On the Parametrized Complexity of Bend-Minimum Orthogonal Planarity

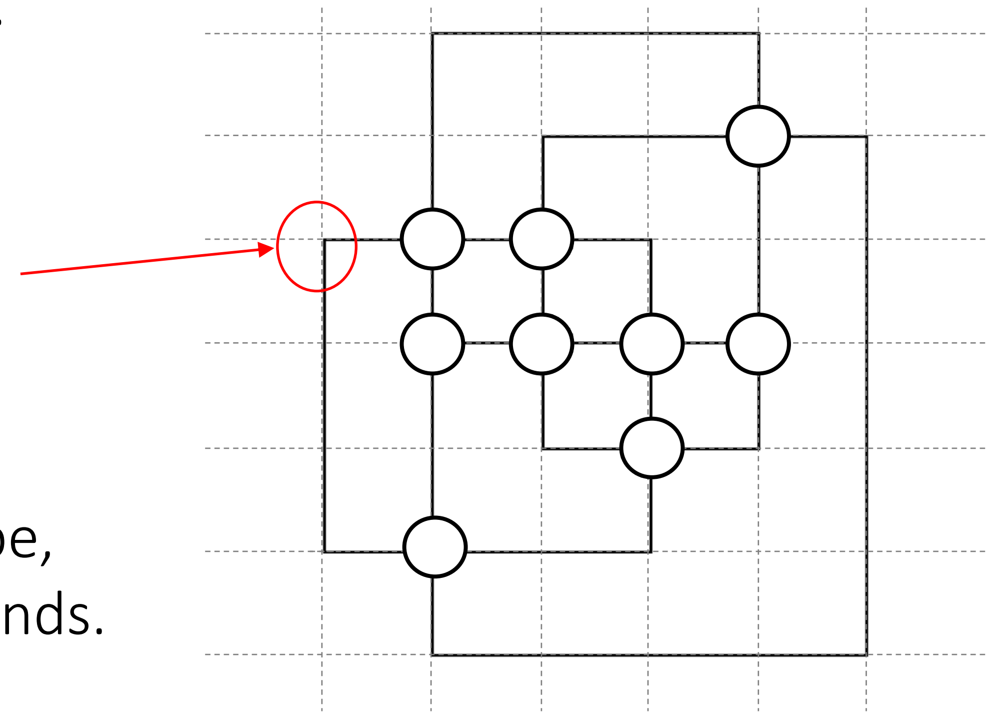
E. Di Giacomo, W. Didimo, G. Liotta, F. Montecchiani, G. Ortali

University of Perugia

Introduction

Orthogonal Drawings

- Vertices are placed at grid points. Edges are chains of horizontal and vertical segments.
- A *bend* is a point where a horizontal and a vertical segment of the same edge touch.
- An *orthogonal representation* is a class of orthogonal drawings having the same shape, i.e., relative position of the vertices and bends.
- We consider planar drawings/representations.



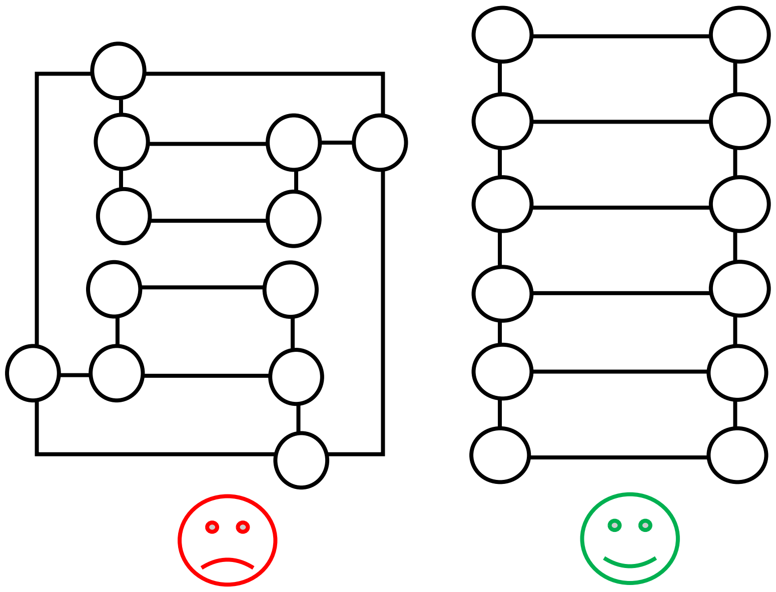
Bend-Minimization Problem

Input: A graph G and a positive integer *b.*
Output: An orthogonal representation of G with at most b bends, if it exists.

- The problem is NP-hard (also for b=0).
  [Garg & Tamassia, SIAM J. Comp. 2001]

- FPT with respect *tw*, *b*, and *k* where:
  - *tw* is the treewidth
  - *b* is the number of bends
  - *k* is the number of vertices of degree 1 or 2

  [Di Giacomo, Liotta, Montecchiani, JCSS 2022]



Our Contribution

Theorem

Let G be an n-vertex graph with *k* vertices of degree at most 2 and *b* be a positive integer. There is an algorithm that solves the Bend-Minimization Problem on G in $O\left(2^{(k+b)\log(k+b)}\right)\mathrm{n}^{O(1)}$ time.

The Algorithm

It is based on Dynamic Programming, and it performs a bottom-up visit of the rooted **SPQ*R-tree** of the graph, for each possible choice of the root.

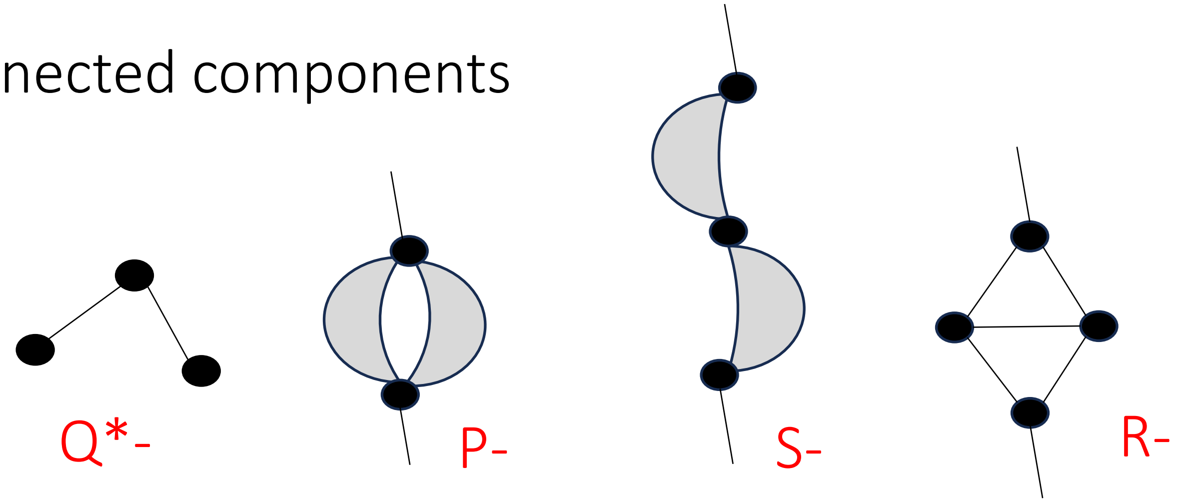
The partial solutions are encoded by using the concept of **spirality**.

Main Ingredients

SPQ*R-tree

Decomposition of the graph into its triconnected components

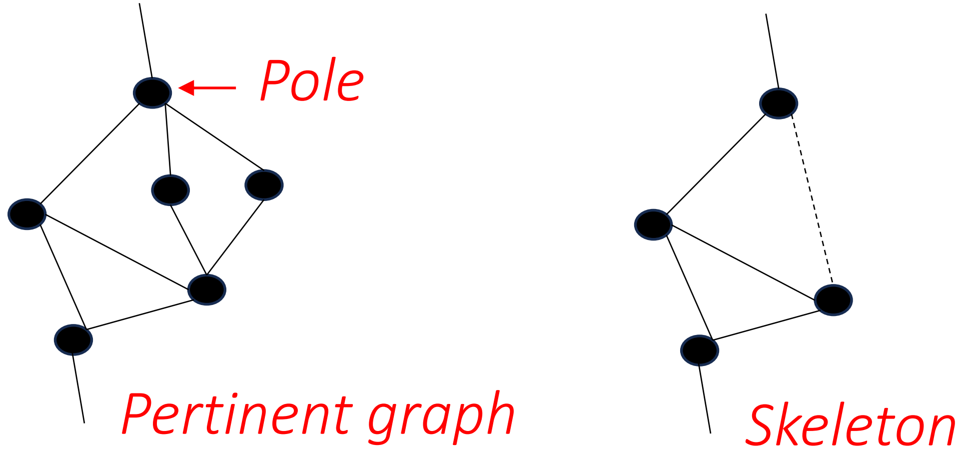
- Q*- : Chain of edges
- P- : Parallel compositions
- S- : Series compositions
- R- : Anything else (the rigids)



Pertinent graph: induced subgraph

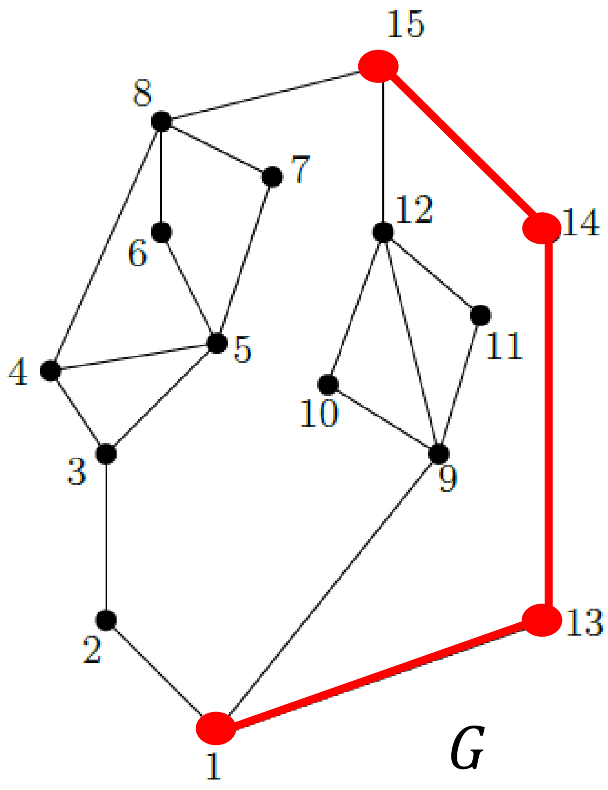
Skeleton: subcomponents are replaced with virtual edges

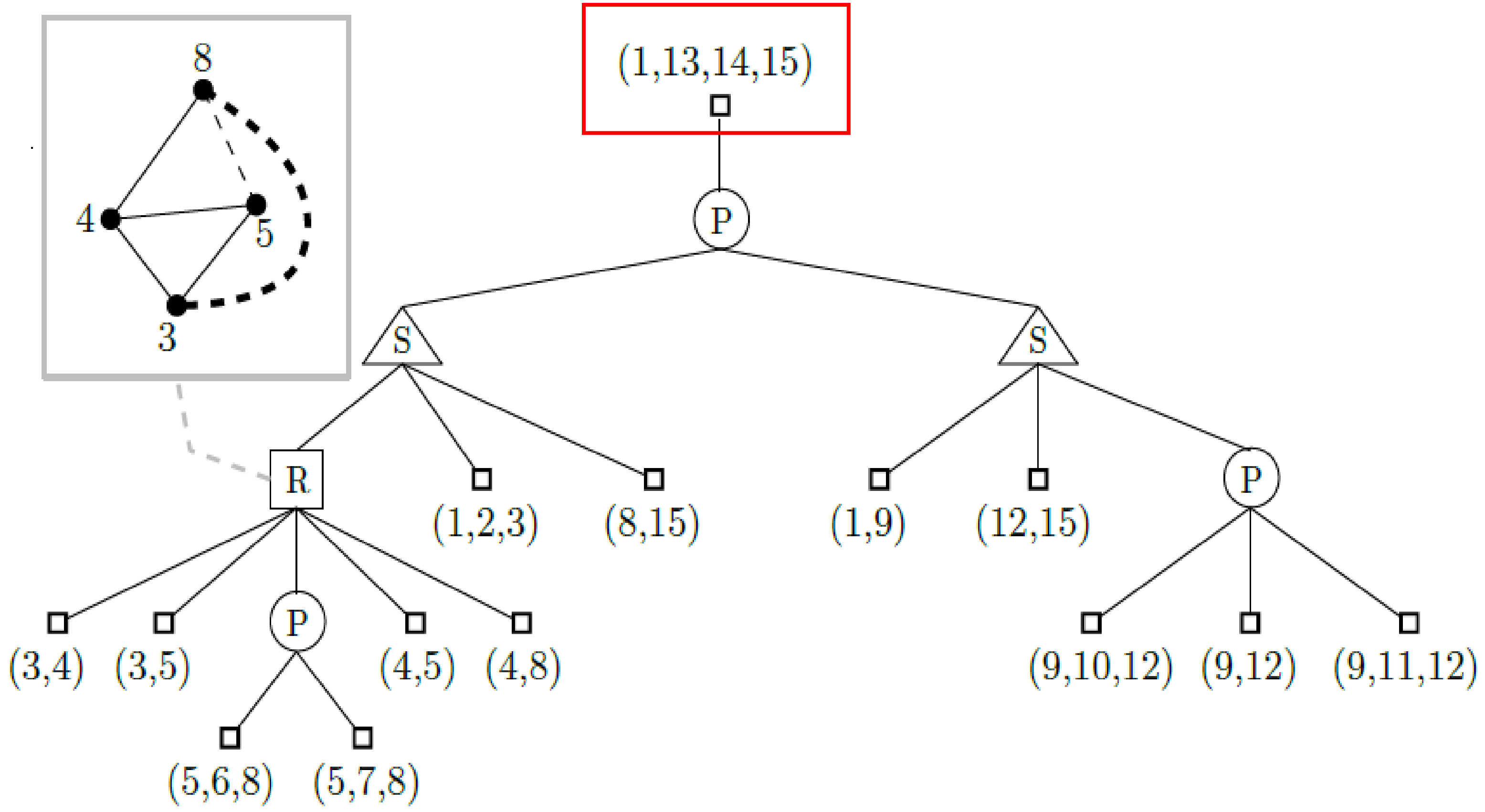
Poles: vertices of the component incident to the rest of the graph



SPQ*R-tree

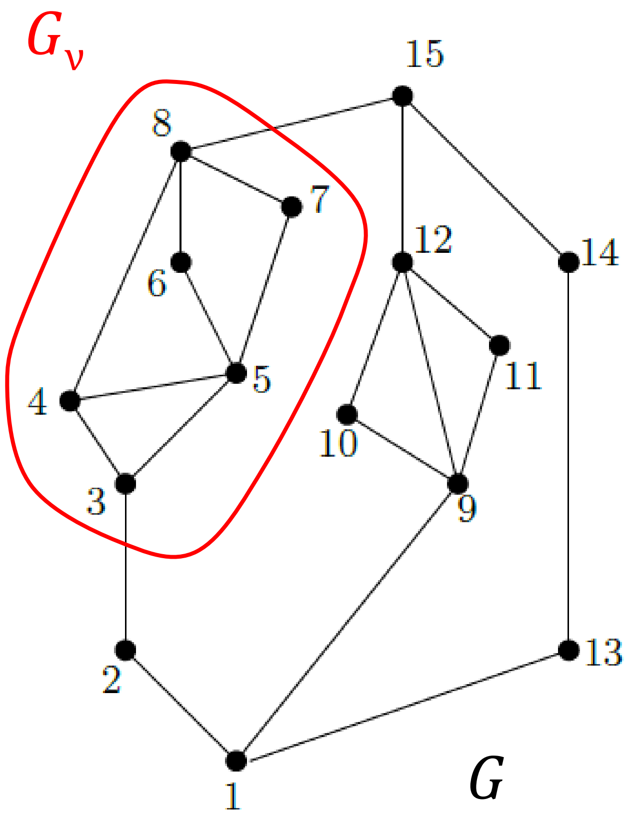
The SPQ*R-tree of G rooted at the Q*-node (1,13,14,15)

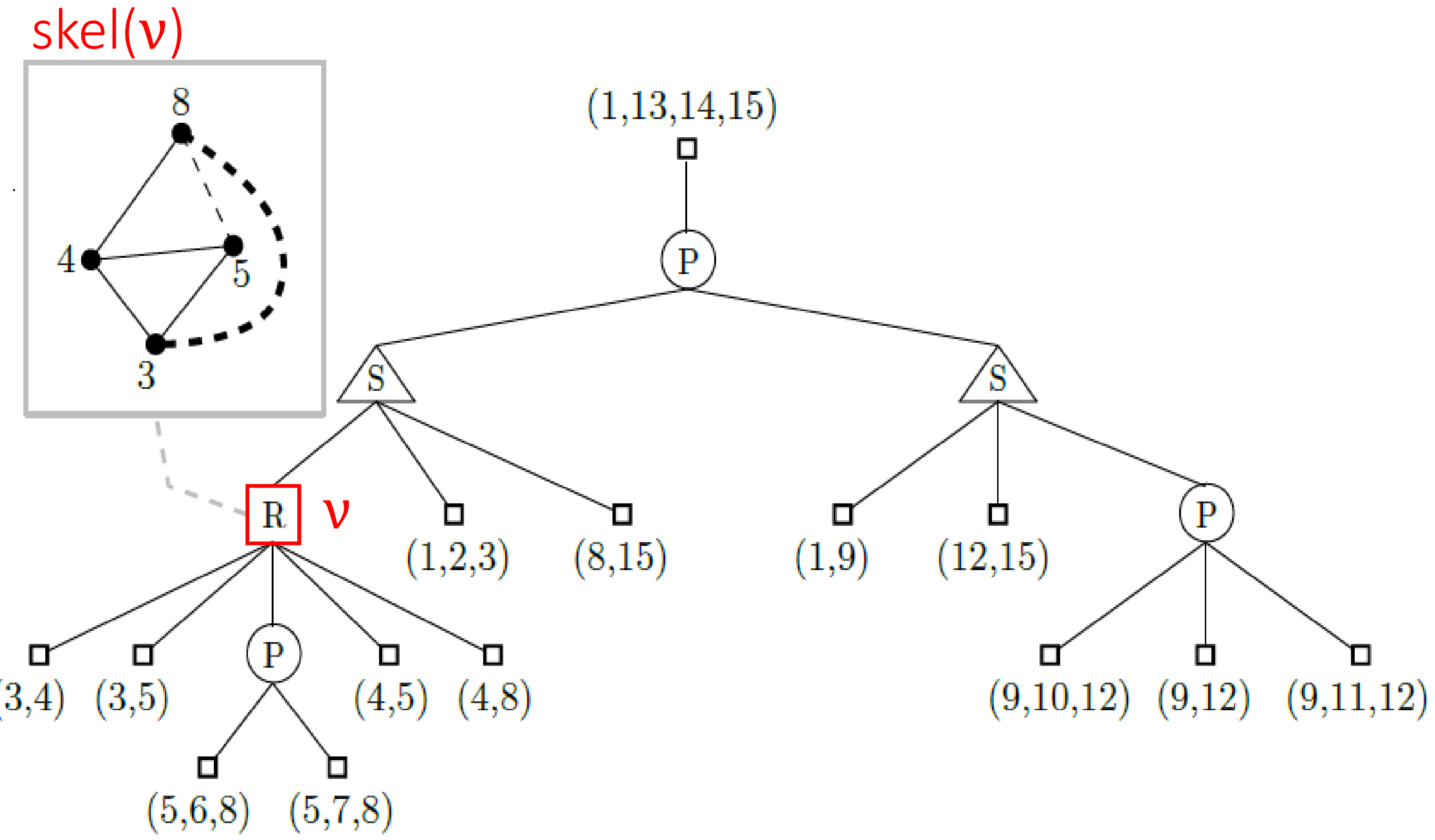




SPQ*R-tree

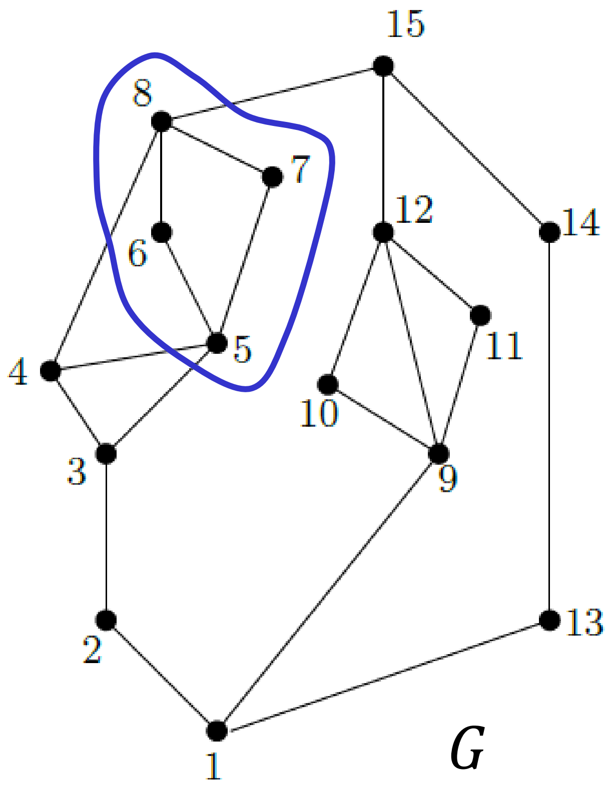
An R-node ν, its pertinent graph G_ν, and its skeleton skel(ν)

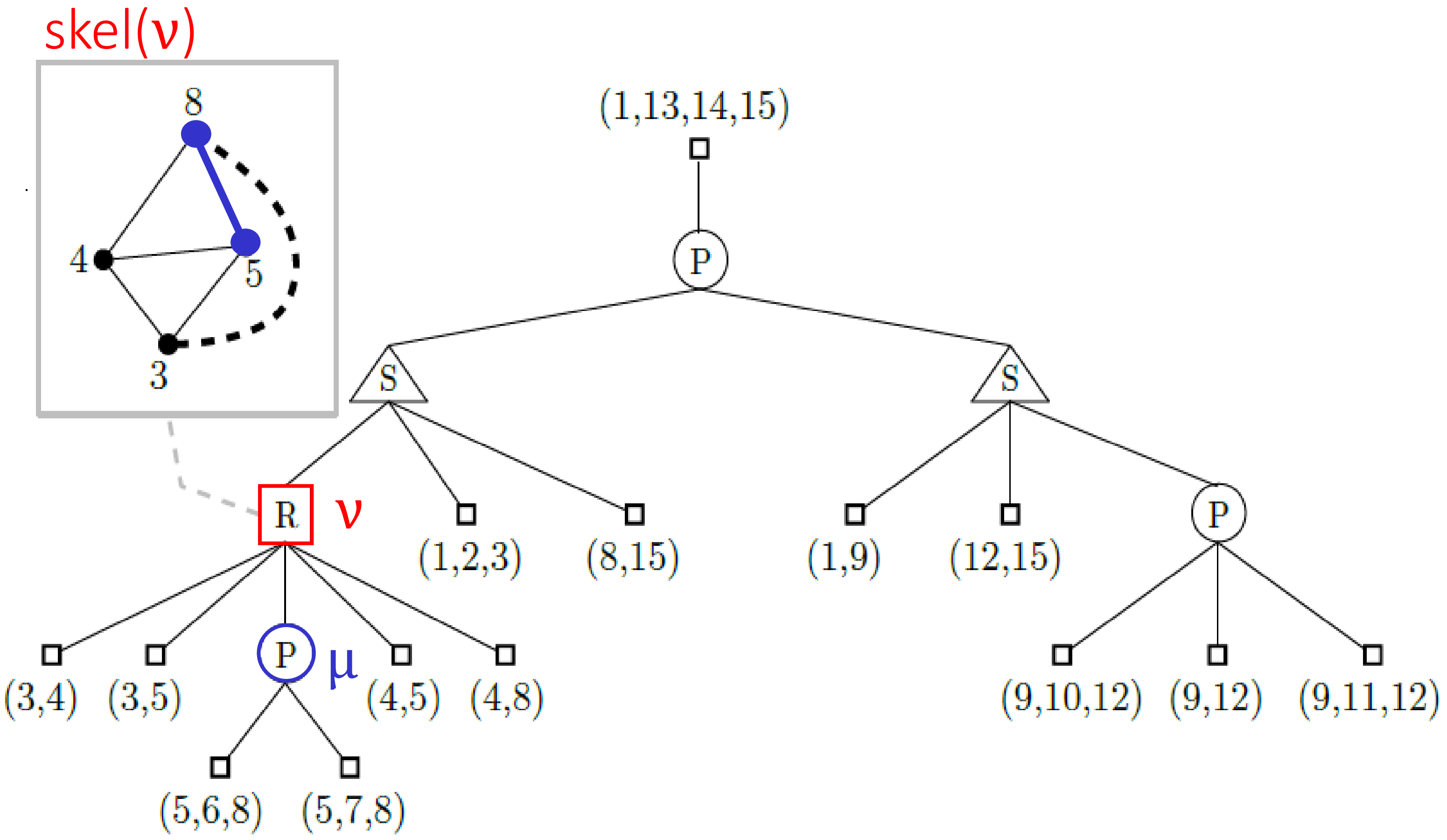




SPQ*R-tree

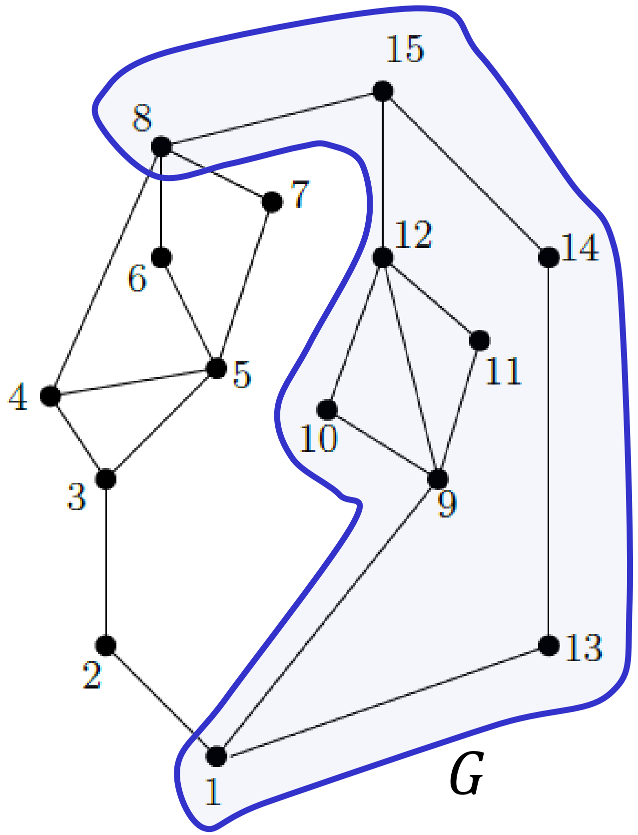
The virtual edge (5,8) corresponds to the P-node μ.

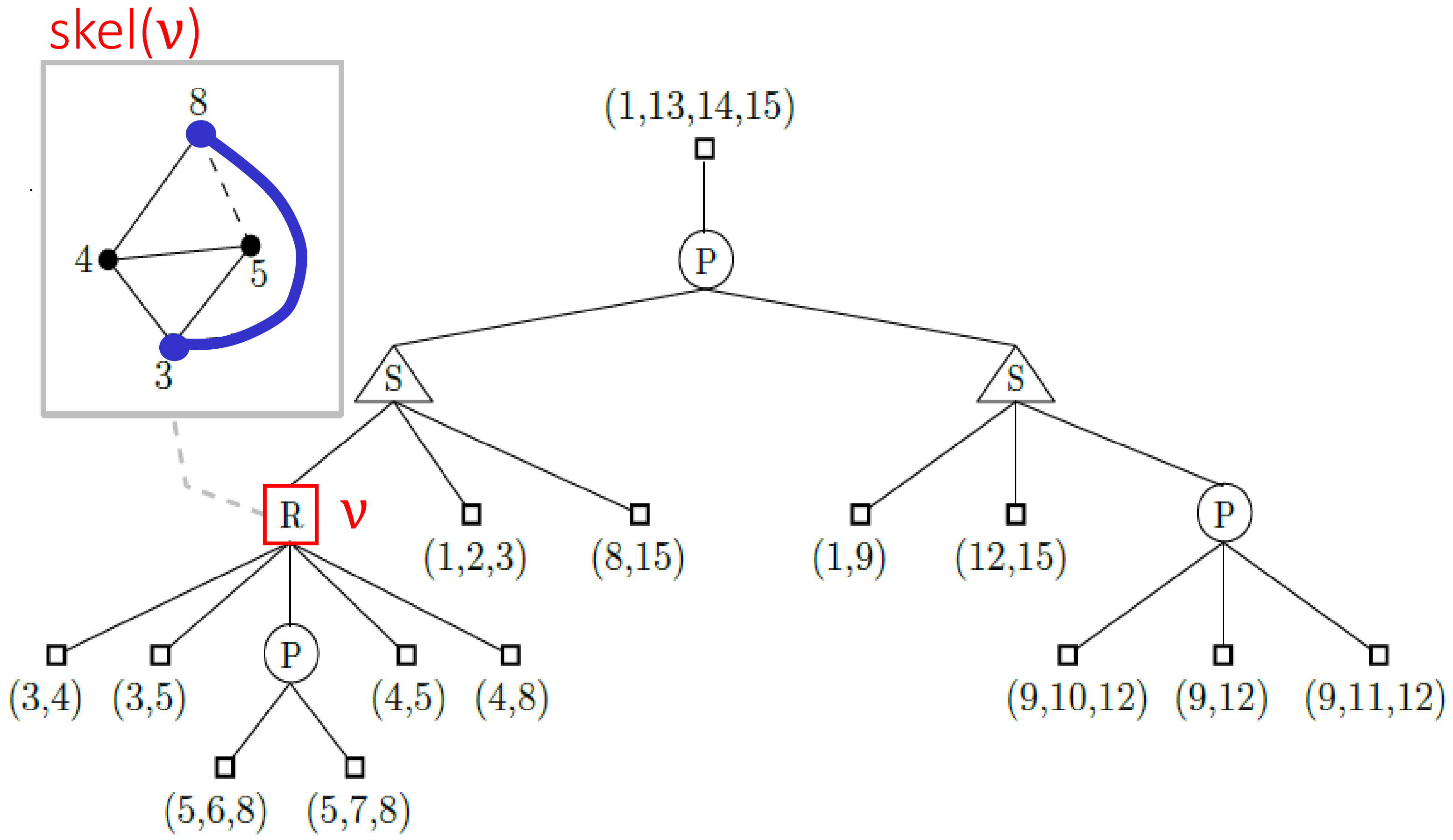




SPQ*R-tree

The virtual edge (3,8) corresponds to the rest of the graph

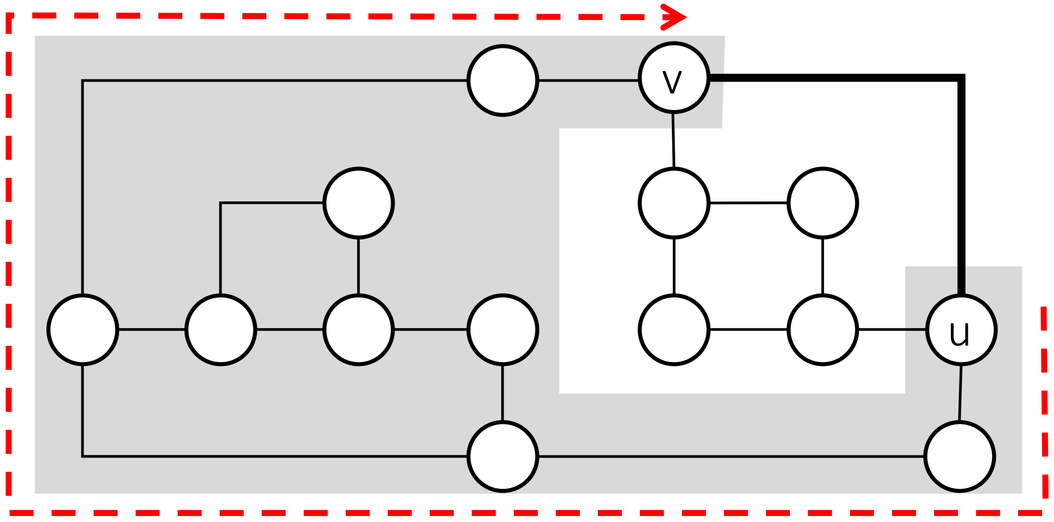




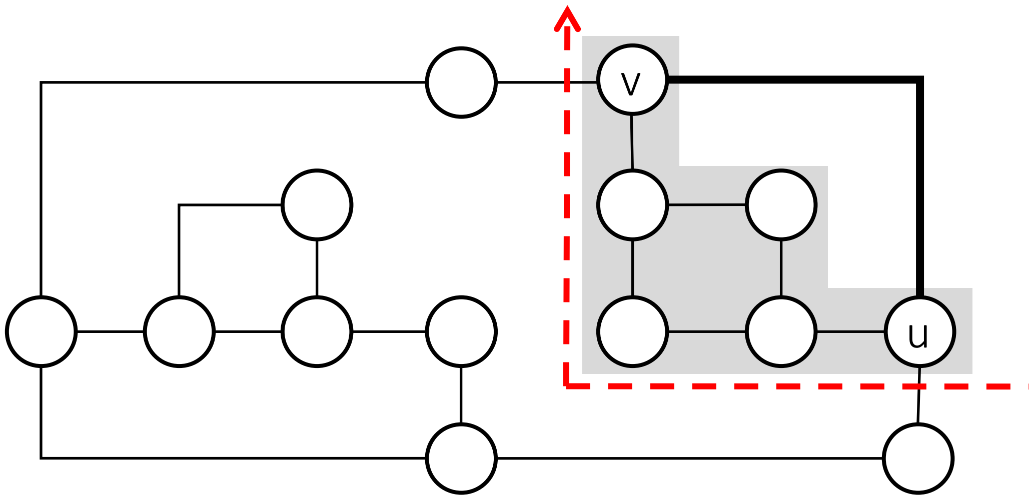
Spirality

Measure of how much a component is ''rolled-up'' in an orthogonal representation.

Number of right turns minus left turns of every path connecting the poles.



The spirality of this component is 3



The spirality of this component is 1

Definition of Record

Definition of record

Every component ν is associated with a spirality set Σ_ν of pairs (σ_ν, X_ν), where:

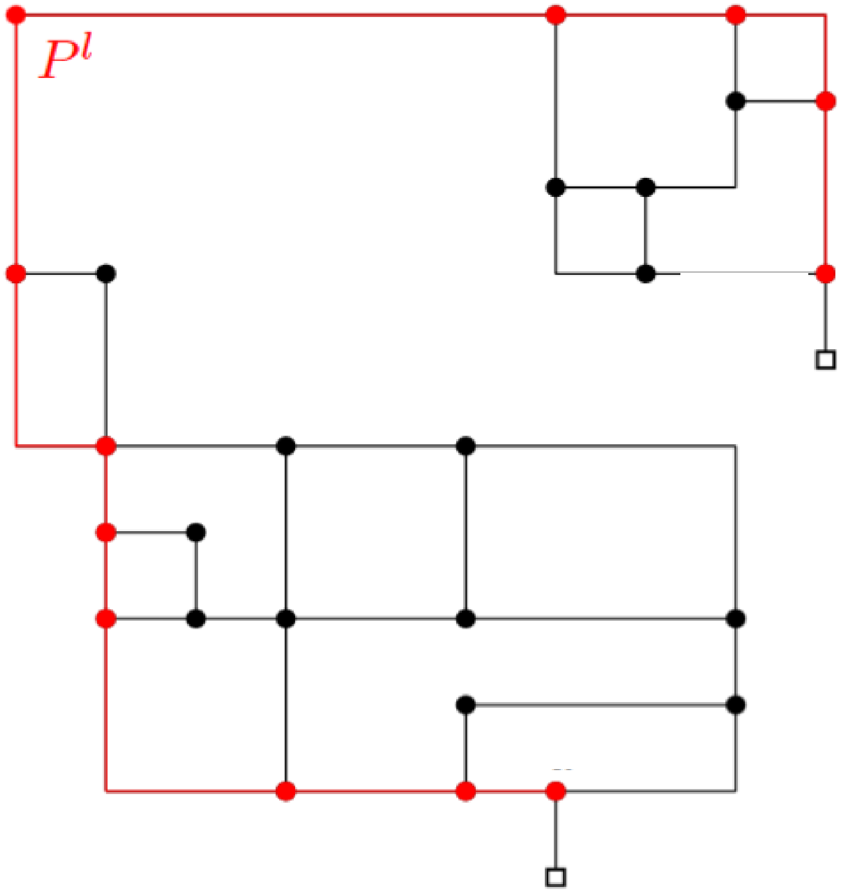
- σ_ν is a value of spirality that ν admits
- X_ν is a pair (b_ν, H_ν), where:
  - b_ν is an integer in the interval *[0,b]*
  - H_ν is an orthogonal representation of G_ν with b_ν bends and spirality σ_ν

Lemma

Let ν be a node of T and H be an orthogonal representation of G with b or less bends and suppose that G_ν contains at most k degree-2 vertices. The spirality σ_ν of the restriction H_ν of H to G_ν belongs to $[-k-b-2, k+b+2]$.

In order to spiralize you need reflex angles. (bends and degree-2 vertices).
<u>The number of possible spiralities is bounded by *k+b*.</u>

Definition of record



σ_ν =2<4<b+k

An orthogonal drawing with spirality 2.
The left path of the component contains 3 bends and 1 vertices of degree 2.

Definition of record

Every component ν is associated with a spirality set $\Sigma_\nu = (\sigma_\nu, X_\nu)$, where:

$|\Sigma_\nu| = f(k+b)$

- σ_ν is a value of spirality that ν admits
- X_ν is a pair (b_ν, H_ν), where:
  - b_ν is an integer in the interval *[0,b]*
  - H_ν is an orthogonal representation of G_ν with b_ν bends and spirality σ_ν

Lemma

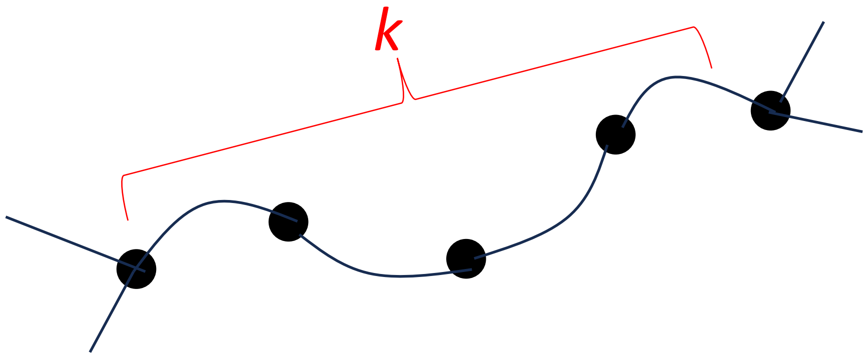
Let v be a node of T and H be an orthogonal representation of G with b or less bends and suppose that G_v contains at most k degree-2 vertices. The spirality σ_v of the restriction H_v of H to G_v belongs to $[-k-b-2, k+b+2]$.

In order to spiralize you need reflex angles! (bends and degree-2 vertices).
The number of possible spiralities is bounded by k and b.

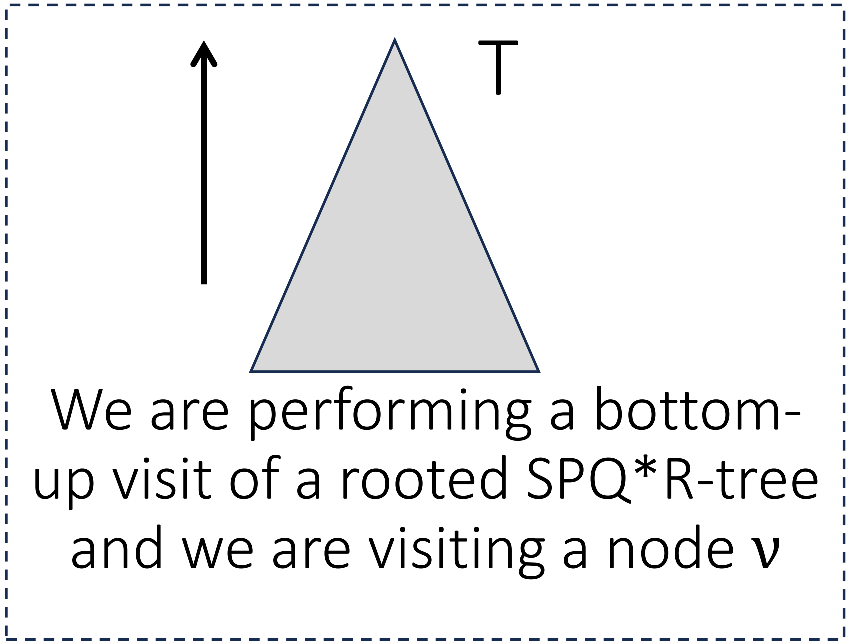
The Algorithm
- Biconnected case -

Q*-, P-, and S-Nodes

- If ν is a Q*-, the spirality set Σ_ν of ν can be computed in *O(k+b)* time, since G_ν has at most k vertices



- If ν is a P-node or a S-node the spirality set Σ_ν of ν can be computed in $O(n(k+b))$ time and $O(n(k+b)^2)$ time, respectively

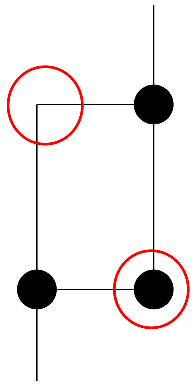


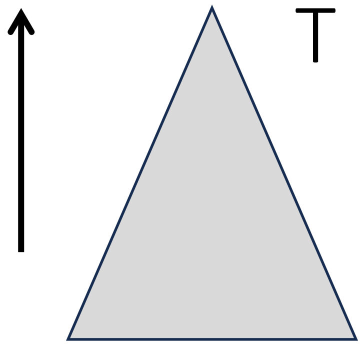
[Di Battista, G., Liotta, G., Vargiu, F. SIAM J. Comput. 27(6), 1998]

R-Nodes

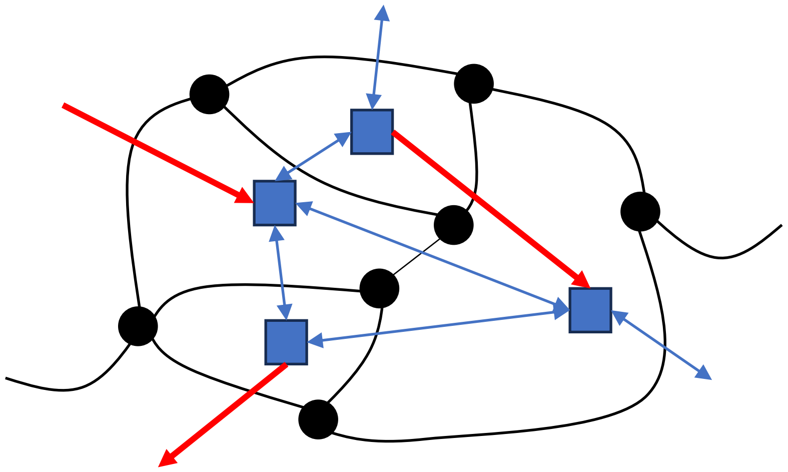
Key Observations

- In this case **ν** has less than k+b children that are not Q*-nodes, since for each one of them at least 2 reflex angles in the outer face are needed.





We are performing a bottom-up visit of a rooted SPQR-tree and we are visiting a node **ν**



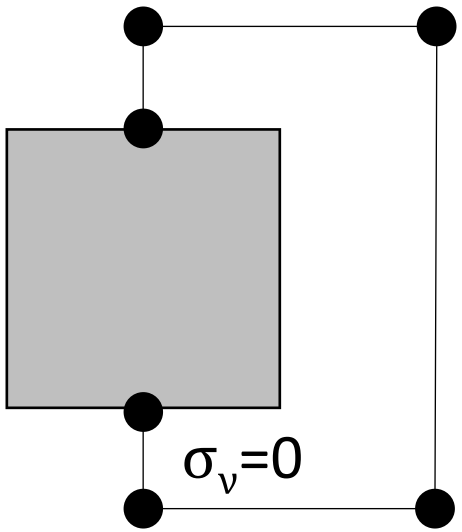
- The fixed embedding algorithm, which is based on the network-flow model from Tamassia, can be easily extended with the variant that some subgraphs of G have a fixed drawing.

R-Nodes

- In order to test spirality σ_ν we attach a chain in the external face of G_ν and we fix its spirality to 4- σ_ν.

- For the children of ν that are not Q*-nodes we select all possible combinations of partial solutions and we fix the representation of each one with the corresponding representative representation.

$$\Sigma_\mu \rightarrow (\sigma_\mu, X_\mu) \rightarrow (b_\mu, H_\mu)$$



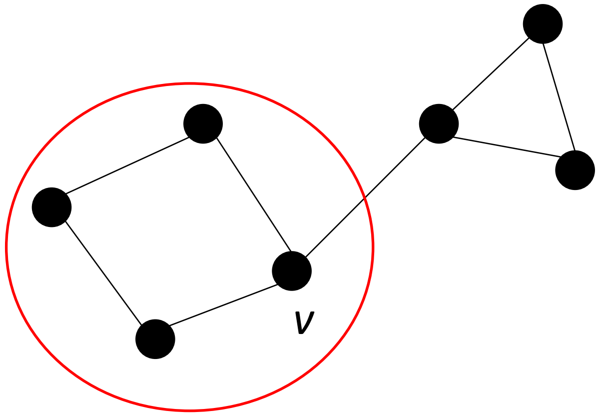
- We perform the above computation for σ_ν and each combination of partial solutions in order to compute the set Σ_ν.

A Glimpse to the
- General case -

The General Case

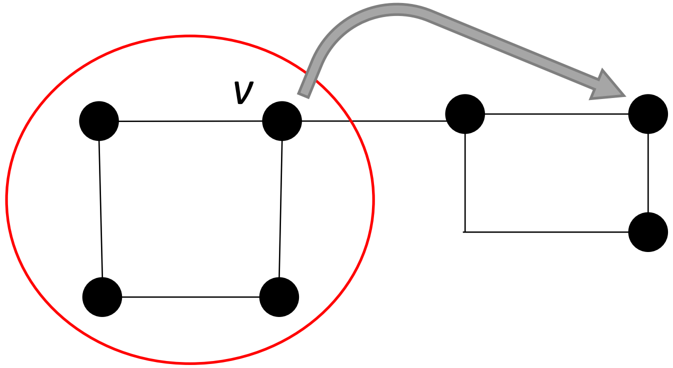
- A degree-2 vertex of a biconnected component (block) may correspond to higher degree vertices in the graph.

- We can associate it to the at least 1 degree-2 vertex or bend associated to the connected component on the opposite side.



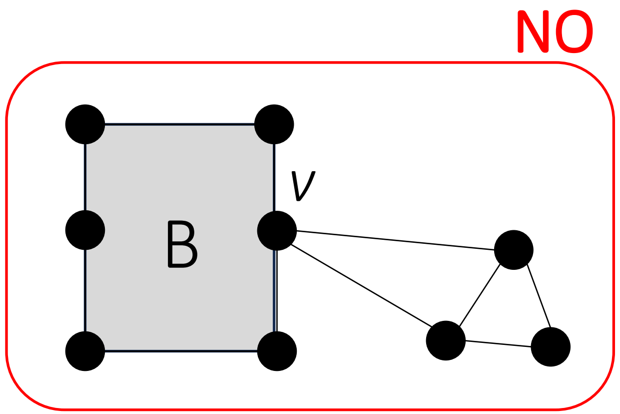
- **Observation:**

  The graph has k+b cutvertices and consequently at most 2k+b vertices of degree-2 vertices in any biconnected component.

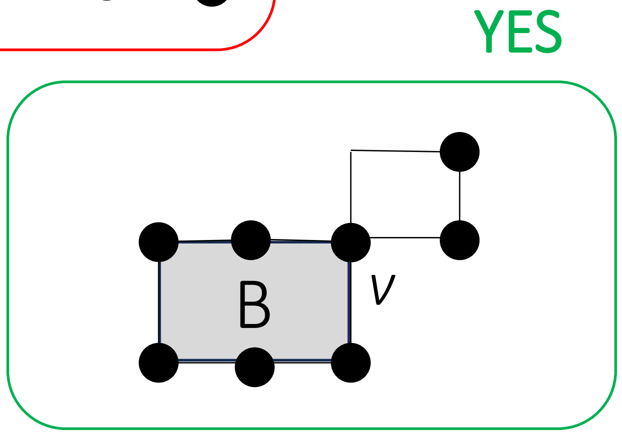


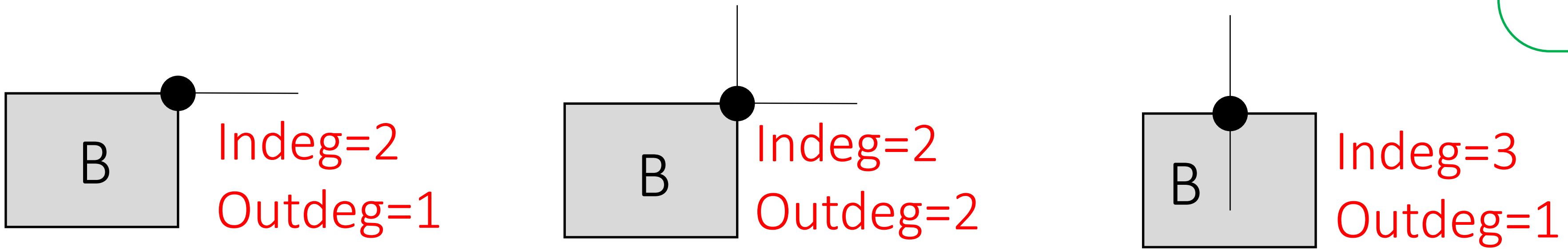
The General Case

- We extend the algorithm in order to handle potential angle constraints at the cutvertices.



- There are several types of angle constraints that we have to consider, depending on the indegree and outdegree of the cutvertex in the block B





- We test these constraint using the *Block-cutvertex tree*, which is a decomposition of the graph into its blocks.

Open Problems

Open Problems

- Can we reduce the number of our parameters from 2 to 1?
- Are there other interesting parameters for this problem?
- Are there other interesting upperbounds for the spirality of a component?

Thanks for your attention!