing Subgraphs by Gadgets

$\deg(s) = 1, \deg(t) = 1$ | $\deg(s) = 1, \deg(t) = 2$ | $\deg(s) = 2, \deg(t) = 2$



$[-\,\mathrm{maxrot}(G), \mathrm{maxrot}(G)]$ | $[-\,\mathrm{maxrot}(G) - 1, \mathrm{maxrot}(G)]$ | $[-\,\mathrm{maxrot}(G) - 2, \mathrm{maxrot}(G)]$

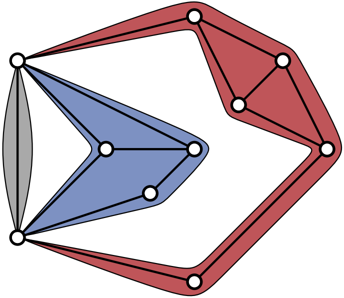
Computing the Maximum Rotation

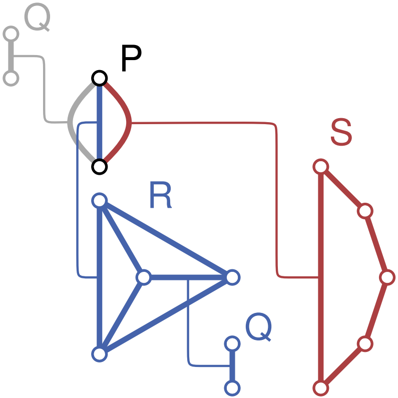
Ingredients for Computing the Maximum Rotation

- Represent all planar embeddings with *s* and *t* on the outer face with the SPQR-tree. [**Di Battista, Tamassia** 1996]
- Traverse the SPQR-tree bottom up and compute the maximum rotation for each node.
- Replace visited nodes by gadgets to reduce the number of embeddings.

The SPQR-Tree

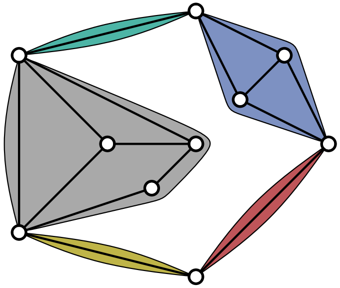
- Decomposition of a biconnected graph in triconnected components.
- Representation of all planar embeddings.

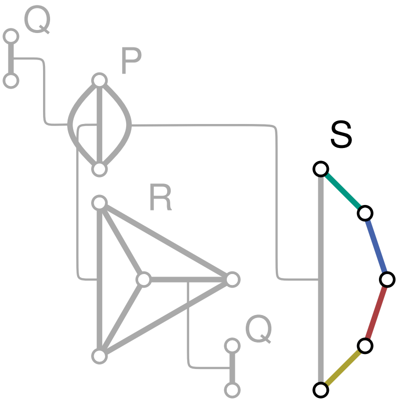




The SPQR-Tree

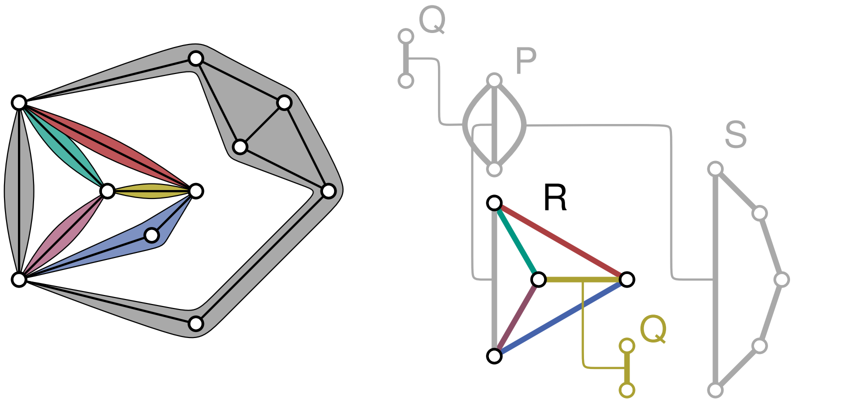
- Decomposition of a biconnected graph in triconnected components.
- Representation of all planar embeddings.





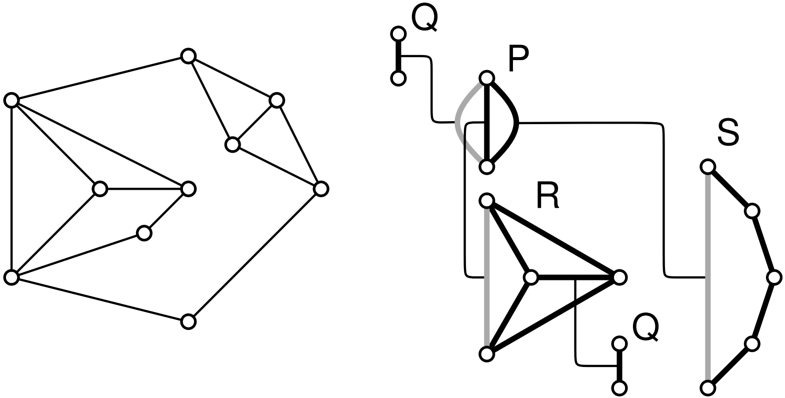
The SPQR-Tree

- Decomposition of a biconnected graph in triconnected components.
- Representation of all planar embeddings.



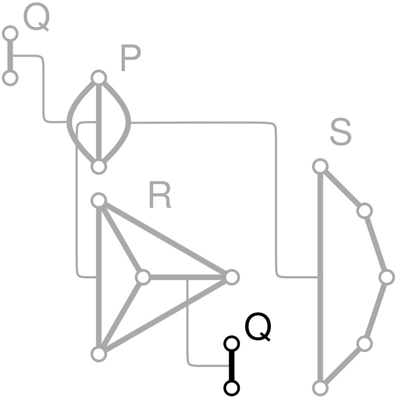
The SPQR-Tree

- Decomposition of a biconnected graph in triconnected components.
- Representation of all planar embeddings.



Maximum Rotation of a Q-node

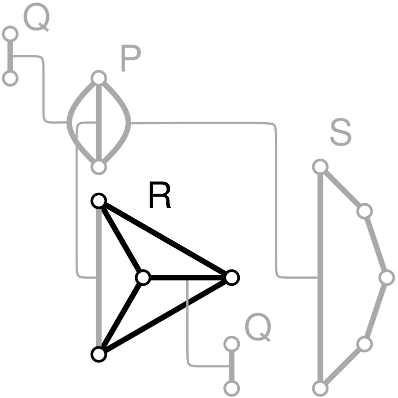
The maximum rotation of a Q-node is simply the flexibility of the corresponding edge.



Maximum Rotation of a R-node

The Skeleton of an R-node is triconnected

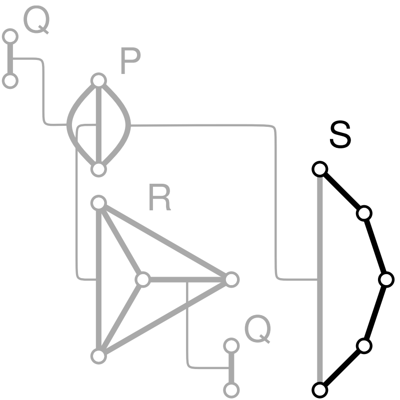
- Replace by gadgets.
- Check both embeddings.



Maximum Rotation of a S-node

An S-node represents a Chain of Graphs

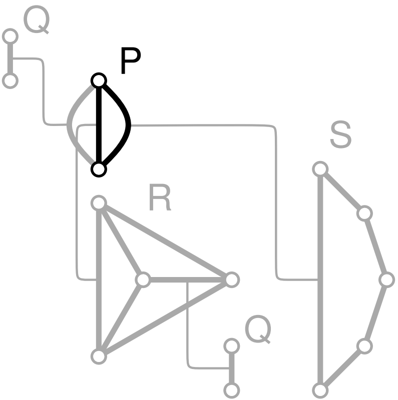
- Maximize the rotation for every graph.
- Put the graphs together with an angle of 90°.
- Only simple calculations.



Maximum Rotation of a P-node

The Skeleton of a P-node has three or four parallel Edges

- Replace by gadgets.
- Check up to six embeddings.

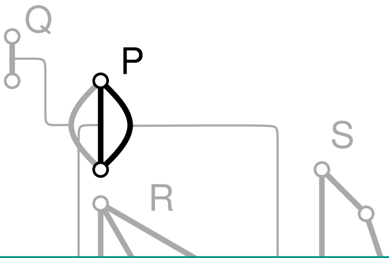


Maximum Rotation of a P-node

Karlsruhe Institute of Technology

The Skeleton of a P-node has three or four parallel Edges

- Replace by gadgets.
- Check up to six embeddings.



Theorem

For a biconnected graph G with designated vertices s and t we can compute maxrot(G) *in* $O(n^{3/2})$ *time or decide that G has no valid drawing with s and t on the outer face.*

Solving FLEXDRAW

Theorem

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

- Choose every pair of adjacent vertices as s and t.
- Use block-cutvertex tree to solve the general case.

Conclusion

Results

- Solve FLEXDRAW for graphs with positive flexibility.
- Complexity gap between $\mathcal{NP}$-hardness for 0-embeddability and polynomial-time algorithms for 2-embeddability closed.
- Maximum rotation yields characterization of possible shapes.

Open Problems

- Speed up from $O(n^{5/2})$ to $O(n^2)$.
- Can FLEXDRAW be solved efficiently if a subgraph has flexibility 0?
  e.g. matchings, trees, series parallel graphs, graphs with max deg 3
- Some kind of optimization.

Conclusion

Karlsruhe Institute of Technology

Results

- Solve FLEXDRAW for graphs with positive flexibility.
- Complexity gap between $\mathcal{NP}$-hardness for 0-embeddability and polynomial-time algorithms for 2-embeddability closed.
- Maximum rotation yields characterization of possible shapes.

Open Problems

- Speed up from $O(n^{5/2})$ to $O(n^2)$.
- Can FLEXDRAW be solved efficiently if a subgraph has flexibility 0?
  e.g. matchings, trees, series parallel graphs, graphs with max deg 3
- Some kind of optimization.