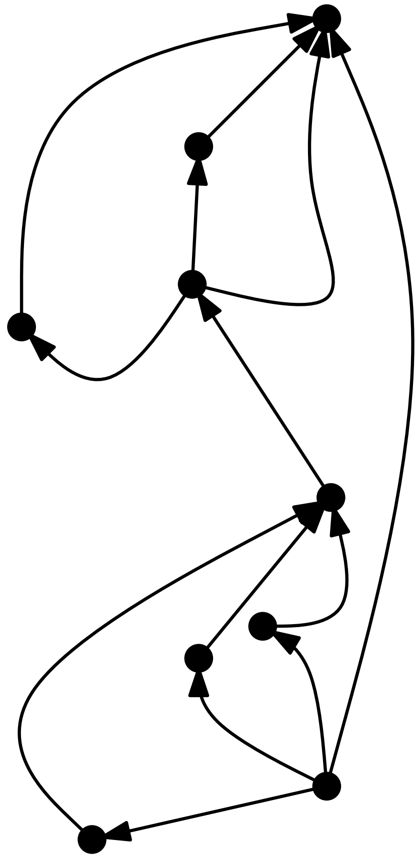
1-bend Upward Planar Drawings of SP-digraphs with the Optimal Number of Slopes

Emilio Di Giacomo, Giuseppe Liotta,
Fabrizio Montecchiani

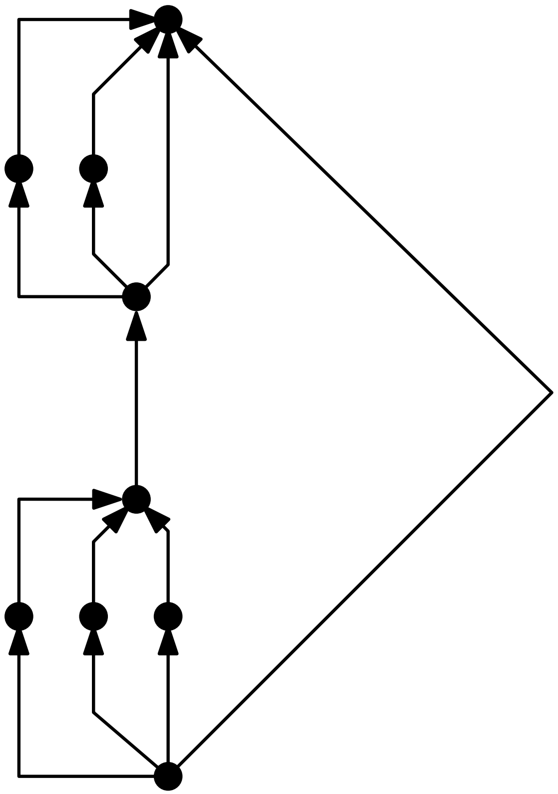
Università degli Studi di Perugia, Italy

GD 2016, September 19-21, Athens

How to draw a planar digraph?

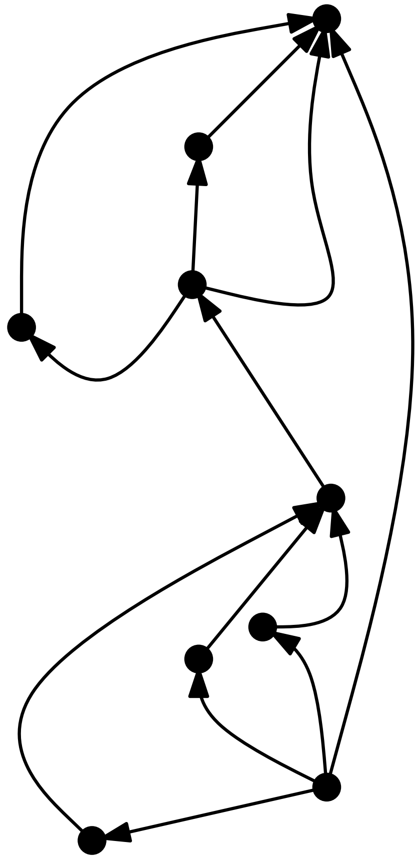


Input: A planar digraph G

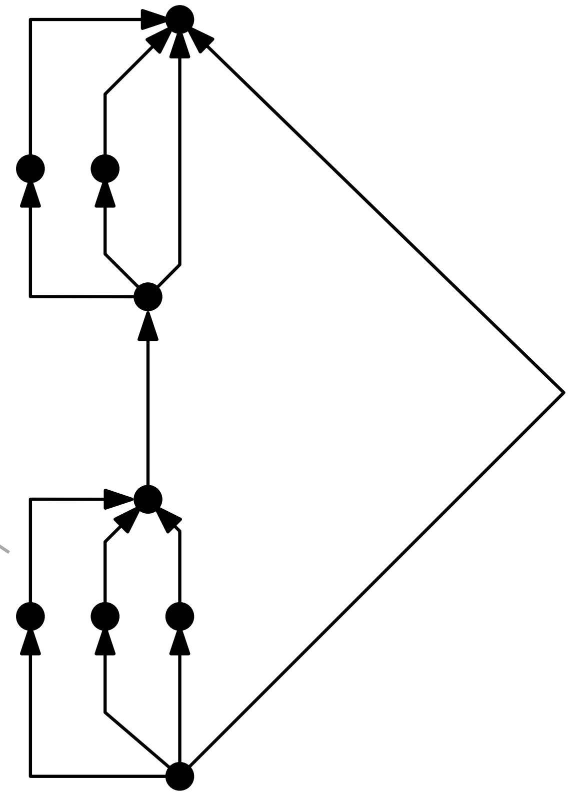


Output: A drawing Γ of G

How to draw a planar digraph?



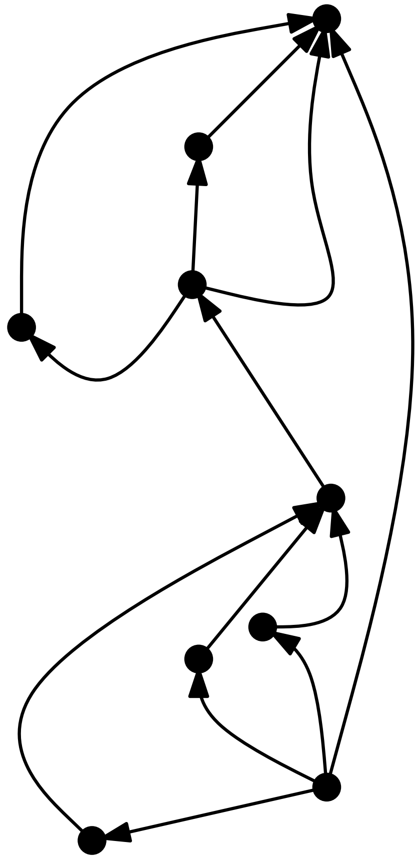
edges are
y-monotone
curves



Input: A planar digraph G

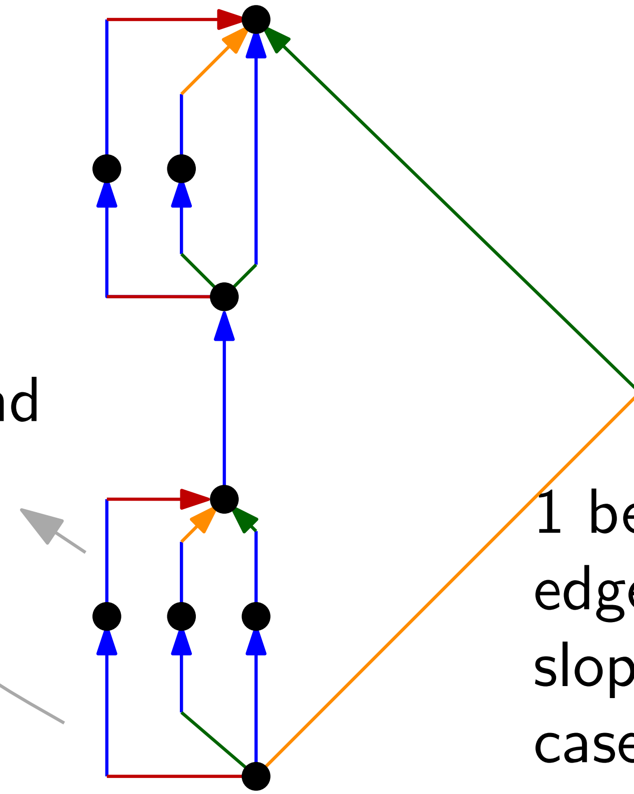
Output: A drawing Γ of G
- upward

How to draw a planar digraph?



Input: A planar digraph G

edges are polylines using few segments and few slopes

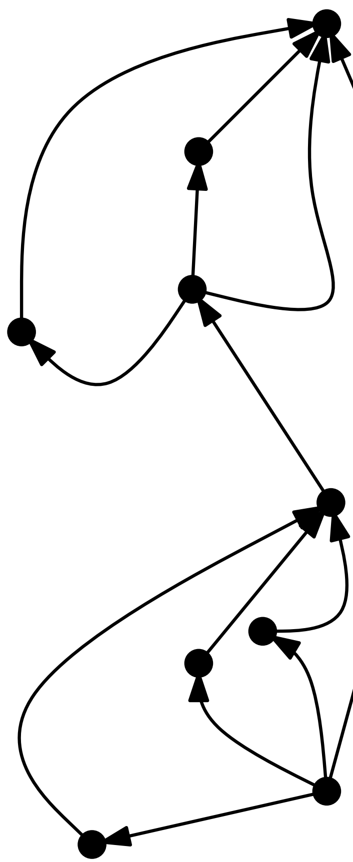


1 bend per edge and four slopes in this case

Output: A drawing Γ of G

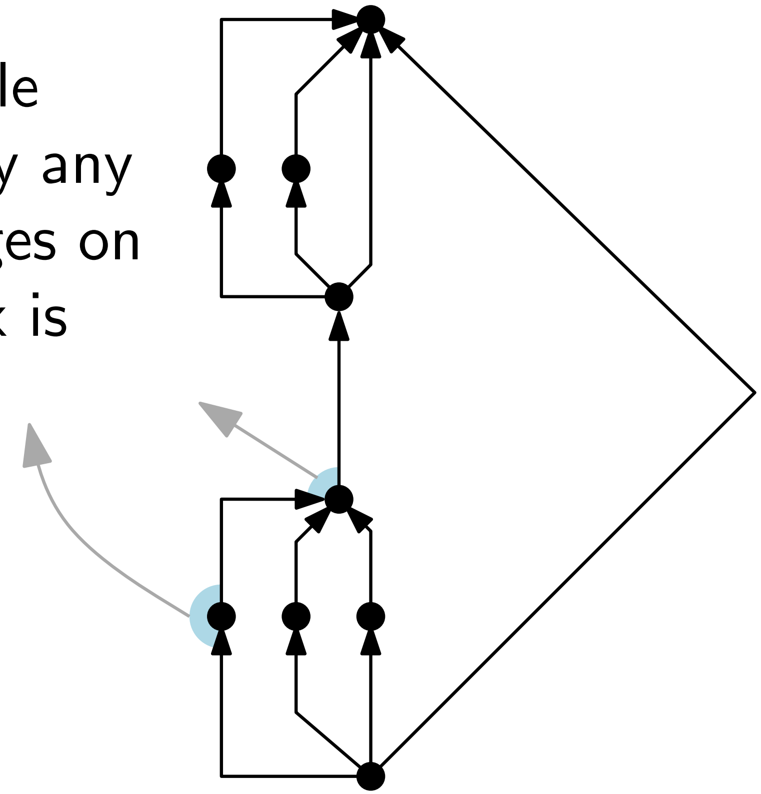
- upward
- with few bends and few edge slopes

How to draw a planar digraph?



Input: A planar digraph G

the angle made by any two edges on a vertex is large



Output: A drawing Γ of G

- upward
- with few bends and few edge slopes
- with large angular resolution

Drawing graphs with few slopes

k-bend planar slope number (k-bpsn) of a graph G: minimum number of slopes needed to compute a planar polyline drawing of G with at most k bends per edge.

$$k\text{-bpsn}(G) \geq \left\lceil \frac{\Delta}{2} \right\rceil$$

where Δ is the maximum vertex degree of G

Drawing graphs with few slopes

- The 2-bpsn of planar graphs is $\left\lceil \frac{\Delta}{2} \right\rceil$ [Keszech et al., 2010]

Drawing graphs with few slopes

- The 2-bpsn of planar graphs is $\lceil \frac{\Delta}{2} \rceil$ [Keszech et al., 2010]
- The 1-bpsn of planar graphs is at most 1.5Δ [Knauer & Walczak, 2015] and at least $0.75(\Delta - 1)$ [Ksezech et al., 2010]
- The 1-bpsn of outerplanar graphs is $\lceil \frac{\Delta}{2} \rceil$ [Knauer & Walczack, 2015]

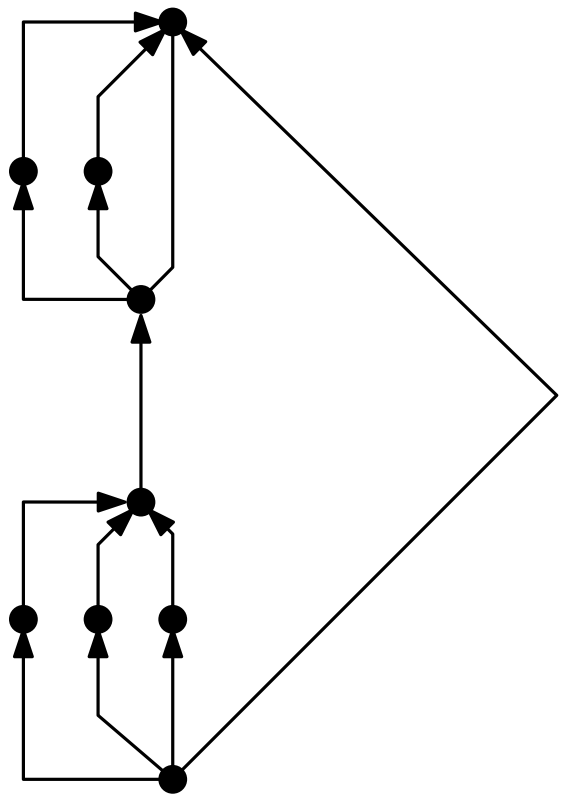
Drawing graphs with few slopes

- The 2-bpsn of planar graphs is $\lceil \frac{\Delta}{2} \rceil$ [Keszech et al., 2010]
- The 1-bpsn of planar graphs is at most 1.5Δ [Knauer & Walczak, 2015] and at least $0.75(\Delta - 1)$ [Ksezech et al., 2010]
- The 1-bpsn of outerplanar graphs is $\lceil \frac{\Delta}{2} \rceil$ [Knauer & Walczack, 2015]
- The psn of planar graphs is at most $O(c^{\Delta})$ and at least $3\Delta - 6$ [Keszech et al., 2010]

Our contribution

The *1-bend upward planar slope number (1-bupsn)* of a graph G is the minimum number of slopes needed to compute a 1-bend upward planar drawing of G

Observation: 1-bupsn(G) $\geq$ 1-bpsn(G)



Our contribution

We show that the **1-bupsn** of any **series-parallel digraph** with maximum vertex degree Δ is at most Δ, and this bound is worst-case optimal

Our contribution

We show that the **1-bupsn** of any **series-parallel digraph** with maximum vertex degree Δ is at most Δ, and this bound is worst-case optimal

This result improves the general upper bound 1.5Δ of the 1-bpsn of planar graphs in the case of series-parallel graphs [Knauer & Walczak, 2015]

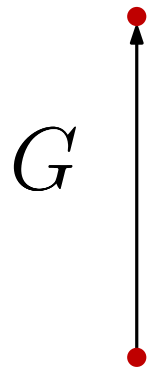
Our drawings can be computed in linear time and have angular resolution at least $\frac{\pi}{\Delta}$ (worst-case optimal)

Preliminary definitions

Series-Parallel Digraphs

A series-parallel digraph (SP-digraph for short) is a simple planar digraph that has one *source* and one *sink*, called *poles*, and it is recursively defined as follows.

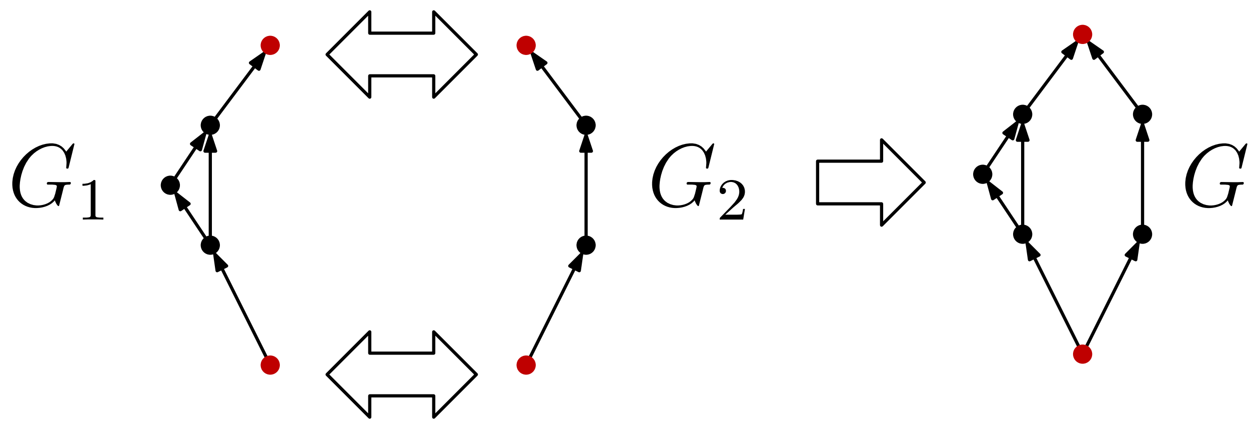
A single edge is an SP-digraph.



Series-Parallel Digraphs

A series-parallel digraph (SP-digraph for short) is a simple planar digraph that has one *source* and one *sink*, called *poles*, and it is recursively defined as follows.

The digraph obtained by identifying the sources and the sinks of two SP-digraphs is an SP-digraph.

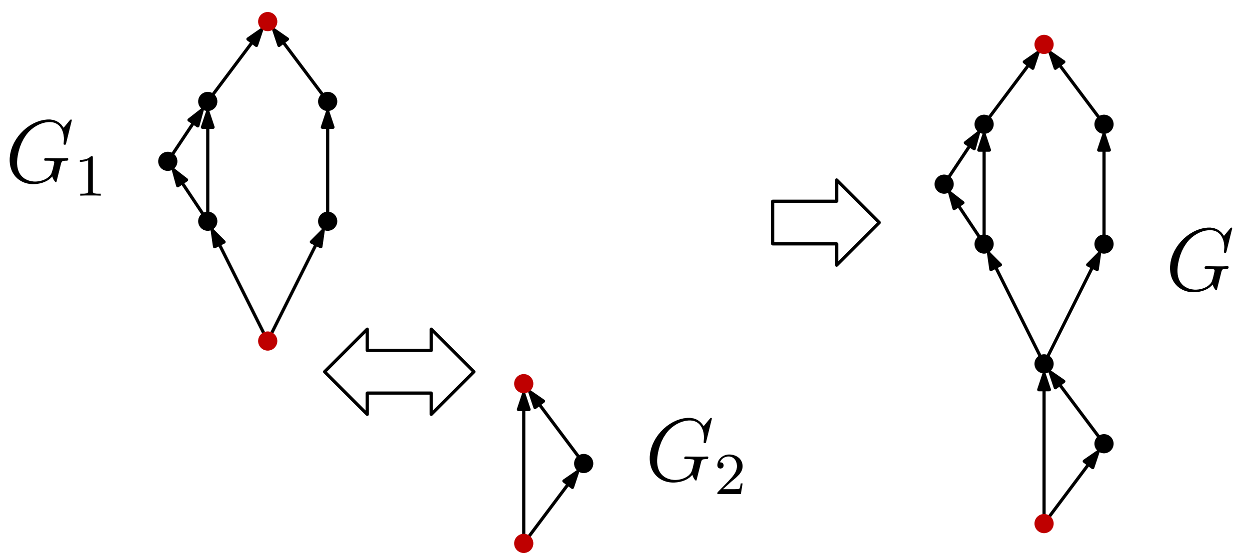


PARALLEL composition

Series-Parallel Digraphs

A series-parallel digraph (SP-digraph for short) is a simple planar digraph that has one *source* and one *sink*, called *poles*, and it is recursively defined as follows.

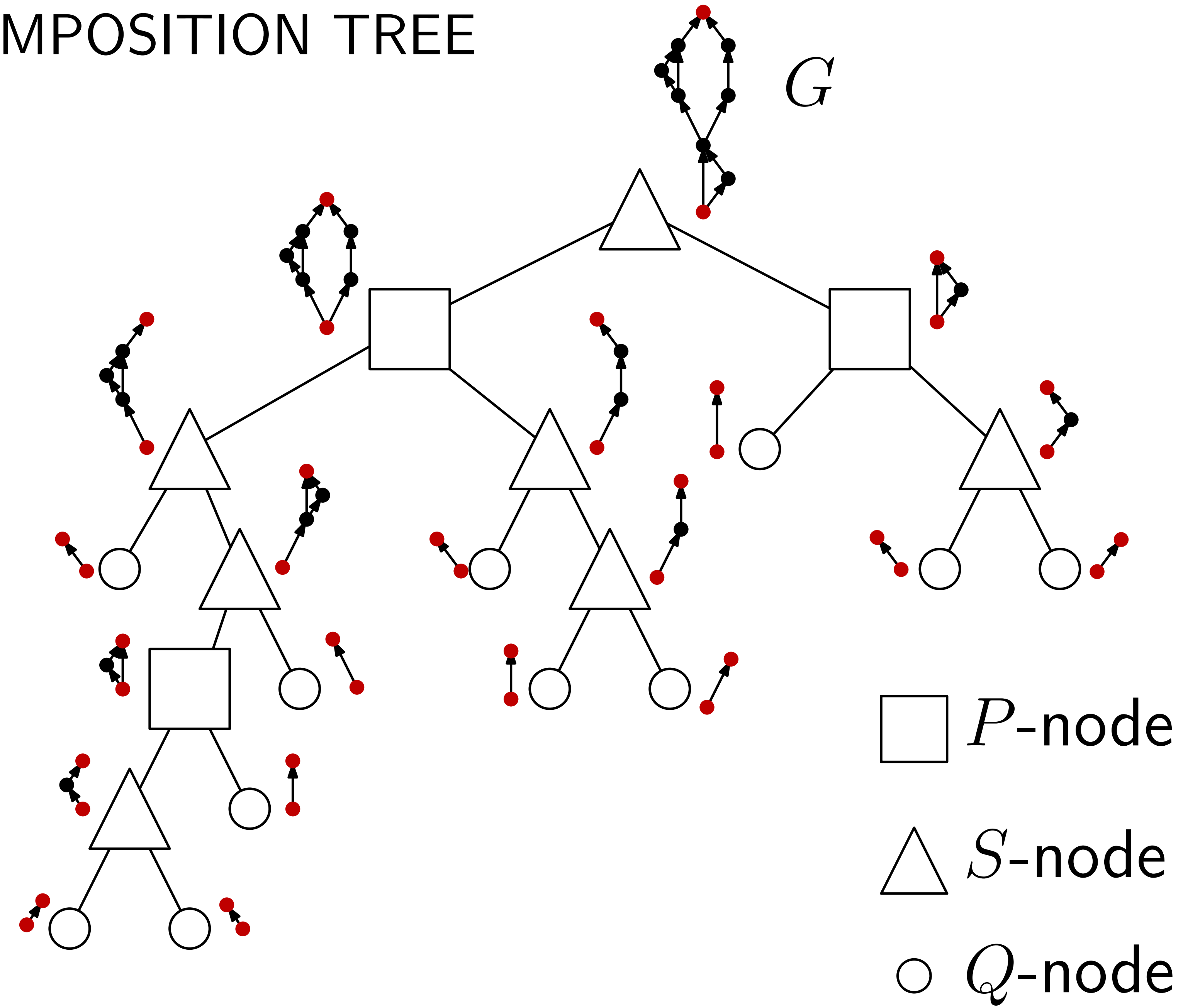
The digraph obtained by identifying the sink of a SP-digraph with the source of another SP- digraph is an SP-digraph.



SERIES composition

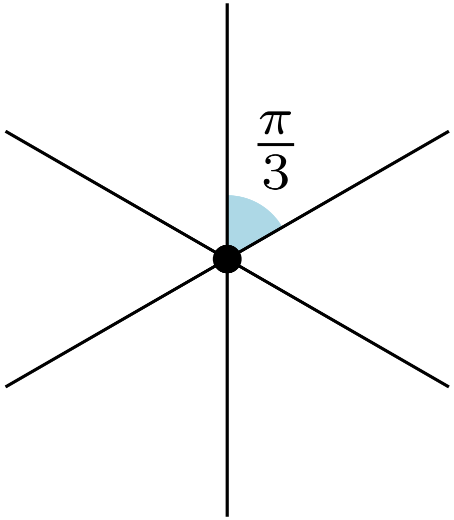
Series-Parallel Digraphs

DECOMPOSITION TREE

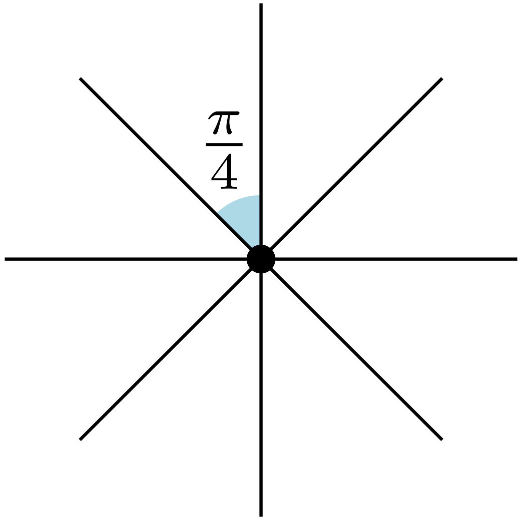


The Slope Set $\mathcal{S}_\Delta$

$$s_i = \frac{\pi}{2} + i\frac{\pi}{\Delta} \text{ for } i = 0, \ldots, \Delta - 1$$



$\Delta = 3$



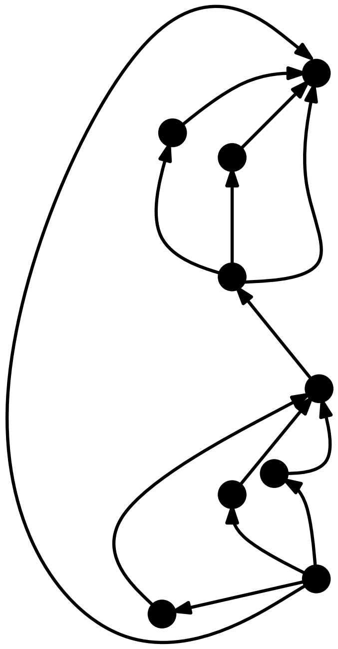
$\Delta = 4$

Proof overview

Proof Overview

Input: an SP-digraph G

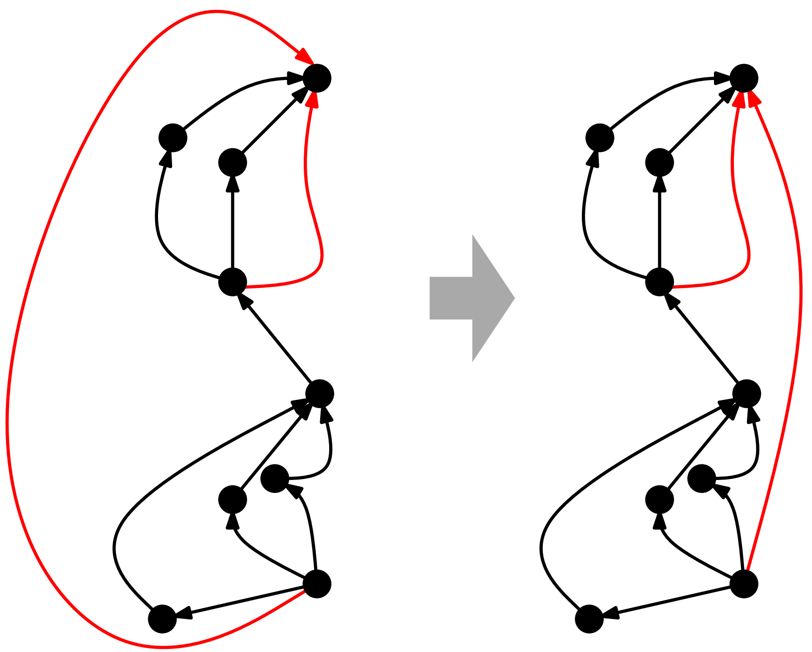
Output: a 1-bend upward planar drawing Γ of G with at most Δ slopes of the slope-set $\mathcal{S}_\Delta$



Proof Overview

Input: an SP-digraph G
Output: a 1-bend upward planar drawing Γ of G with at most Δ slopes of the slope-set $\mathcal{S}_\Delta$

- right push transitive edges

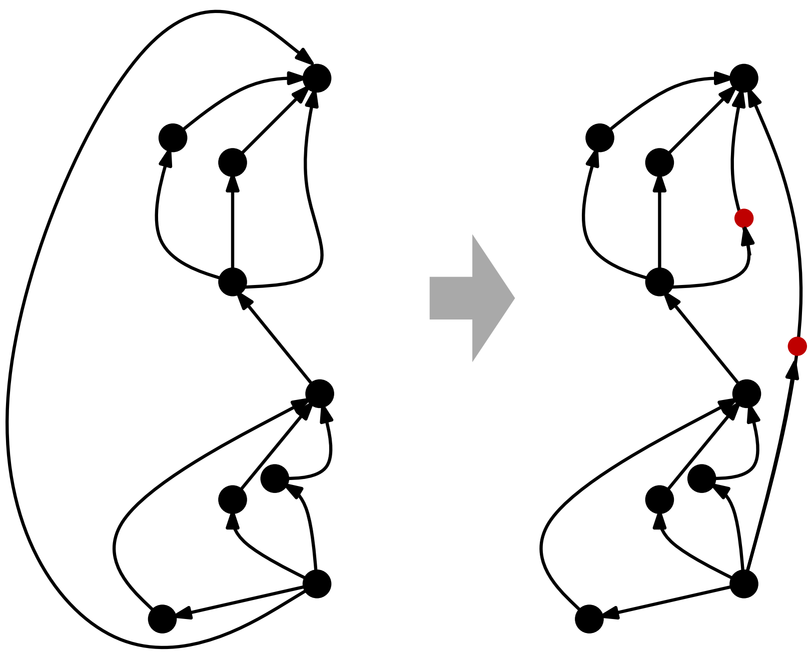


Proof Overview

Input: an SP-digraph G

Output: a 1-bend upward planar drawing Γ of G with at most Δ slopes of the slope-set $\mathcal{S}_\Delta$

- right push + subdivide transitive edges

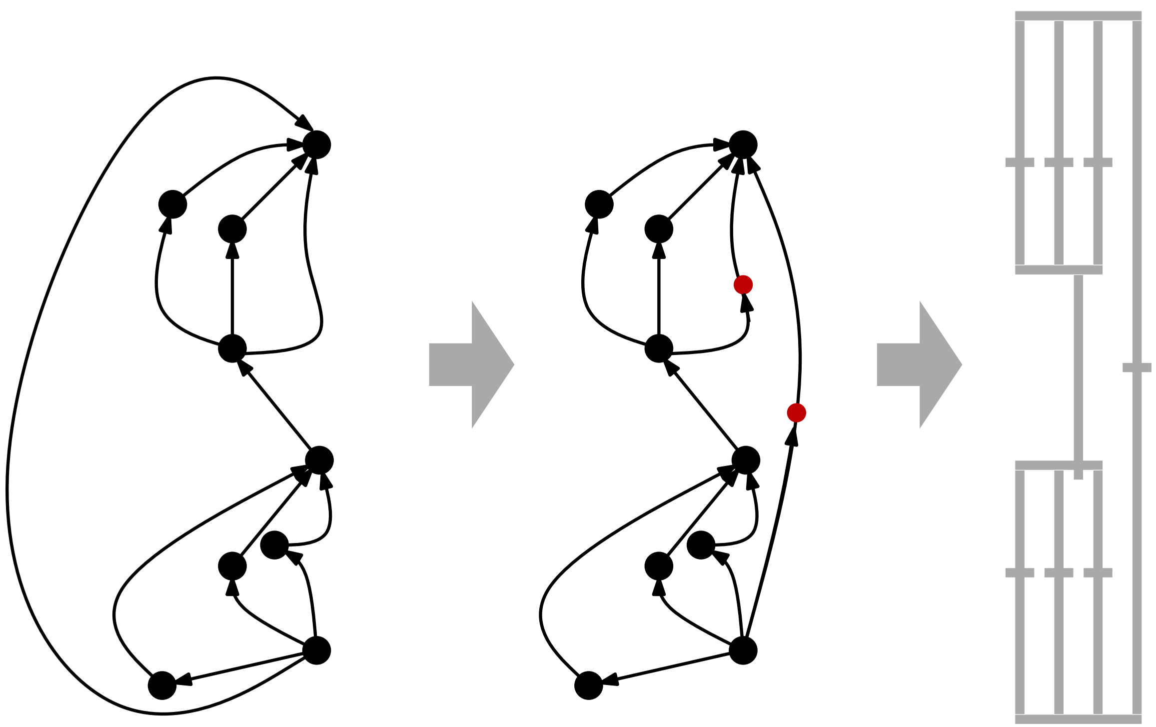


Proof Overview

Input: an SP-digraph G

Output: a 1-bend upward planar drawing Γ of G with at most Δ slopes of the slope-set $\mathcal{S}_\Delta$

- right push $+$ subdivide transitive edges
- Construct a cross-contact representation γ of G

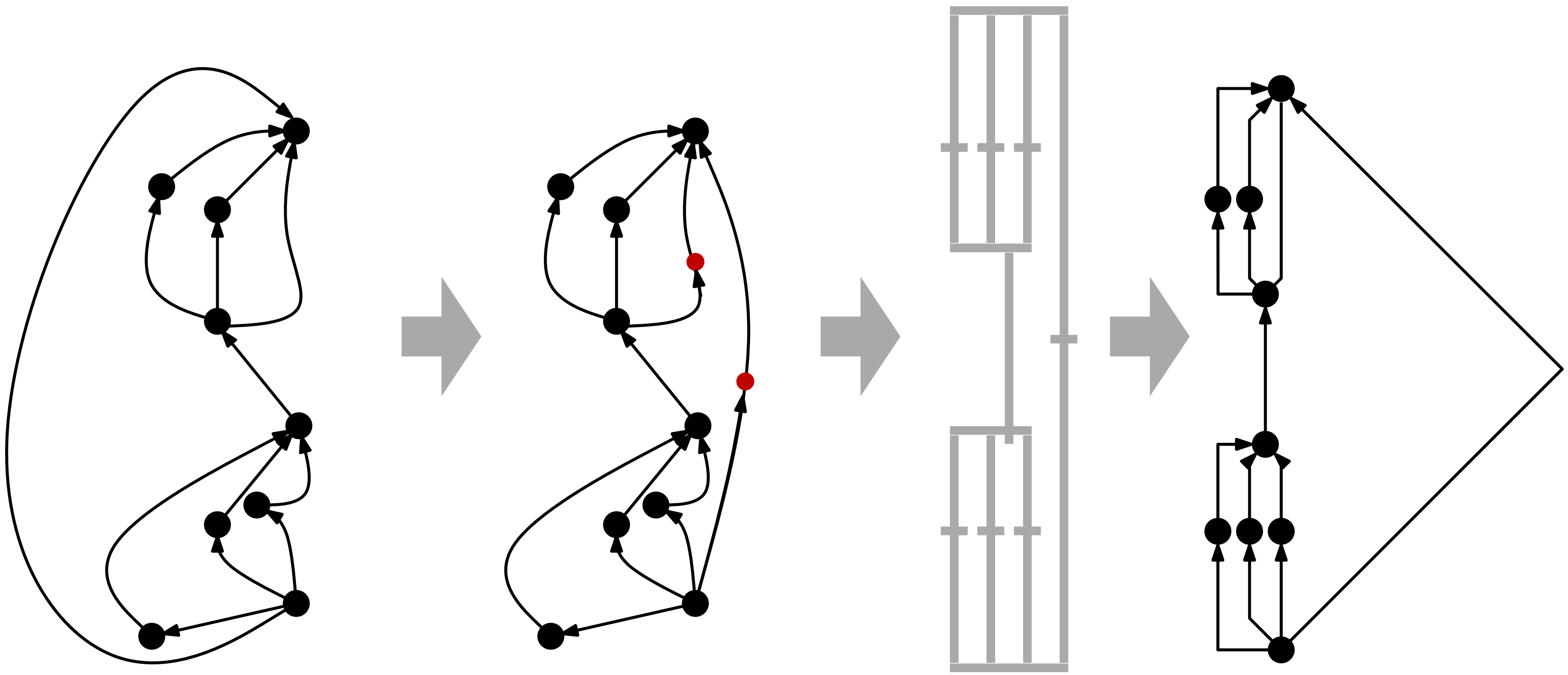


Proof Overview

Input: an SP-digraph G

Output: a 1-bend upward planar drawing Γ of G with at most Δ slopes of the slope-set $\mathcal{S}_\Delta$

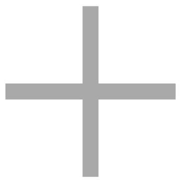
- right push + subdivide transitive edges
- Construct a cross-contact representation γ of G
- Transform Γ into the desired representation
- Remove subdivsion vertices



Upward Cross-Contact Representations

Upward Cross-Contact Representations (UCCR)

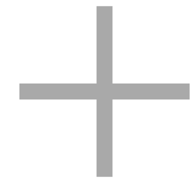
cross $=$ a horizontal segment and a vertical segment sharing an inner point



degenerate cross $=$ a horizontal/vertical segment

Upward Cross-Contact Representations (UCCR)

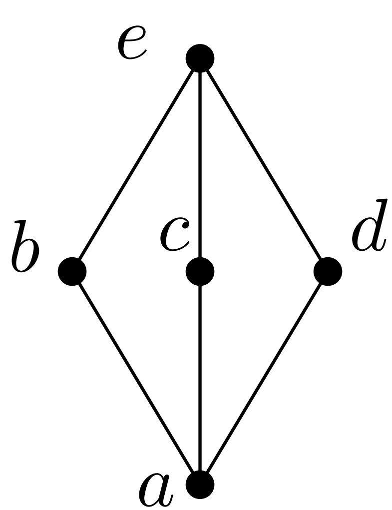
cross = a horizontal segment and a vertical segment sharing an inner point

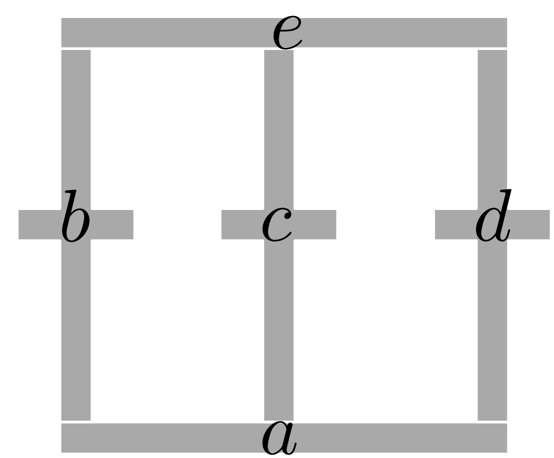


degenerate cross = a horizontal/vertical segment

cross-contact representation (CCR) γ of a graph G:

- Vertices = (Degenerate) Crosses
- Edges $\iff$ Contacts

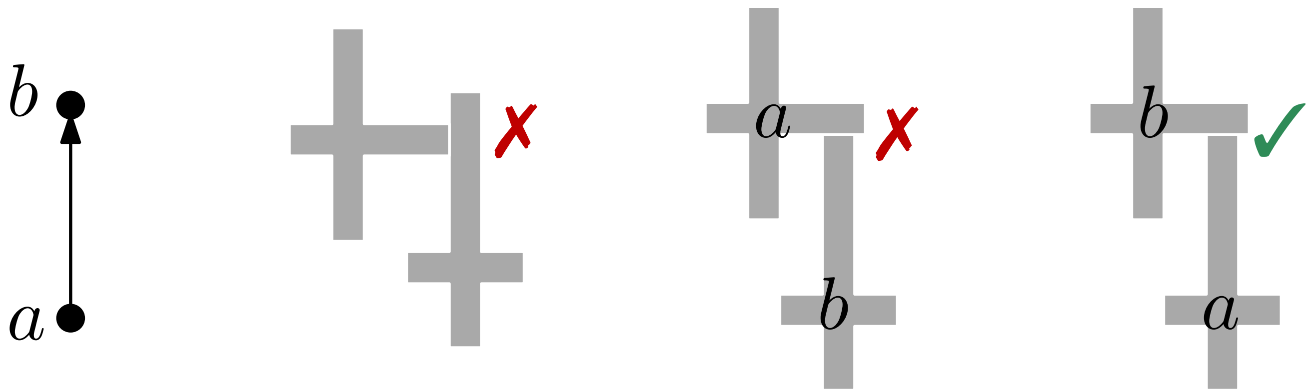




Upward Cross-Contact Representations (UCCR)

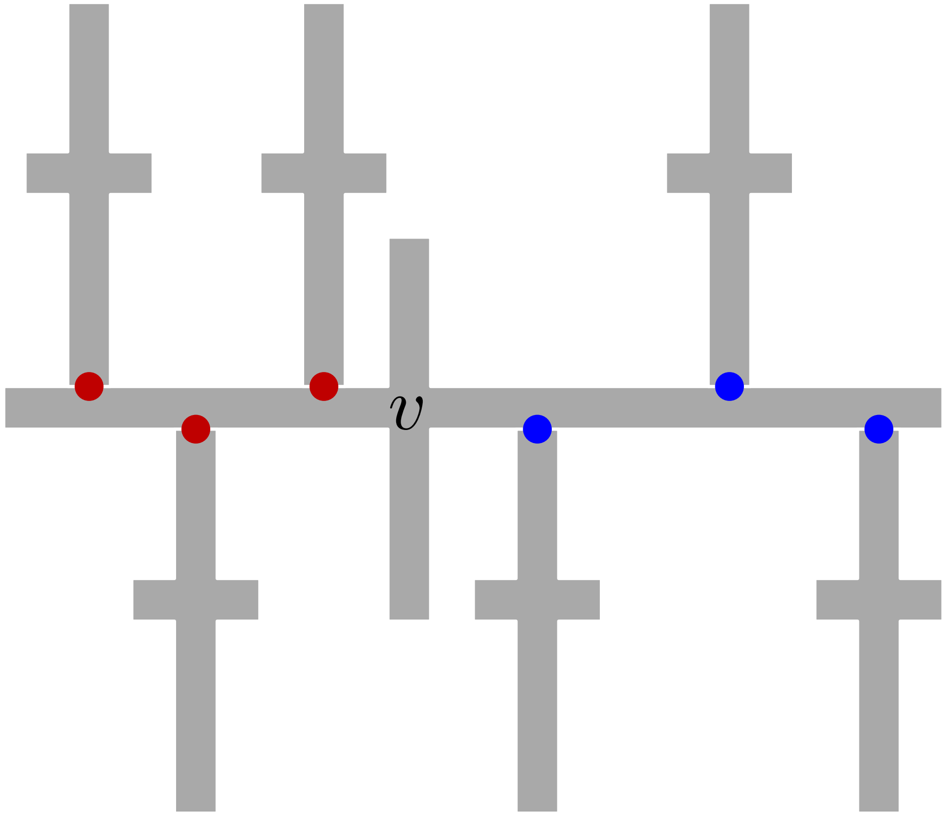
center of a cross = the point shared by its horizontal and vertical segment, or its midpoint if degenerate

upward CCR (UCCR) of a digraph G



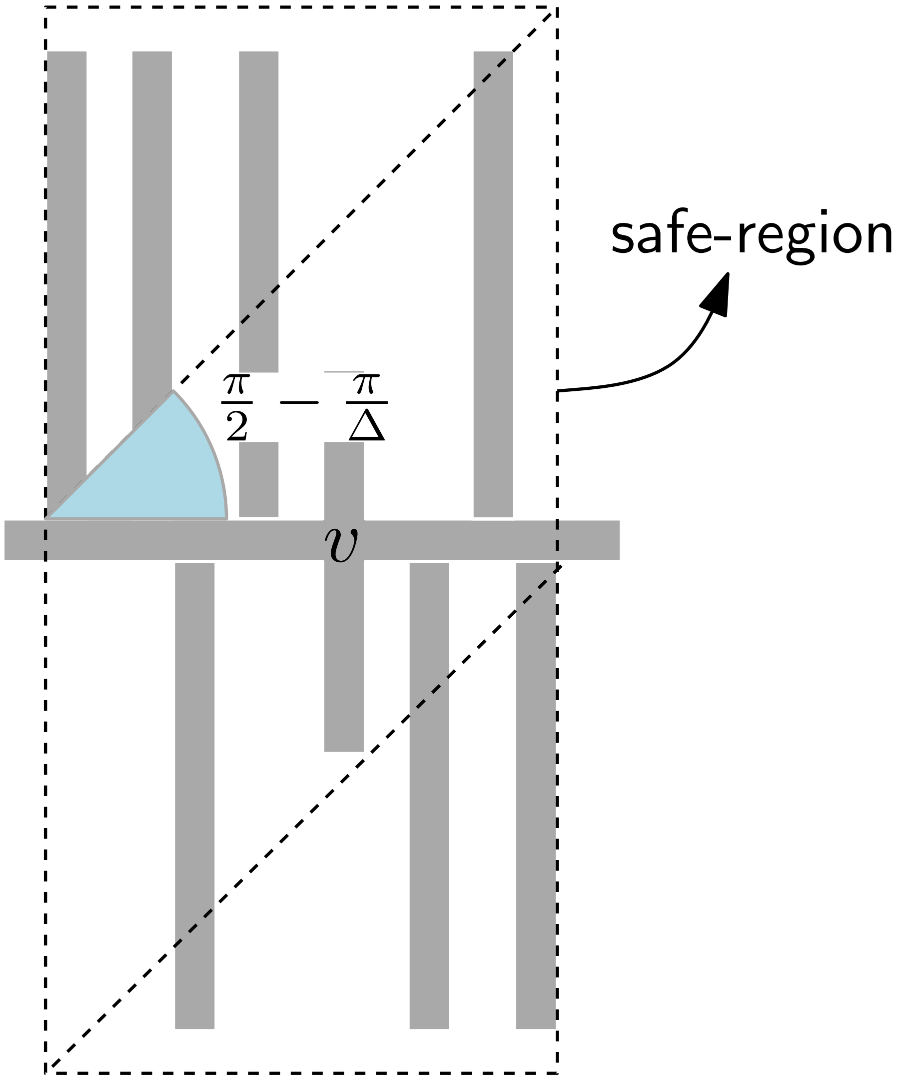
Upward Cross-Contact Representations (UCCR)

balanced UCCR = for every cross, we have the same number of contacts to the left and to the right of its center, except for at most one



Upward Cross-Contact Representations (UCCR)

well-spaced UCCR $=$ no two *safe-regions* intersect



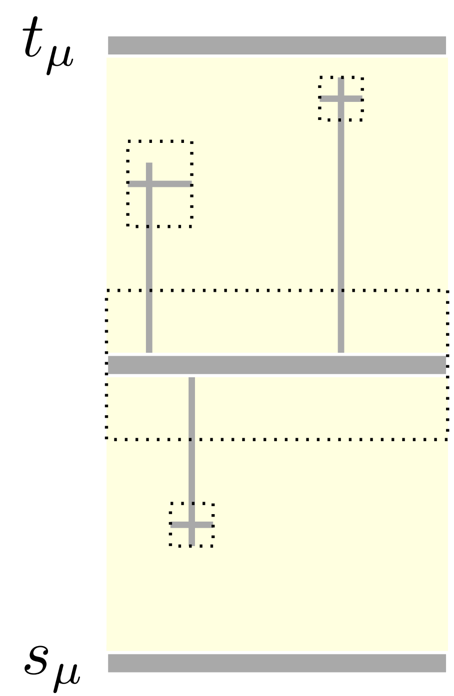
Sketch of the Algorithm

Drawing Algorithm

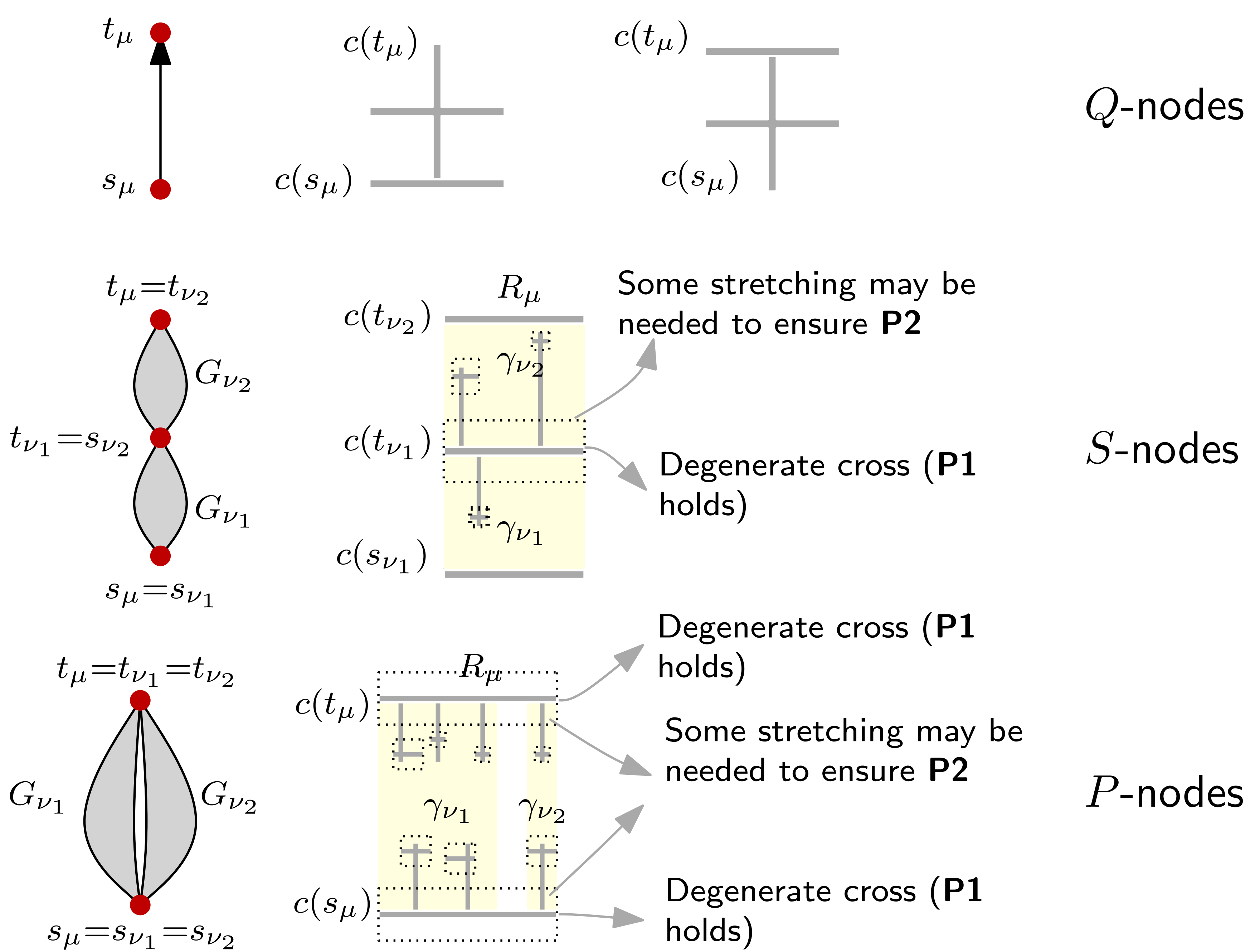
Input: an SP-digraph G with no transitive edges (subdivided before) and its decomposition tree T
Output: a balanced and well-spaced UCCR γ of G

The algorithm computes γ through a bottom-up visit of T. For each node μ of T computes an UCCT γ_μ of the graph G_μ associated with μ s.t. the following properties hold:

P1. γ_μ is balanced
P2. γ_μ is well-spaced
P3. if μ is an S-/P-node, than γ_μ fits in a rectangle R_μ with the two poles as opposite sides



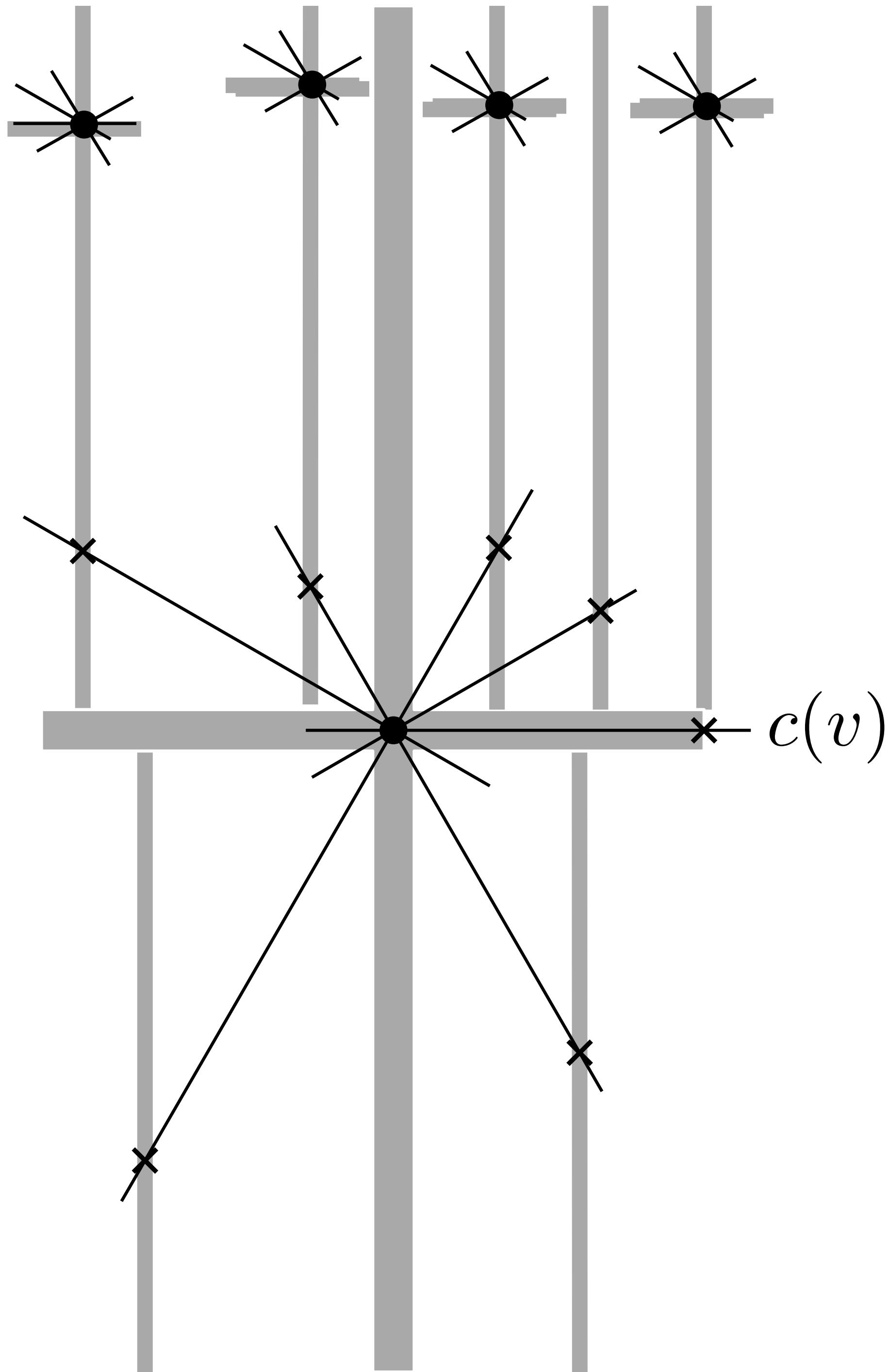
Q-/S-/P-nodes



t_μ
s_μ
$c(t_\mu)$
$c(s_\mu)$
$c(t_\mu)$
$c(s_\mu)$
Q-nodes
$t_\mu = t_{\nu_2}$
G_{ν_2}
$t_{\nu_1} = s_{\nu_2}$
G_{ν_1}
$s_\mu = s_{\nu_1}$
R_μ
$c(t_{\nu_2})$
γ_{ν_2}
$c(t_{\nu_1})$
γ_{ν_1}
$c(s_{\nu_1})$
Some stretching may be needed to ensure **P2**
Degenerate cross (**P1** holds)
S-nodes
$t_\mu = t_{\nu_1} = t_{\nu_2}$
G_{ν_1}
G_{ν_2}
$s_\mu = s_{\nu_1} = s_{\nu_2}$
R_μ
$c(t_\mu)$
γ_{ν_1}
γ_{ν_2}
$c(s_\mu)$
Degenerate cross (**P1** holds)
Some stretching may be needed to ensure **P2**
Degenerate cross (**P1** holds)
P-nodes

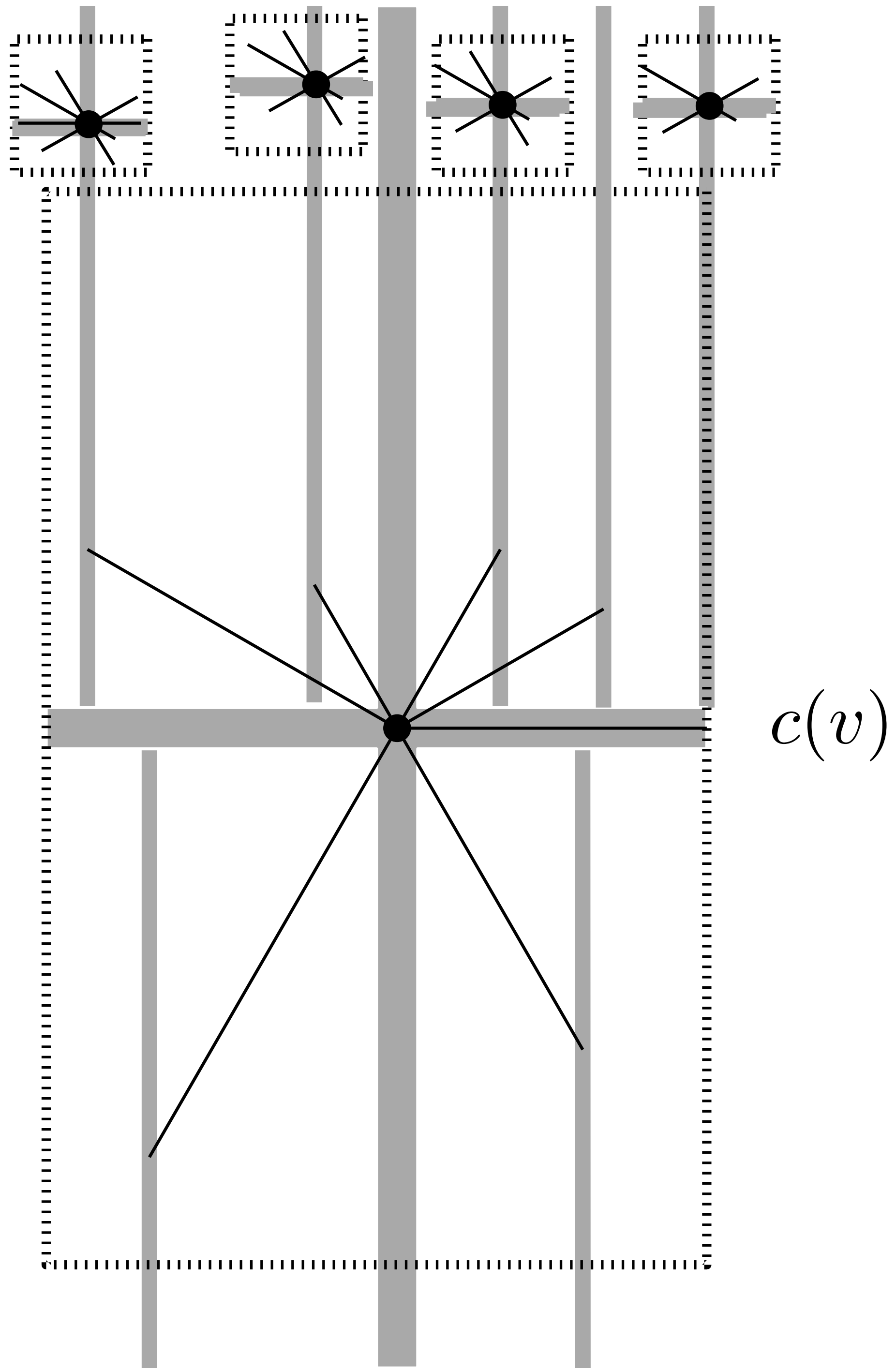
UCCR $\rightarrow$ 1-bend drawing

each cross is balanced so we have enough slopes...



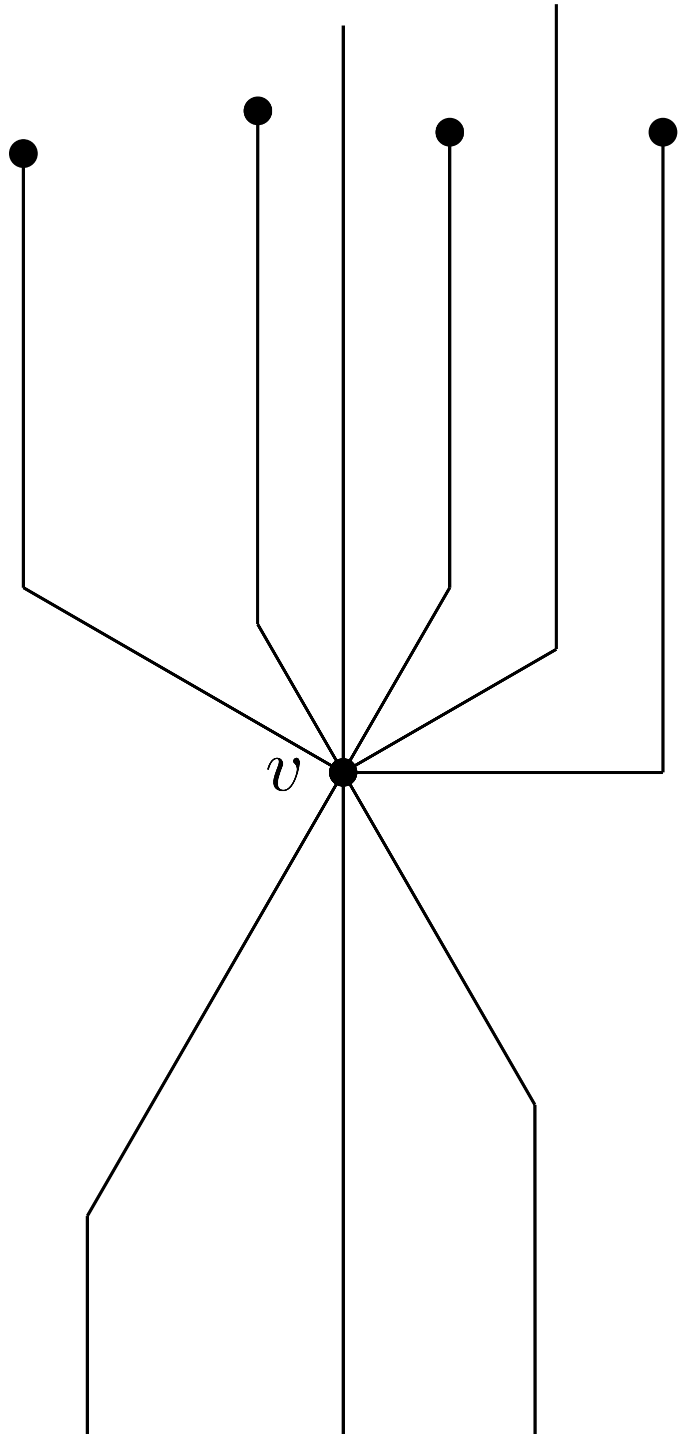
UCCR $\rightarrow$ 1-bend drawing

safe-regions do not intersect, so we do not introduce crossings...

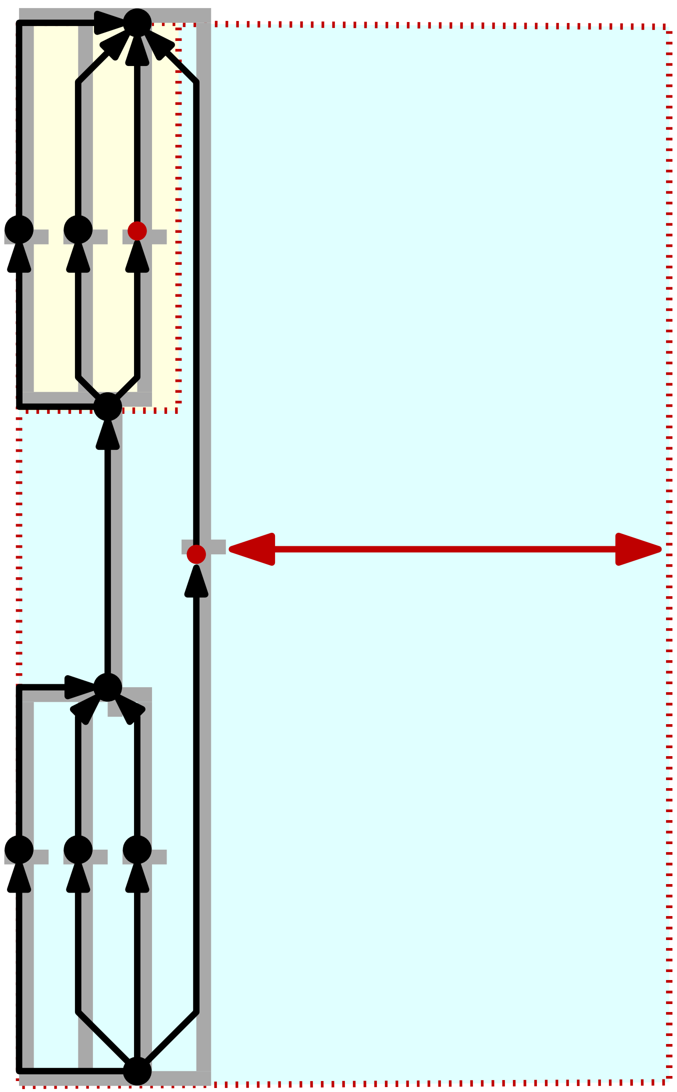


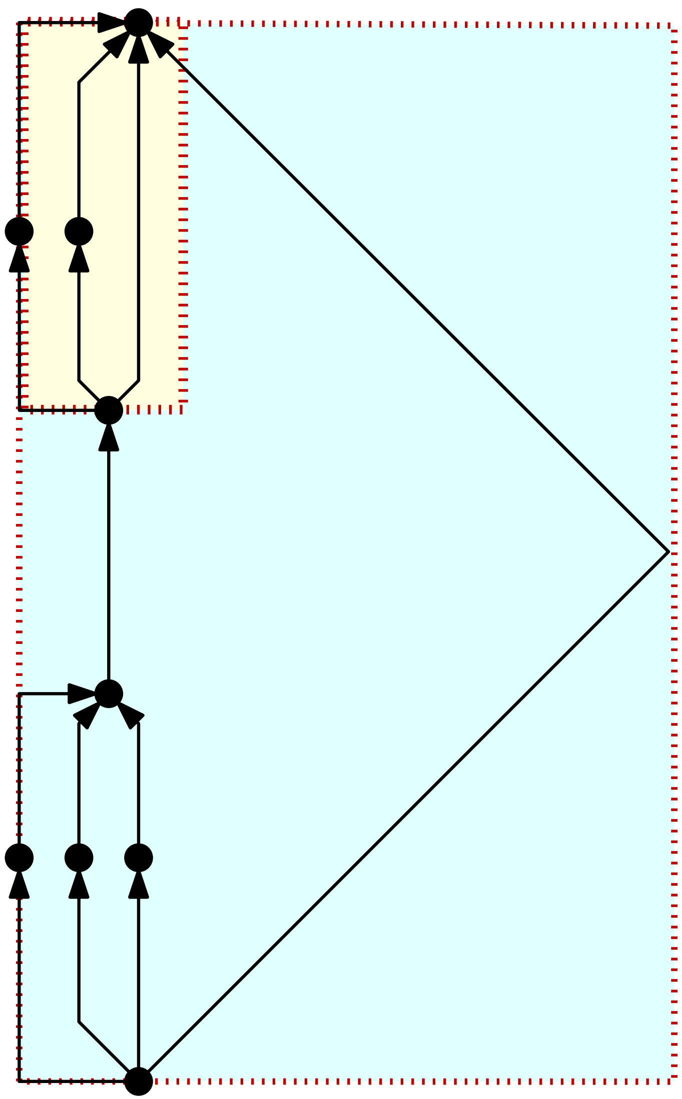
UCCR $\rightarrow$ 1-bend drawing

the vertical slope
is always part of
our set of slopes...



Removing subdivision vertices

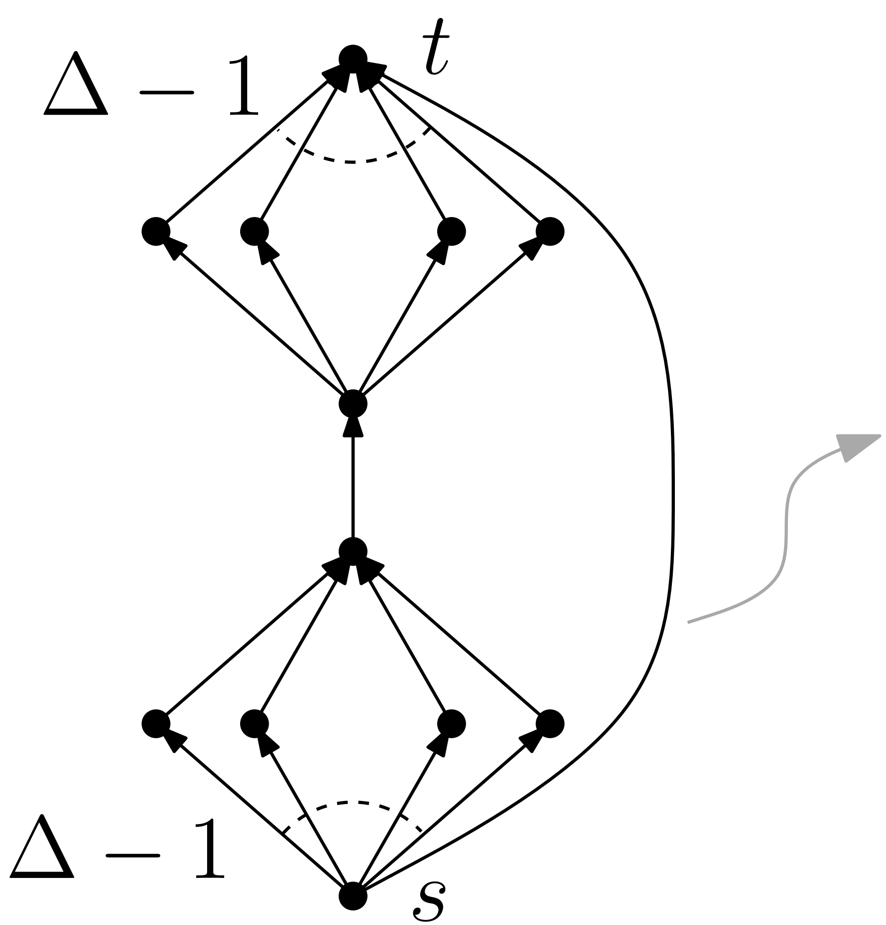




Lower bound

if the source (sink) has out-degree (in-degree) Δ, then $\Delta - 1$ slopes are necessary for an upward drawing

To achieve this bound, the horizontal slope must be used twice



The horizontal slope cannot be reused as otherwise 2 bends are needed

Open Problems

Is Δ a tight bound for the 1-bpsn of SP-graphs?

Can we extend this bound to all partial 2-trees?

What about upward planar graphs?

Open Problems

Is Δ a tight bound for the 1-bpsn of SP-graphs?

Can we extend this bound to all partial 2-trees?

What about upward planar graphs?

THANK YOU