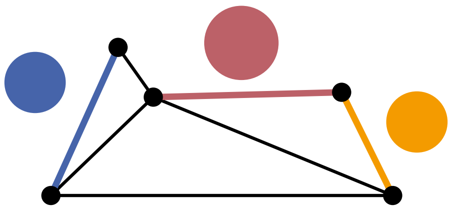
Strongly Monotone Drawings of Planar Graphs

Stefan Felsner	Technische Universität Berlin
Alexander Igamberdiev	FernUniversität in Hagen
Philipp Kindermann	FernUniversität in Hagen
Boris Klemz	Freie Universität Berlin
Tamara Mchedlidze	Karlsruhe Institute of Technology
Manfred Scheucher	Graz University of Technology

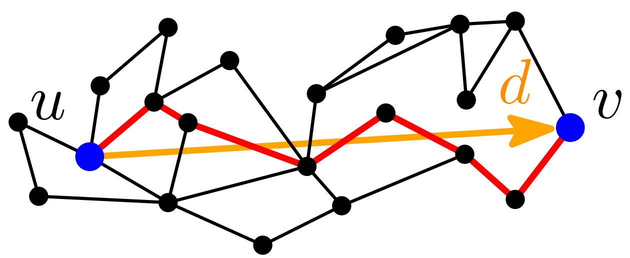


EuroCG 2016, Lugano

Strongly Monotone Drawings

We want planar straight-line graph drawings that are ...

Strongly monotone: between each vertex pair (u, v) exists a path that is monotonically increasing in direction $d = \overrightarrow{uv}$

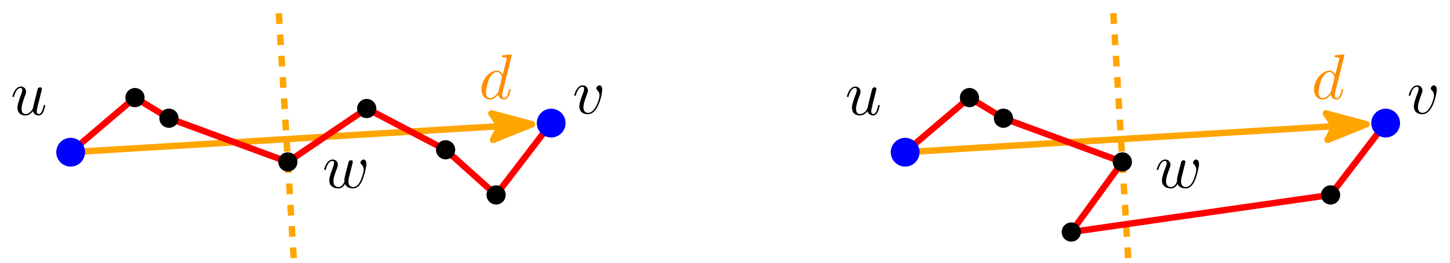


strongly monotone

Strongly Monotone Drawings

We want planar straight-line graph drawings that are ...

Strongly monotone: between each vertex pair (u, v) exists a path that is monotonically increasing in direction $d = \overrightarrow{uv}$

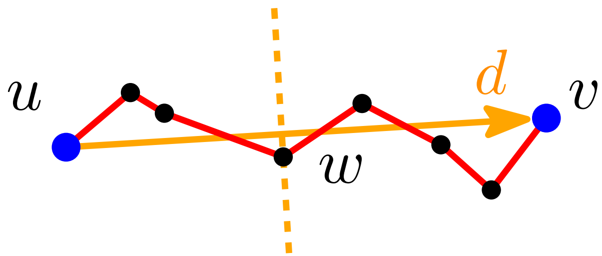


strongly monotone　　　　not strongly monotone

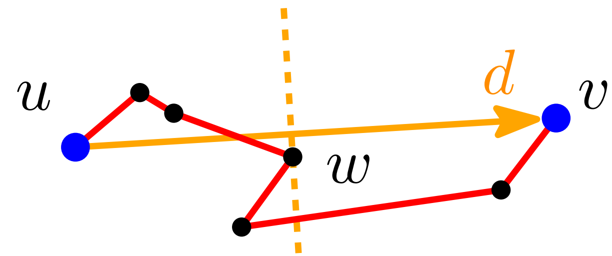
Strongly Monotone Drawings

We want planar straight-line graph drawings that are ...

Strongly monotone: between each vertex pair (u, v) exists a path that is monotonically increasing in direction $d = \overrightarrow{uv}$



strongly monotone



not strongly monotone

Motivation: Path-finding tasks become easy!

Related Work & Results

Strongly monotone drawings:

- do not exist for every planar graph [Kindermann et al. '14]
- exist for every 2-connected outerplanar graph [Kindermann et al. '14]
- exist for every tree [Kindermann et al. '14]
  - area required can be exponential [Nöllenburg et al. '14]

Related Work & Results

Strongly monotone drawings:

- do not exist for every planar graph [Kindermann et al. '14]
- exist for every 2-connected outerplanar graph [Kindermann et al. '14]
- exist for every tree [Kindermann et al. '14]
  - area required can be exponential [Nöllenburg et al. '14]

- exist for every planar 3-connectd graph
- exist for every outerplanar graph
- exist for every 2-tree

} **our results**

3-connected Graphs

Theorem: Every 3-connected planar graph has a strongly monotone drawing.

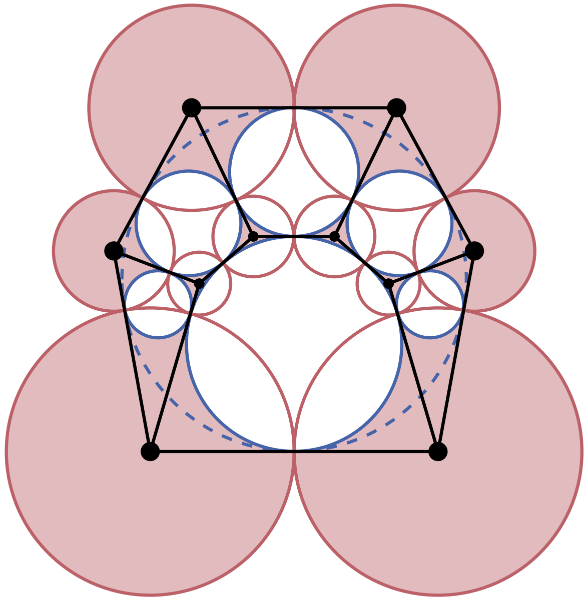
Proof idea:

Every 3-connected planar graph G admits a primal-dual circle packing $\mathcal{P}$.
[Brightwell, Scheinerman 1993]

Drawing induced by $\mathcal{P}$ is strongly monotone.

Primal Dual Circle Packings

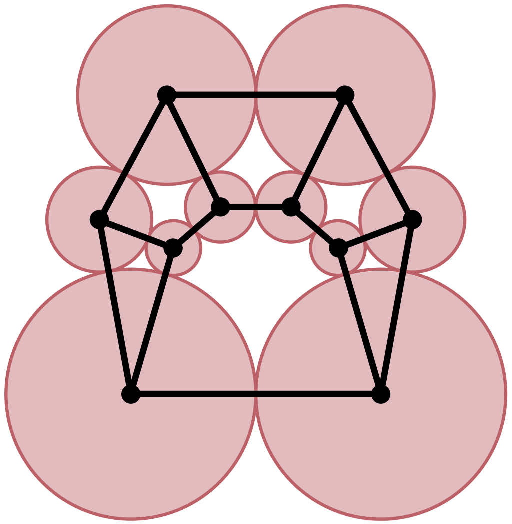
primal dual circle packing $\mathcal{P}$ of G



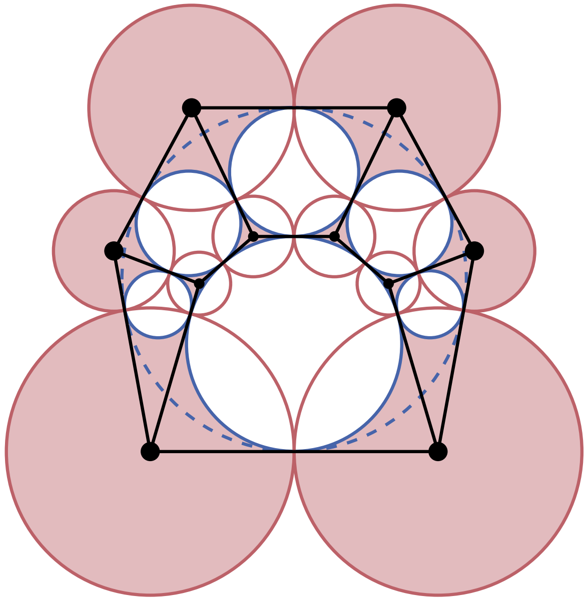
Primal Dual Circle Packings

circle contact representations of ...

(**primal**) graph G



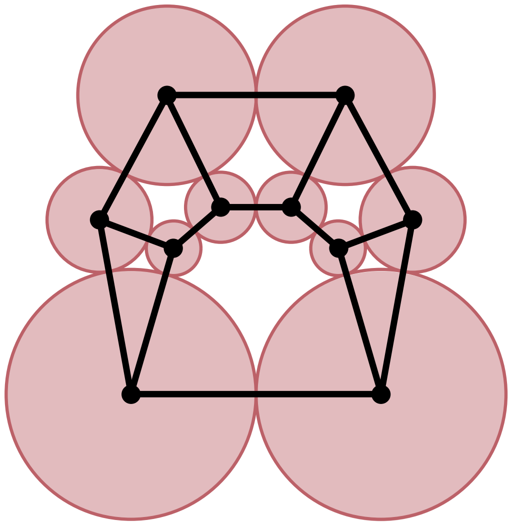
primal dual circle packing $\mathcal{P}$ of G



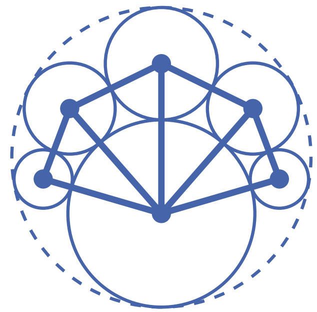
Primal Dual Circle Packings

circle contact representations of ...

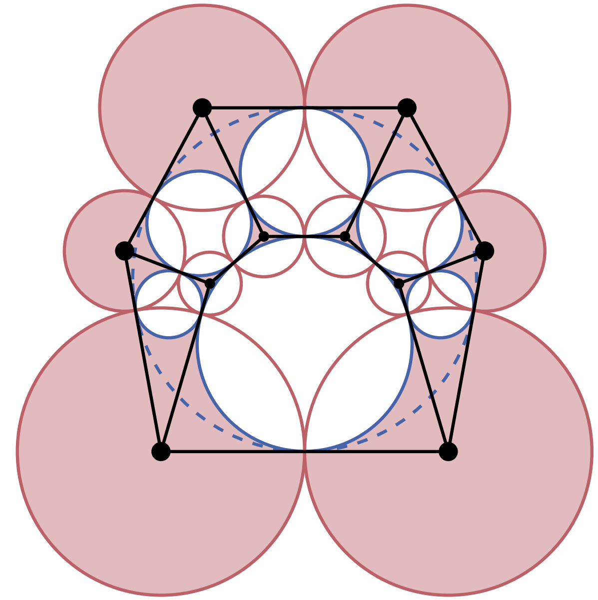
(**primal**) graph G



dual graph of G



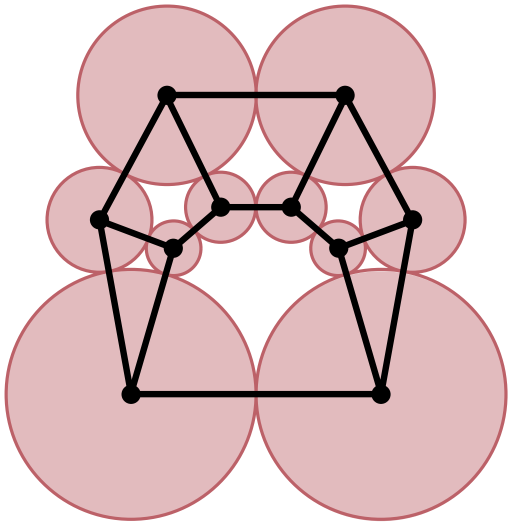
primal dual circle packing $\mathcal{P}$ of G



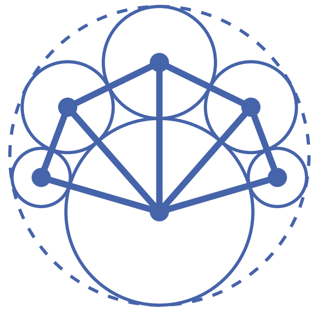
Primal Dual Circle Packings

circle contact representations of ...

(**primal**) graph G

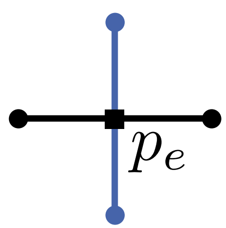


dual graph of G

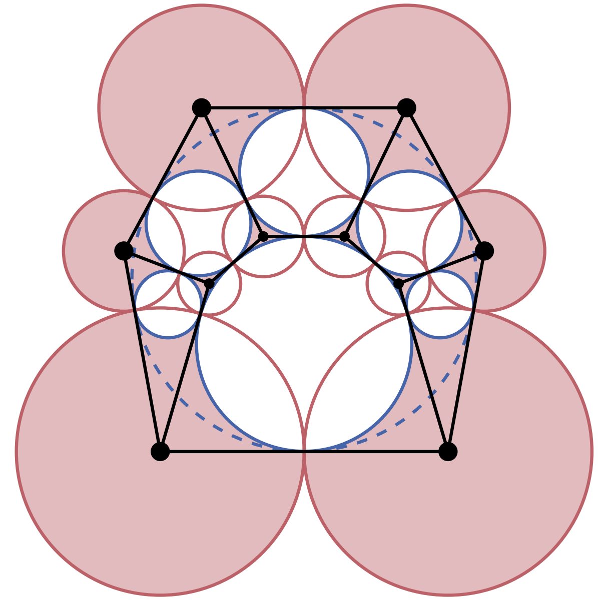


... which are **orthogonal**:

edge e **crosses**
its dual edge e^*
perpendicularly
in **edge point** p_e



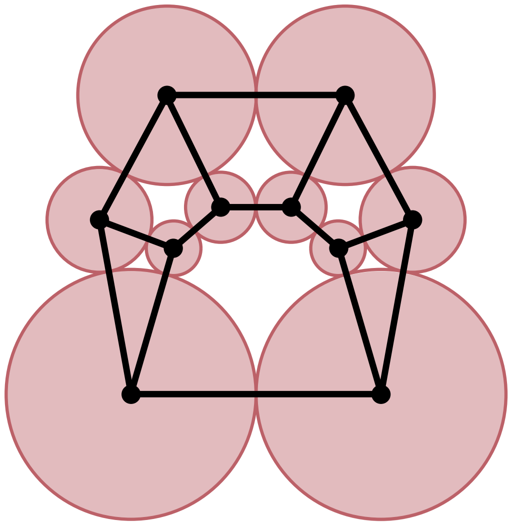
primal dual circle packing $\mathcal{P}$ of G



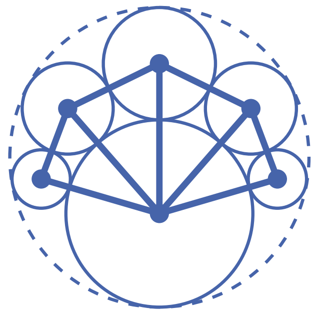
Primal Dual Circle Packings

circle contact representations of ...

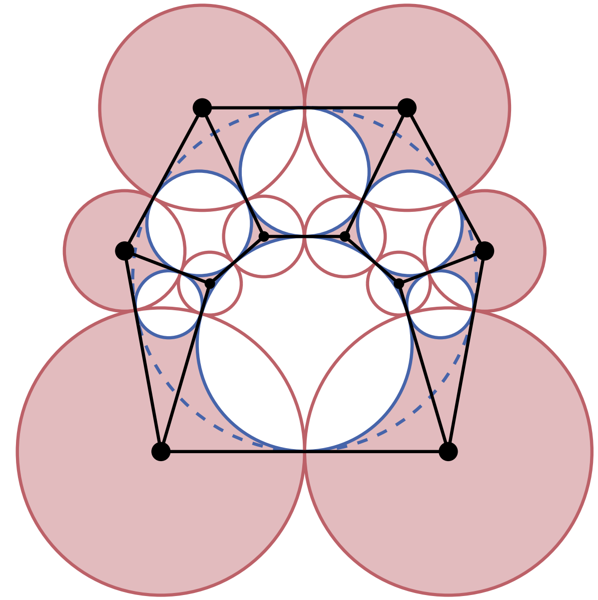
(**primal**) graph G



dual graph of G

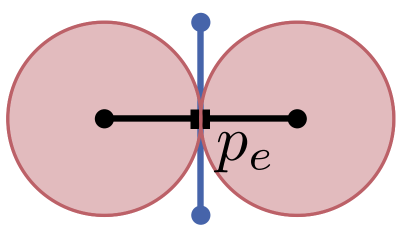


primal dual circle packing $\mathcal{P}$ of G



... which are **orthogonal**:

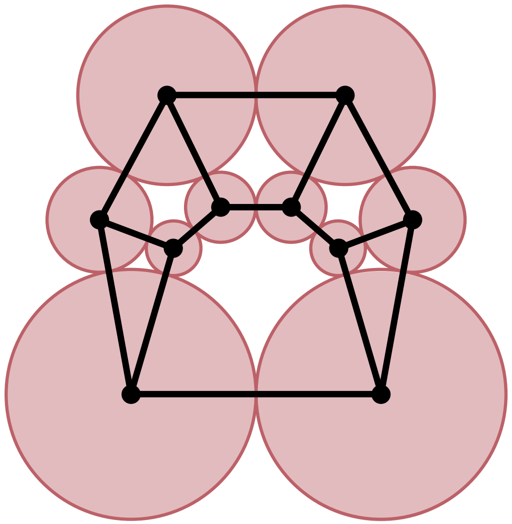
edge e **crosses**
its dual edge e^*
perpendicularly
in **edge point** p_e

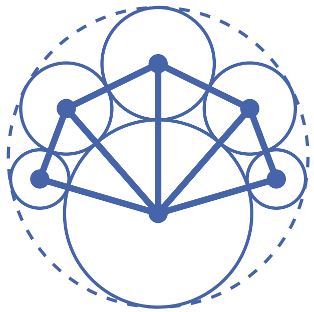


Primal Dual Circle Packings

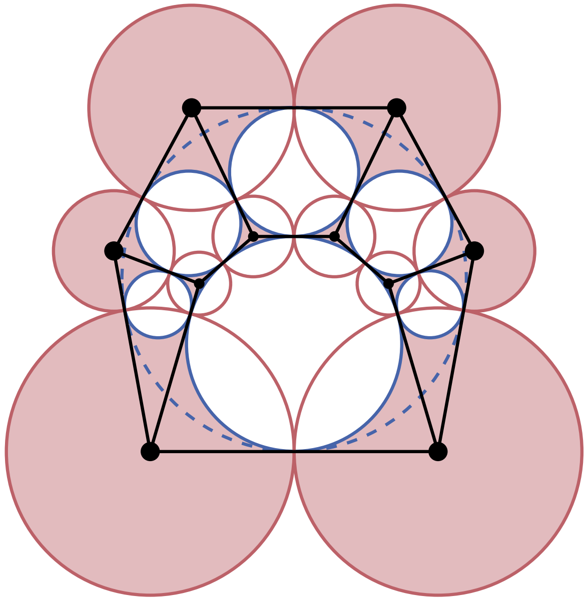
circle contact representations of ...

(**primal**) graph G **dual** graph of G



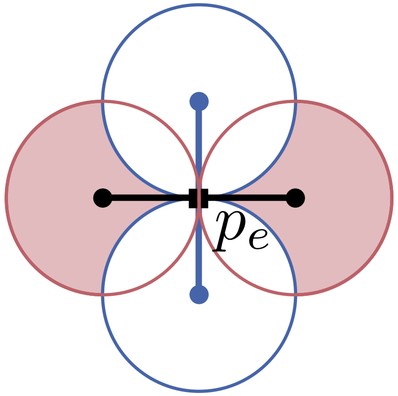


primal dual circle packing $\mathcal{P}$ of G



... which are **orthogonal**:

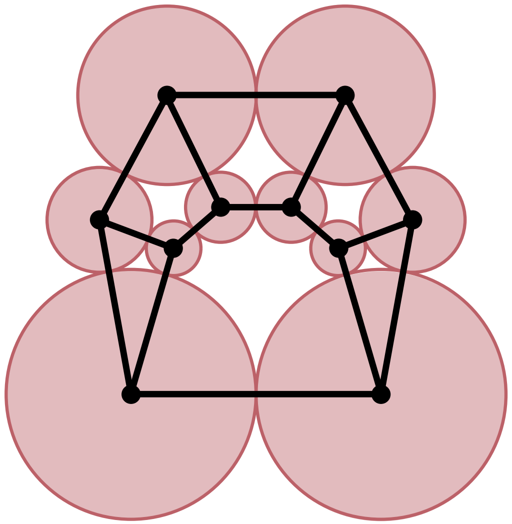
edge e **crosses**
its dual edge e^*
perpendicularly
in **edge point** p_e



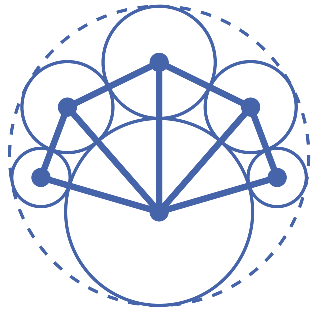
Primal Dual Circle Packings

circle contact representations of ...

(**primal**) graph G

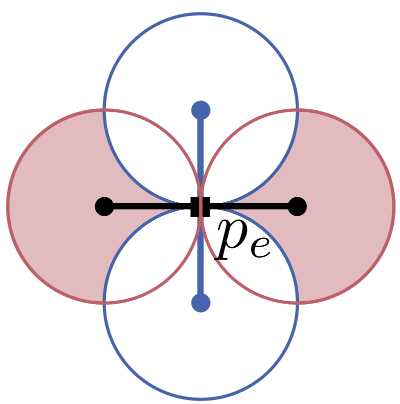


dual graph of G



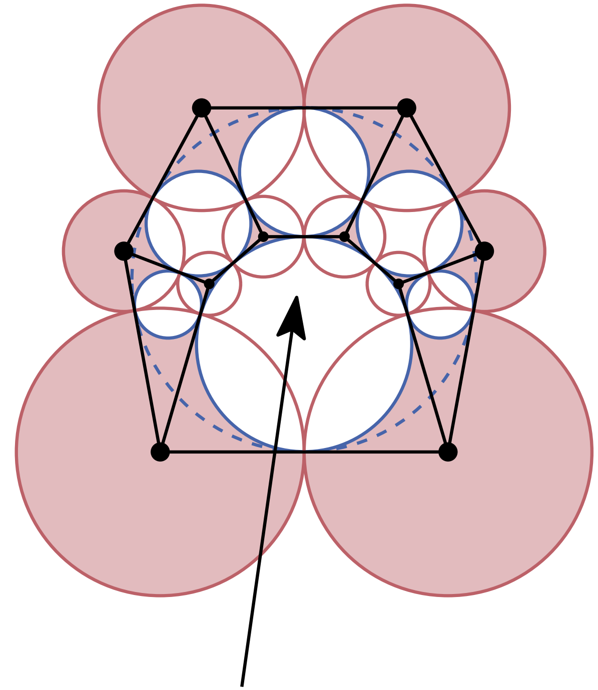
... which are **orthogonal**:

edge e **crosses**
its dual edge e^*
perpendicularly
in **edge point** p_e



$\Rightarrow$

primal dual circle packing $\mathcal{P}$ of G

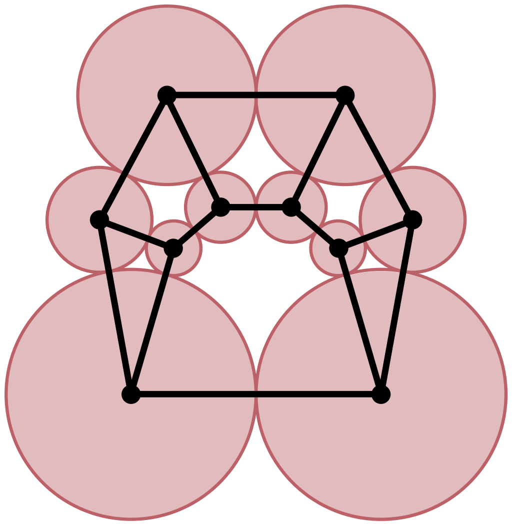


face circle = inscribed circle of face

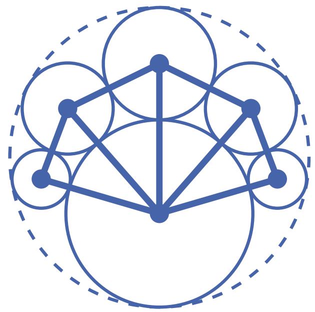
Primal Dual Circle Packings

circle contact representations of ...

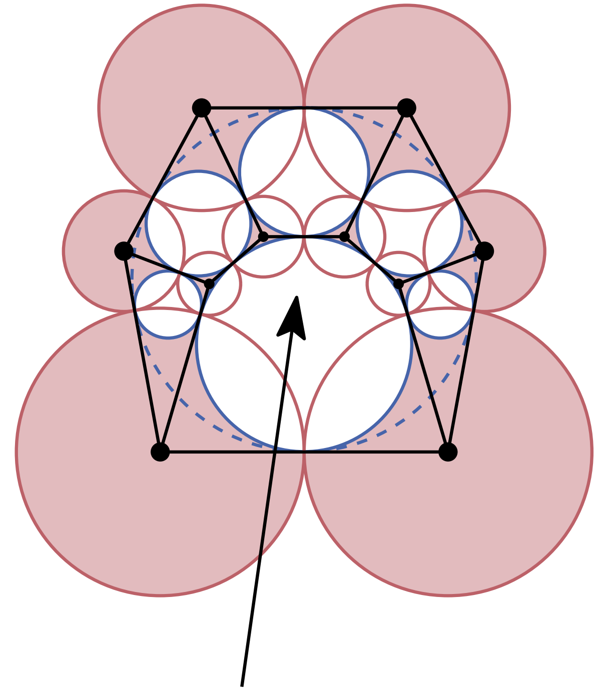
(**primal**) graph G



dual graph of G

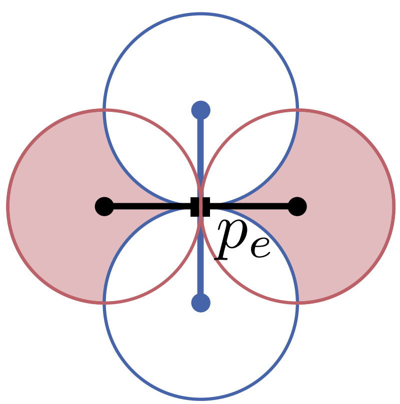


primal dual circle packing $\mathcal{P}$ of G



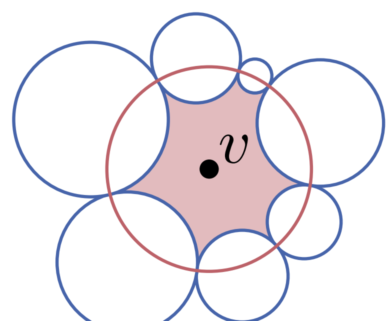
... which are **orthogonal**:

edge e **crosses**
its dual edge e^*
perpendicularly
in **edge point** p_e



$\Rightarrow$ **face circle** $=$ inscribed circle of face

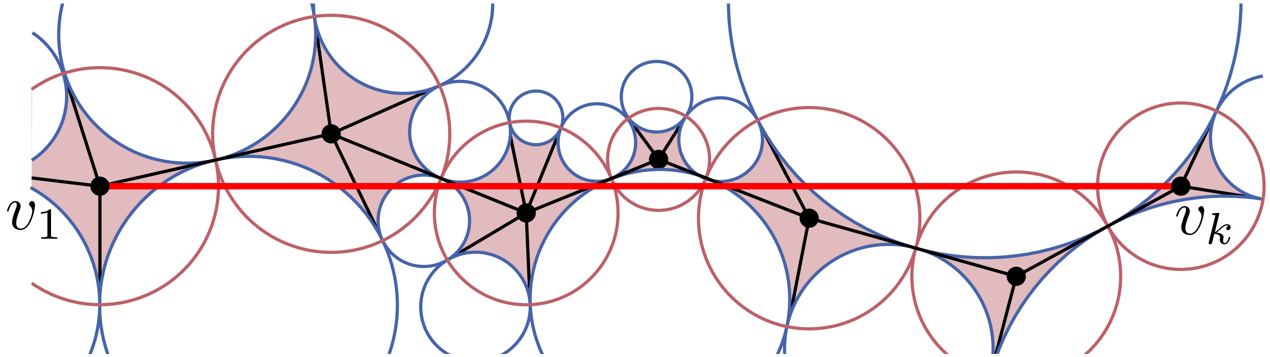
$\Rightarrow$ vertex v has 'star-shaped' **Region** R_v



v can see all of R_v!

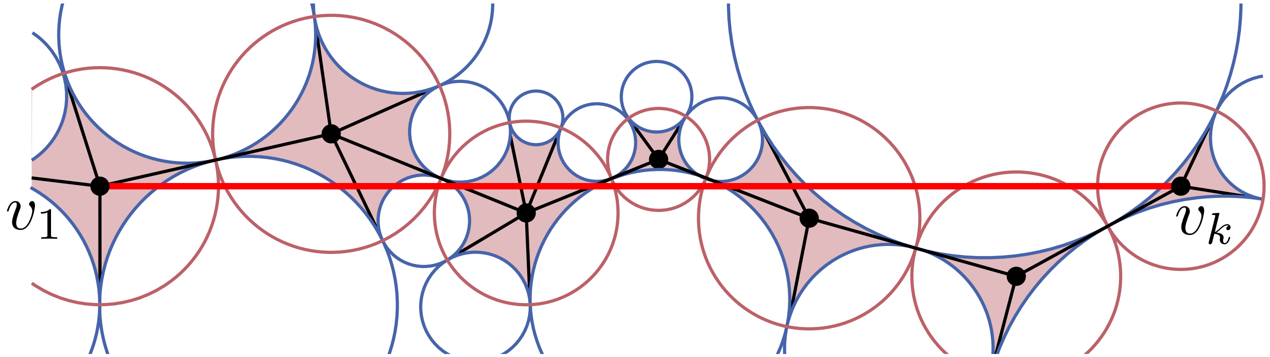
Constructing Strongly Monotone Paths

Consider vertices v_1, v_k where w.l.o.g. $s = \overline{v_1 v_k}$ horizontal.



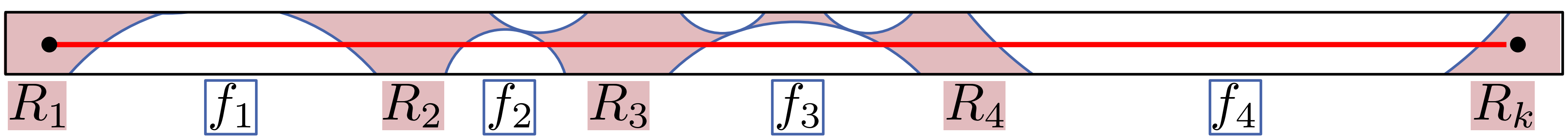
Constructing Strongly Monotone Paths

Consider vertices v_1, v_k where w.l.o.g. $s = \overline{v_1 v_k}$ horizontal.



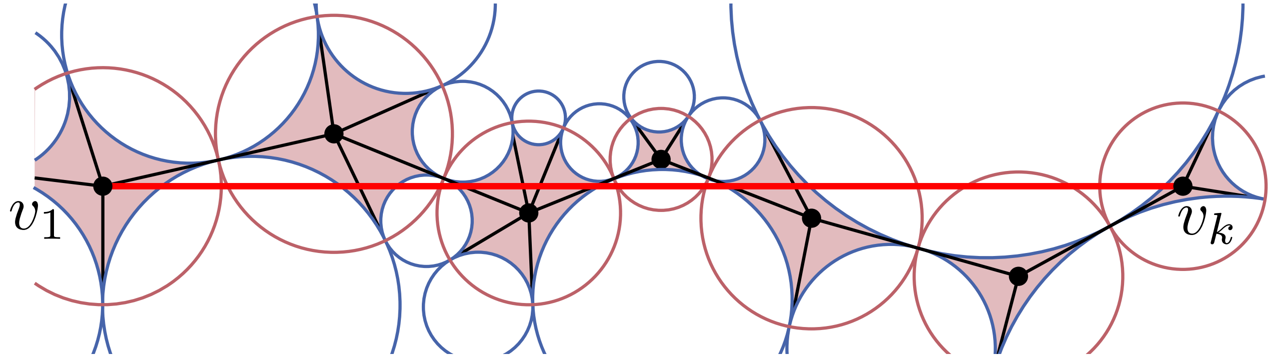
General position: s does not pass through circle centers or edge points.

$\Rightarrow$ s intersects **alternating sequence** of vertex regions and face circles:



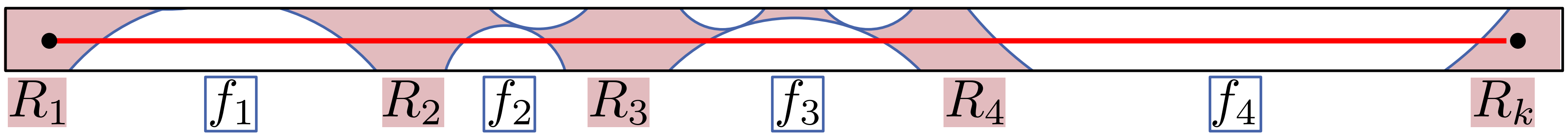
Constructing Strongly Monotone Paths

Consider vertices v_1, v_k where w.l.o.g. $s = \overline{v_1 v_k}$ horizontal.



General position: s does not pass through circle centers or edge points.

$\Rightarrow$ s intersects **alternating sequence** of vertex regions and face circles:

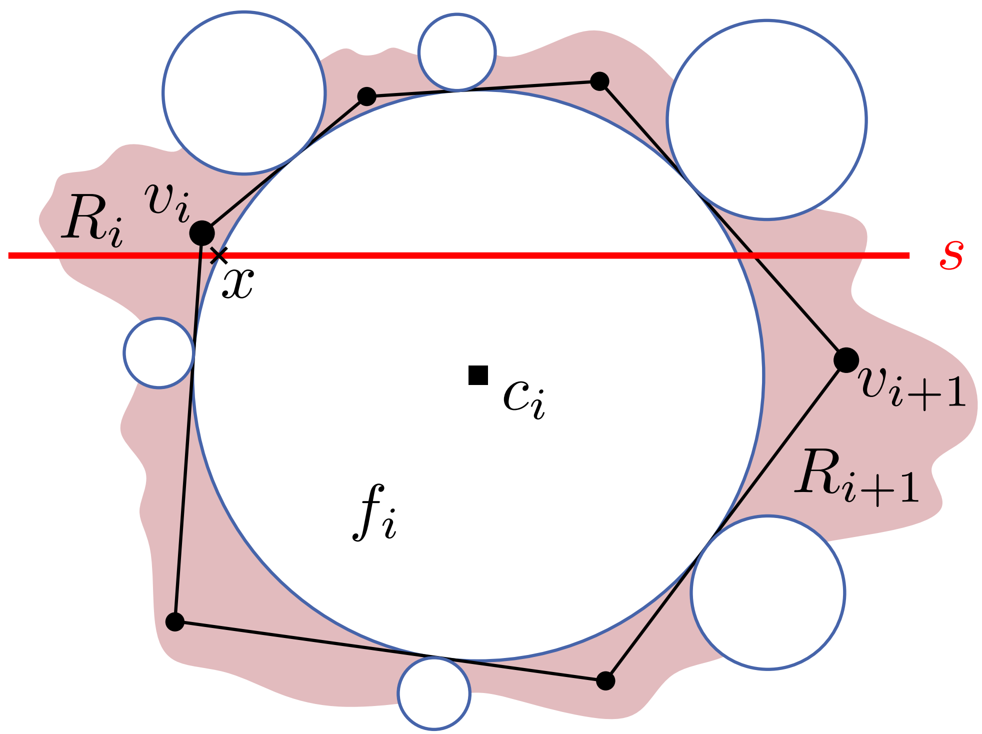


Each region R_i has some vertex v_i in its center.

Idea: Find path P_i from v_i of R_i to v_{i+1} of R_{i+1}, $1 \leq i < k$.

Concatenation yields strongly monotone (v_1, v_k)-path!

Walking around a Face Circle

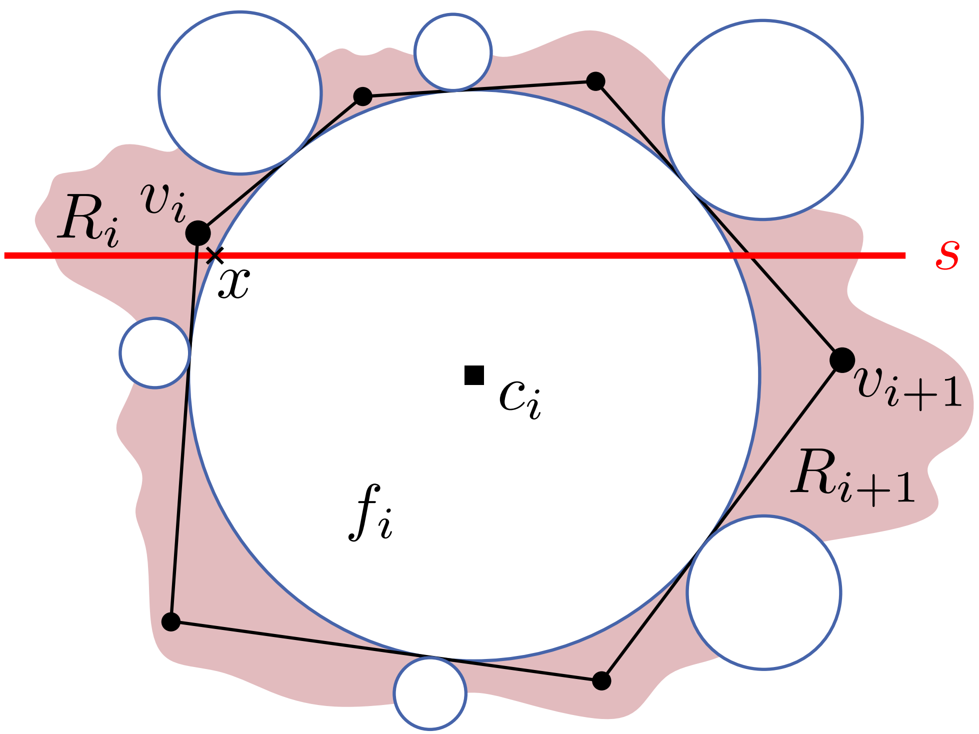


construction of $P_i = (e_1, \ldots, e_r)$

Walking around a Face Circle

Circle f_i is **inscribed circle of inner face**

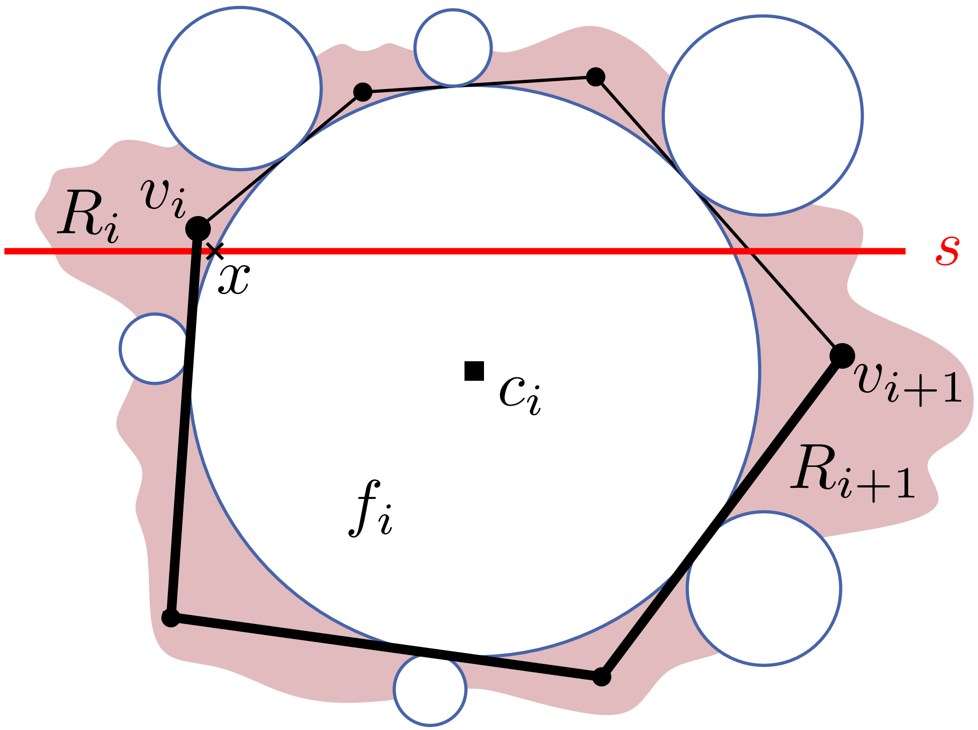
$\Rightarrow$ **two paths** leading from v_i to v_{i+1}



construction of $P_i = (e_1, \ldots, e_r)$

Walking around a Face Circle

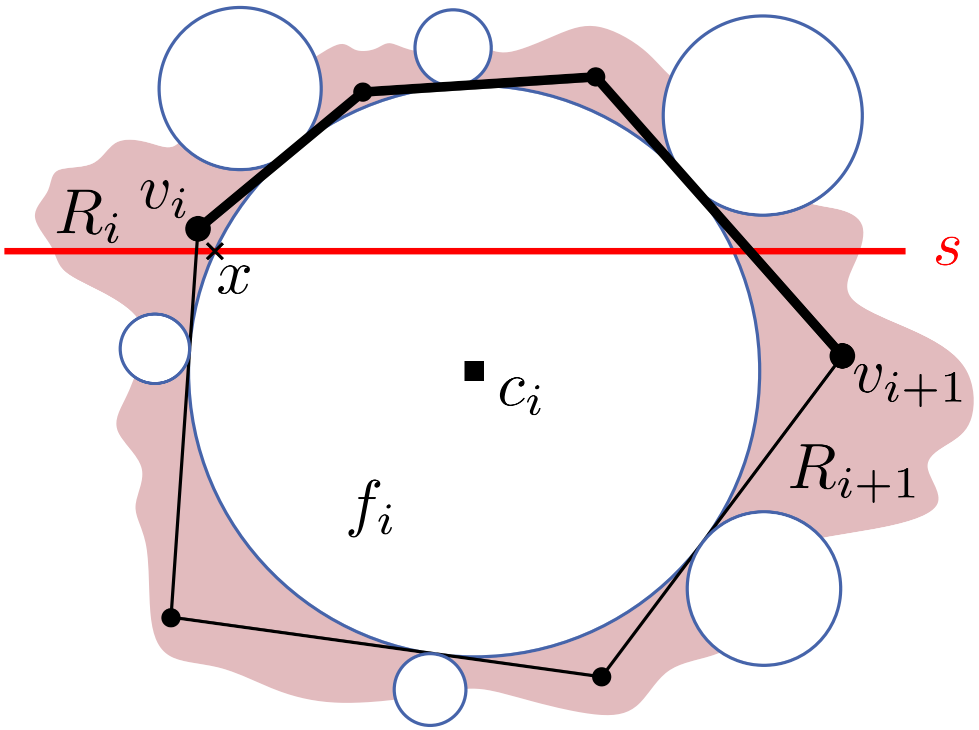
Circle f_i is **inscribed circle of inner face**
$\Rightarrow$ **two paths** leading from v_i to v_{i+1}



construction of $P_i = (e_1, \ldots, e_r)$

Walking around a Face Circle

Circle f_i is **inscribed circle of inner face**
$\Rightarrow$ **two paths** leading from v_i to v_{i+1}

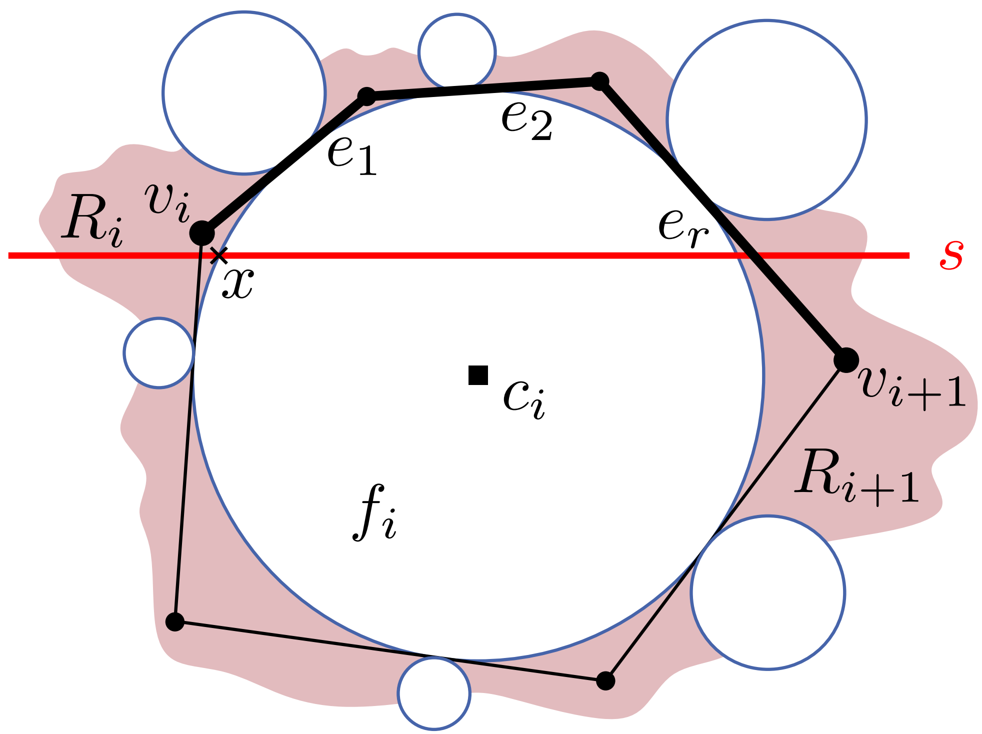


construction of $P_i = (e_1, \ldots, e_r)$

Walking around a Face Circle

Circle f_i is **inscribed circle of inner face**
$\Rightarrow$ **two paths** leading from v_i to v_{i+1}

If c_i is below s, pick the upper path.
Otherwise the lower path.



construction of $P_i = (e_1, \ldots, e_r)$

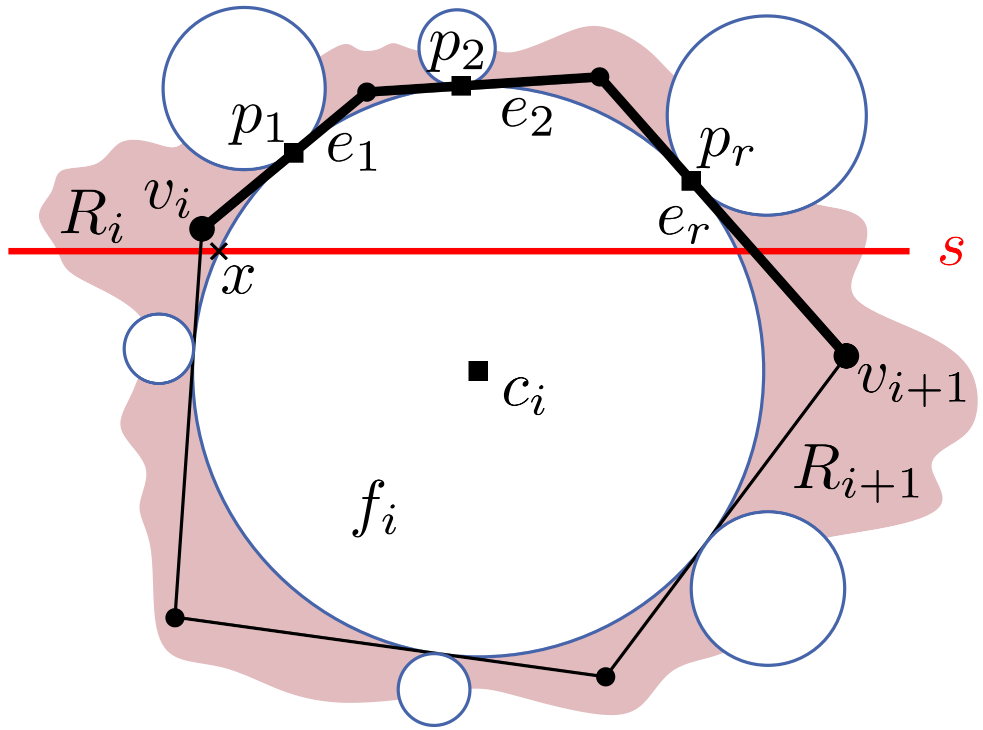
Walking around a Face Circle

Circle f_i is **inscribed circle of inner face**
$\Rightarrow$ **two paths** leading from v_i to v_{i+1}

If c_i is below s, pick the upper path.
Otherwise the lower path.

$e_1, \ldots, e_r$ tangent to f_i.

Edge points $p_1, \ldots, p_r$ **are above** s:



construction of $P_i = (e_1, \ldots, e_r)$

Walking around a Face Circle

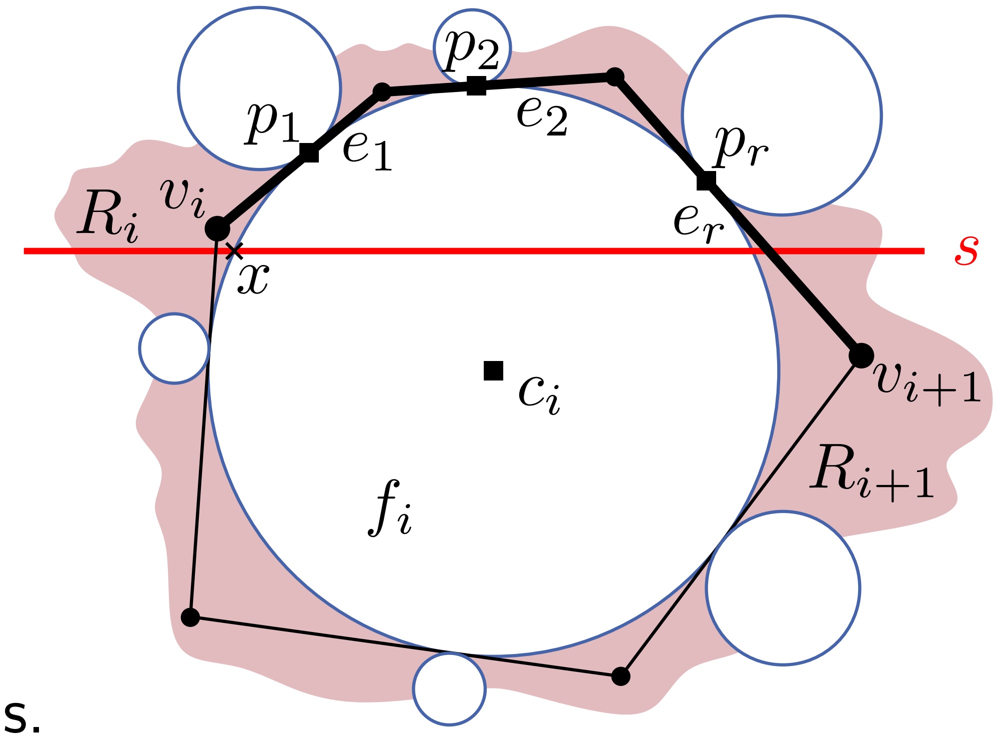
Circle f_i is **inscribed circle of inner face**
$\Rightarrow$ **two paths** leading from v_i to v_{i+1}

If c_i is below s, pick the upper path.
Otherwise the lower path.

$e_1, \ldots, e_r$ tangent to f_i.

Edge points $p_1, \ldots, p_r$ **are above** s:

- Edge point of e_1 is above s
  $\Leftarrow$ v_i sees x and e_1 points upwards.



construction of $P_i = (e_1, \ldots, e_r)$

Walking around a Face Circle

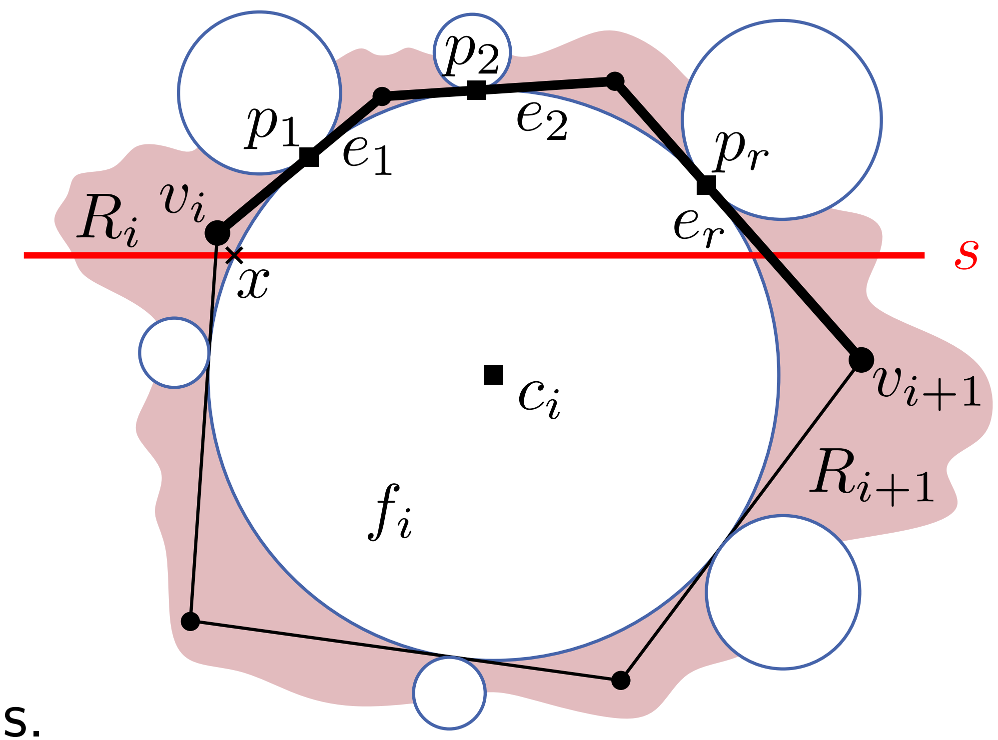
Circle f_i is **inscribed circle of inner face**
$\Rightarrow$ **two paths** leading from v_i to v_{i+1}

If c_i is below s, pick the upper path.
Otherwise the lower path.

$e_1, \ldots, e_r$ tangent to f_i.

Edge points $p_1, \ldots, p_r$ **are above** s:

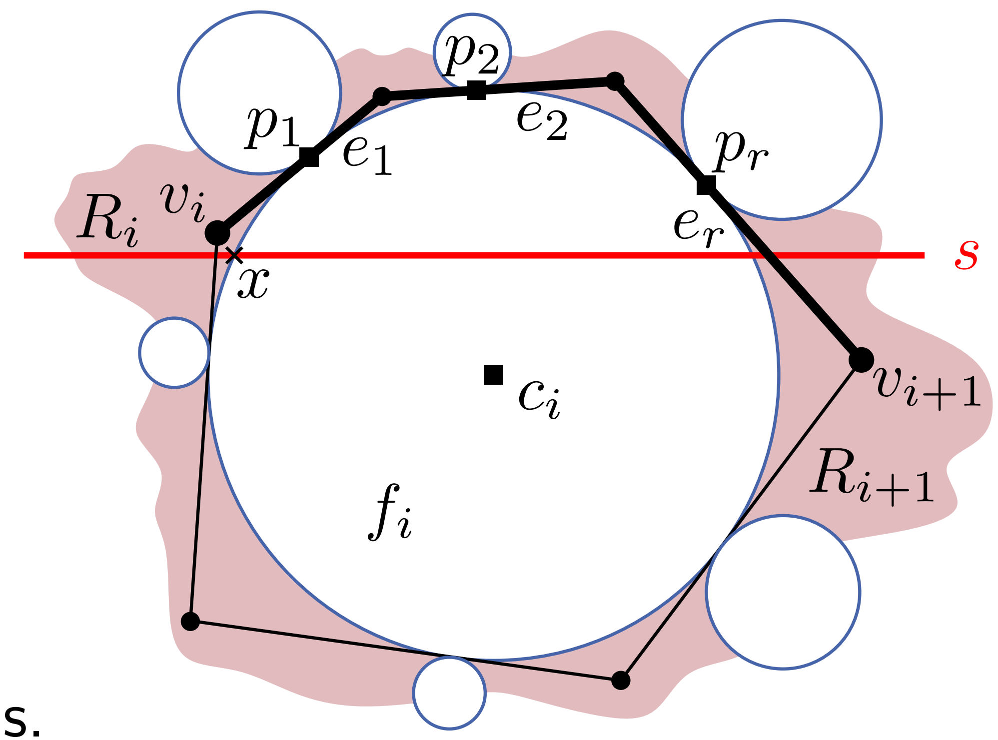
- Edge point of e_1 is above s
  $\Leftarrow$ v_i sees x and e_1 points upwards.
- Same for edge point of e_r.



construction of $P_i = (e_1, \ldots, e_r)$

Walking around a Face Circle

Circle f_i is **inscribed circle of inner face**
$\Rightarrow$ **two paths** leading from v_i to v_{i+1}



construction of $P_i = (e_1, \ldots, e_r)$

If c_i is below s, pick the upper path.
Otherwise the lower path.

$e_1, \ldots, e_r$ tangent to f_i.

Edge points $p_1, \ldots, p_r$ **are above** s:

- Edge point of e_1 is above s
  $\Leftarrow$ v_i sees x and e_1 points upwards.
- Same for edge point of e_r.

$\Rightarrow$ all **edge points** $p_1, \ldots p_r$ **above** c_i.

$p_1, \ldots, p_r$ **in this order** around f_i.

Walking around a Face Circle

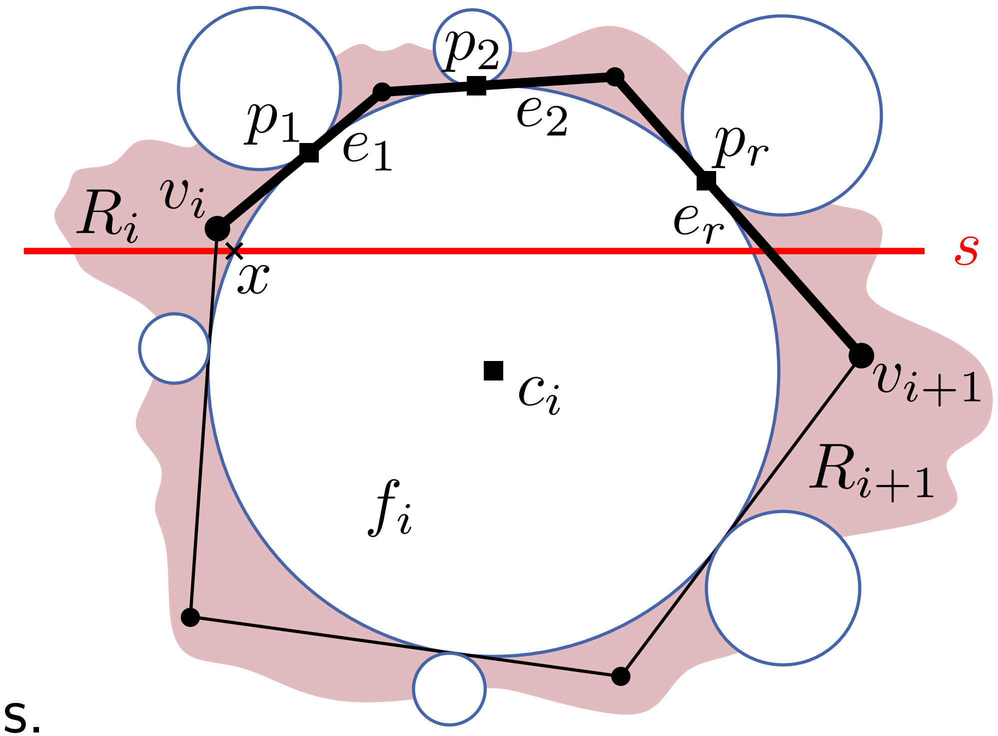
Circle f_i is **inscribed circle of inner face**
$\Rightarrow$ **two paths** leading from v_i to v_{i+1}

If c_i is below s, pick the upper path.
Otherwise the lower path.

$e_1, \ldots, e_r$ tangent to f_i.

Edge points $p_1, \ldots, p_r$ **are above** s:

- Edge point of e_1 is above s
  $\Leftarrow$ v_i sees x and e_1 points upwards.
- Same for edge point of e_r.



construction of $P_i = (e_1, \ldots, e_r)$

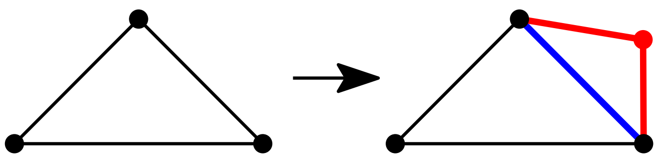
$\Rightarrow$ all **edge points** $p_1, \ldots p_r$ **above** c_i.
$p_1, \ldots, p_r$ **in this order** around f_i.

$\Rightarrow$ x-coordinates increase

$\Rightarrow$ strongly monotone

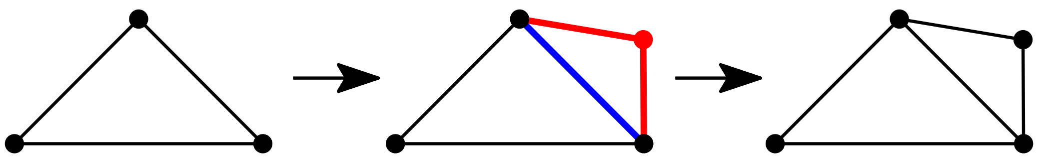
2-Trees

Graph generated from a K_3 by **stacking** vertices on edges.



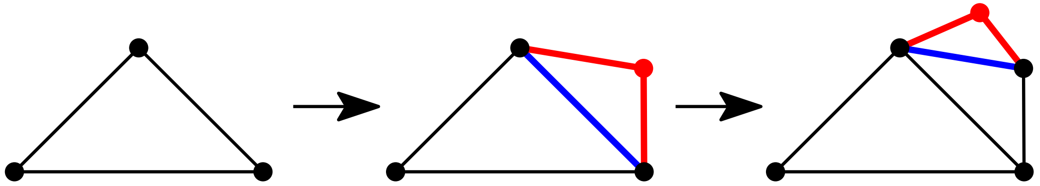
2-Trees

Graph generated from a K_3 by **stacking** vertices on edges.



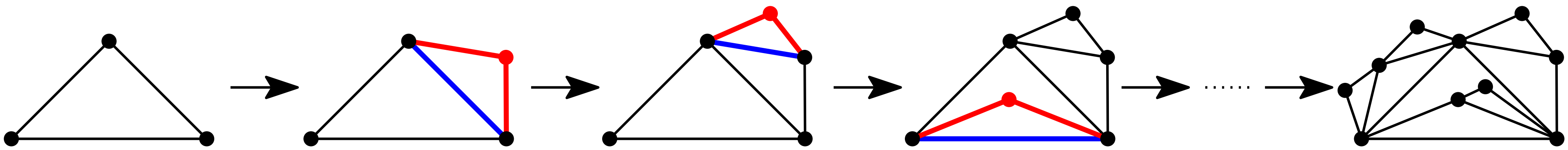
2-Trees

Graph generated from a K_3 by **stacking** vertices on edges.



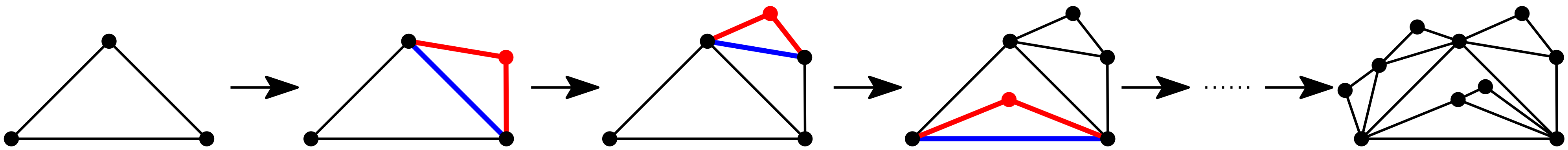
2-Trees

Graph generated from a K_3 by **stacking** vertices on edges.



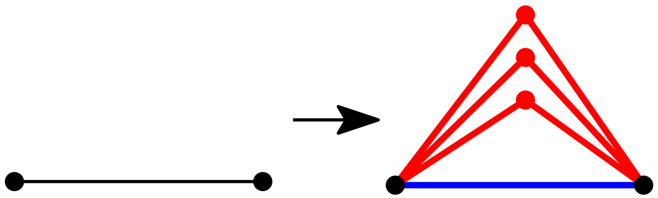
2-Trees

Graph generated from a K_3 by **stacking** vertices on edges.



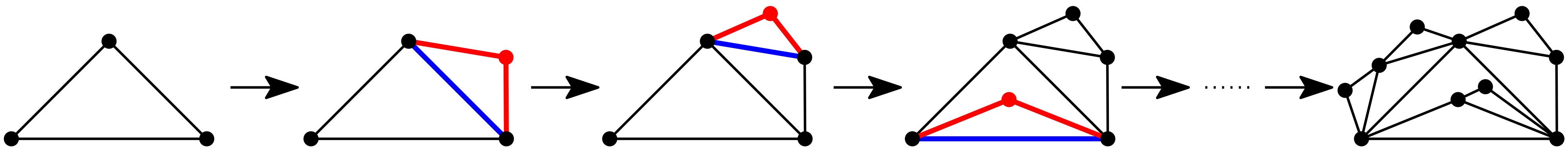
Equivalently:

Graph generated from an edge by **multistacking once per edge**.



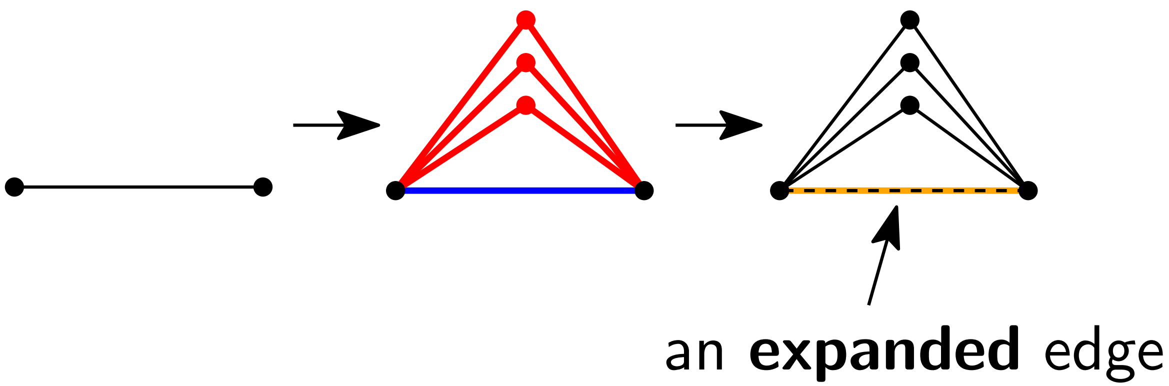
2-Trees

Graph generated from a K_3 by **stacking** vertices on edges.



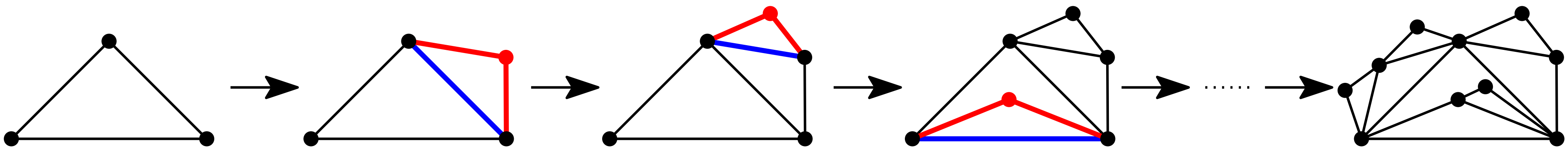
Equivalently:

Graph generated from an edge by **multistacking once per edge**.



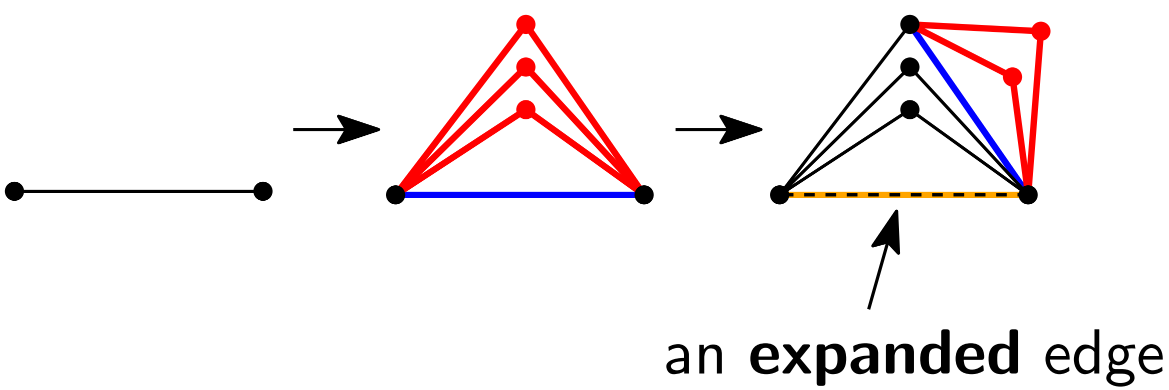
2-Trees

Graph generated from a K_3 by **stacking** vertices on edges.



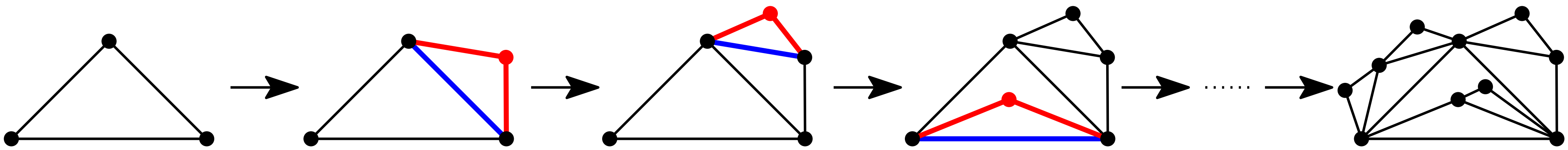
Equivalently:

Graph generated from an edge by **multistacking once per edge**.



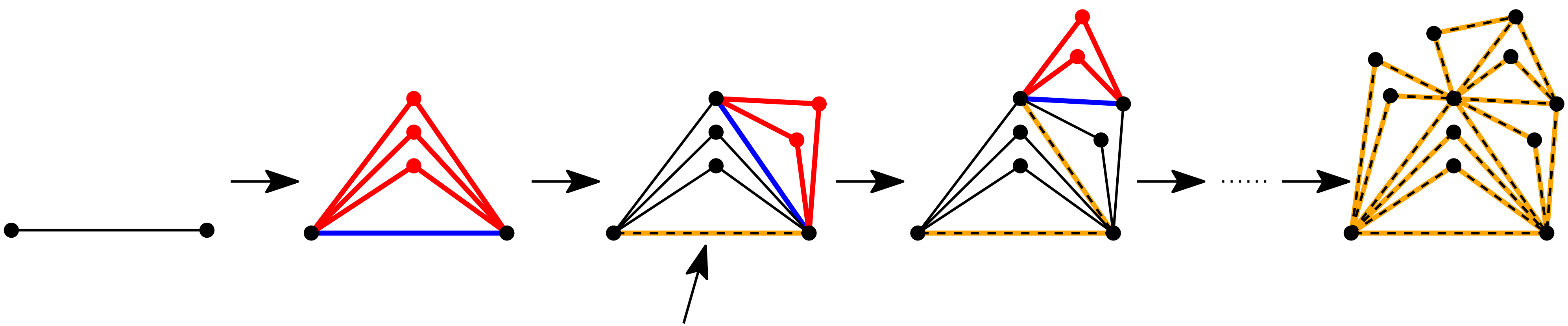
2-Trees

Graph generated from a K_3 by **stacking** vertices on edges.



Equivalently:

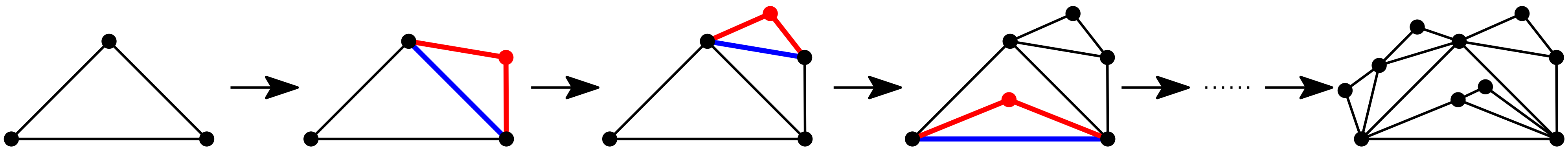
Graph generated from an edge by **multistacking once per edge**.



an **expanded** edge

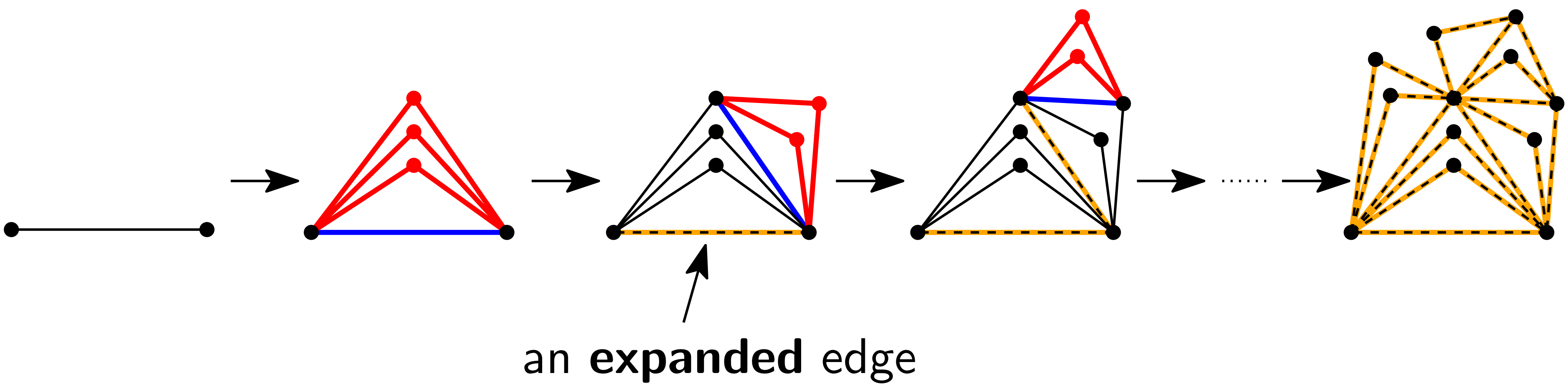
2-Trees

Graph generated from a K_3 by **stacking** vertices on edges.



Equivalently:

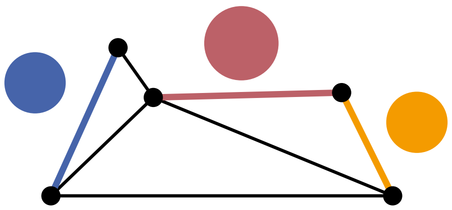
Graph generated from an edge by **multistacking once per edge**.



Theorem: Every 2-tree has a strongly monotone drawing.

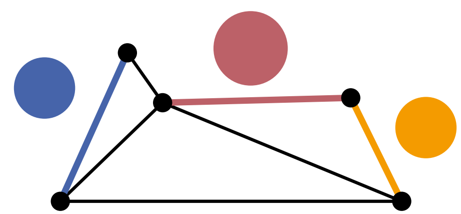
Definitions

A **drawing with bubbles** (**DWB**) is a straight-line drawing of $G = (V, E)$ where every edge in some $E' \subseteq E$ is associated with a **bubble**.

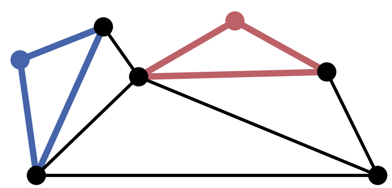


Definitions

A **drawing with bubbles** (**DWB**) is a straight-line drawing of $G = (V, E)$ where every edge in some $E' \subseteq E$ is associated with a **bubble**.

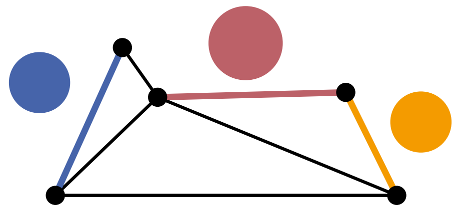


An **extension** of a DWB is a drawing obtained by stacking one vertex into each bubble of some $E'' \subseteq E'$.

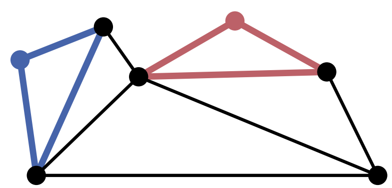


Definitions

A **drawing with bubbles** (**DWB**) is a straight-line drawing of $G = (V, E)$ where every edge in some $E' \subseteq E$ is associated with a **bubble**.



An **extension** of a DWB is a drawing obtained by stacking one vertex into each bubble of some $E'' \subseteq E'$.



A DWB is **strongly monotone** if every extension is strongly monotone.

i.e.
- the drawing without bubbles is strongly monotone
- stacking into bubbles maintains strong monotonicty

Drawing a 2-Tree

Idea: Draw according to **multistacking** sequence.

Invariant: Before / after each multistack ...

- every non-expanded edge has a bubble
- the DWB is strongly monotone
- stacking gives obtuse angles

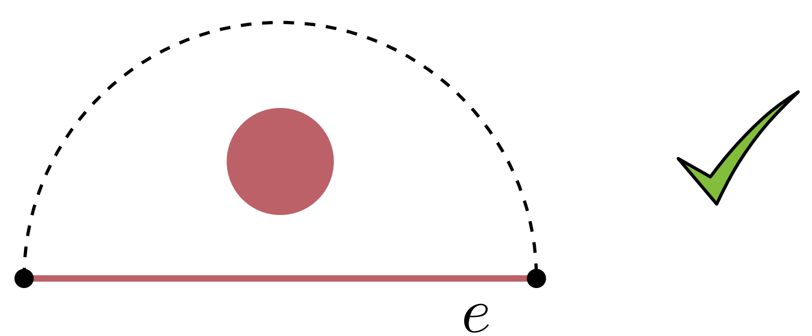
Drawing a 2-Tree

Idea: Draw according to **multistacking** sequence.

Invariant: Before / after each multistack ...

- every non-expanded edge has a bubble
- the DWB is strongly monotone
- stacking gives obtuse angles

Base case:



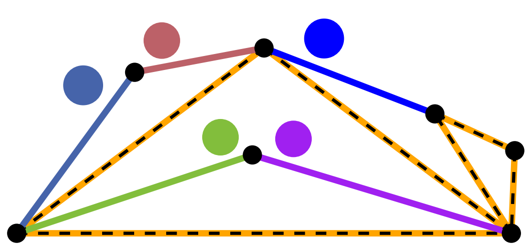
Drawing a 2-Tree

Idea: Draw according to **multistacking** sequence.

Invariant: Before / after each multistack ...

- every non-expanded edge has a bubble
- the DWB is strongly monotone
- stacking gives obtuse angles

Induction step:



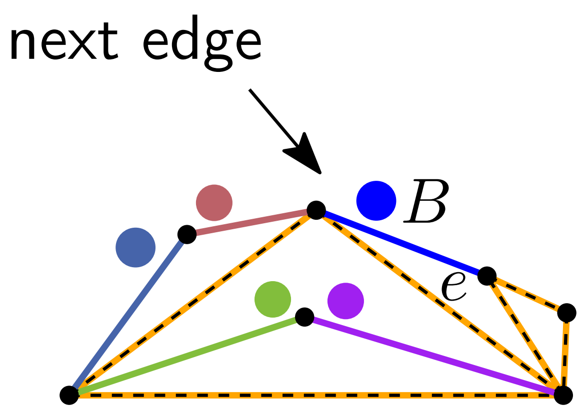
Drawing a 2-Tree

Idea: Draw according to **multistacking** sequence.

Invariant: Before / after each multistack ...

- every non-expanded edge has a bubble
- the DWB is strongly monotone
- stacking gives obtuse angles

Induction step:



Drawing a 2-Tree

Idea: Draw according to **multistacking** sequence.

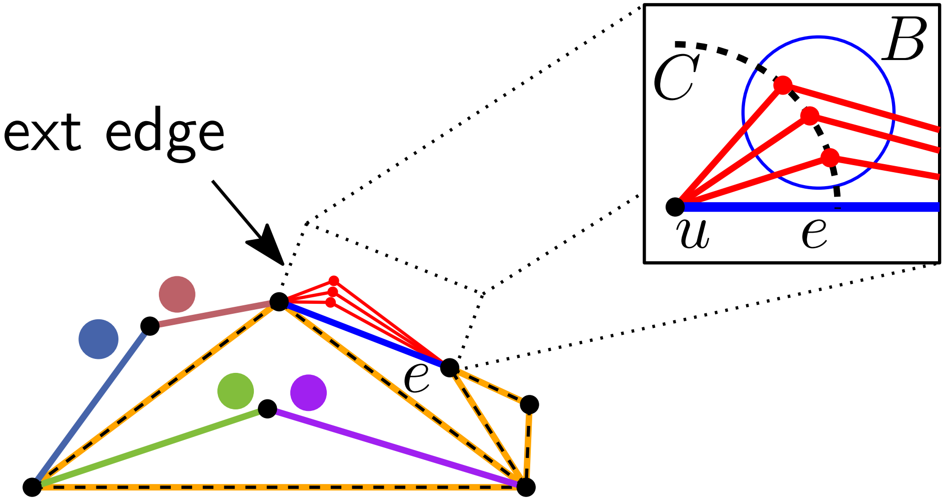
Invariant: Before / after each multistack ...

- every non-expanded edge has a bubble
- the DWB is strongly monotone
- stacking gives obtuse angles

Induction step:

place vertices on circular arc C centered at u

next edge



Drawing a 2-Tree

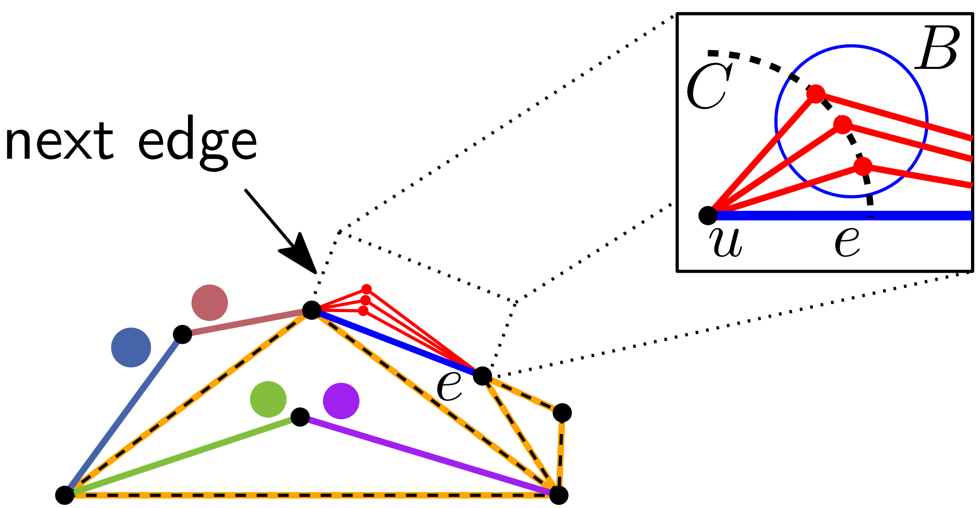
Idea: Draw according to **multistacking** sequence.

Invariant: Before / after each multistack ...

- every non-expanded edge has a bubble
- the DWB is strongly monotone
- stacking gives obtuse angles

Induction step:

place vertices on circular arc C centered at u



New DWB is strongly monotone:

- paths between 2 new vertices
- paths between new vertex and old vertex / bubble

Drawing a 2-Tree

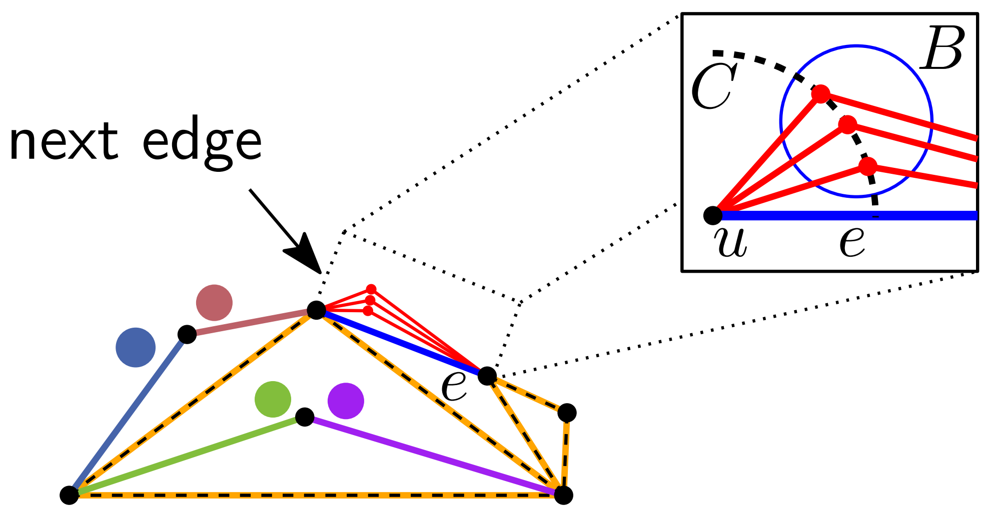
Idea: Draw according to **multistacking** sequence.

Invariant: Before / after each multistack ...

- every non-expanded edge has a bubble
- the DWB is strongly monotone
- stacking gives obtuse angles

Induction step:

place vertices on circular arc C centered at u



New DWB is strongly monotone:

- paths between 2 new vertices ✓
- paths between new vertex and old vertex / bubble

Drawing a 2-Tree

Idea: Draw according to **multistacking** sequence.

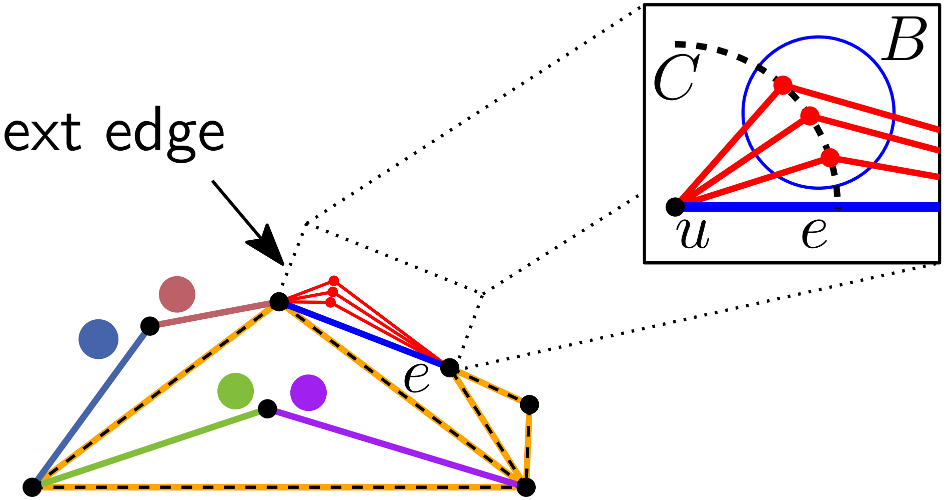
Invariant: Before / after each multistack ...

- every non-expanded edge has a bubble
- the DWB is strongly monotone
- stacking gives obtuse angles

Induction step:

place vertices on circular arc C centered at u

next edge



New DWB is strongly monotone:

- paths between 2 new vertices ✓
- paths between new vertex and old vertex / bubble ✓

Drawing a 2-Tree

Idea: Draw according to **multistacking** sequence.

Invariant: Before / after each multistack ...

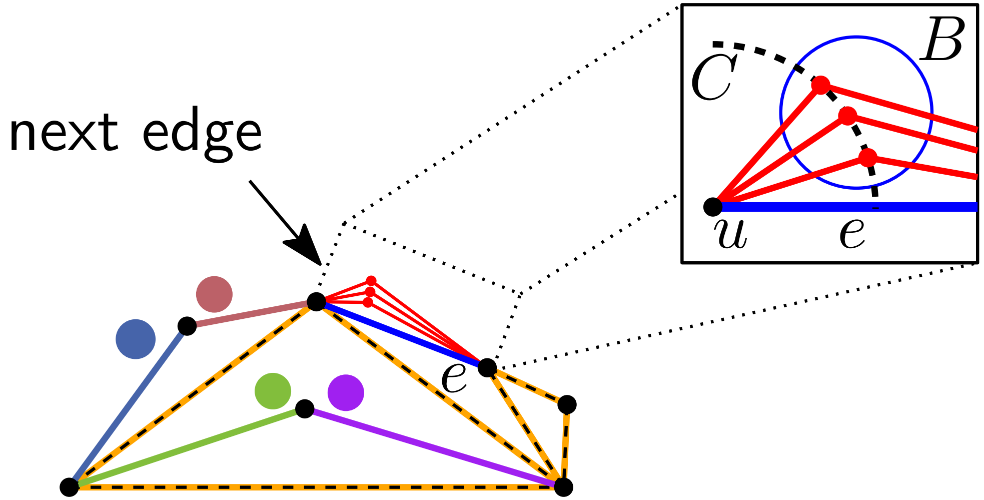
- every non-expanded edge has a bubble
- the DWB is strongly monotone
- stacking gives obtuse angles

Not done yet!

How to add bubbles for new edges?

Induction step:

place vertices on circular arc C centered at u

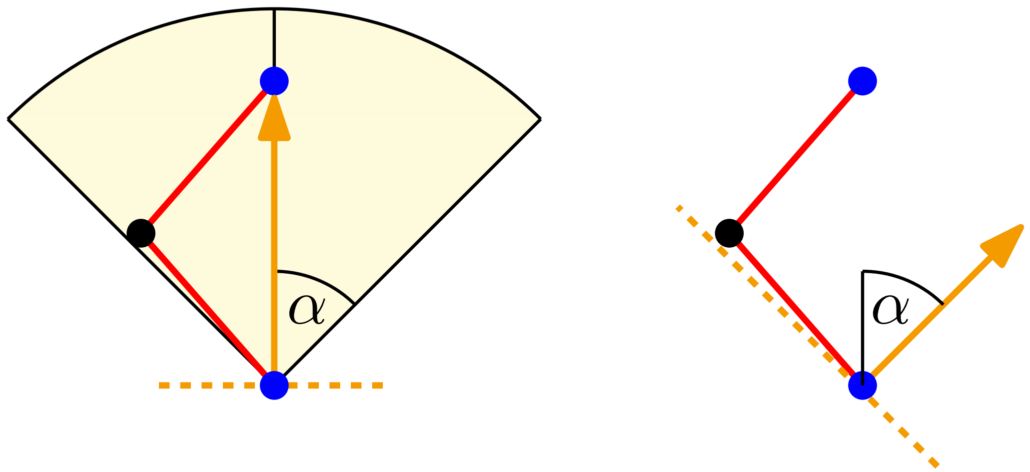


New DWB is strongly monotone:

- paths between 2 new vertices ✓
- paths between new vertex and old vertex / bubble ✓

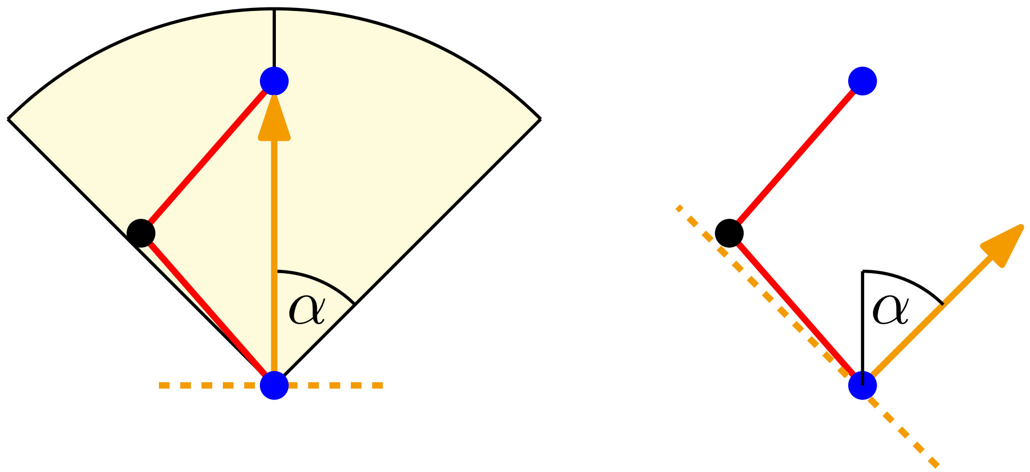
Adding Bubbles

A strongly monotone path P is α-**safe** if it remains monotone while the direction is tilted by at most α.



Adding Bubbles

A strongly monotone path P is α-**safe** if it remains monotone while the direction is tilted by at most α.

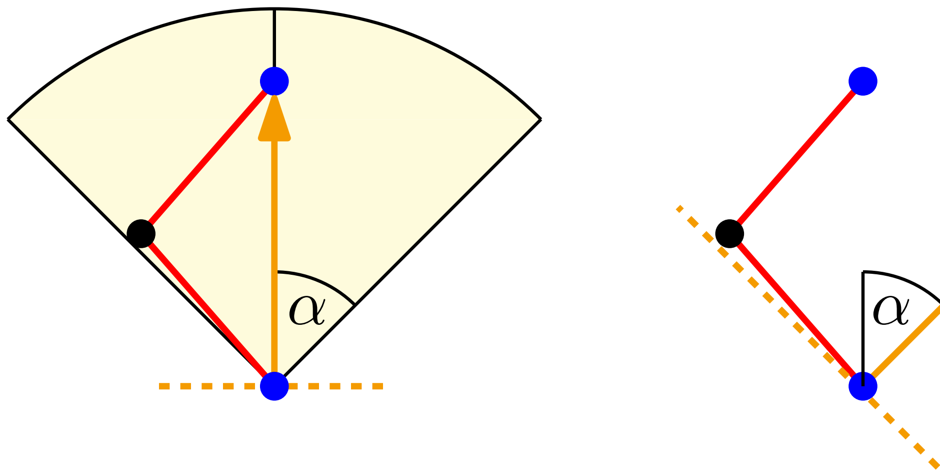


Idea:

- Find α s.t. between any vertex / bubble pair exists an α-safe strongly monotone path.

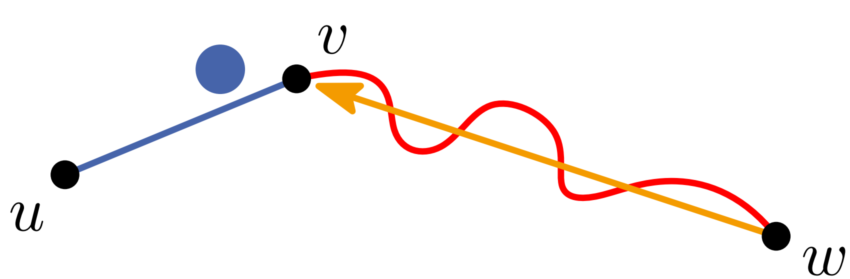
Adding Bubbles

A strongly monotone path P is α-**safe** if it remains monotone while the direction is tilted by at most α.



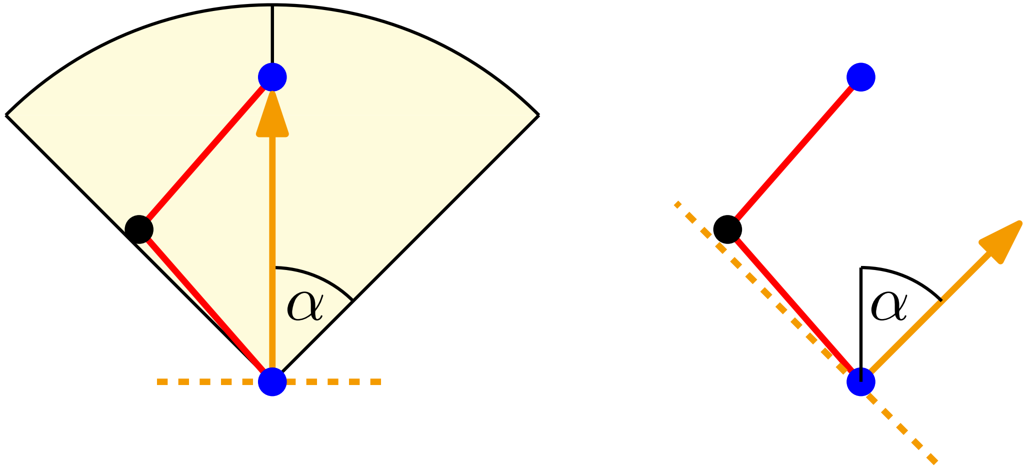
Idea:

- Find α s.t. between any vertex / bubble pair exists an α-safe strongly monotone path.
- Place bubble for $e = (u, v)$ close to v s.t. prolonging paths leading to v changes the direction by $< \alpha$.



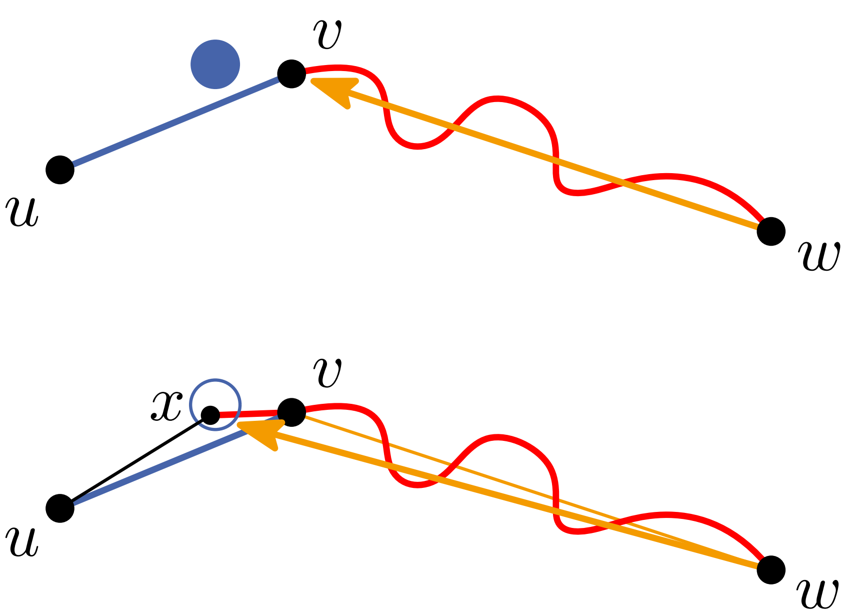
Adding Bubbles

A strongly monotone path P is α-**safe** if it remains monotone while the direction is tilted by at most α.



Idea:

- Find α s.t. between any vertex / bubble pair exists an α-safe strongly monotone path.
- Place bubble for $e = (u, v)$ close to v s.t. prolonging paths leading to v changes the direction by $< \alpha$.



Related Work & Results & Open Problems

Strongly monotone drawings:

- do not exist for every planar graph [Kindermann et al. '14]
- exist for every 2-connected outerplanar graph [Kindermann et al. '14]
- exist for every tree [Kindermann et al. '14]
  - area required can be exponential [Nöllenburg et al. '14]

- exist for every planar 3-connectd graph
- exist for every outerplanar graph
- exist for every 2-tree

} **our results**

- exist for every planar 2-connected graph? **open problem!**

Related Work & Results & Open Problems

Strongly monotone drawings:

- do not exist for every planar graph [Kindermann et al. '14]
- exist for every 2-connected outerplanar graph [Kindermann et al. '14]
- exist for every tree [Kindermann et al. '14]
  - area required can be exponential [Nöllenburg et al. '14]

- exist for every planar 3-connectd graph
- exist for every outerplanar graph
- exist for every 2-tree

} **our results**

- exist for every planar 2-connected graph? **open problem!**

Thank you!