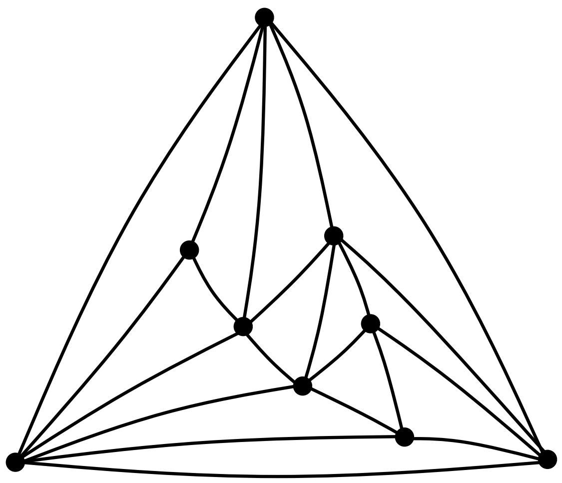
Canonical Ordering for Triangulations on the Cylinder, with Applications to Periodic Straight-line Drawings

Luca Castelli Aleardi, Olivier Devillers and Éric Fusy

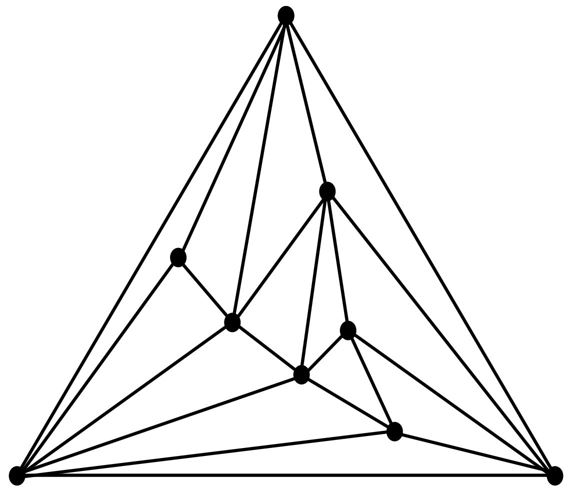
LIX, École Polytechnique, Palaiseau, France

GD'12, Redmond

Planar straight-line drawings

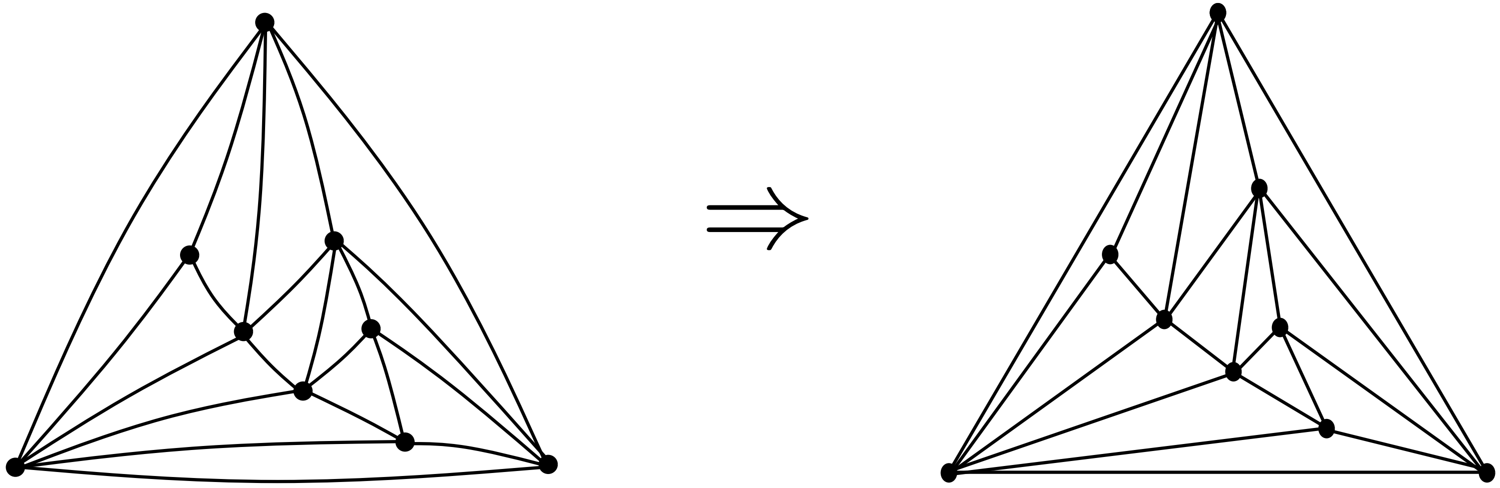


$\Rightarrow$



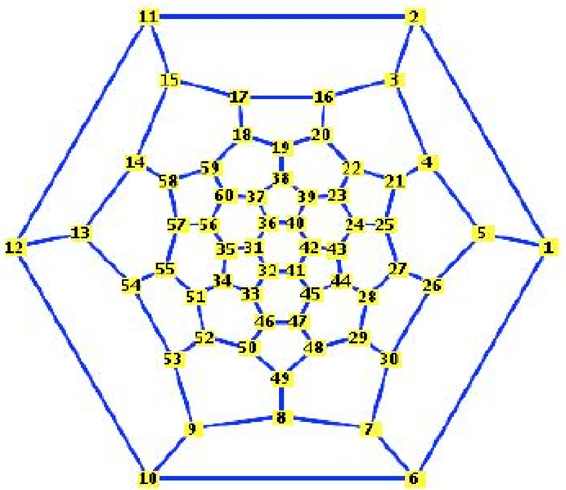
[Wagner'36]
[Fary'48]
[Stein'51]

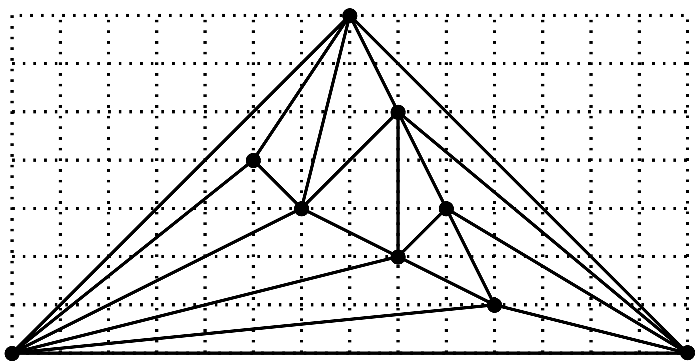
Planar straight-line drawings

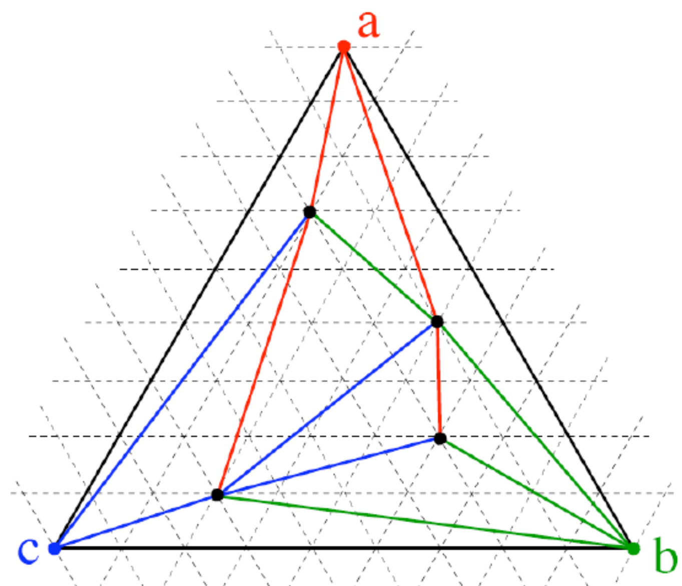


[Wagner'36]
[Fary'48]
[Stein'51]

Classical algorithms:







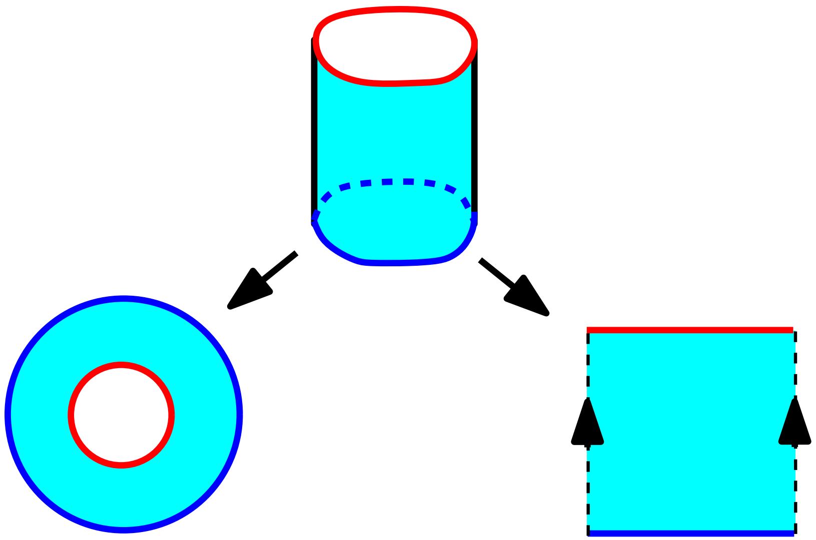
[Tutte'63] spring-embedding

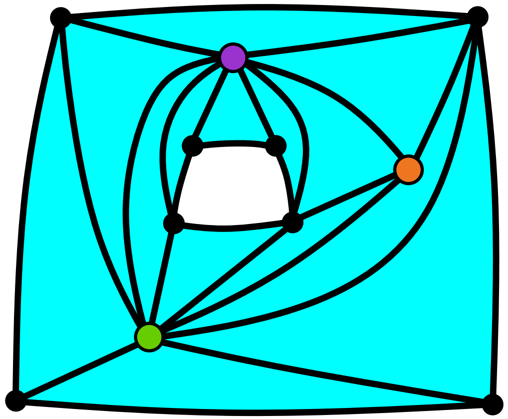
[De Fraysseix,Pach,Pollack'88] "FPP algo" incremental

[Schnyder'90] face-counting principle

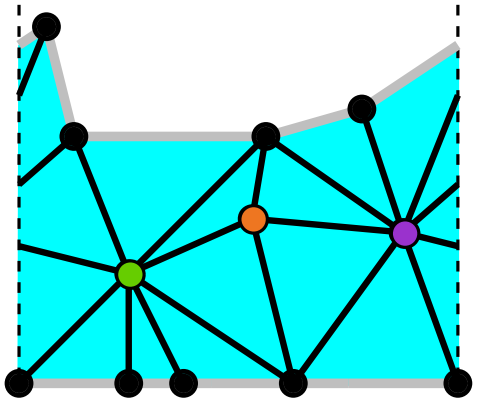
Periodic straight-line drawings

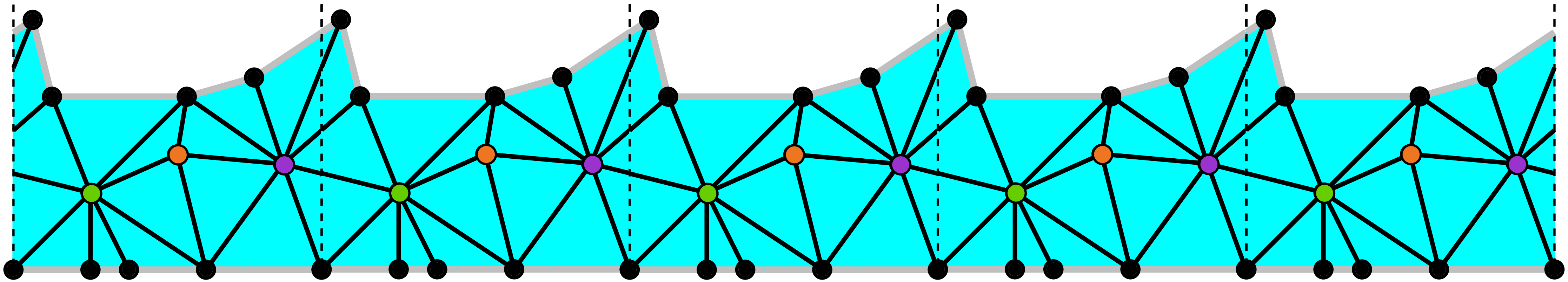
On the cylinder ($\Leftrightarrow$ annulus)





$\Rightarrow$

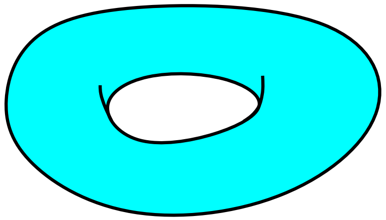




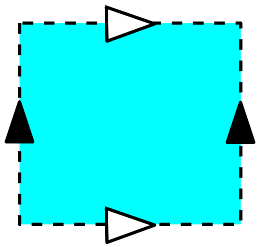
lifts to x-periodic drawing

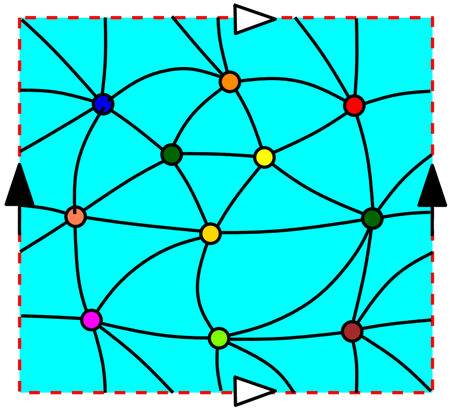
Periodic straight-line drawings

On the (flat) torus

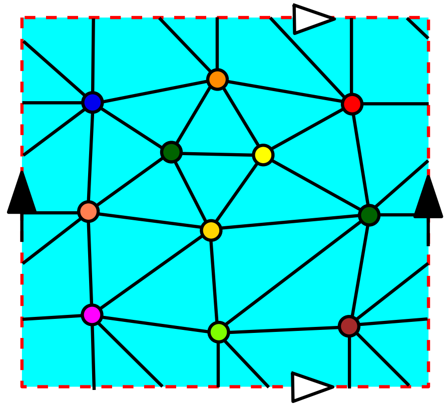


$\Updownarrow$





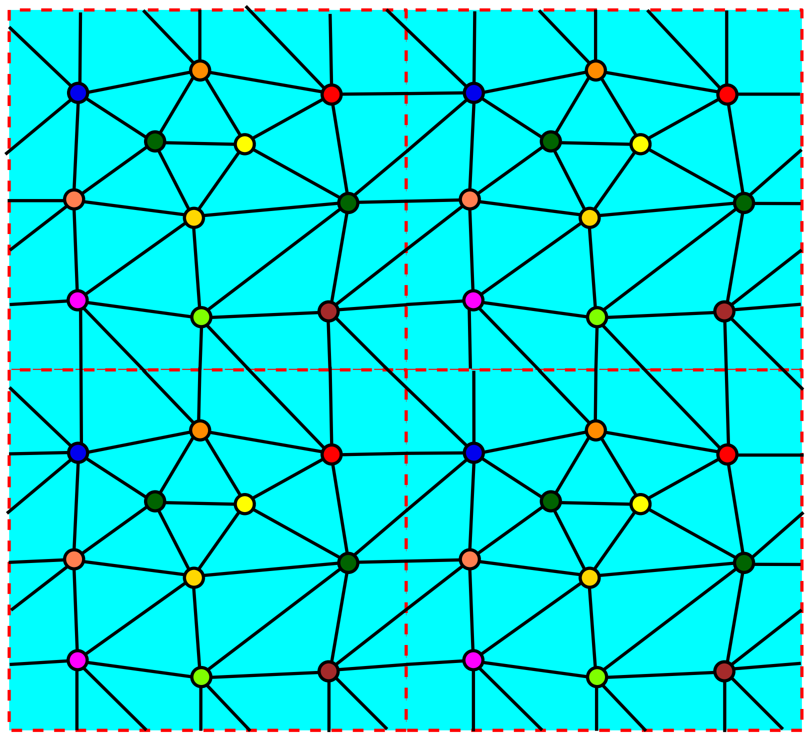
$\Rightarrow$



[Kocay et al'01]
[Gortler et al'06]
[Gonçalves-Lévêque'12]

Grid $O(n^2 \times n^2)$

lifts to
x-periodic &
y-periodic
drawing



Overview

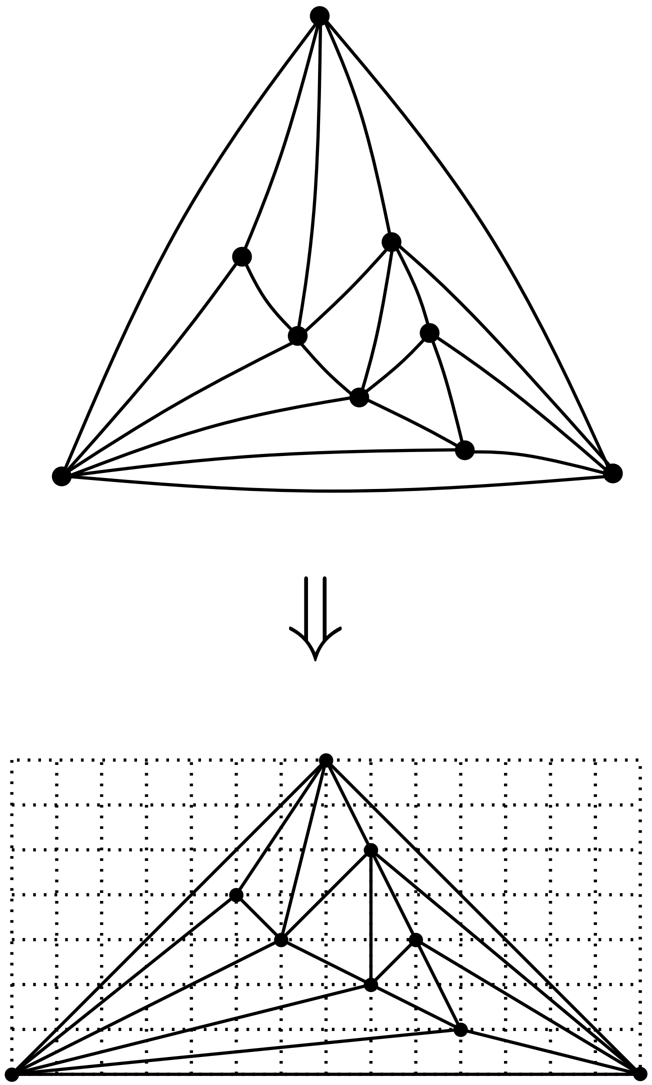
1. Recall FPP algorithm

2. Extend to the cylinder

3. Get toroidal drawings

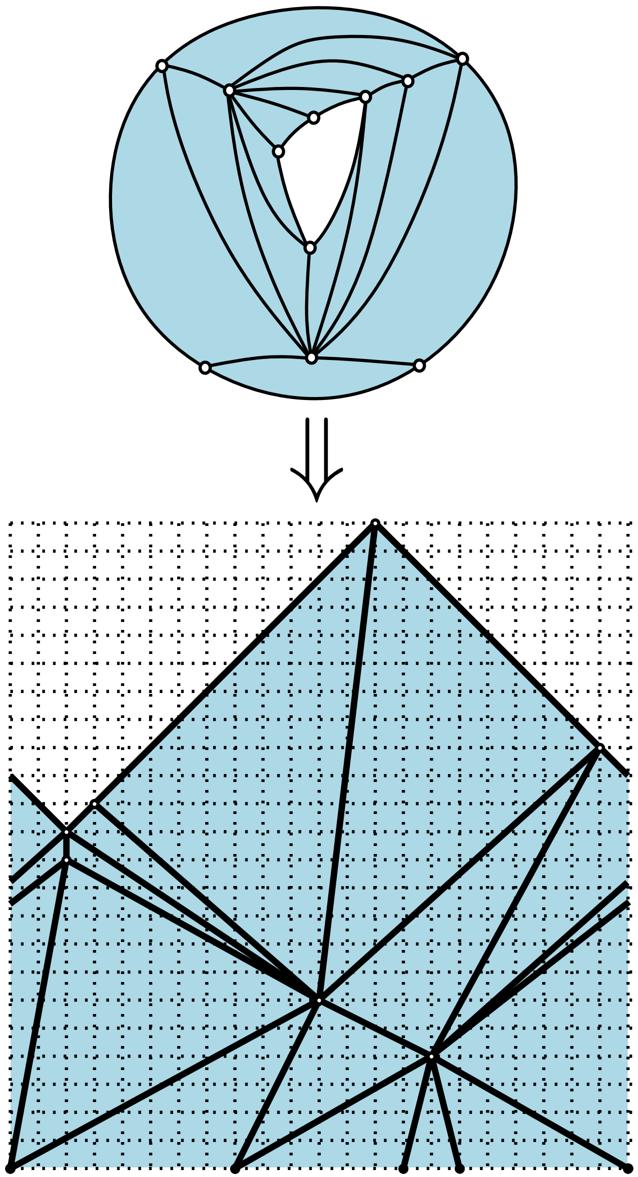
[Castelli,Devillers,F'12]

Plane

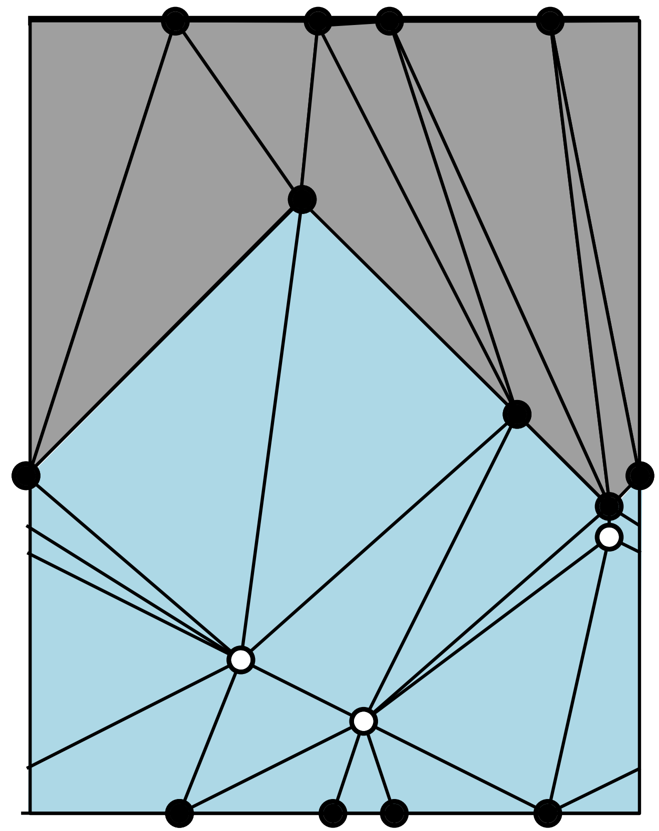


Grid $(2n-4) \times (n-2)$

Cylinder



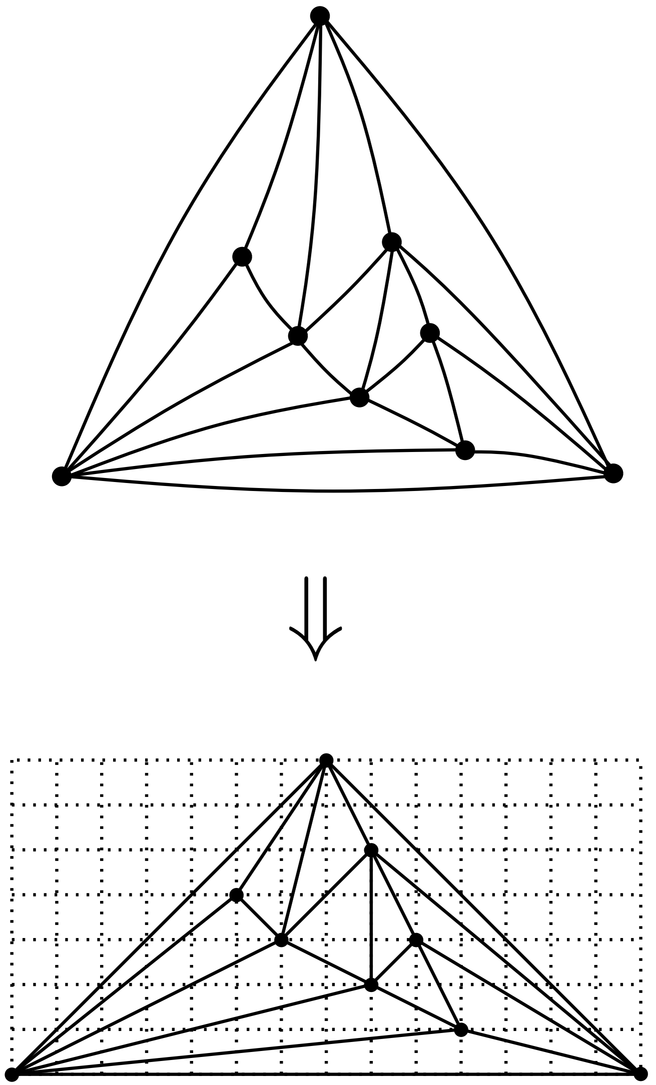
Torus



Overview

1. Recall FPP algorithm

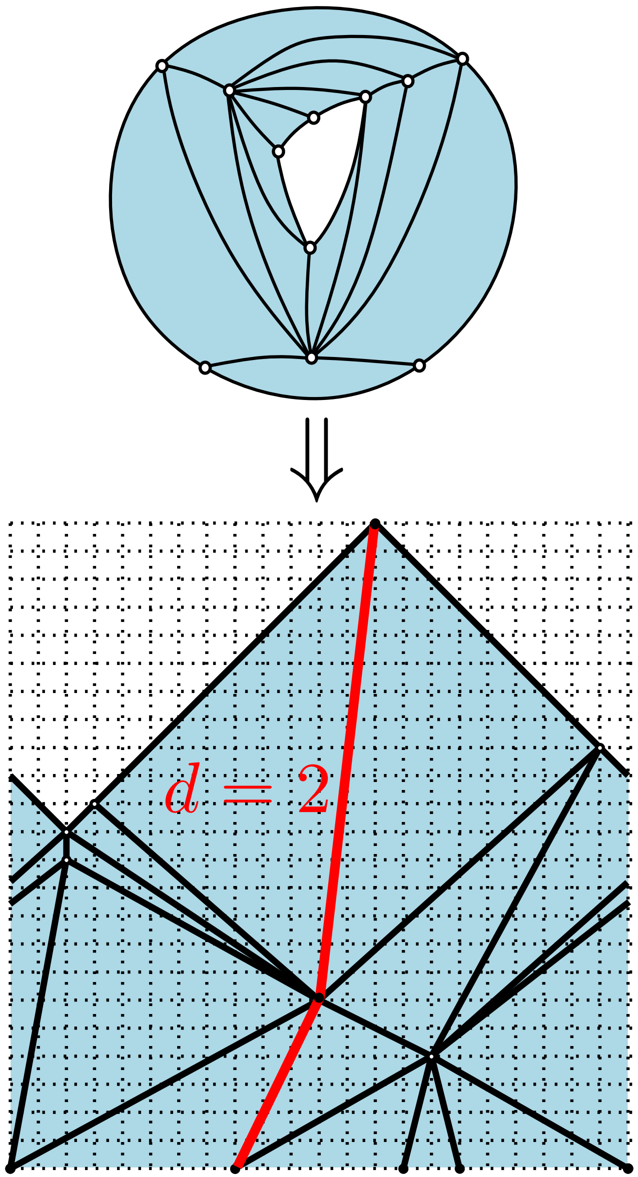
Plane



Grid $(2n-4) \times (n-2)$

2. Extend to the cylinder

Cylinder

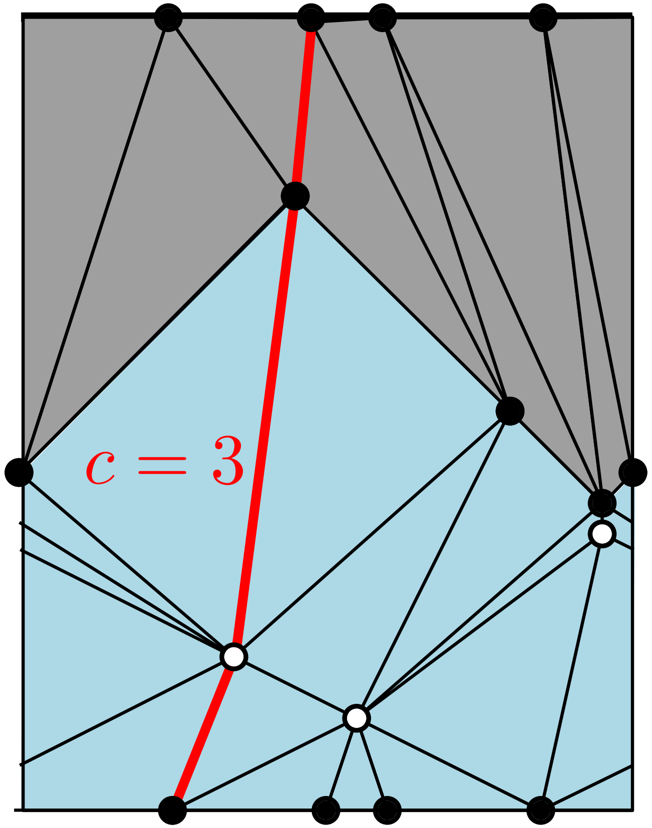


Grid $\leq 2n \times n(2d+1)$

3. Get toroidal drawings

[Castelli,Devillers,F'12]

Torus



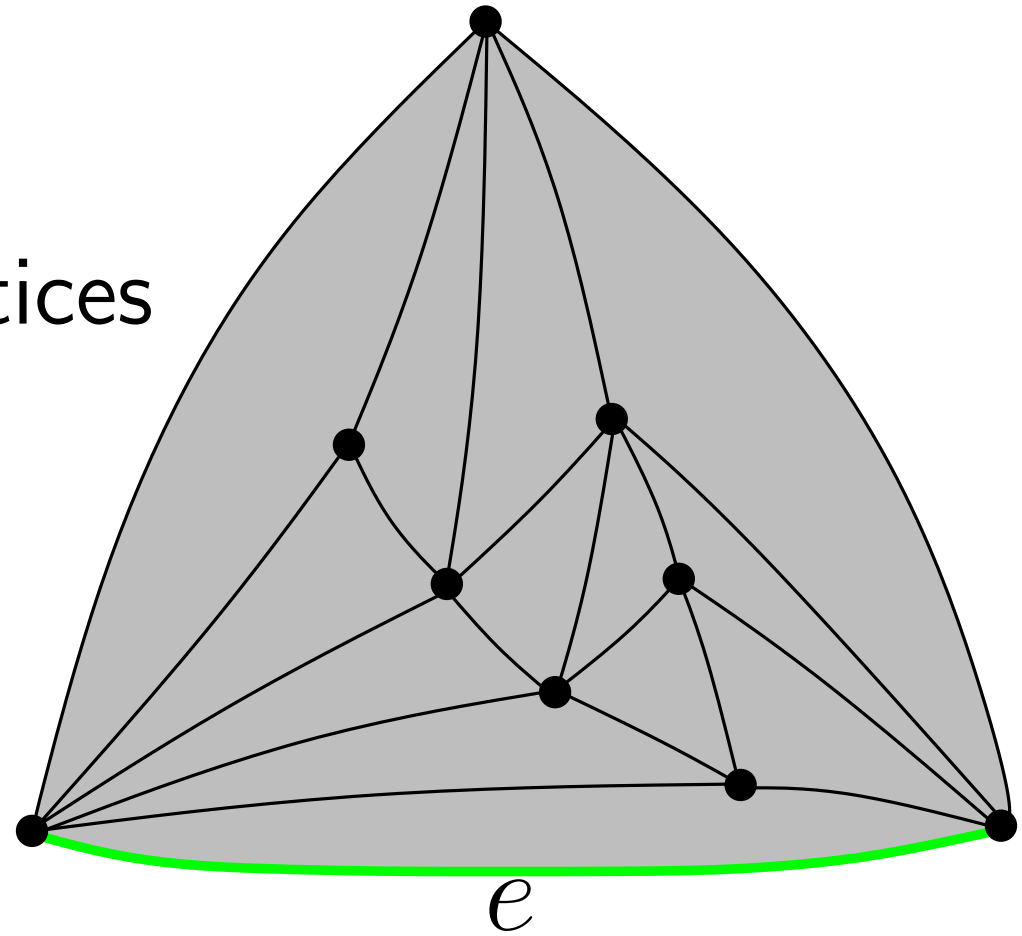
Grid$\leq 2n \times (1+n(2c+1))$

Canonical ordering for planar triangulations

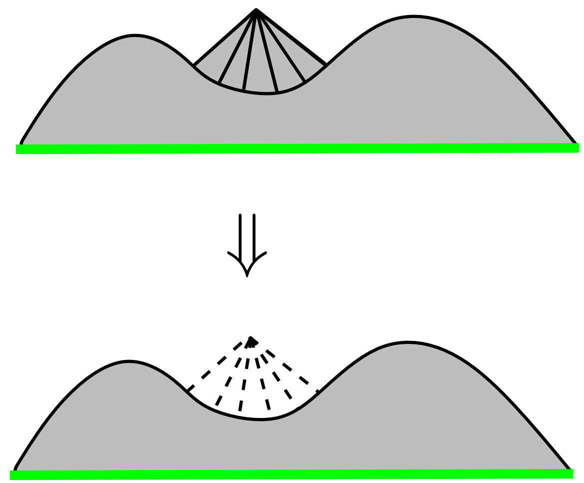
T a planar triangulation with marked **bottom-edge** e

Canonical ordering = shelling order (from top to bottom)

$T \backslash e$ has 7 vertices



At each step:

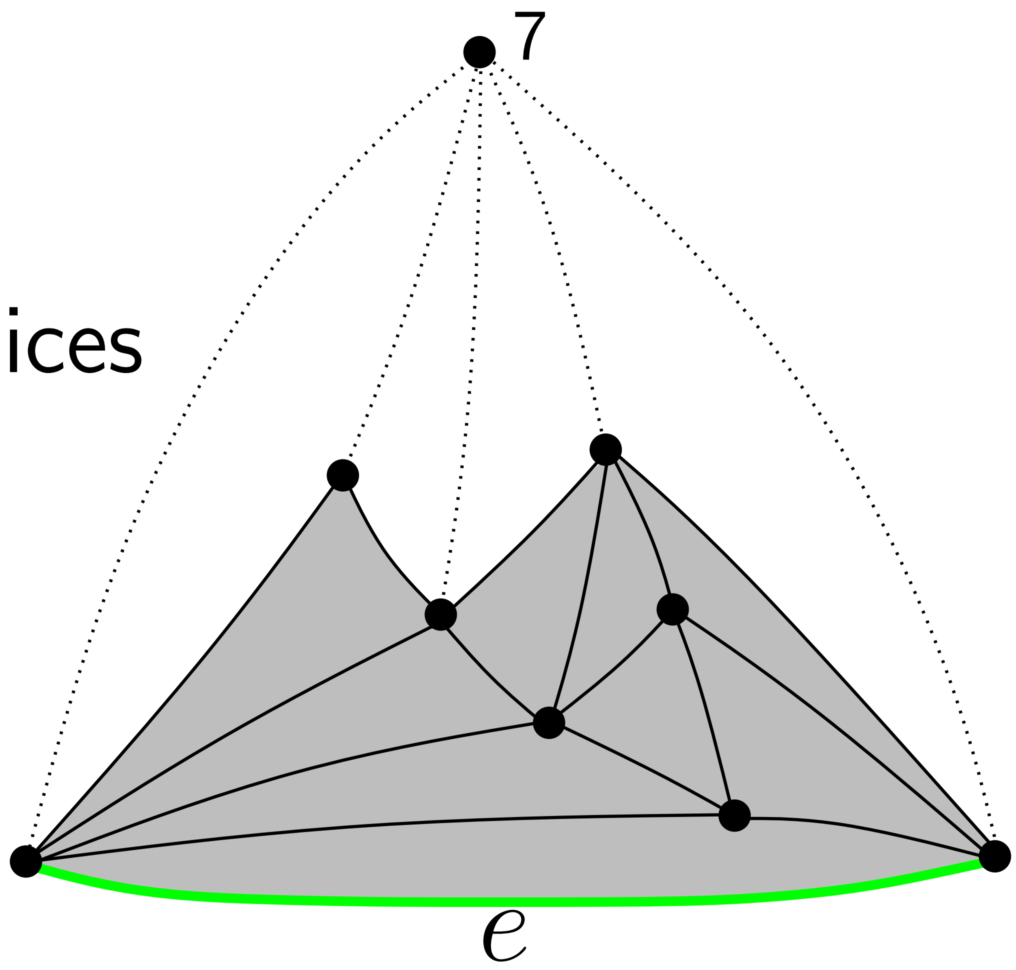


Canonical ordering for planar triangulations

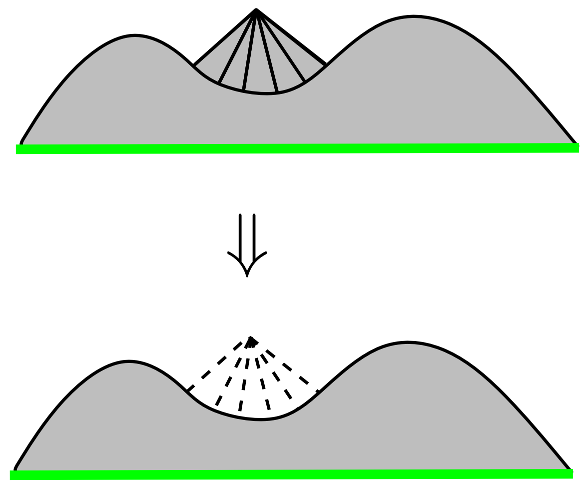
T a planar triangulation with marked **bottom-edge** e

Canonical ordering $=$ shelling order (from top to bottom)

$T \backslash e$ has 7 vertices



At each step:

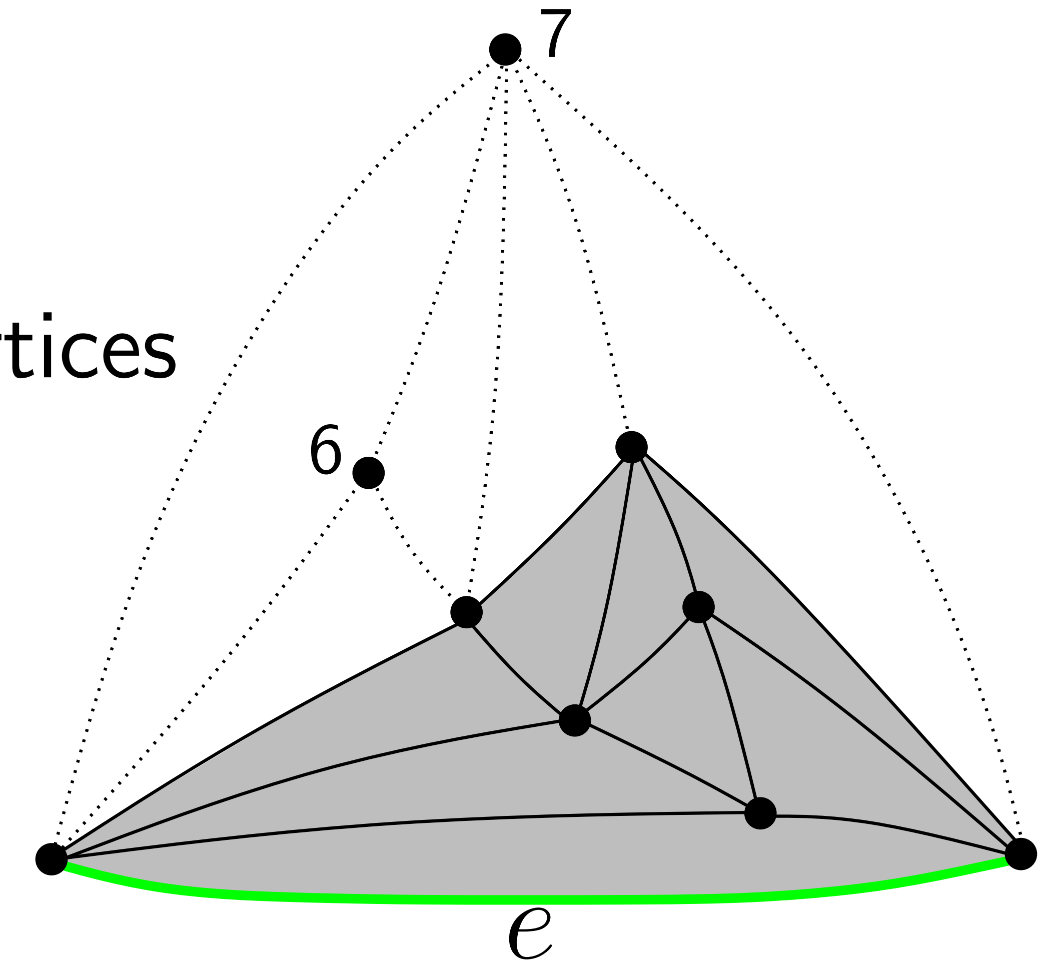


Canonical ordering for planar triangulations

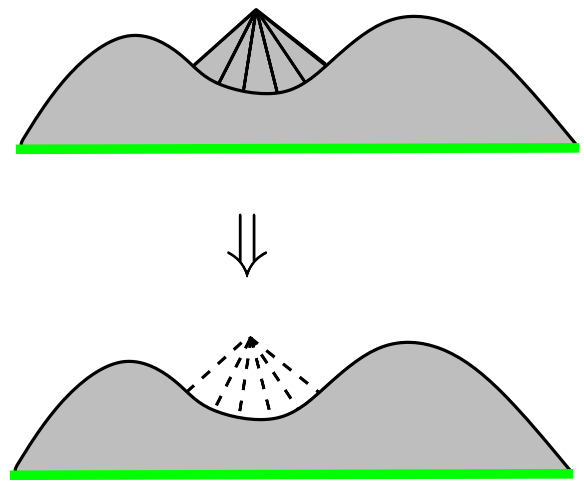
T a planar triangulation with marked **bottom-edge** e

Canonical ordering = shelling order (from top to bottom)

$T \backslash e$ has 7 vertices



At each step:

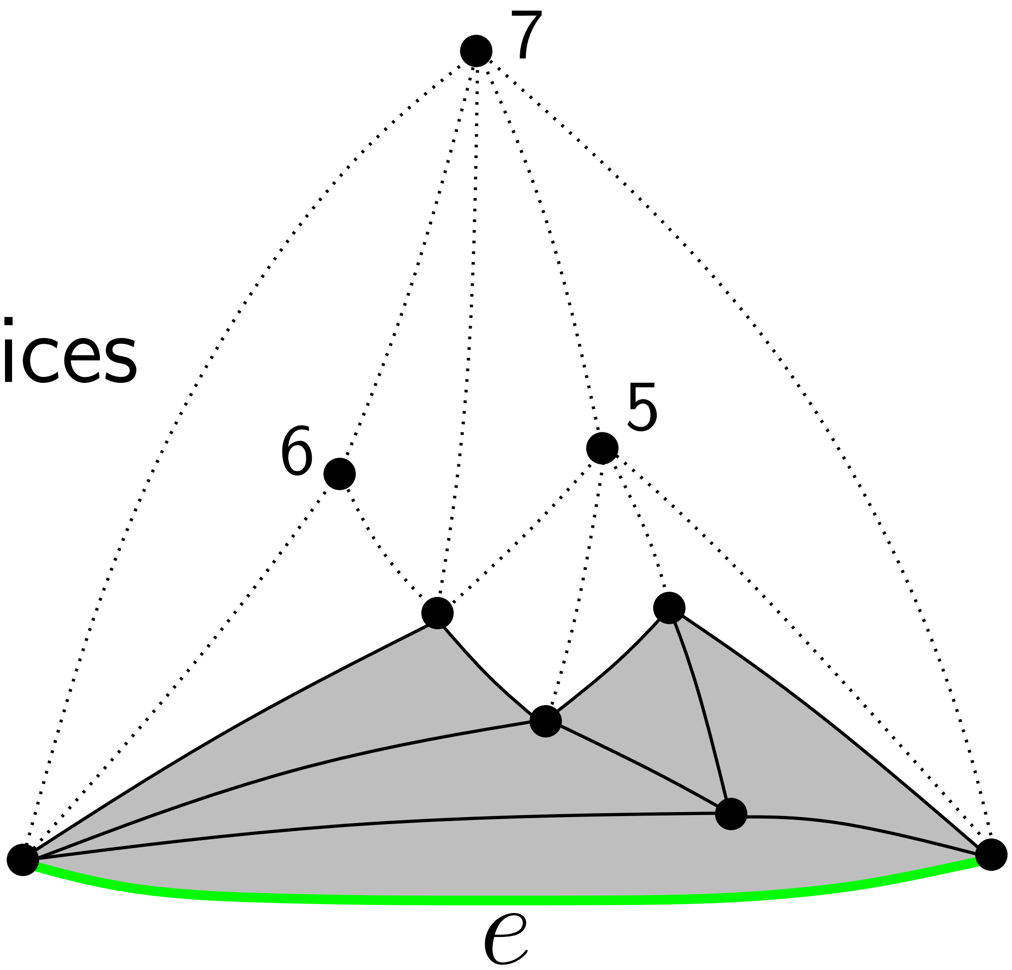


Canonical ordering for planar triangulations

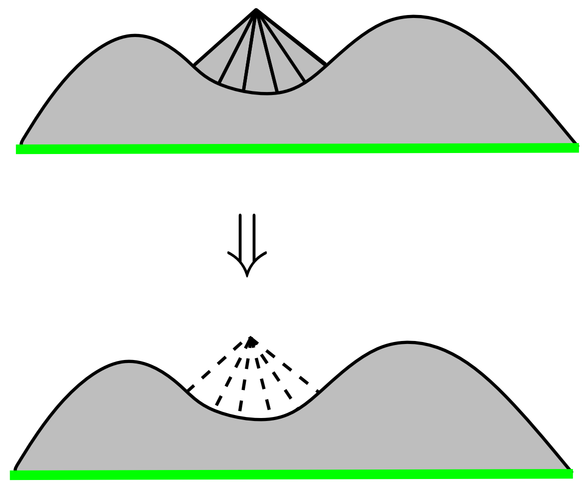
T a planar triangulation with marked **bottom-edge** e

Canonical ordering = shelling order (from top to bottom)

$T \backslash e$ has 7 vertices



At each step:

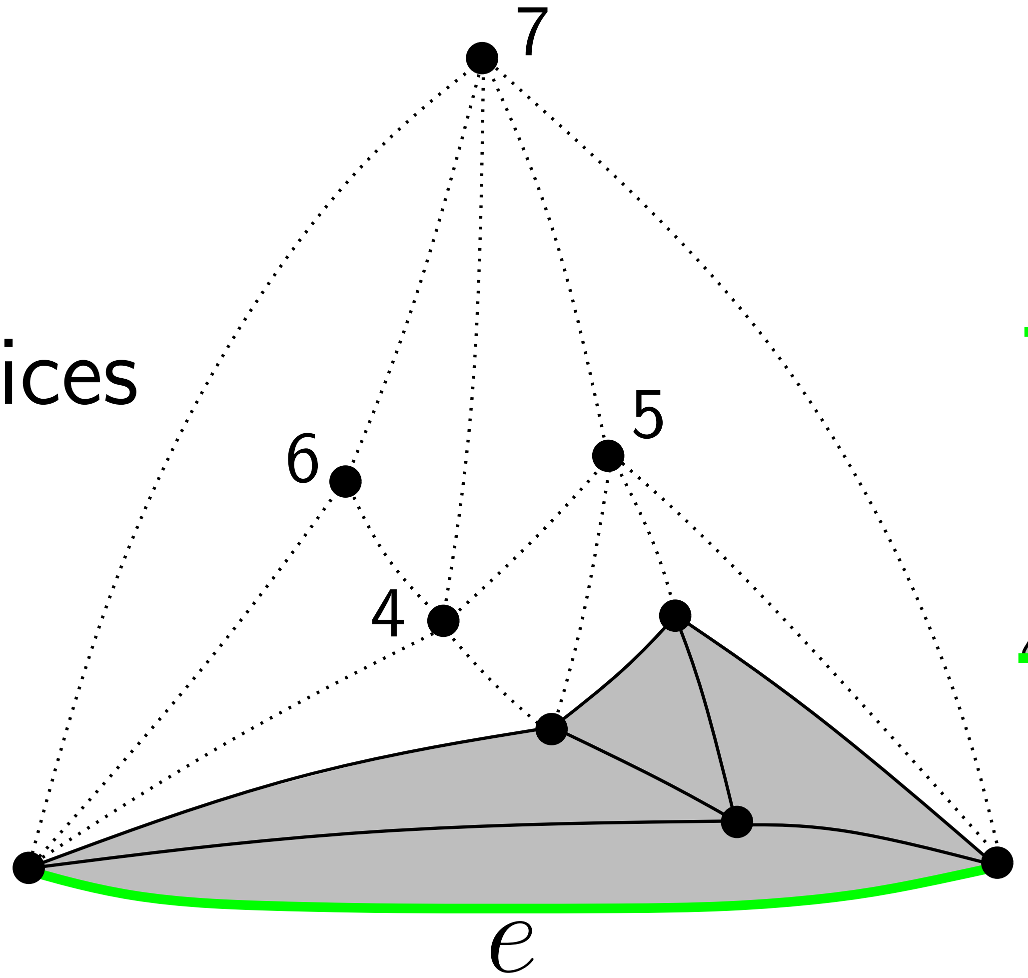


Canonical ordering for planar triangulations

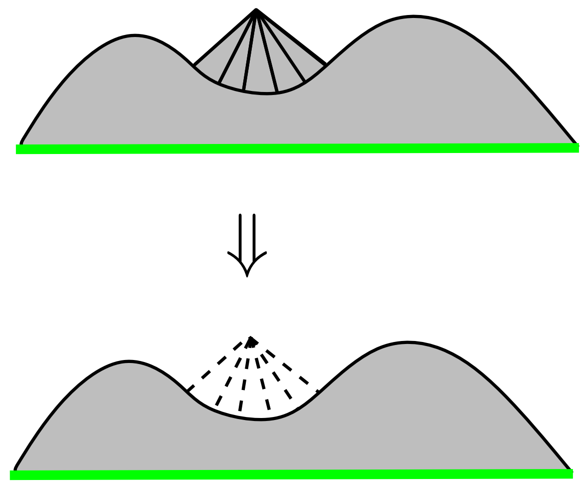
T a planar triangulation with marked **bottom-edge** e

Canonical ordering $=$ shelling order (from top to bottom)

$T \backslash e$ has 7 vertices



At each step:

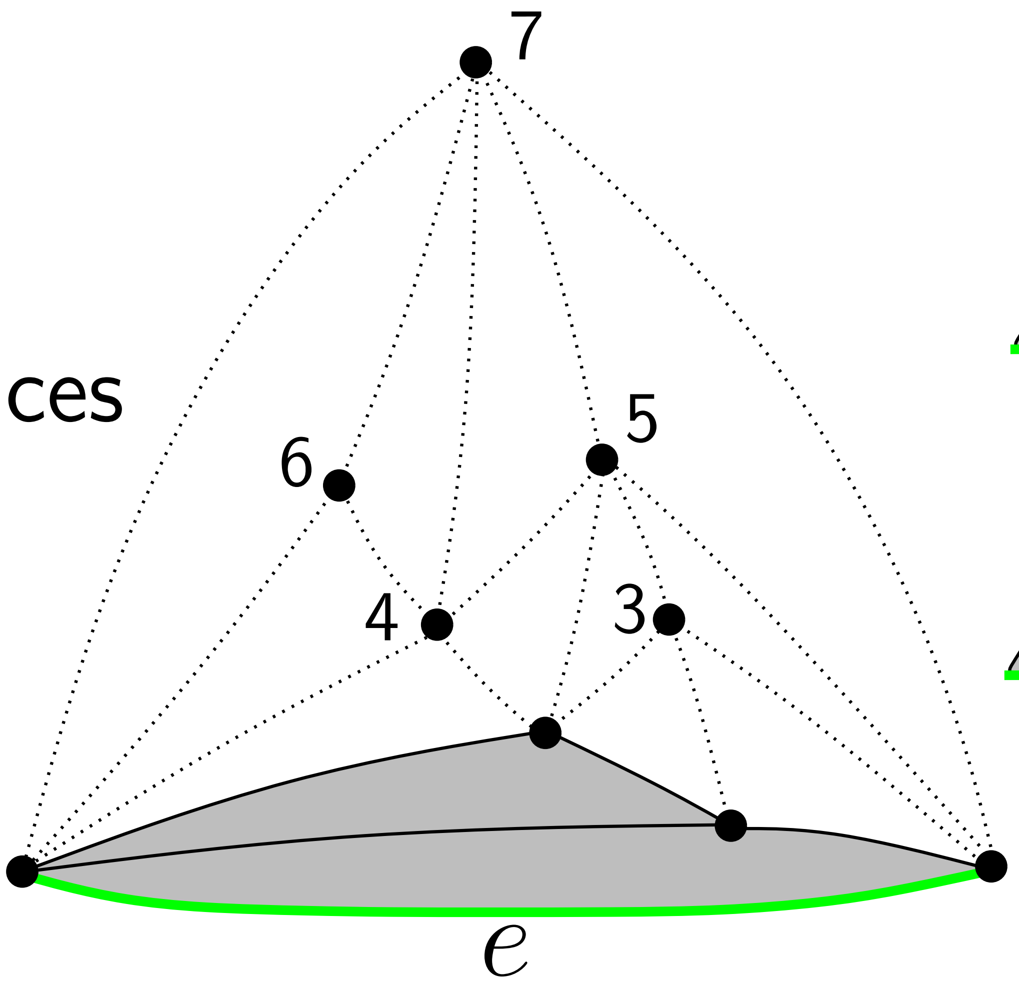


Canonical ordering for planar triangulations

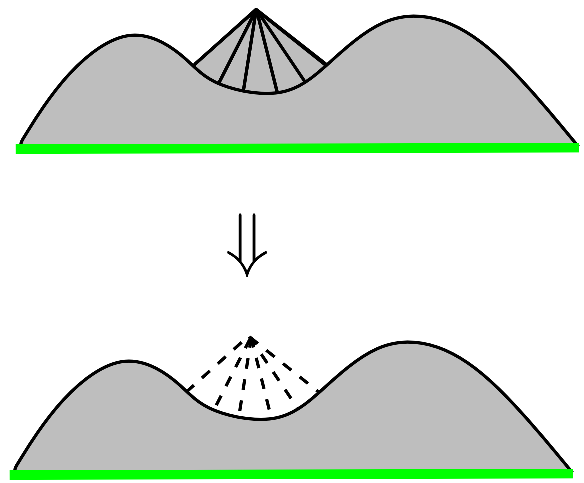
T a planar triangulation with marked **bottom-edge** e

Canonical ordering = shelling order (from top to bottom)

$T \backslash e$ has 7 vertices



At each step:

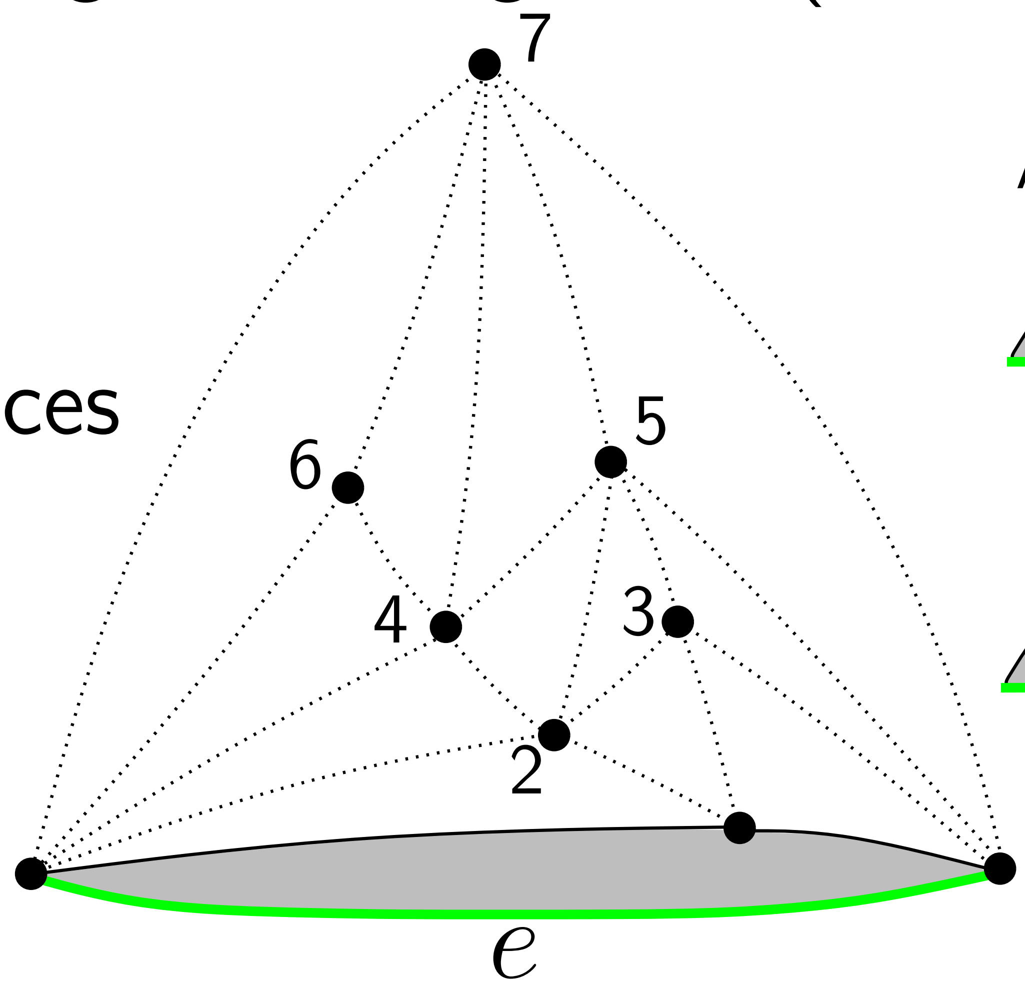


Canonical ordering for planar triangulations

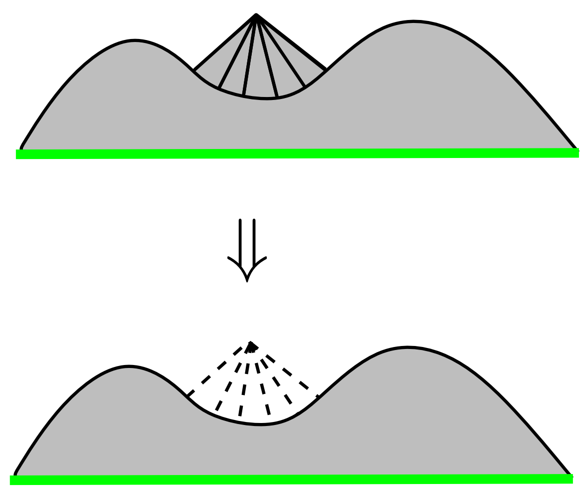
T a planar triangulation with marked **bottom-edge** e

Canonical ordering = shelling order (from top to bottom)

$T \backslash e$ has 7 vertices



At each step:

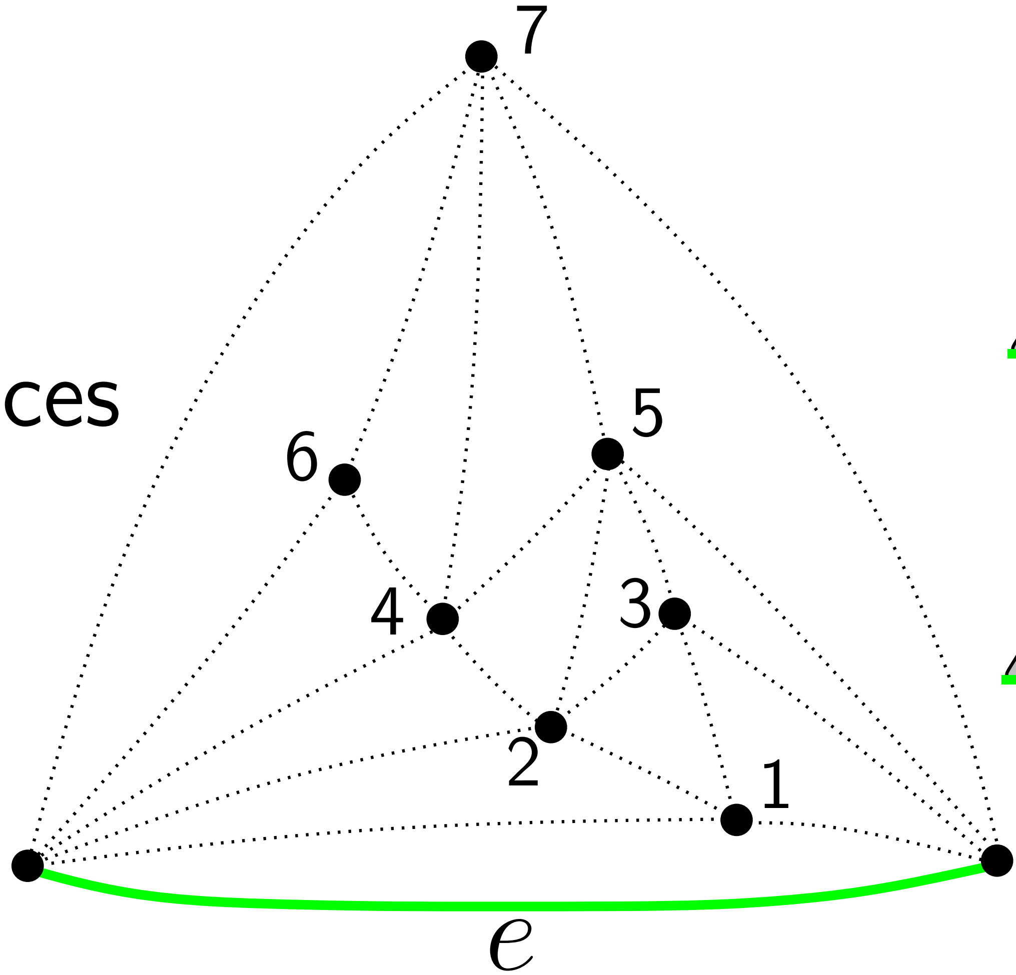


Canonical ordering for planar triangulations

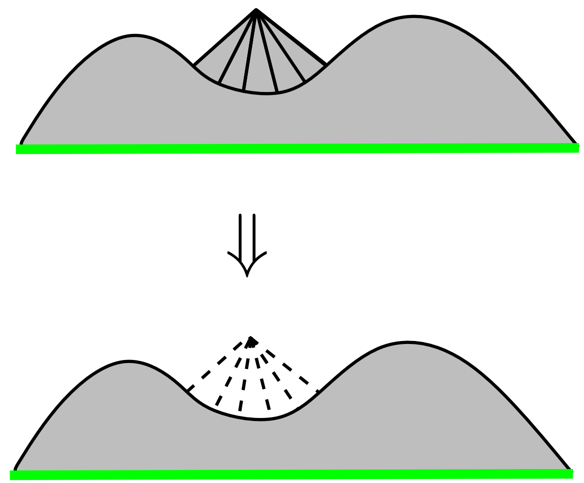
T a planar triangulation with marked **bottom-edge** e

Canonical ordering = shelling order (from top to bottom)

$T \backslash e$ has 7 vertices

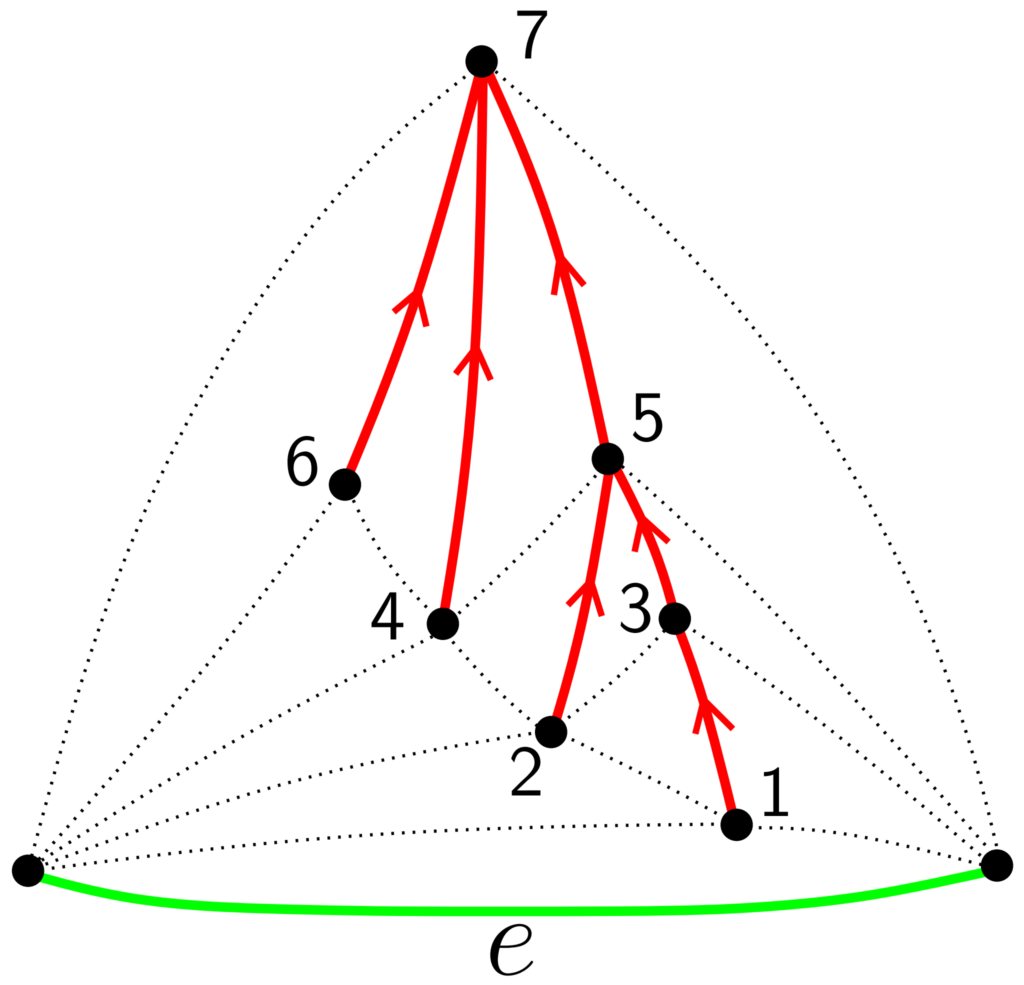


At each step:



Canonical ordering: primal tree & dual tree

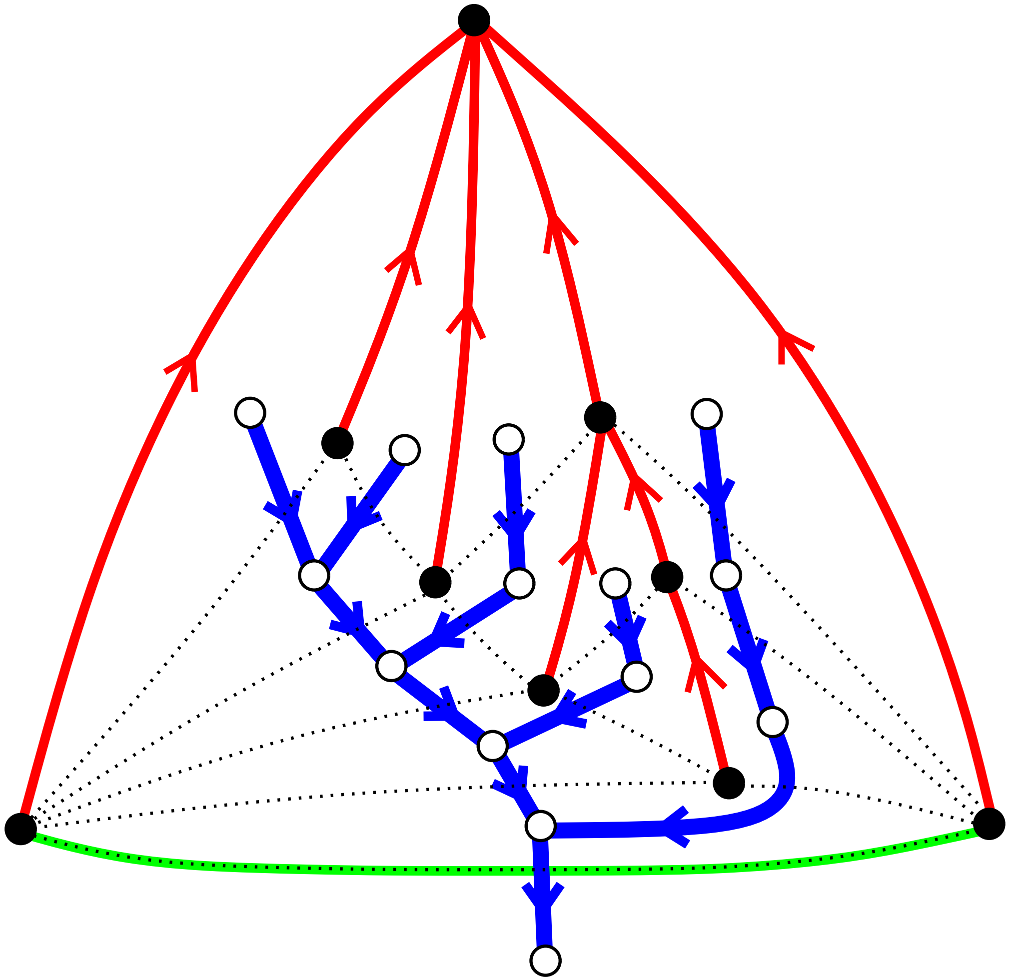
primal tree: parent of each vertex v = neighbour of v of largest label



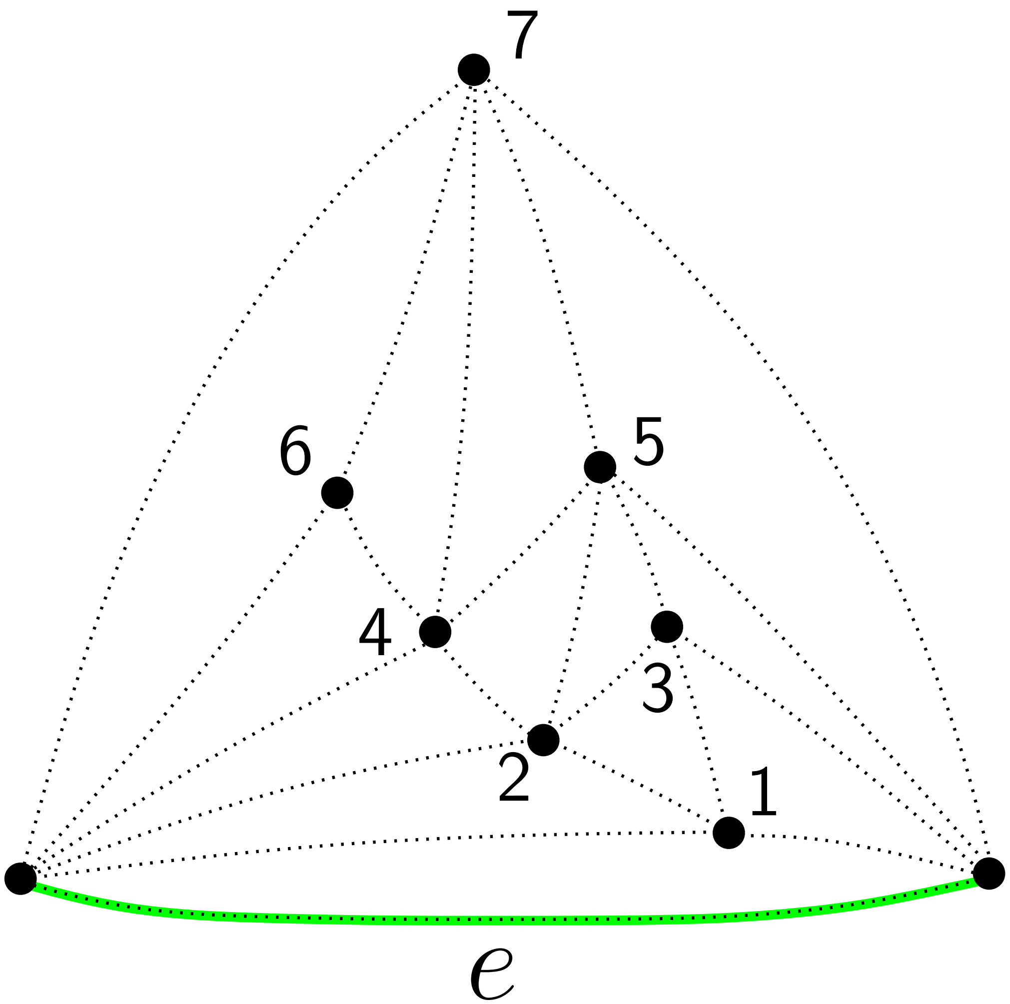
Canonical ordering: primal tree & dual tree

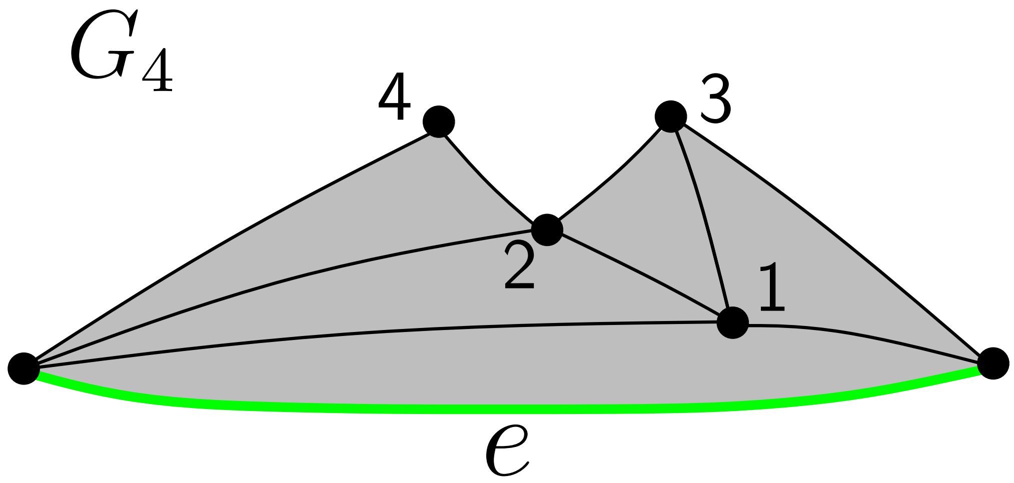
primal tree: parent of each vertex v = neighbour of v of largest label

dual tree: dual of primal tree (augmented with two outer edges)



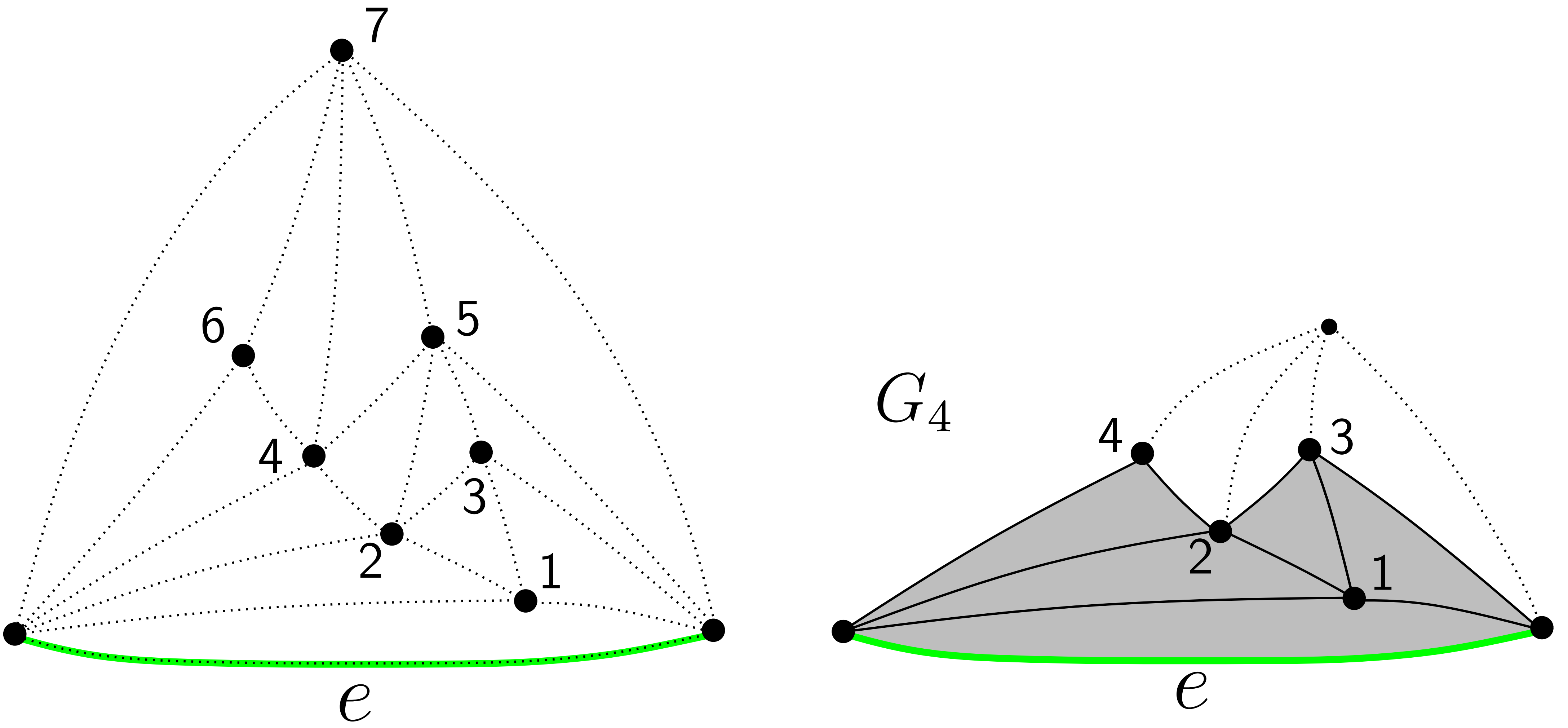
Canonical ordering: successive induced graphs





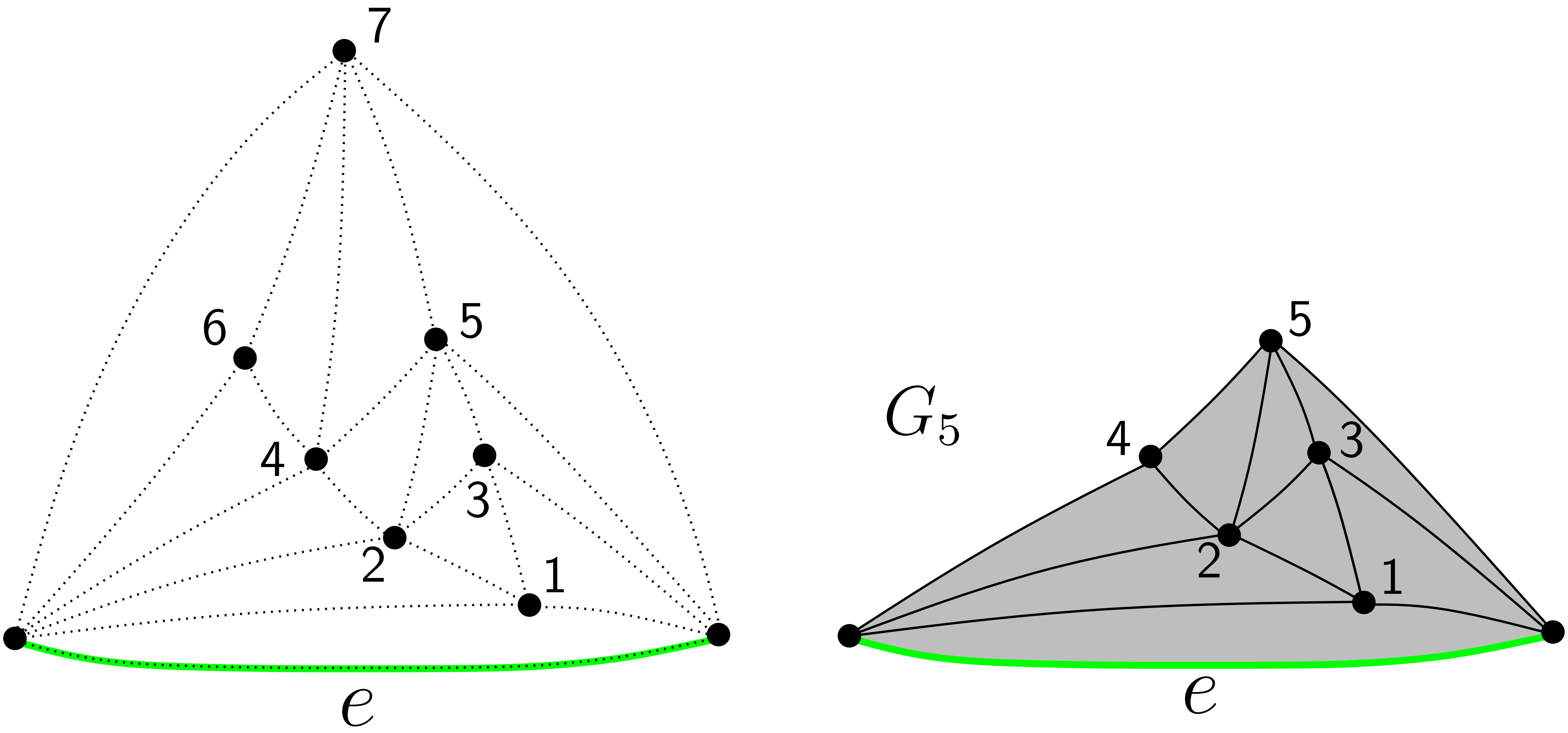
Notation: G_k is the graph formed by e and $\{1, \ldots, k\}$

Canonical ordering: successive induced graphs



Notation: G_k is the graph formed by e and $\{1, \ldots, k\}$

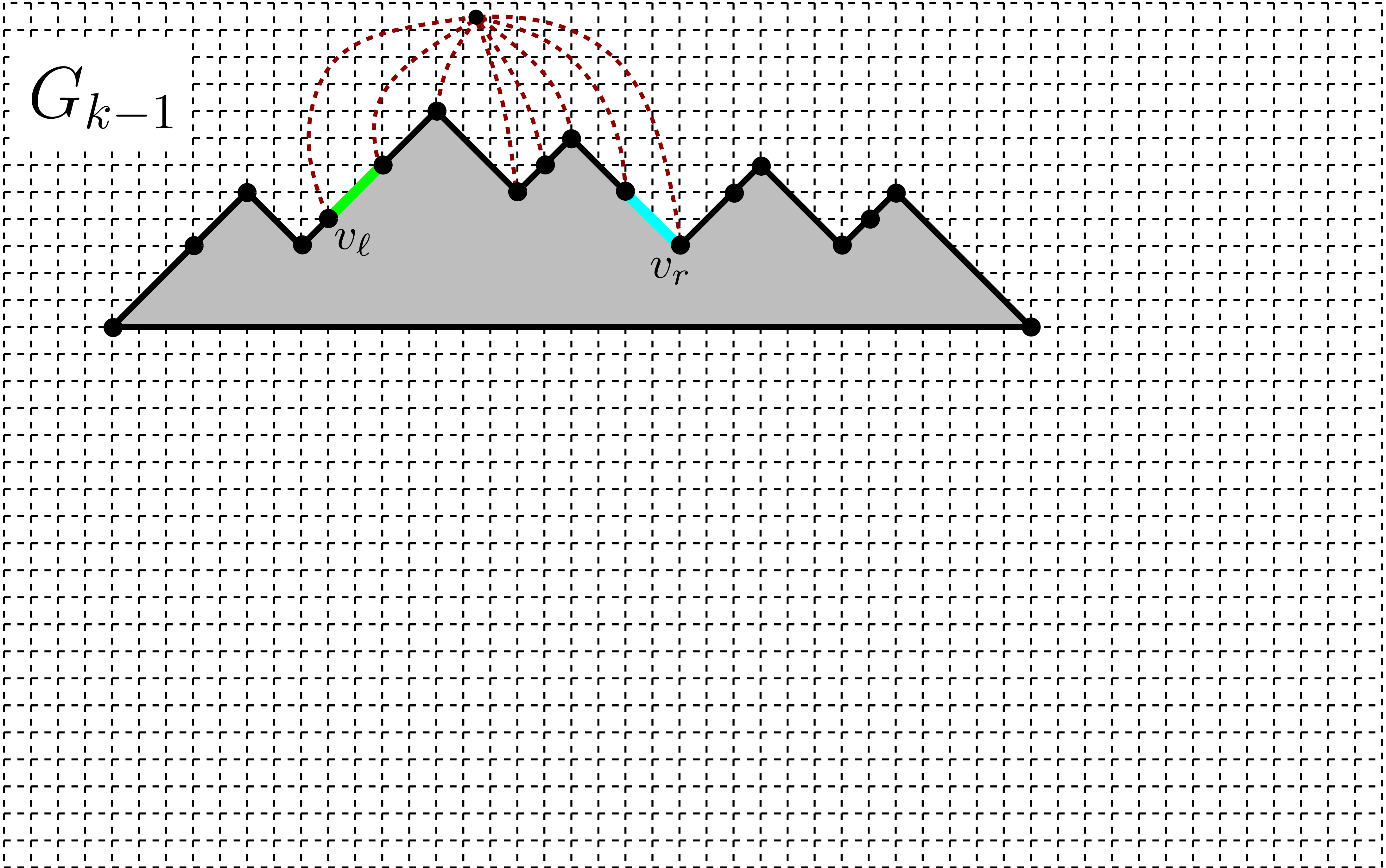
Canonical ordering: successive induced graphs



Notation: G_k is the graph formed by e and $\{1, \dots, k\}$

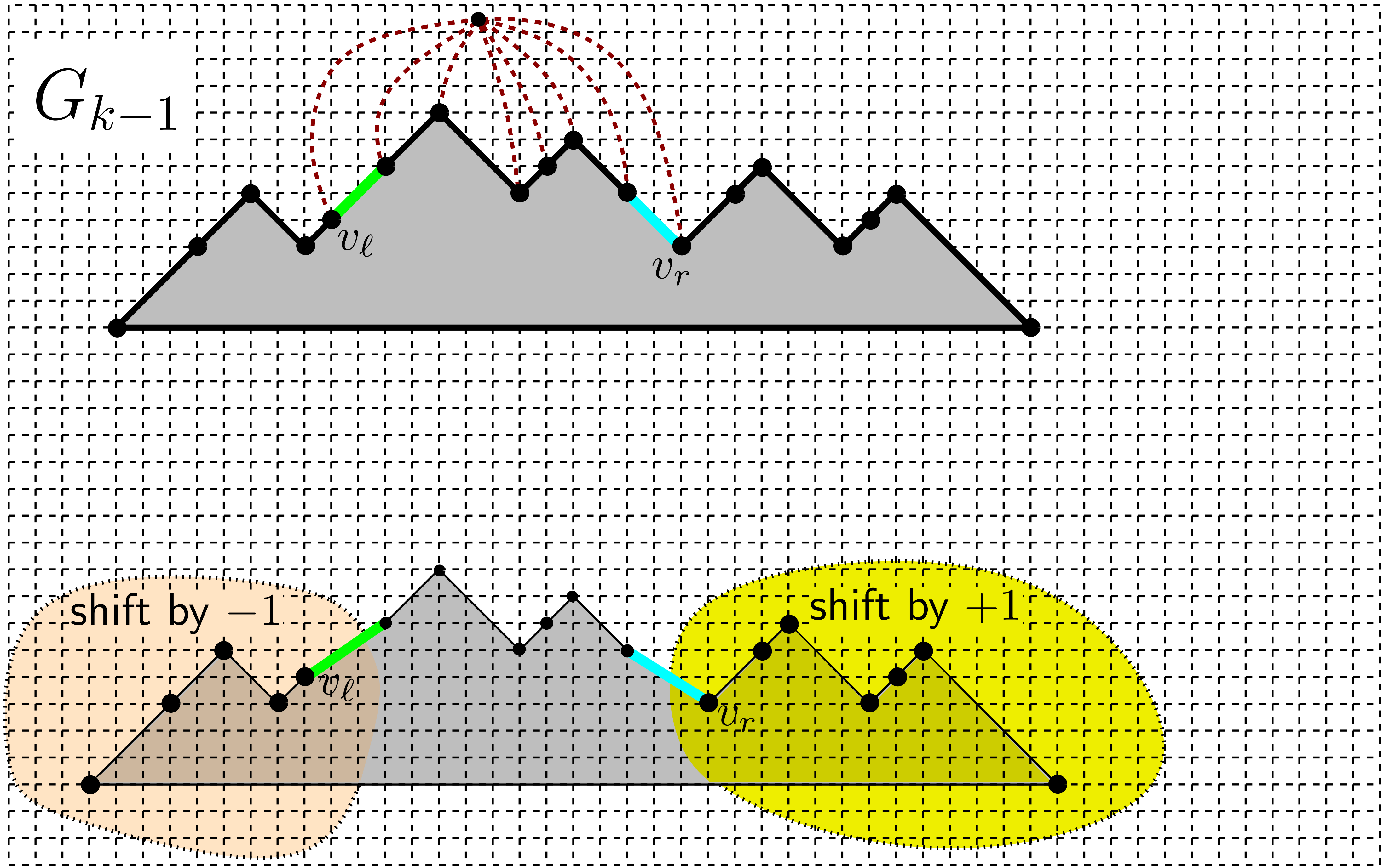
Incremental drawing algorithm

[de Fraysseix, Pollack, Pach'89]



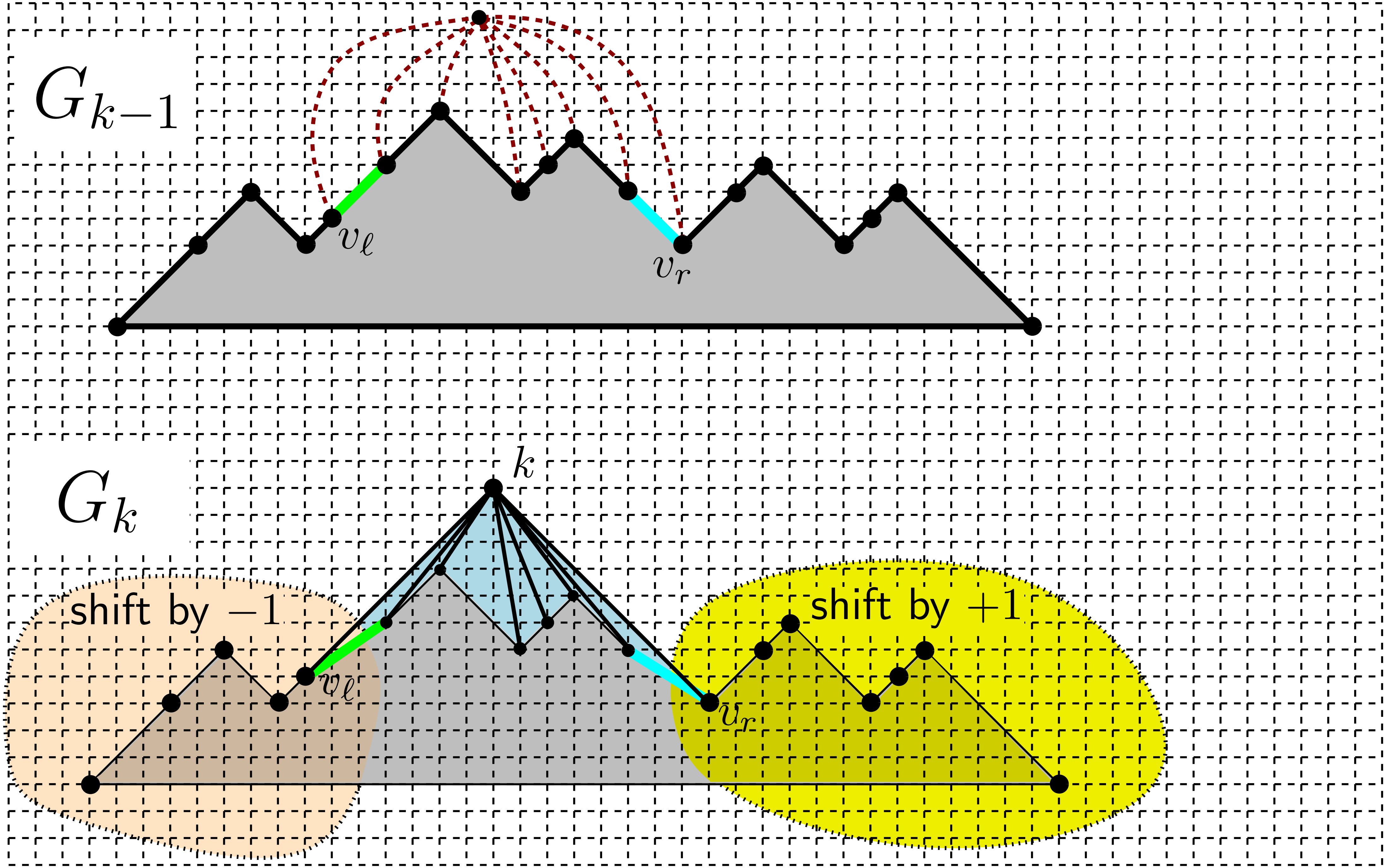
Incremental drawing algorithm

[de Fraysseix, Pollack, Pach'89]



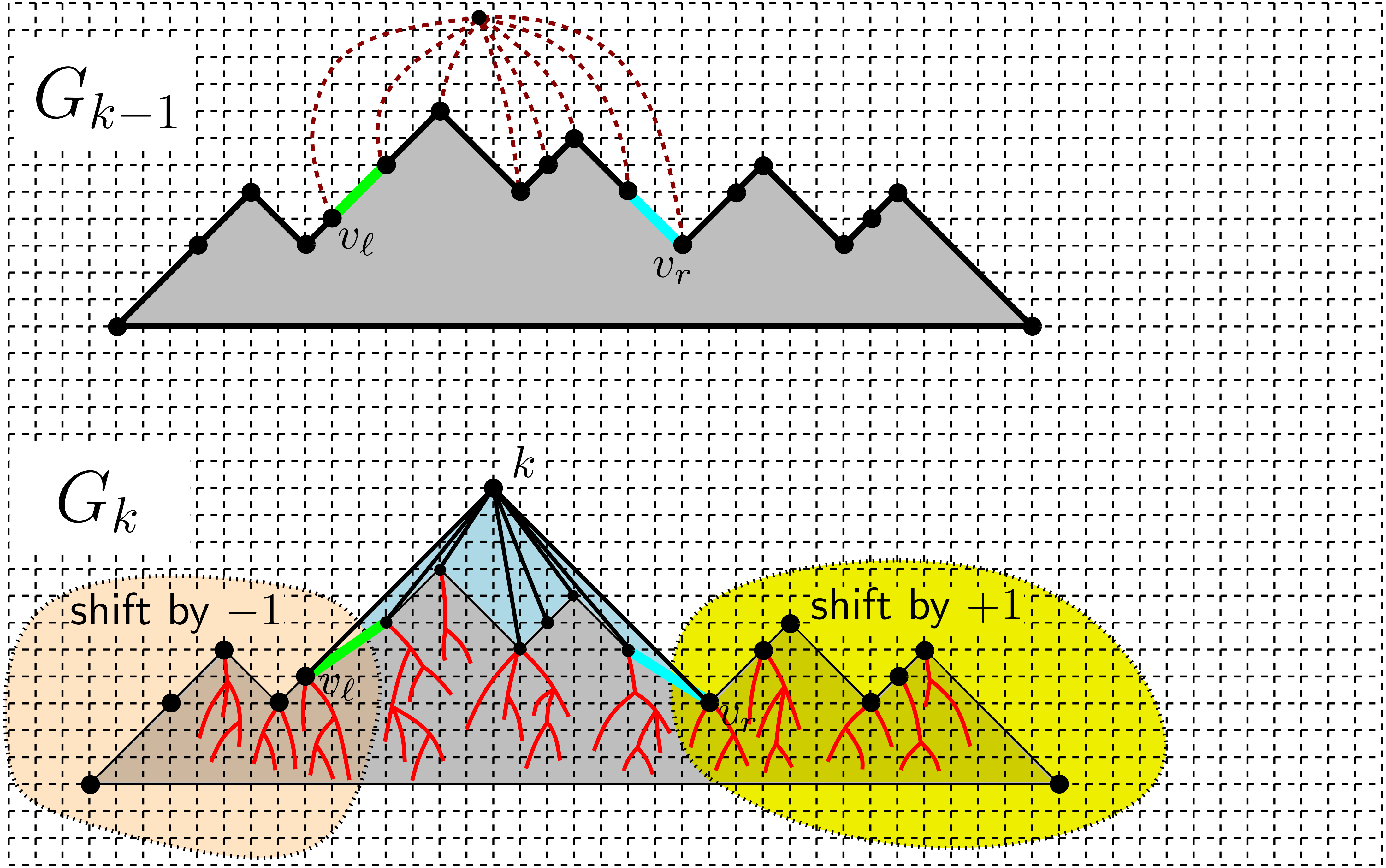
Incremental drawing algorithm

[de Fraysseix, Pollack, Pach'89]



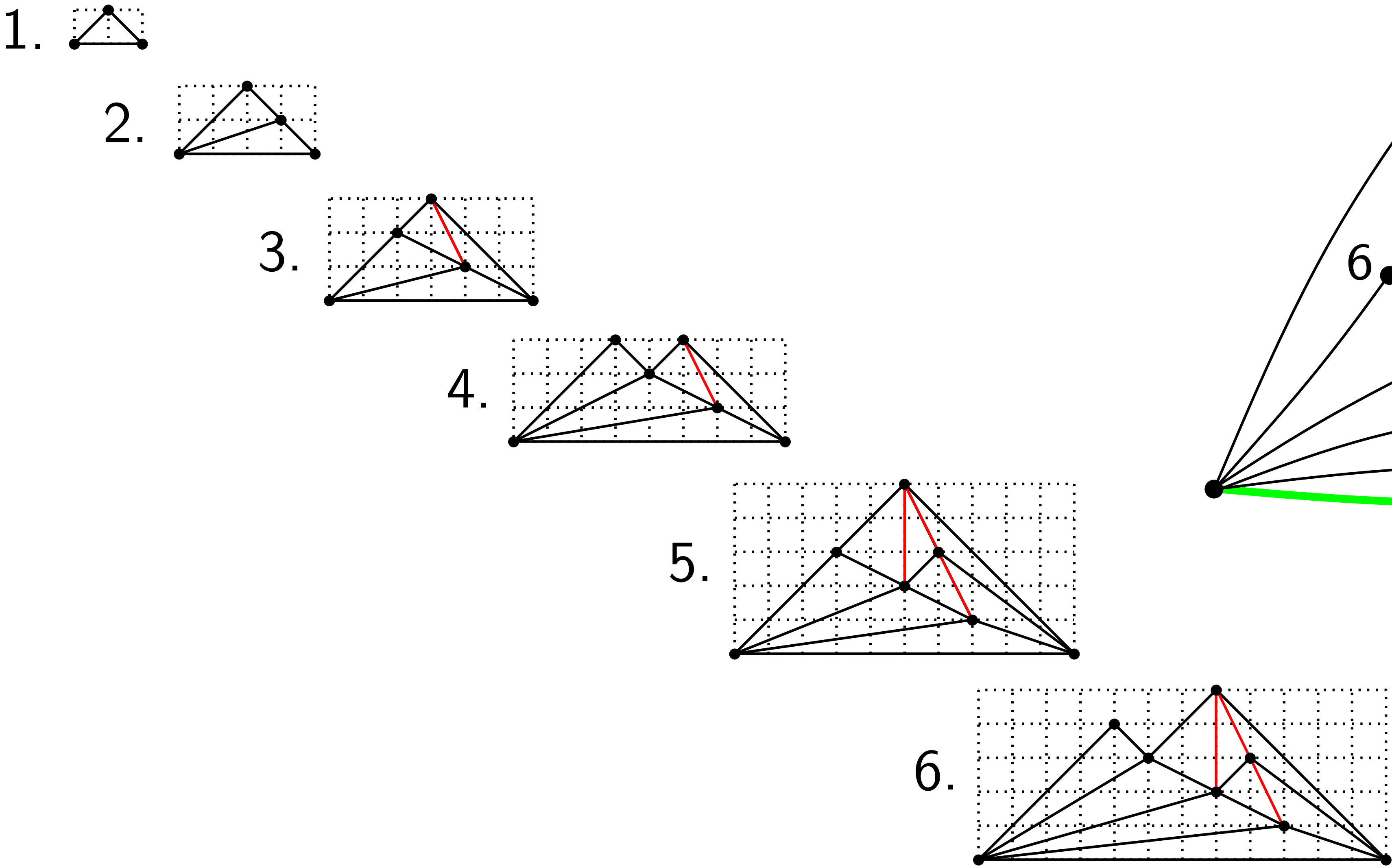
Incremental drawing algorithm

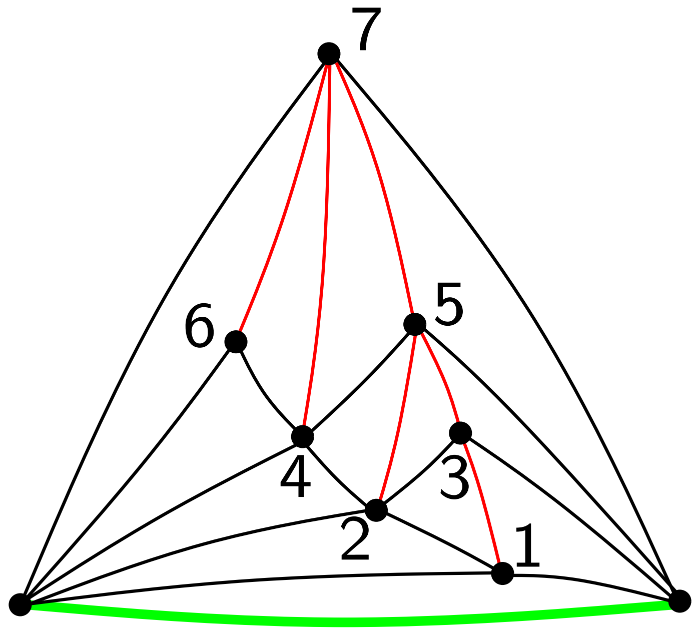
[de Fraysseix, Pollack, Pach'89]

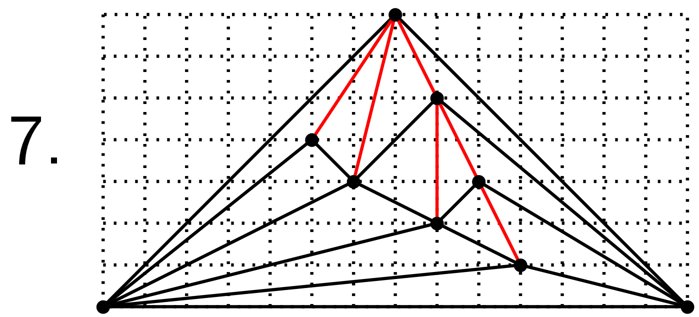


Incremental drawing algorithm

[de Fraysseix, Pollack, Pach'89]





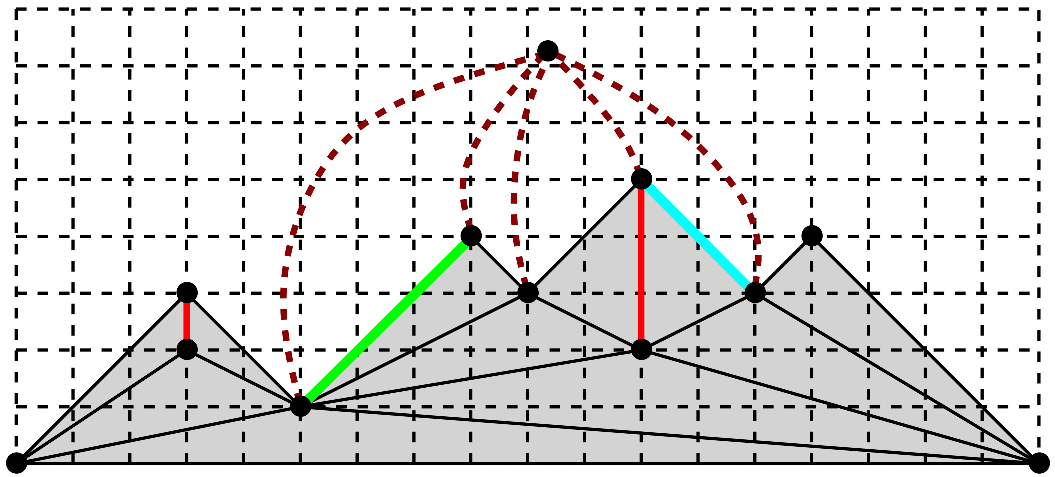


Grid size of G_k: $2k \times k$

Reformulation of the shift-step

At each step: insert two vertical strips of width 1 using the dual tree

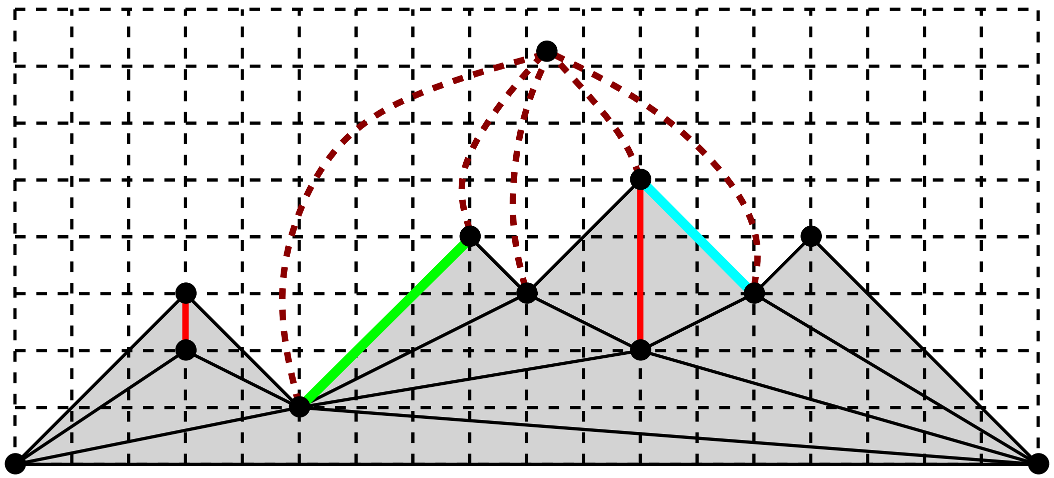
G_{k-1}

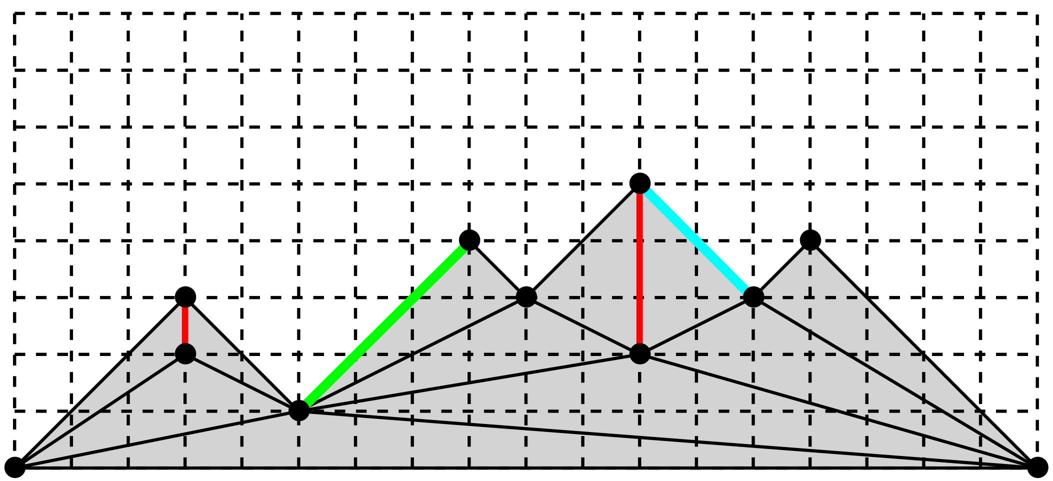


Reformulation of the shift-step

At each step: insert two vertical strips of width 1 using the dual tree

G_{k-1}

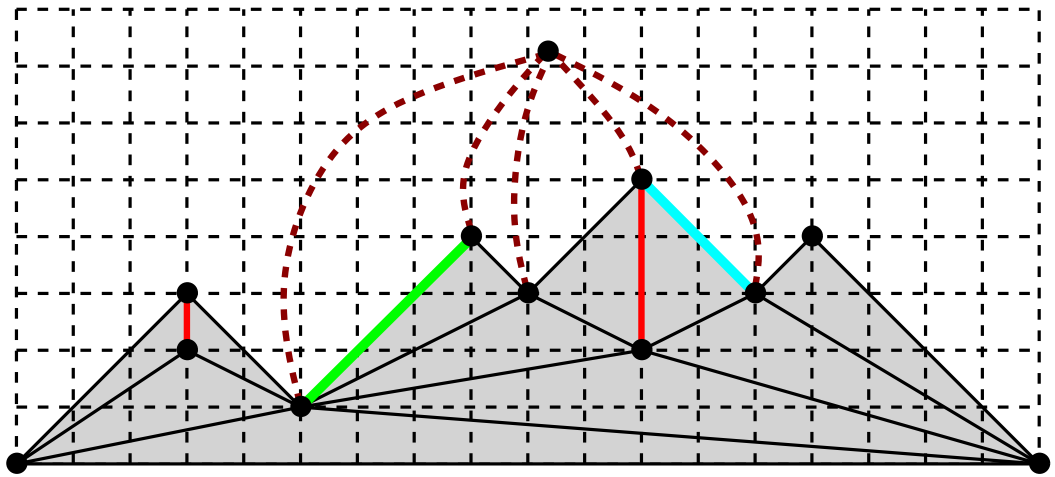


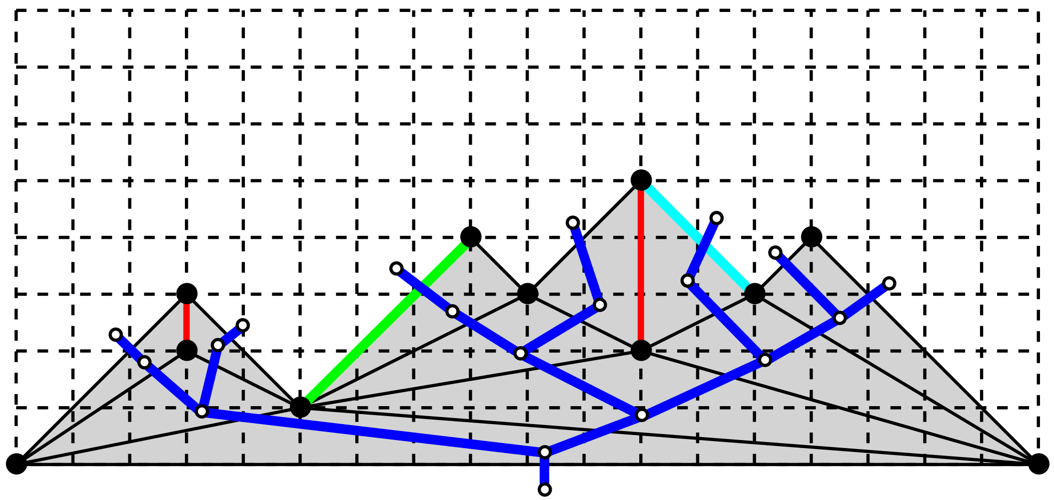


Reformulation of the shift-step

At each step: insert two vertical strips of width 1 using the dual tree

G_{k-1}

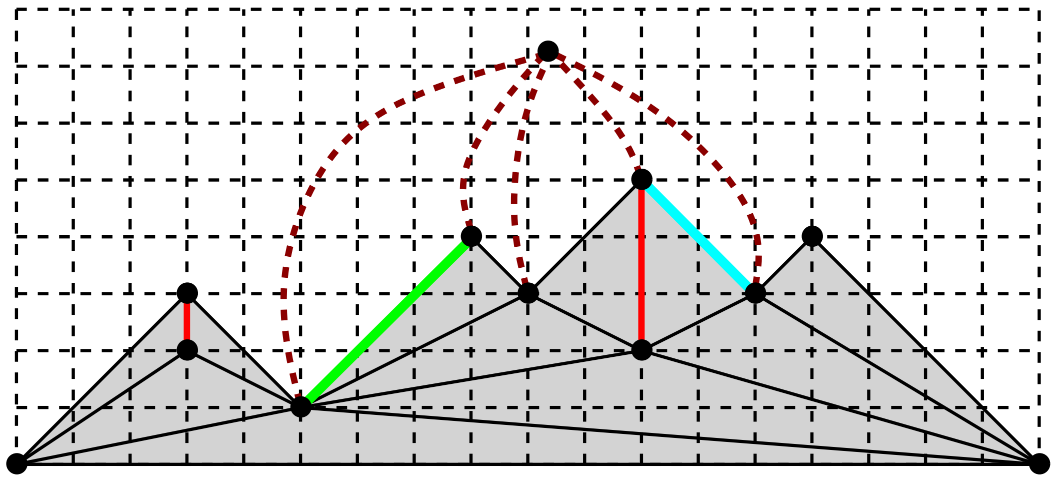


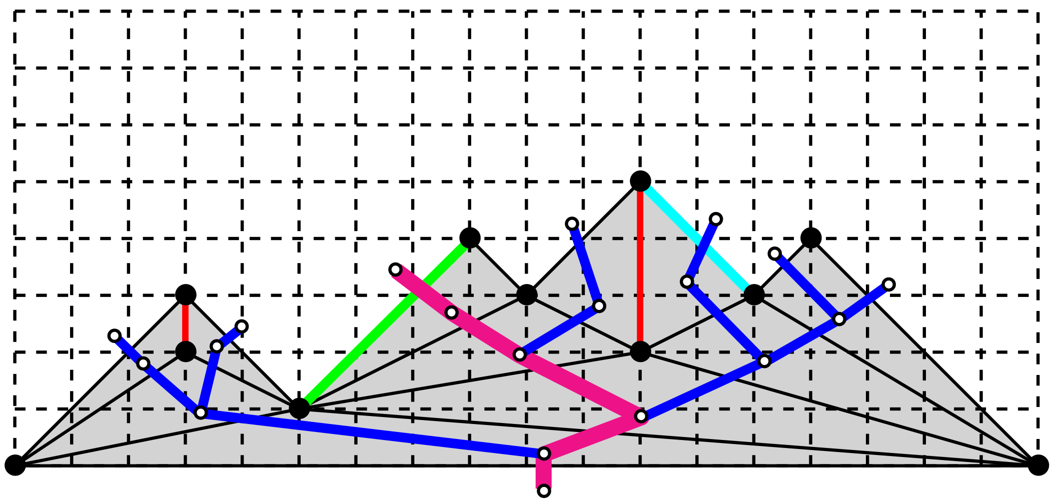


Reformulation of the shift-step

At each step: insert two vertical strips of width 1 using the dual tree

G_{k-1}

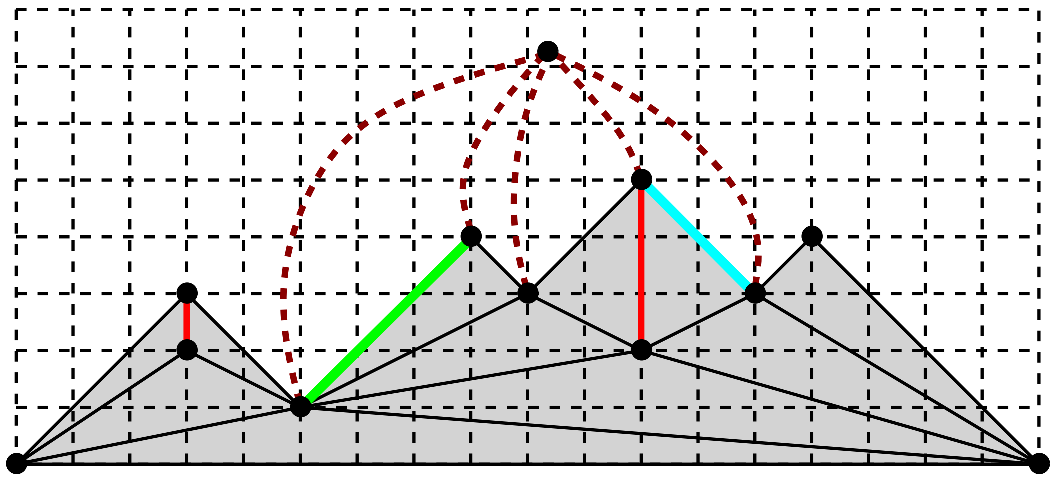


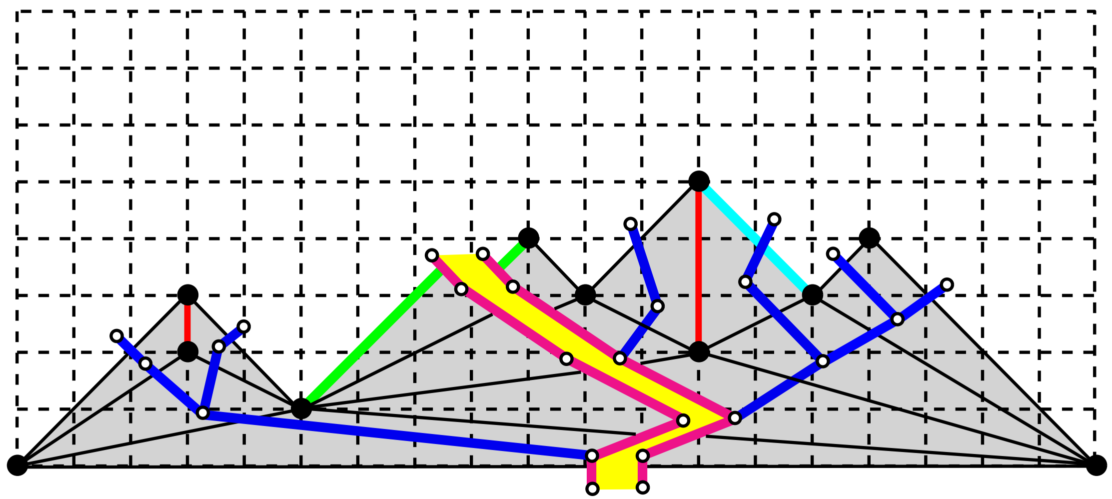


Reformulation of the shift-step

At each step: insert two vertical strips of width 1 using the dual tree

G_{k-1}

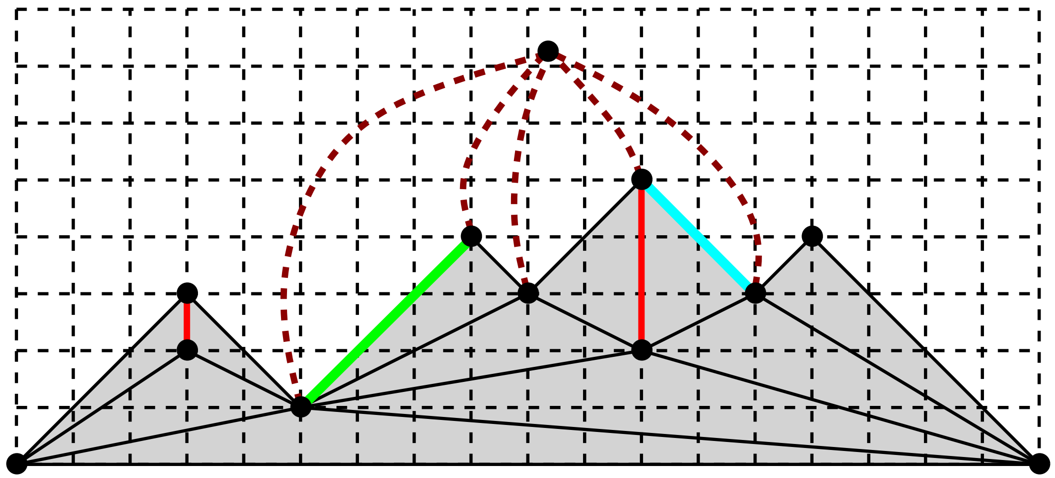


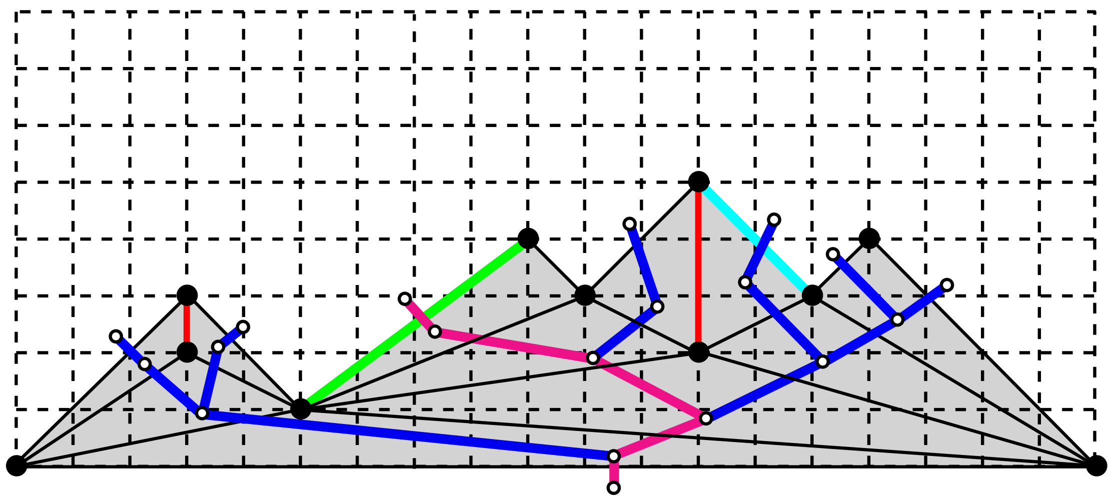


Reformulation of the shift-step

At each step: insert two vertical strips of width 1 using the dual tree

G_{k-1}

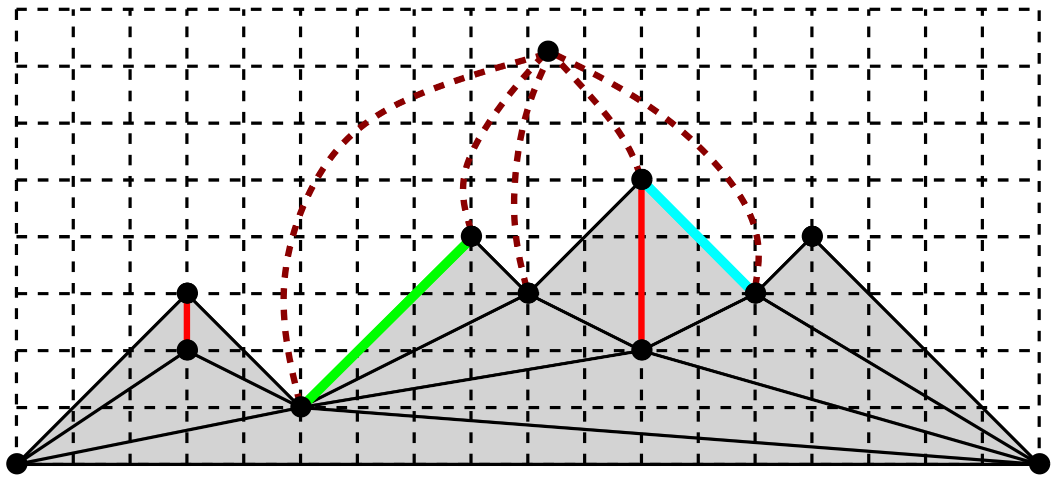


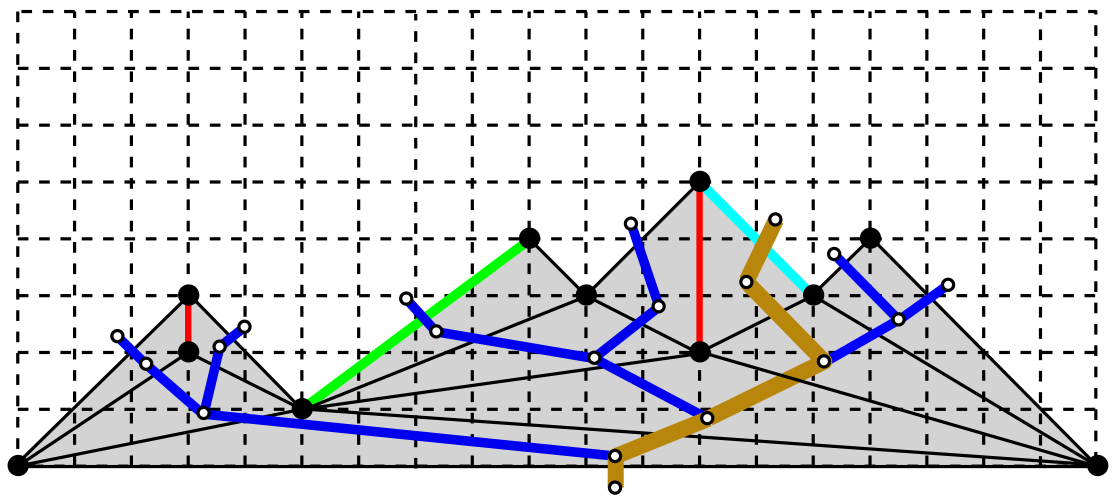


Reformulation of the shift-step

At each step: insert two vertical strips of width 1 using the dual tree

G_{k-1}

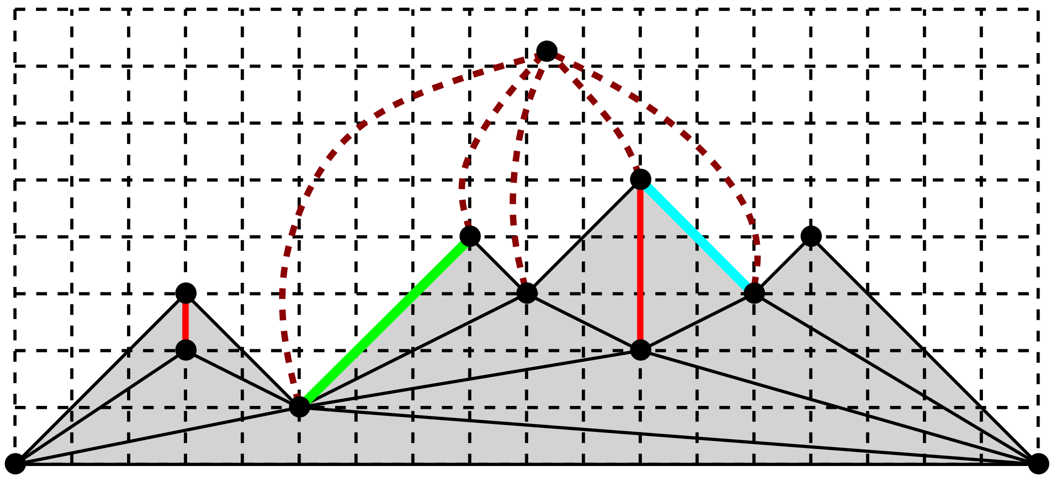


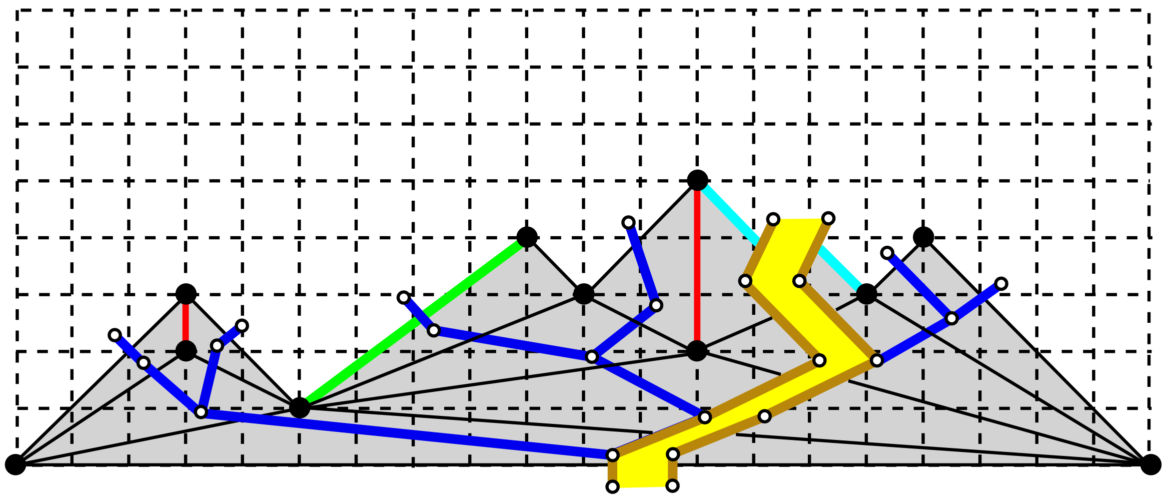


Reformulation of the shift-step

At each step: insert two vertical strips of width 1 using the dual tree

G_{k-1}

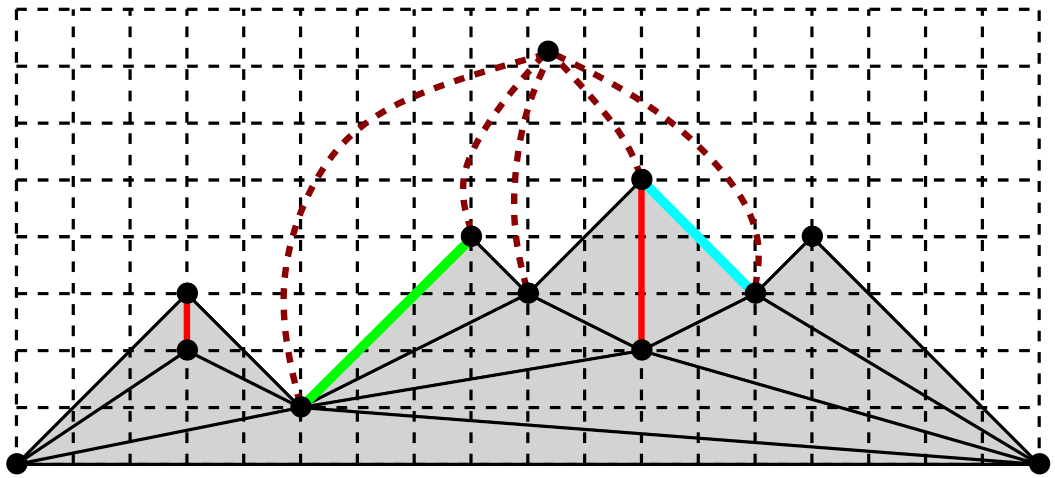


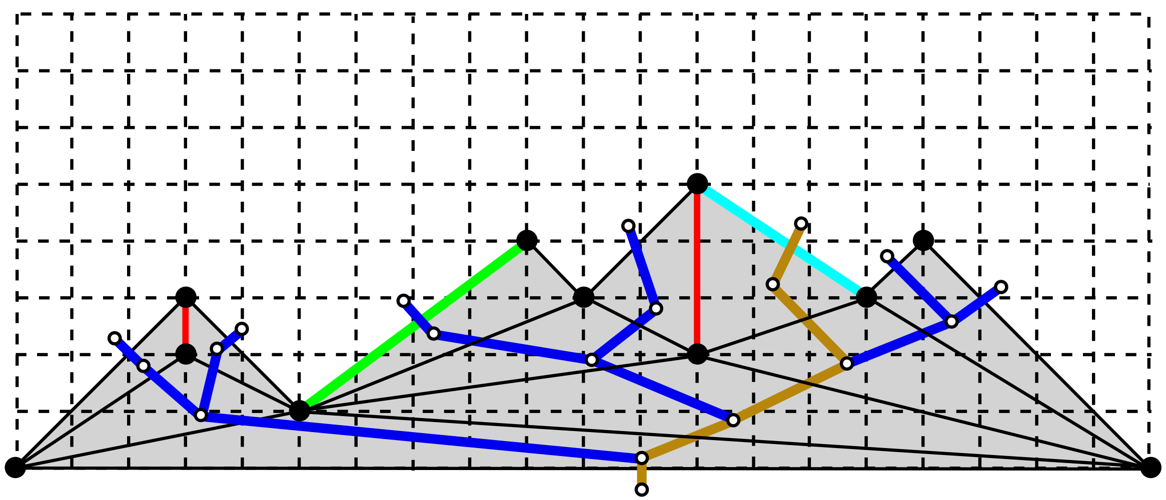


Reformulation of the shift-step

At each step: insert two vertical strips of width 1 using the dual tree

G_{k-1}

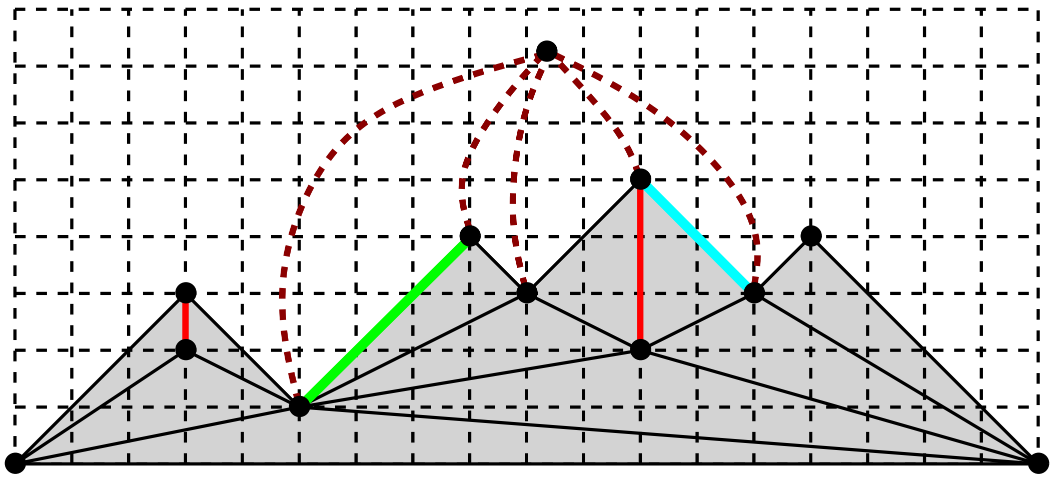


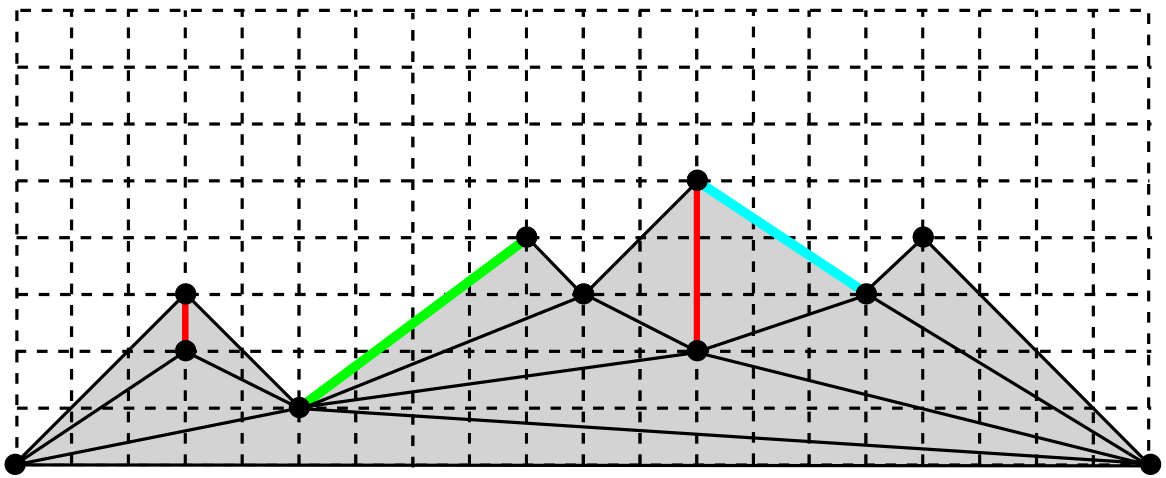


Reformulation of the shift-step

At each step: insert two vertical strips of width 1 using the dual tree

G_{k-1}

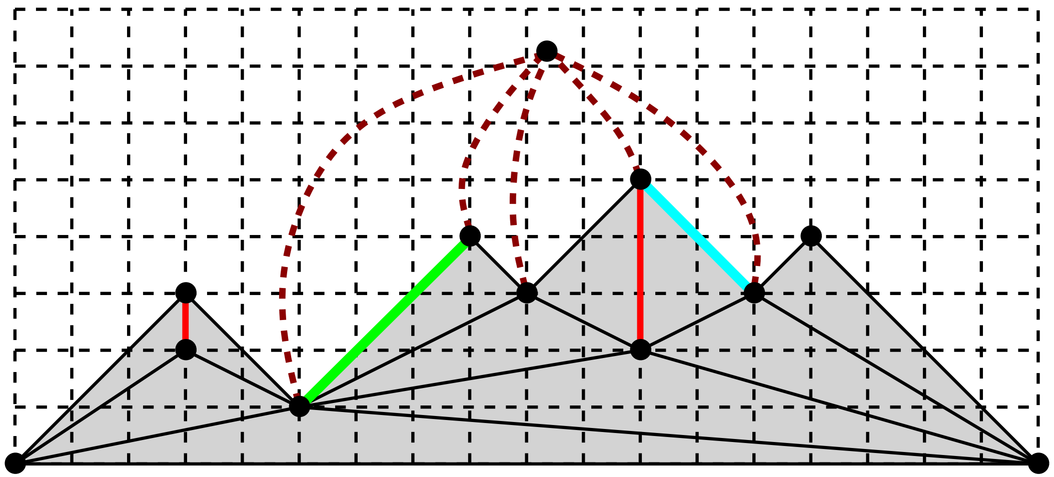




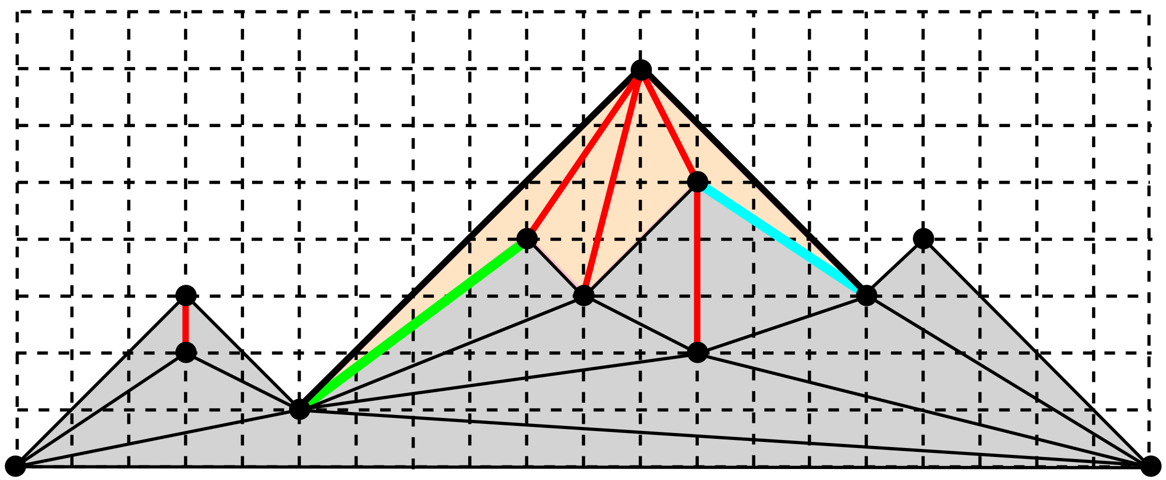
Reformulation of the shift-step

At each step: insert two vertical strips of width 1 using the dual tree

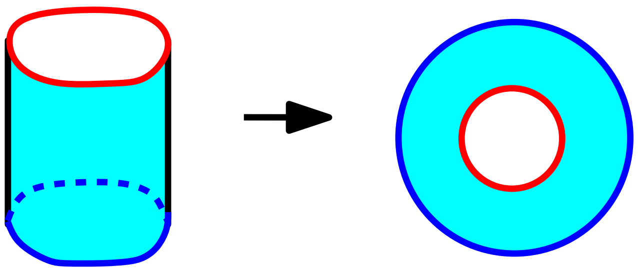
G_{k-1}



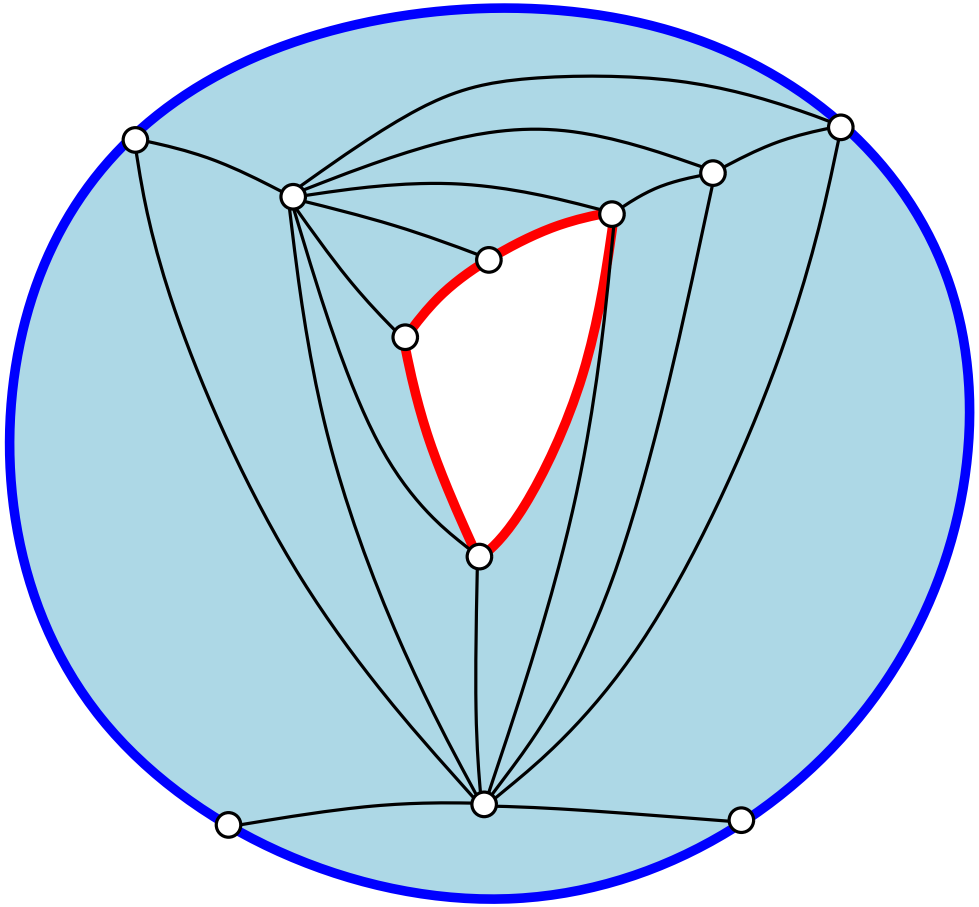
G_k



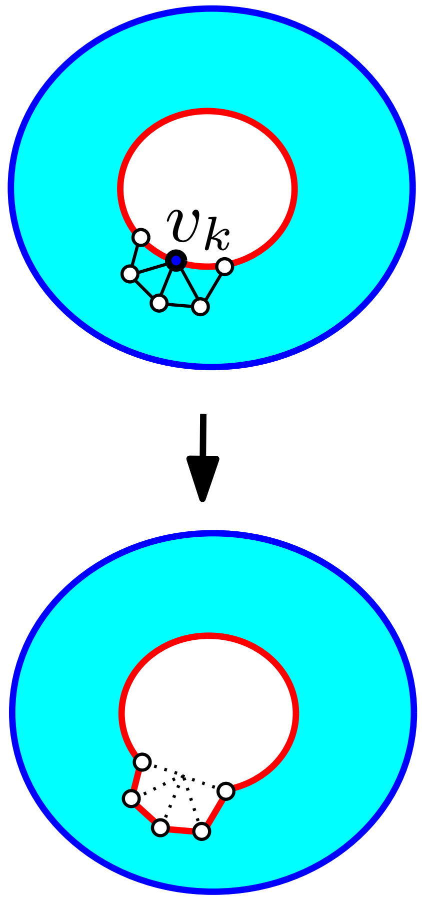
Extension to the cylinder: canonical ordering



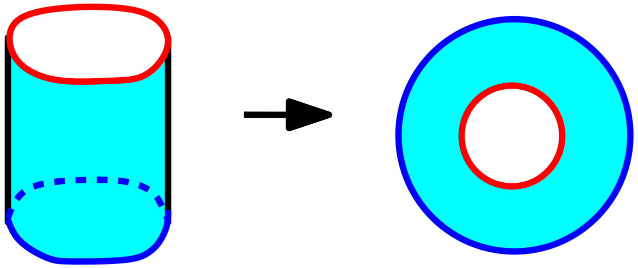
annular representation



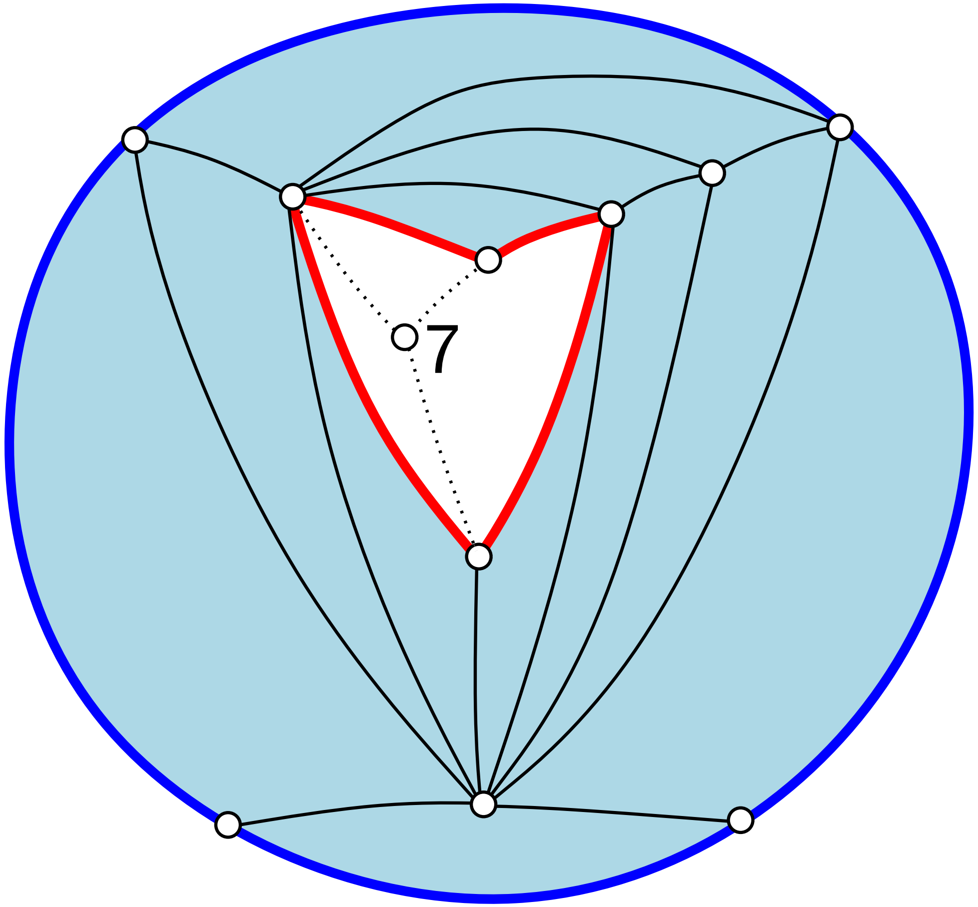
At each step:



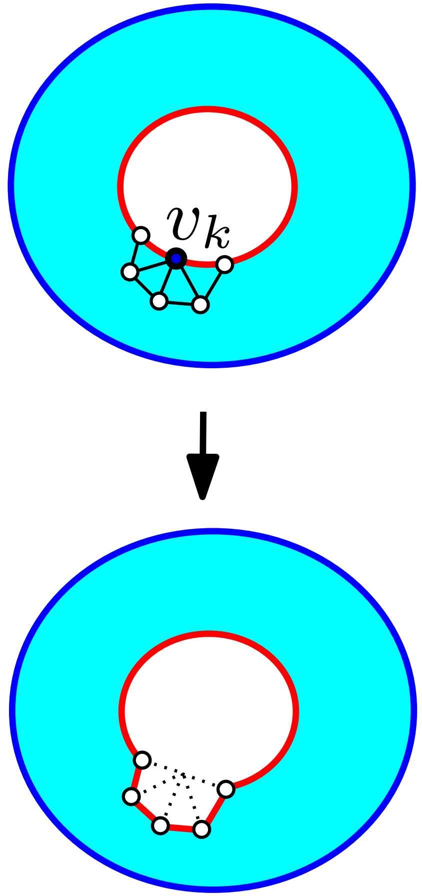
Extension to the cylinder: canonical ordering



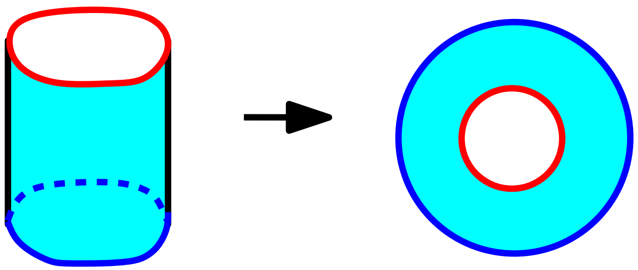
annular representation



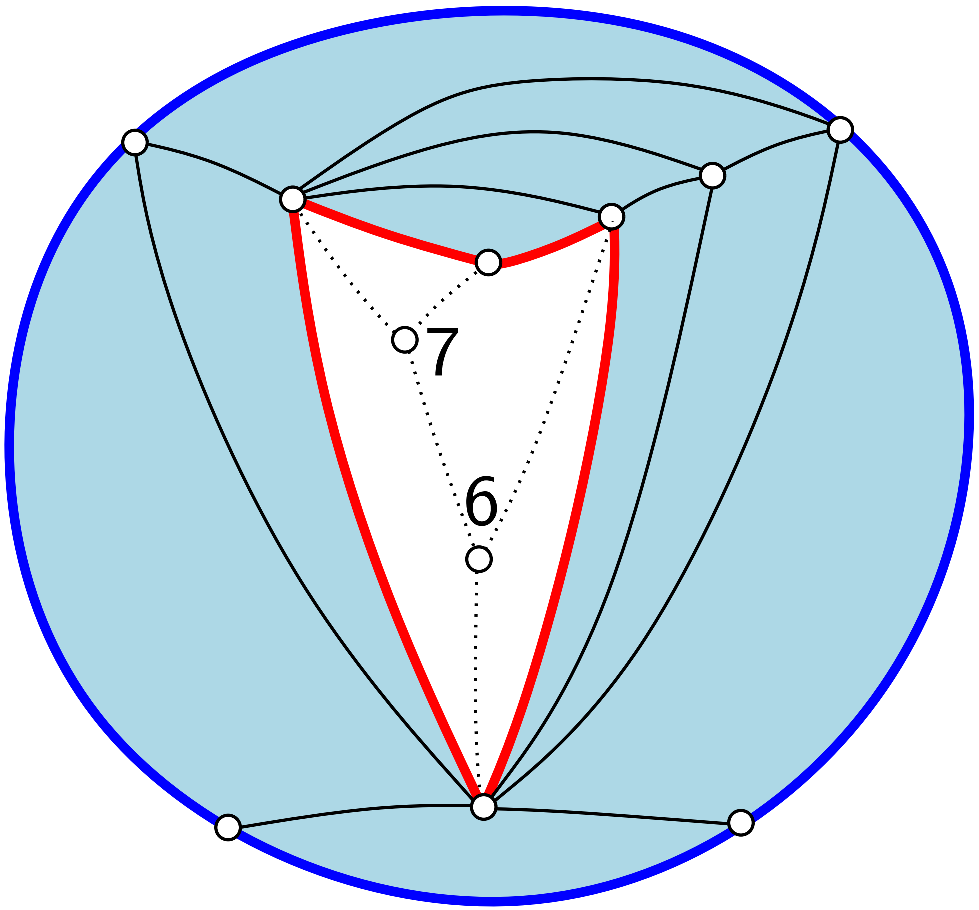
At each step:



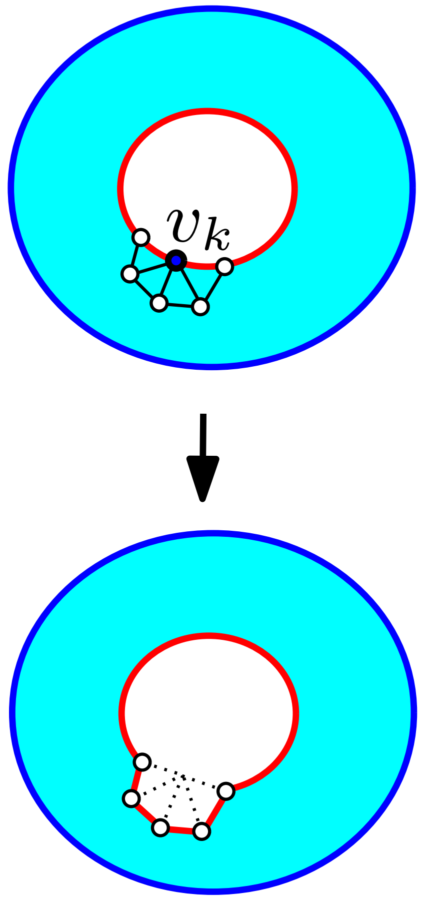
Extension to the cylinder: canonical ordering



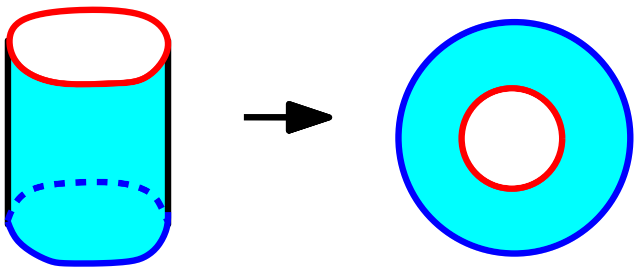
annular representation



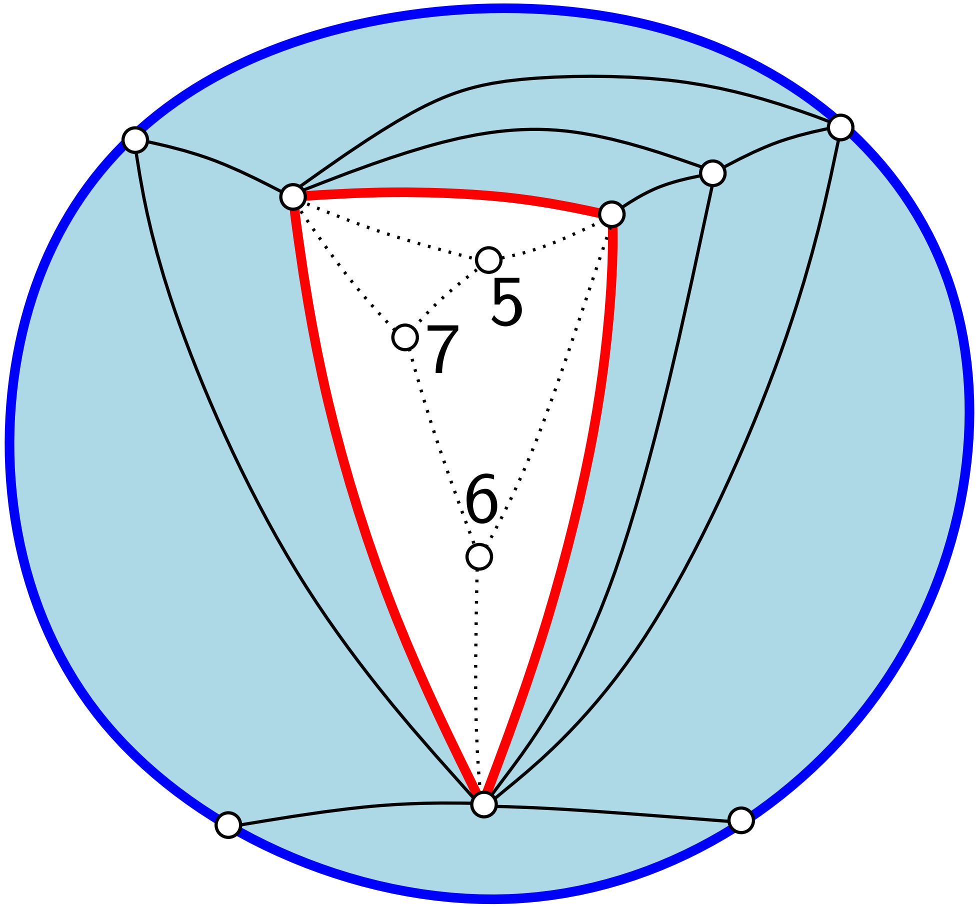
At each step:



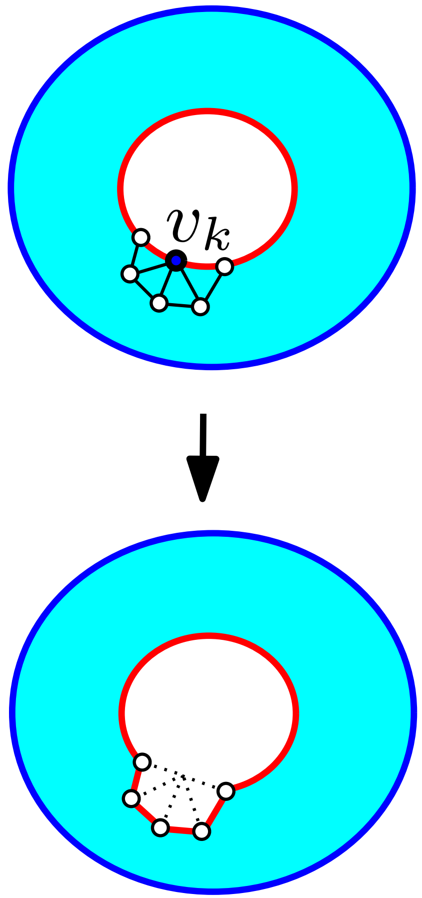
Extension to the cylinder: canonical ordering



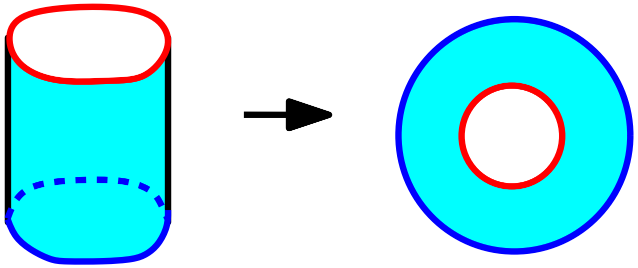
annular representation



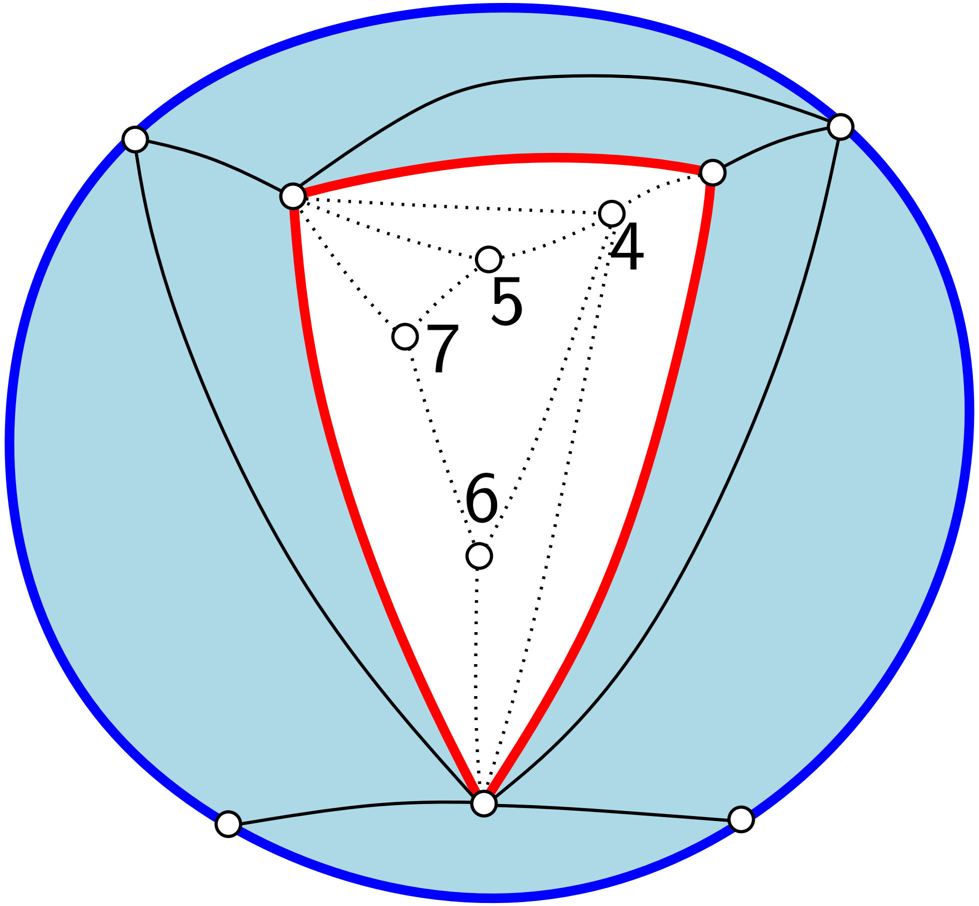
At each step:



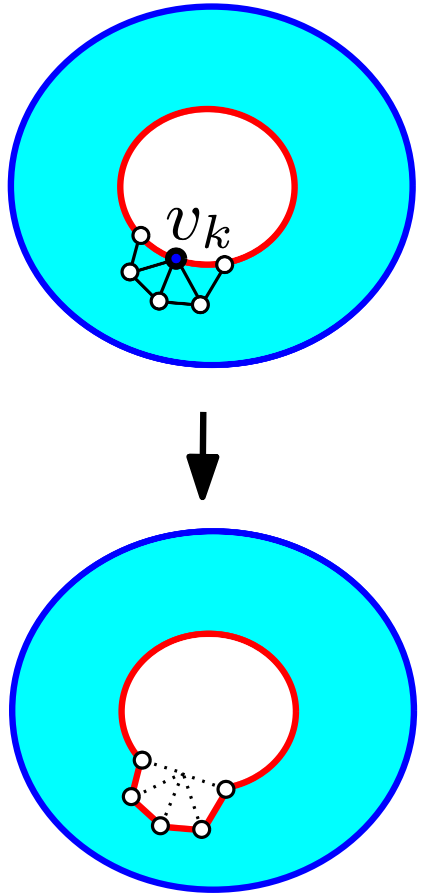
Extension to the cylinder: canonical ordering



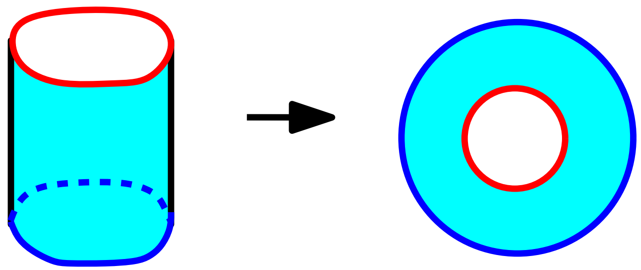
annular representation



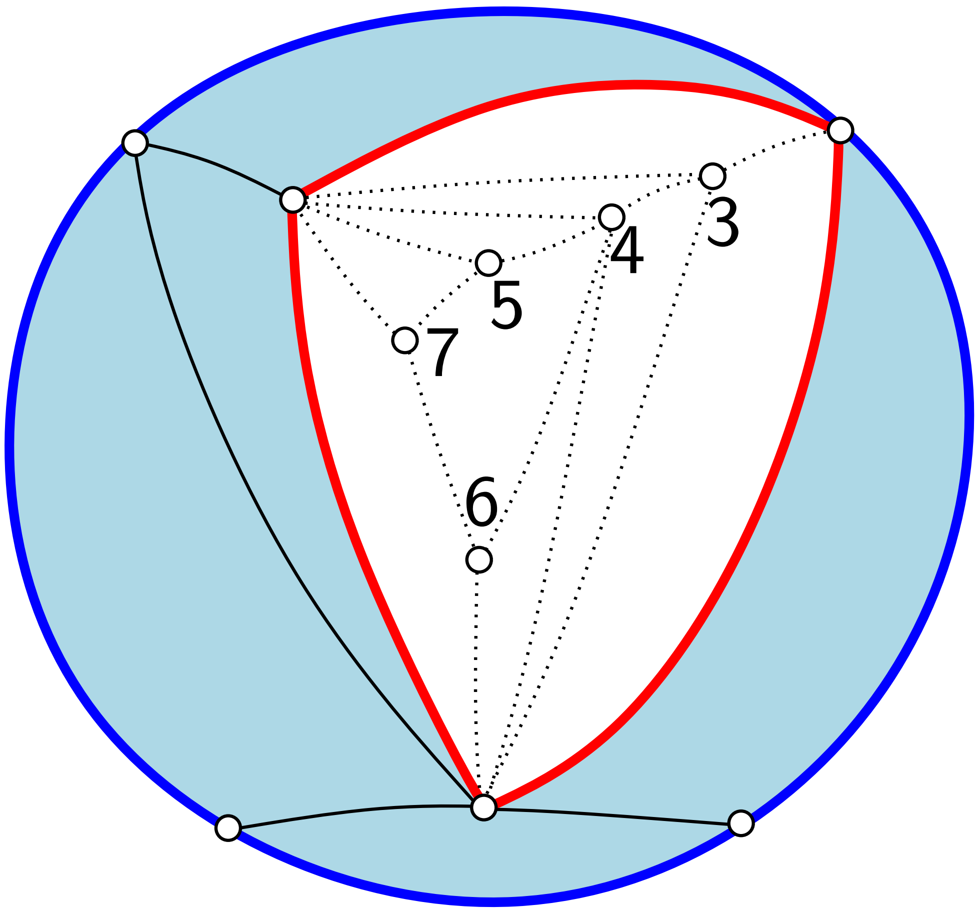
At each step:



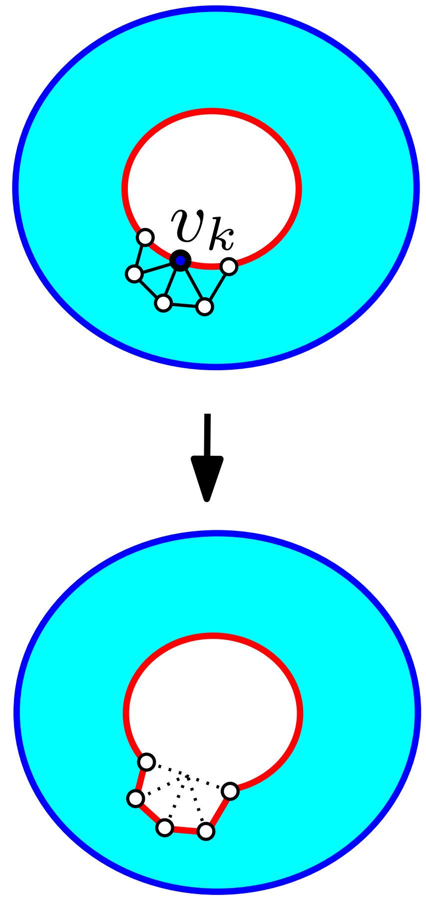
Extension to the cylinder: canonical ordering



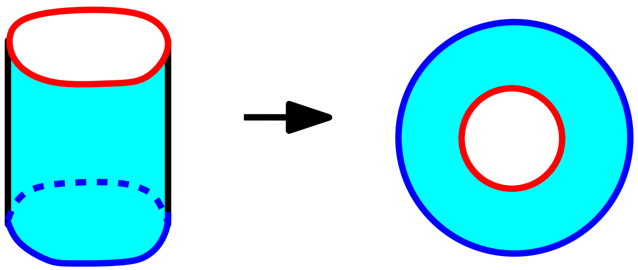
annular representation



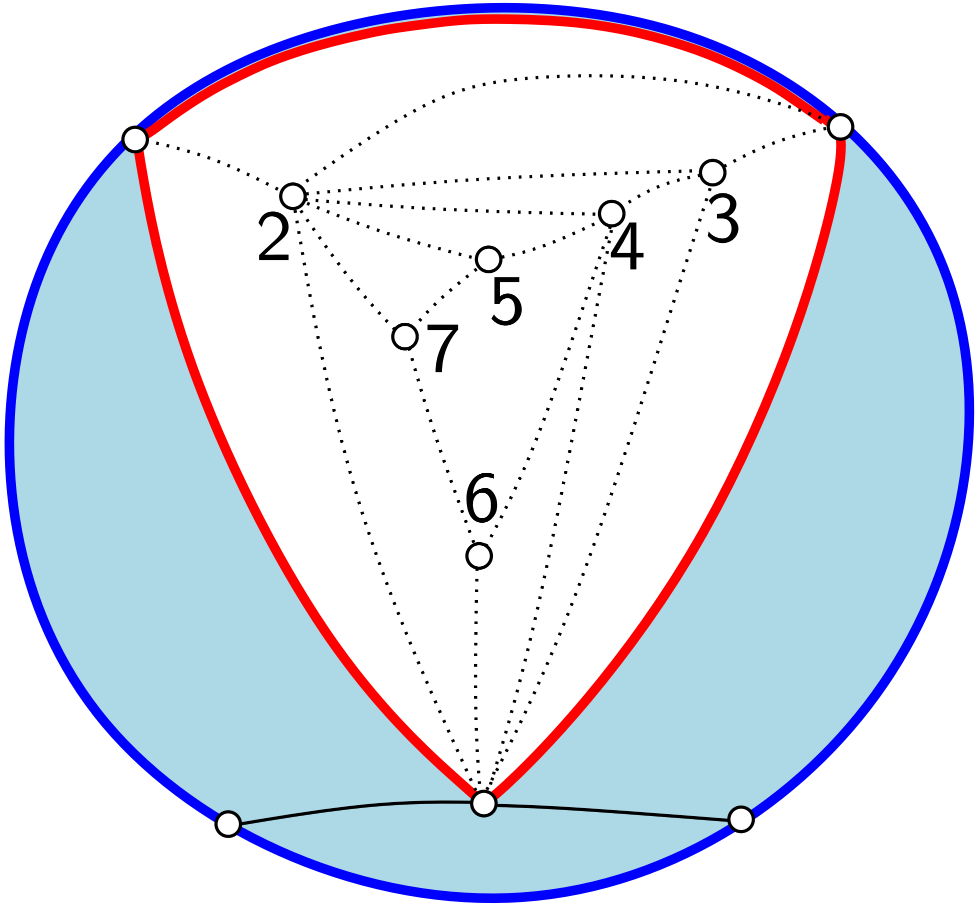
At each step:



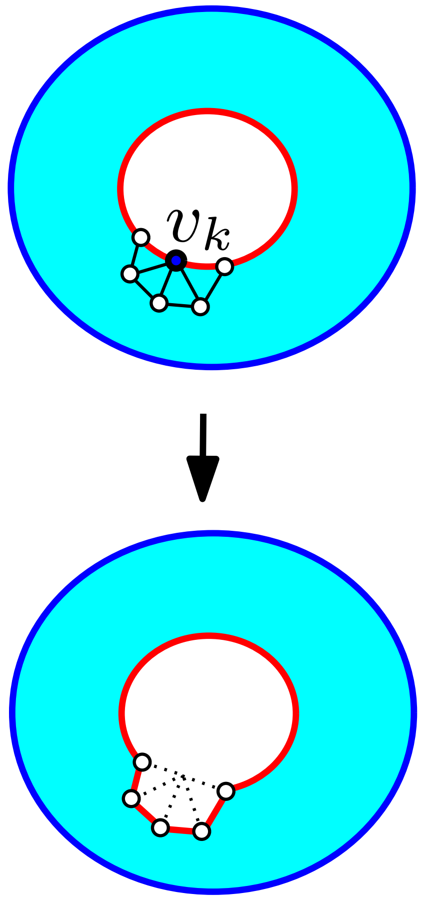
Extension to the cylinder: canonical ordering



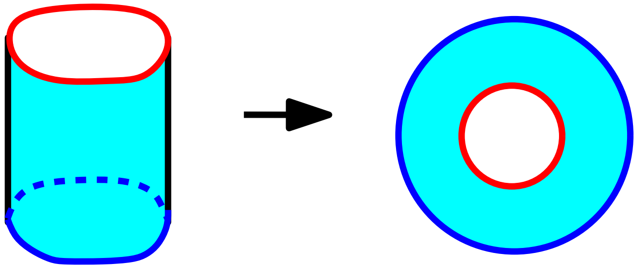
annular representation



At each step:

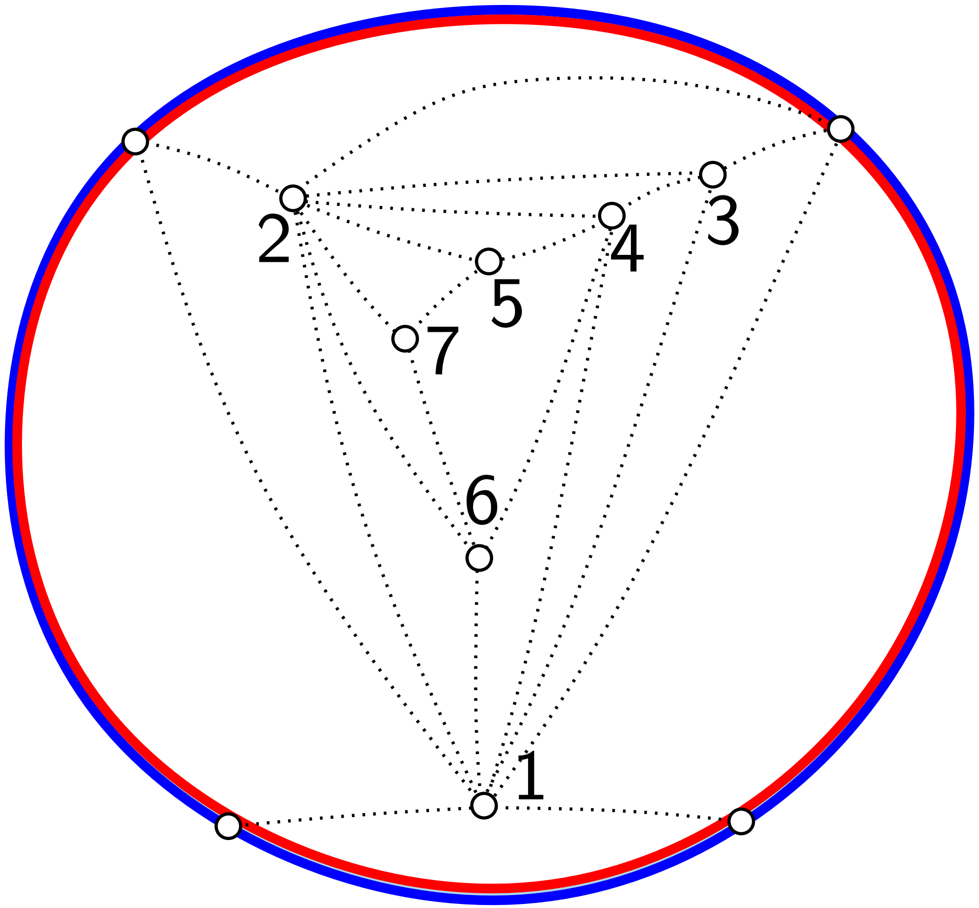


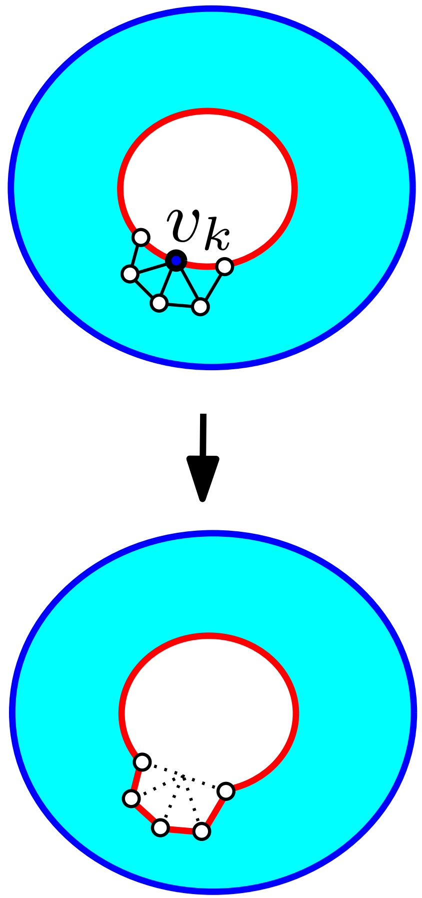
Extension to the cylinder: canonical ordering



annular representation

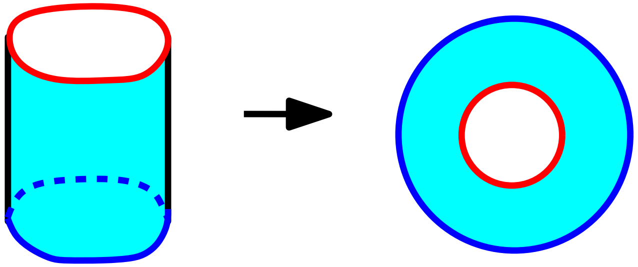
At each step:



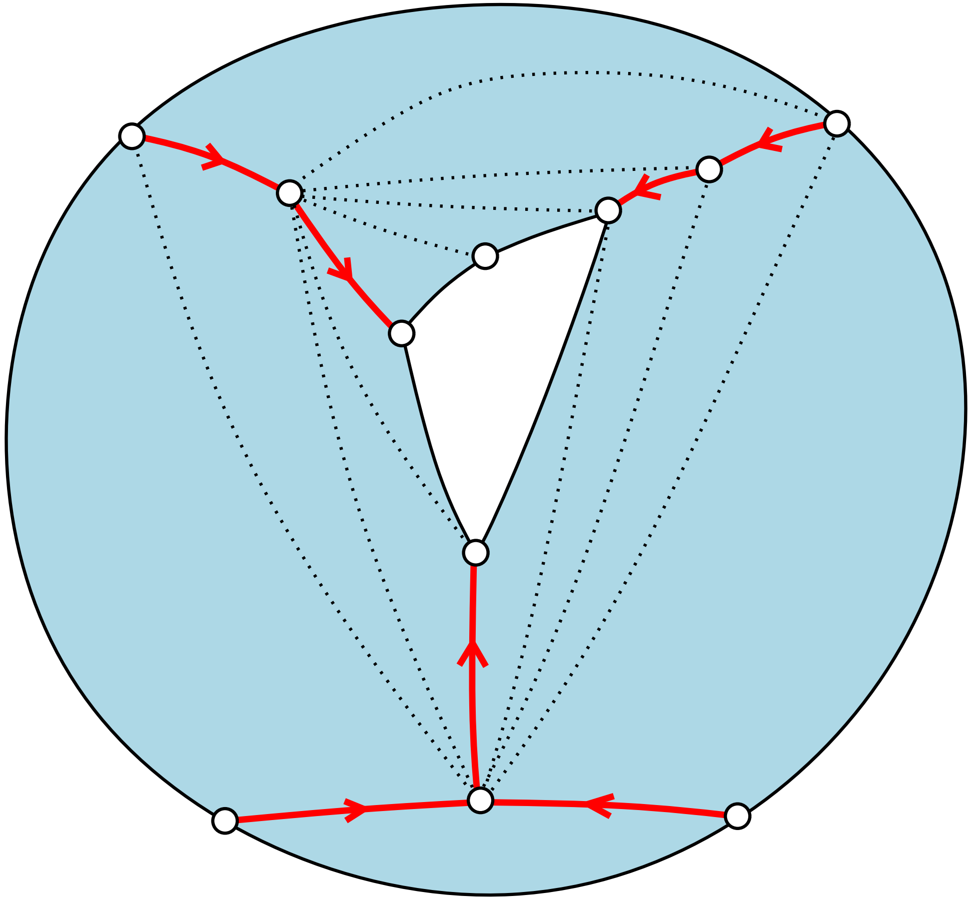


Terminates if no chordal edge incident to bottom cycle

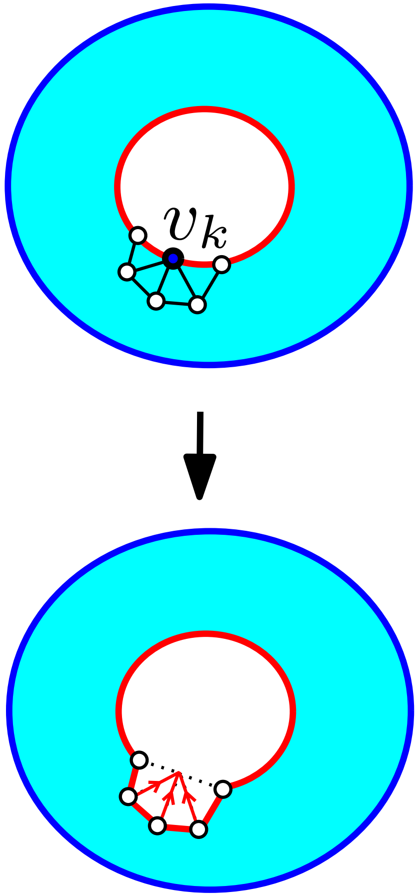
Extension to the cylinder: canonical ordering



annular representation

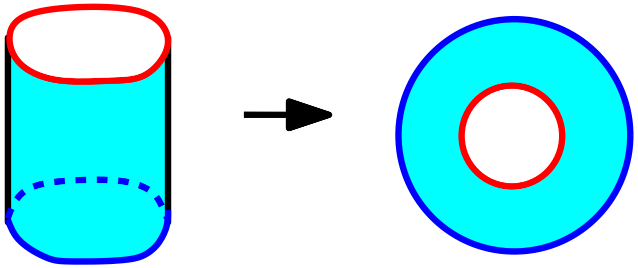


At each step:

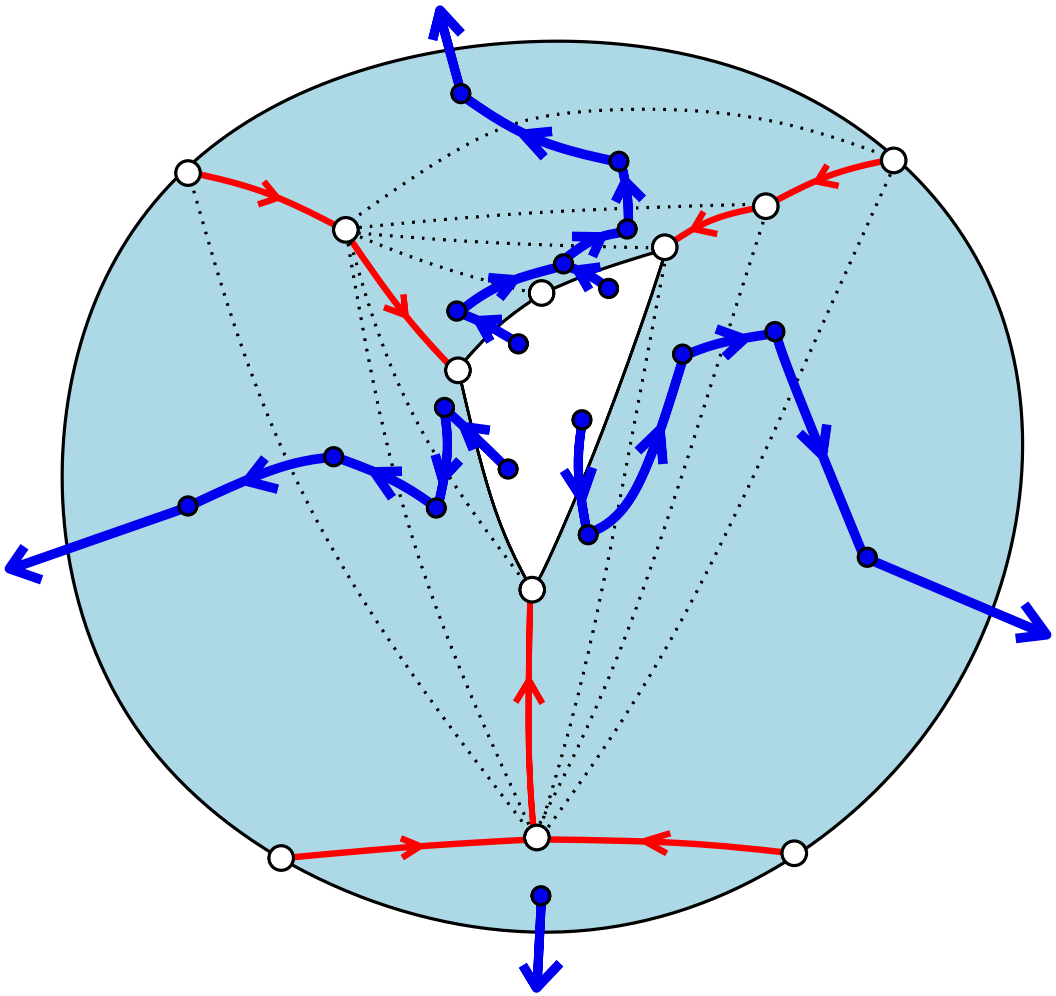


Underlying forest

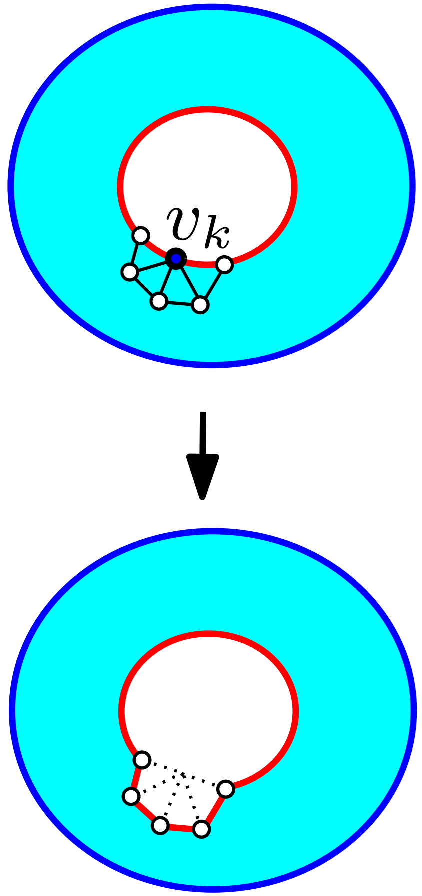
Extension to the cylinder: canonical ordering



annular representation

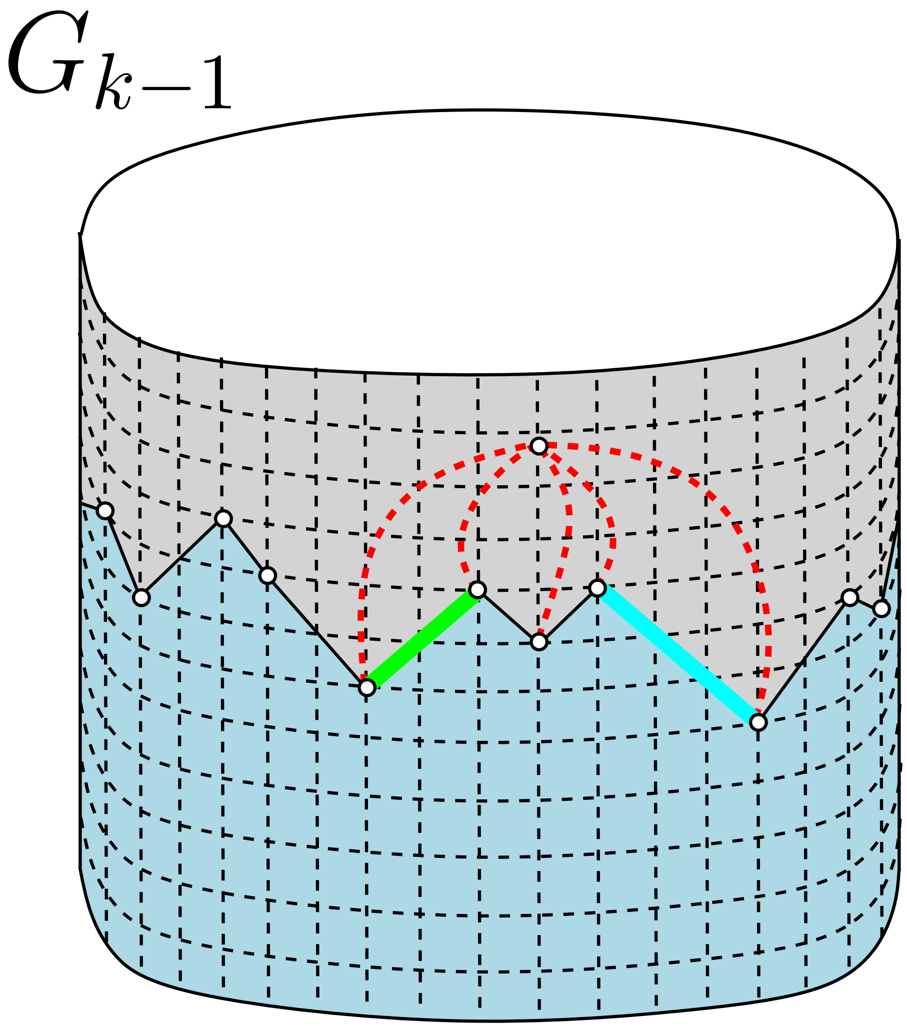


At each step:



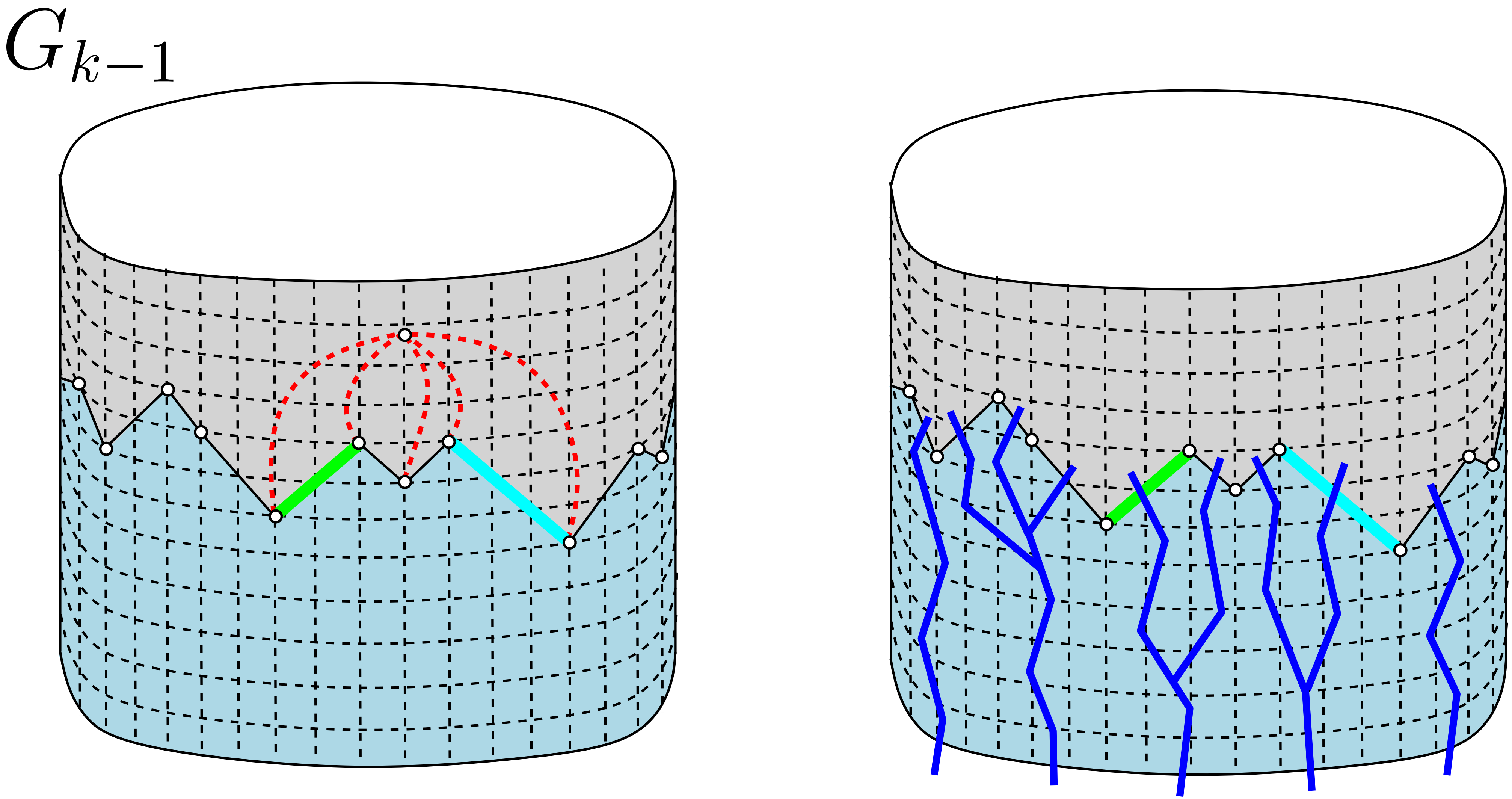
Underlying forest + dual forest

Extension to the cylinder: drawing algorithm



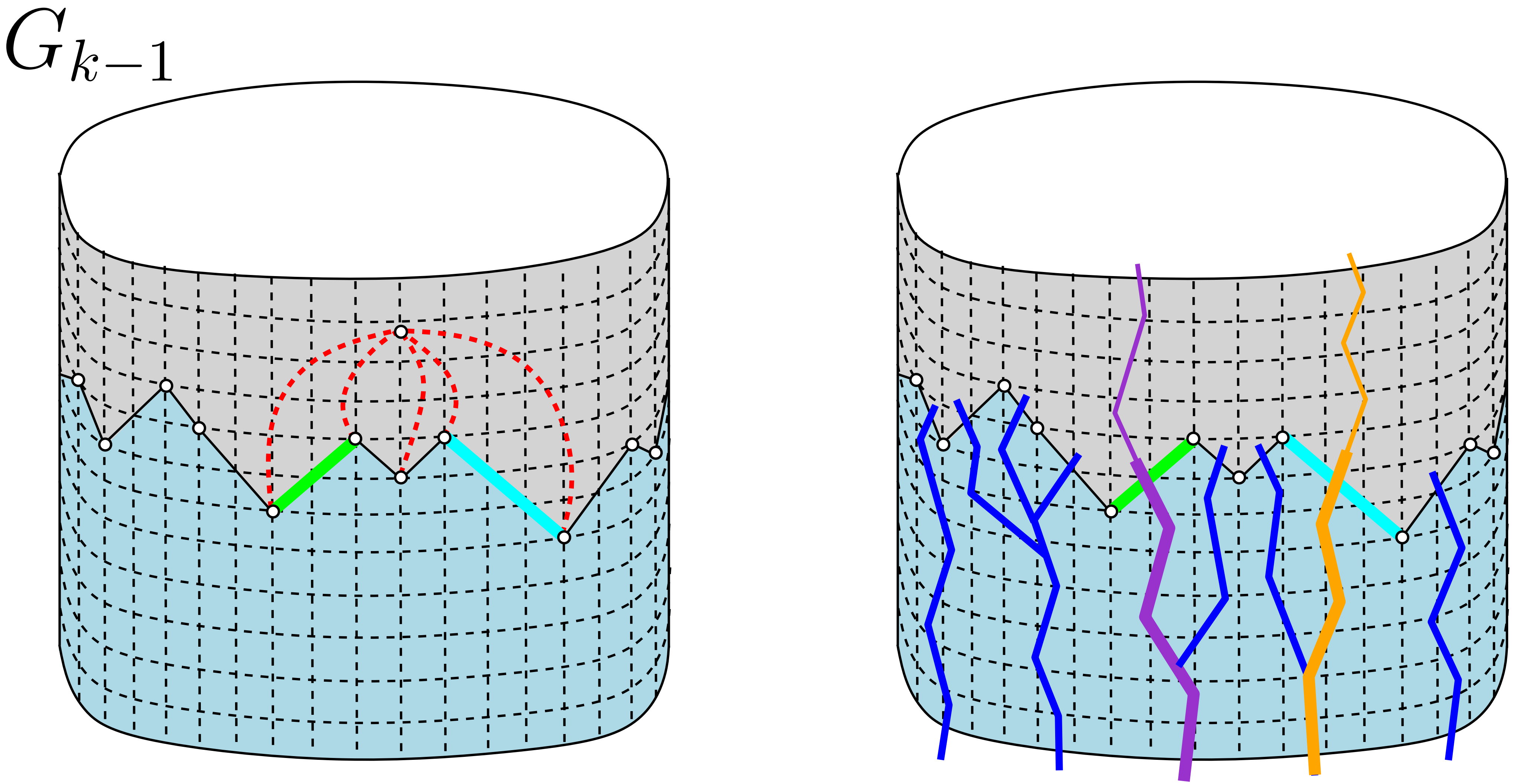
At each step: - insert two vertical strips of width 1
- insert next vertex as in the planar case

Extension to the cylinder: drawing algorithm



At each step: - insert two vertical strips of width 1
- insert next vertex as in the planar case

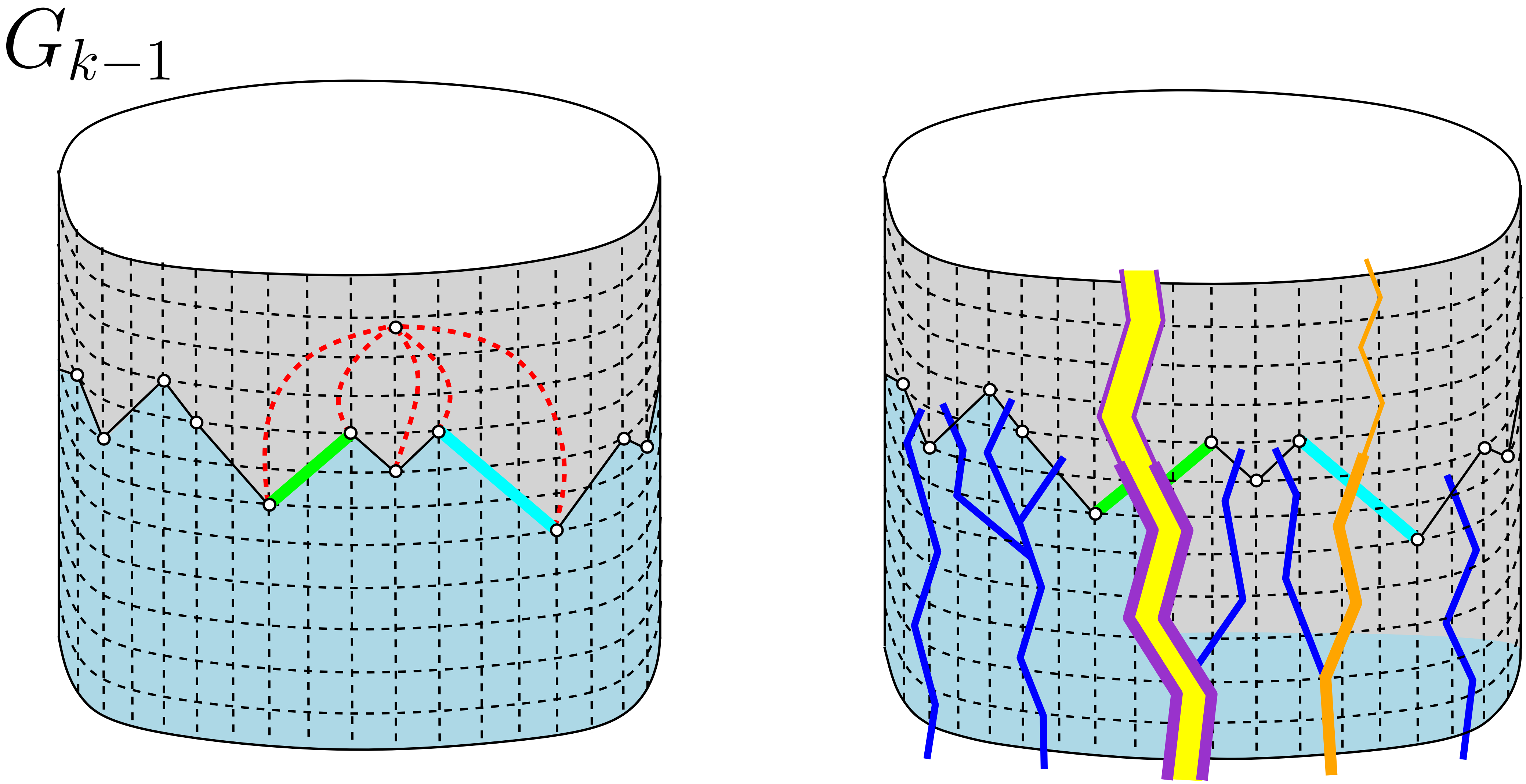
Extension to the cylinder: drawing algorithm



At each step: - insert two vertical strips of width 1

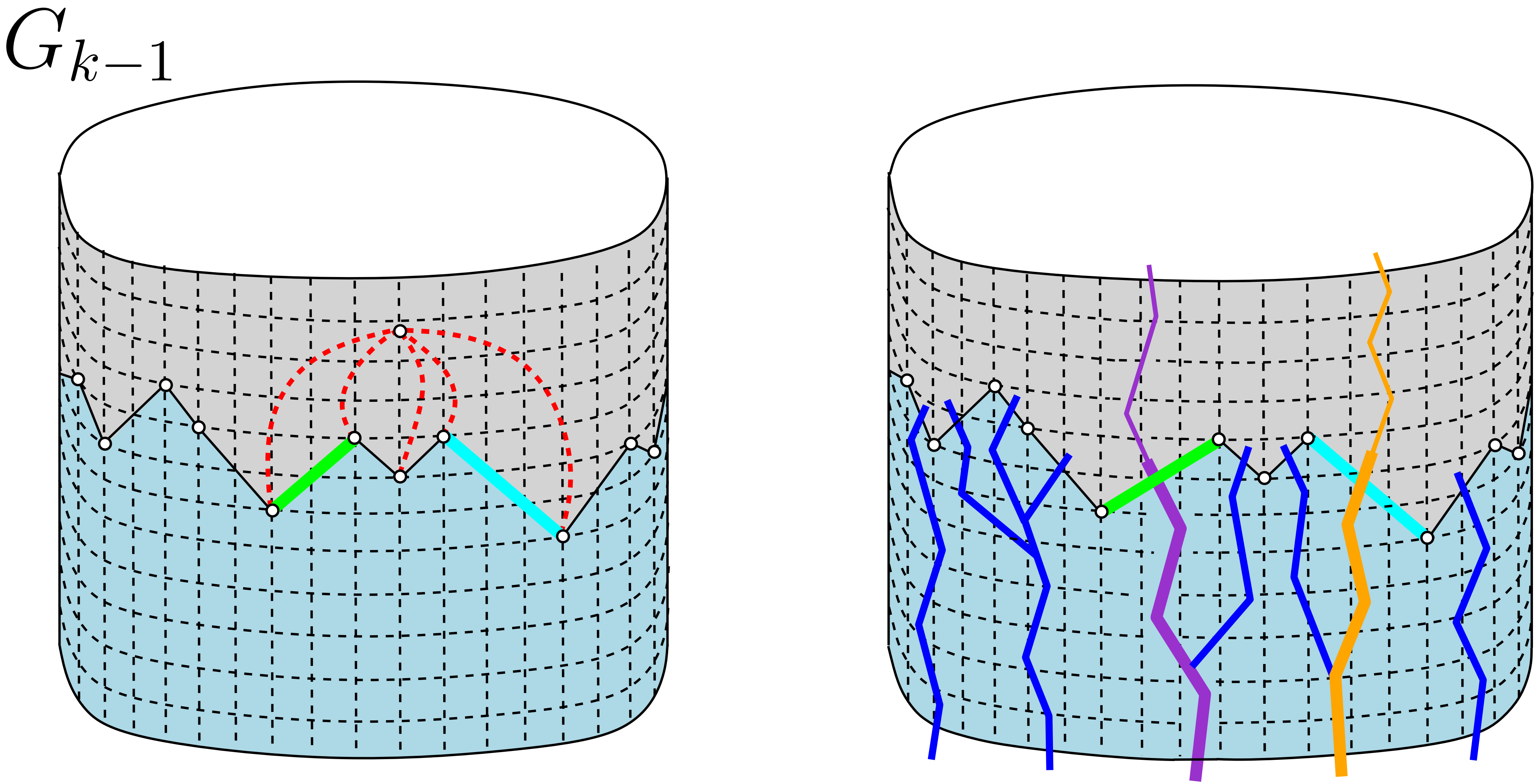
- insert next vertex as in the planar case

Extension to the cylinder: drawing algorithm



At each step: - insert two vertical strips of width 1
- insert next vertex as in the planar case

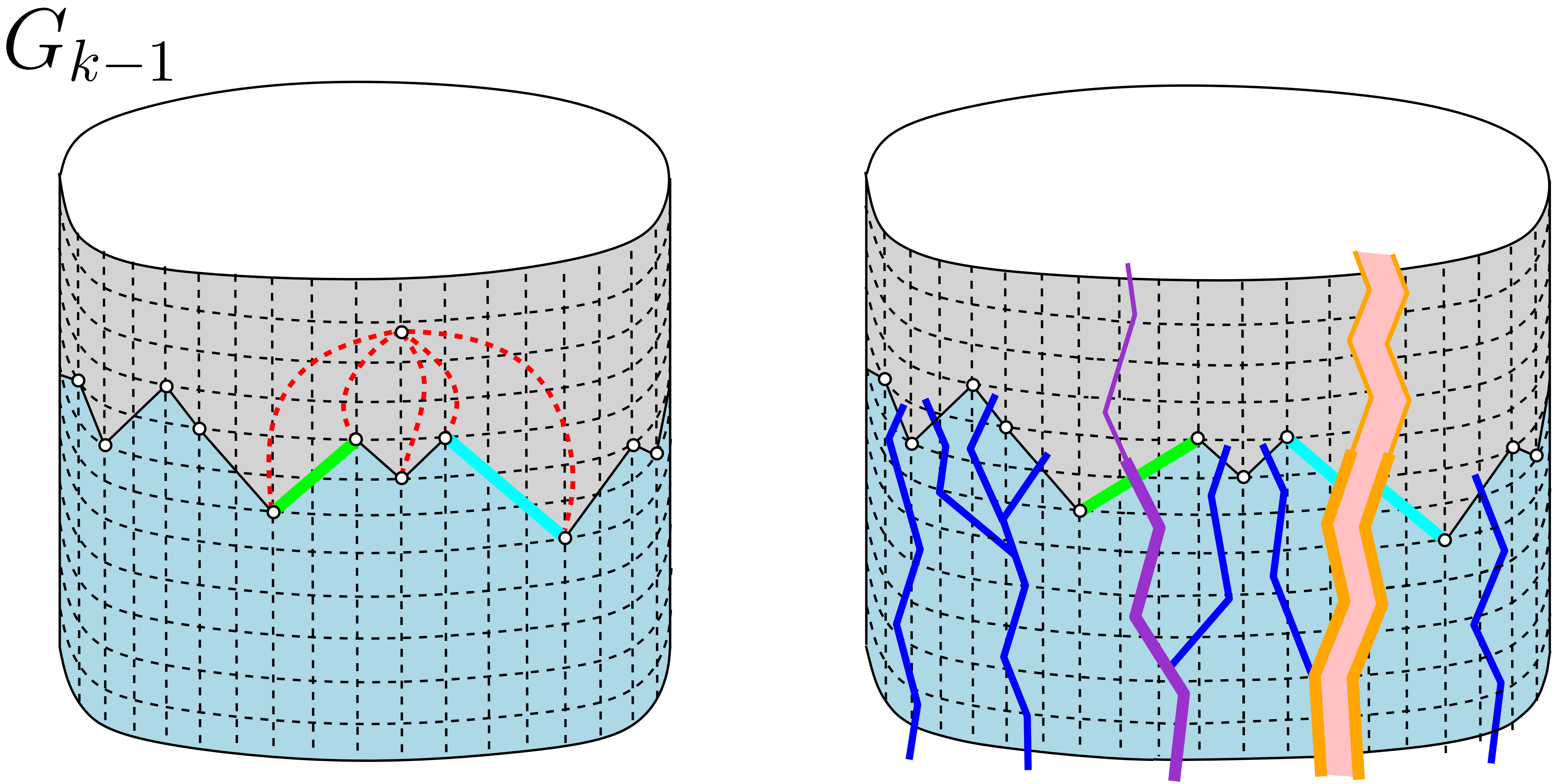
Extension to the cylinder: drawing algorithm



At each step: - insert two vertical strips of width 1

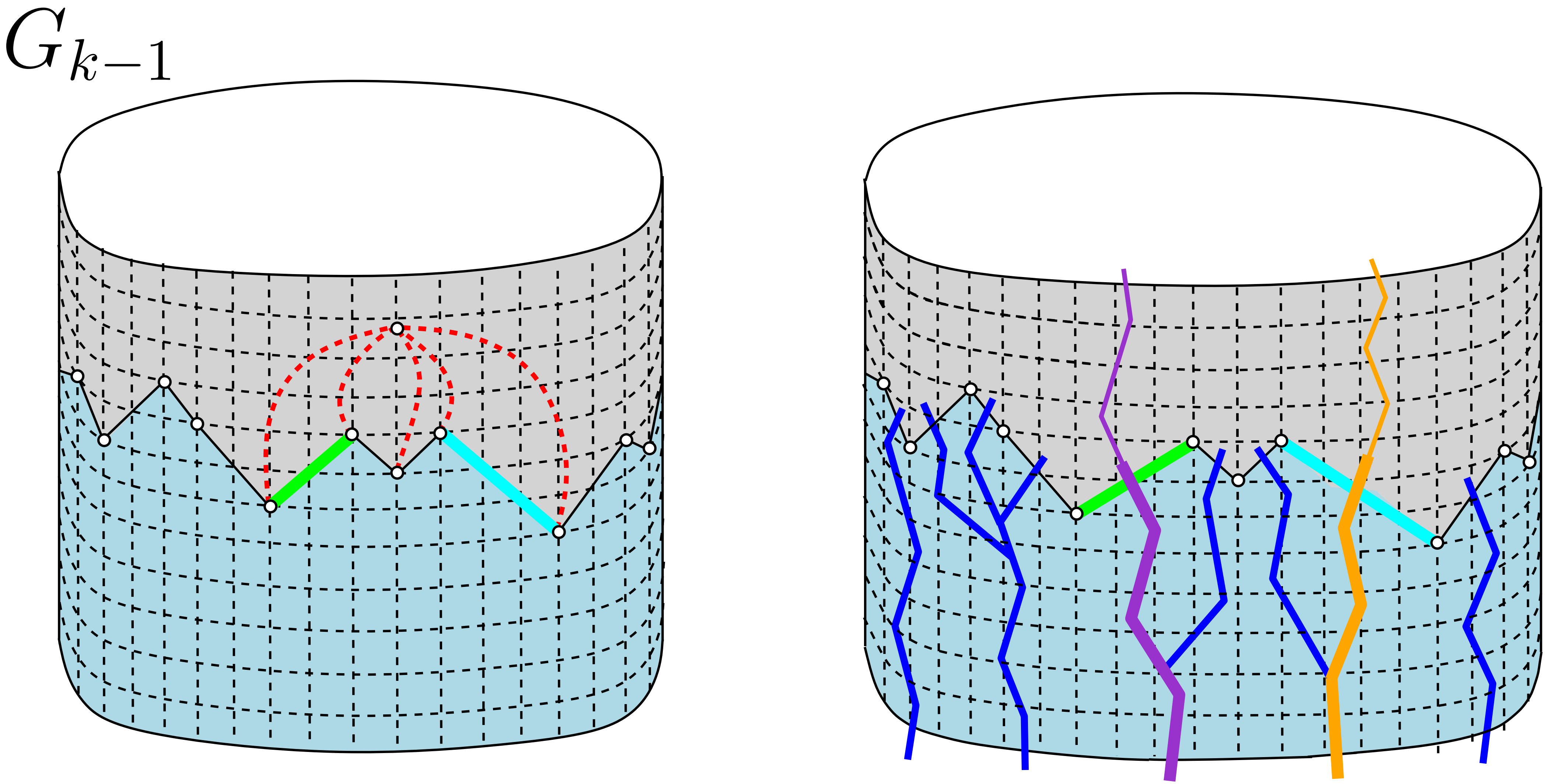
- insert next vertex as in the planar case

Extension to the cylinder: drawing algorithm



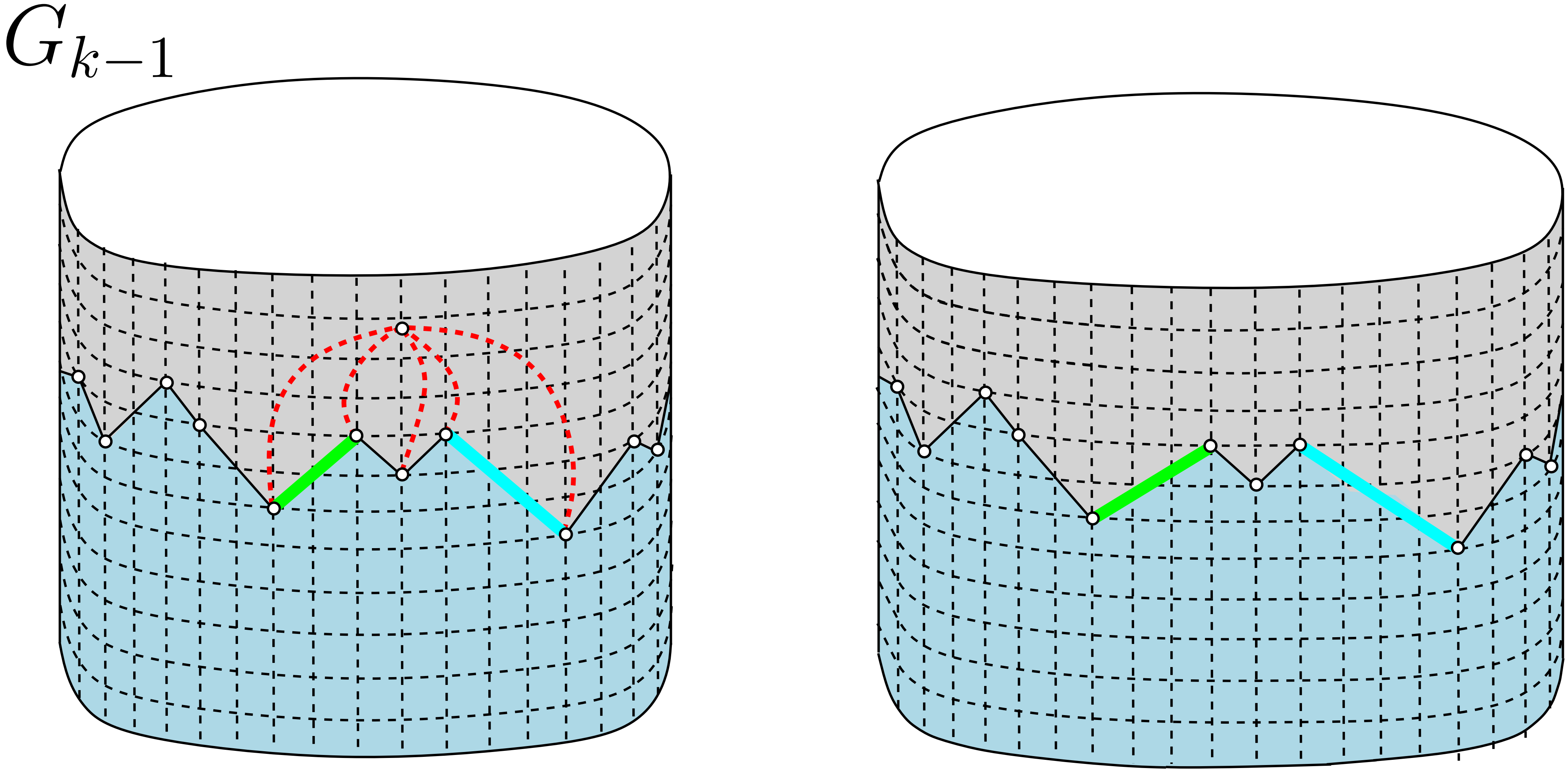
At each step: - insert two vertical strips of width 1
- insert next vertex as in the planar case

Extension to the cylinder: drawing algorithm



At each step: - insert two vertical strips of width 1
- insert next vertex as in the planar case

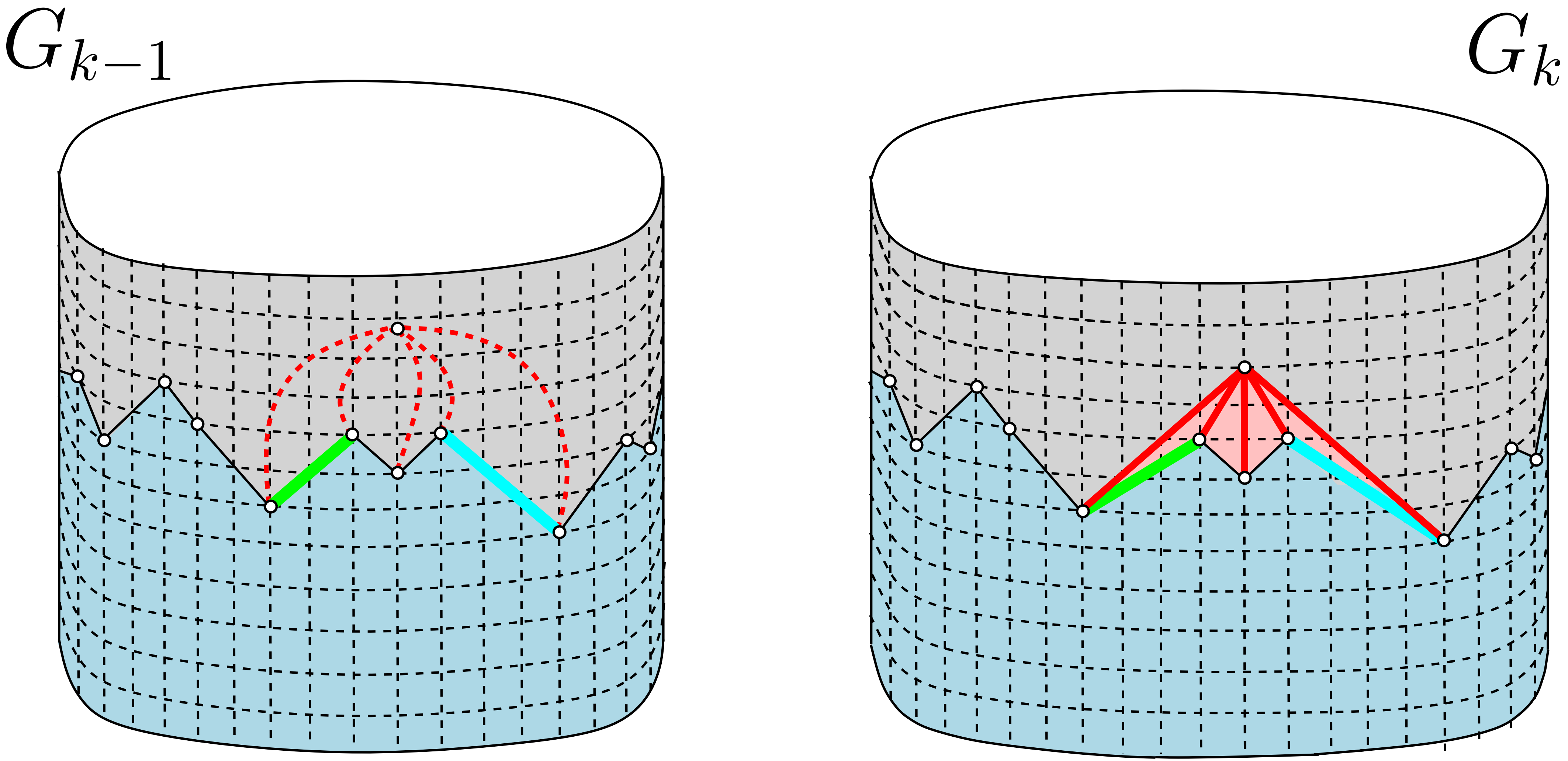
Extension to the cylinder: drawing algorithm



At each step: - insert two vertical strips of width 1

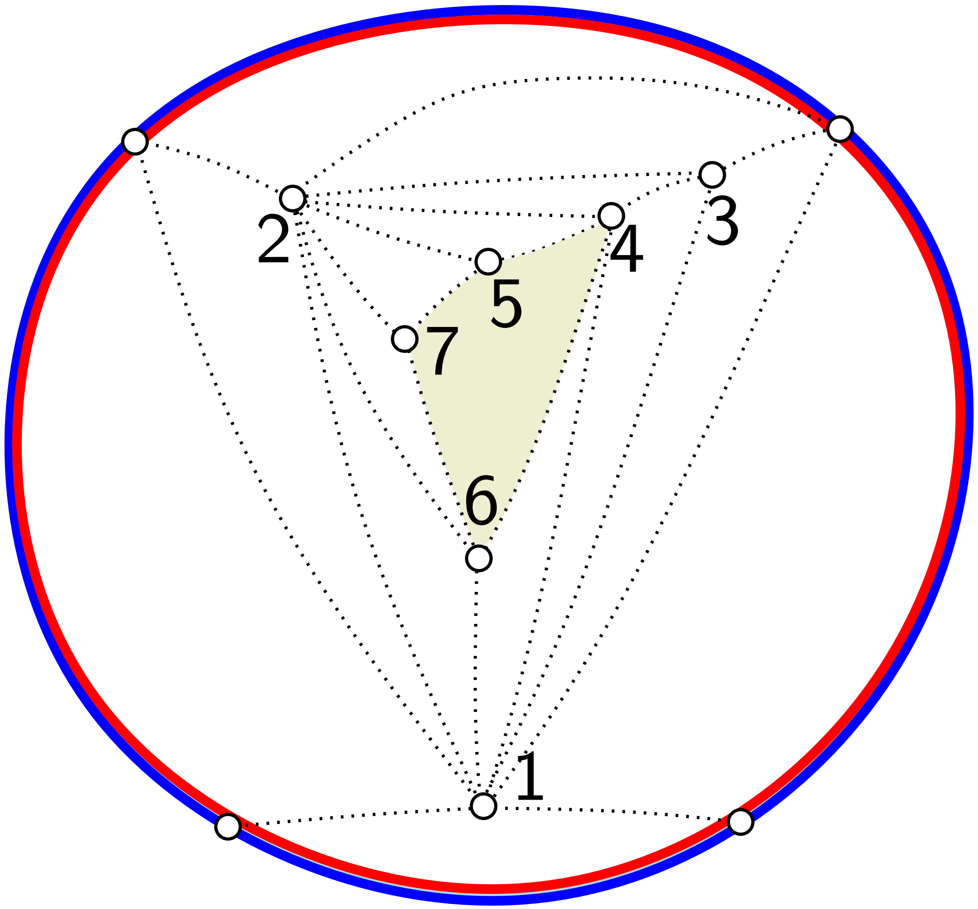
- insert next vertex as in the planar case

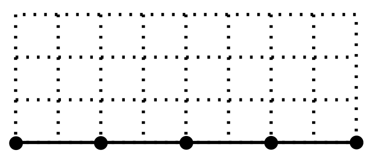
Extension to the cylinder: drawing algorithm



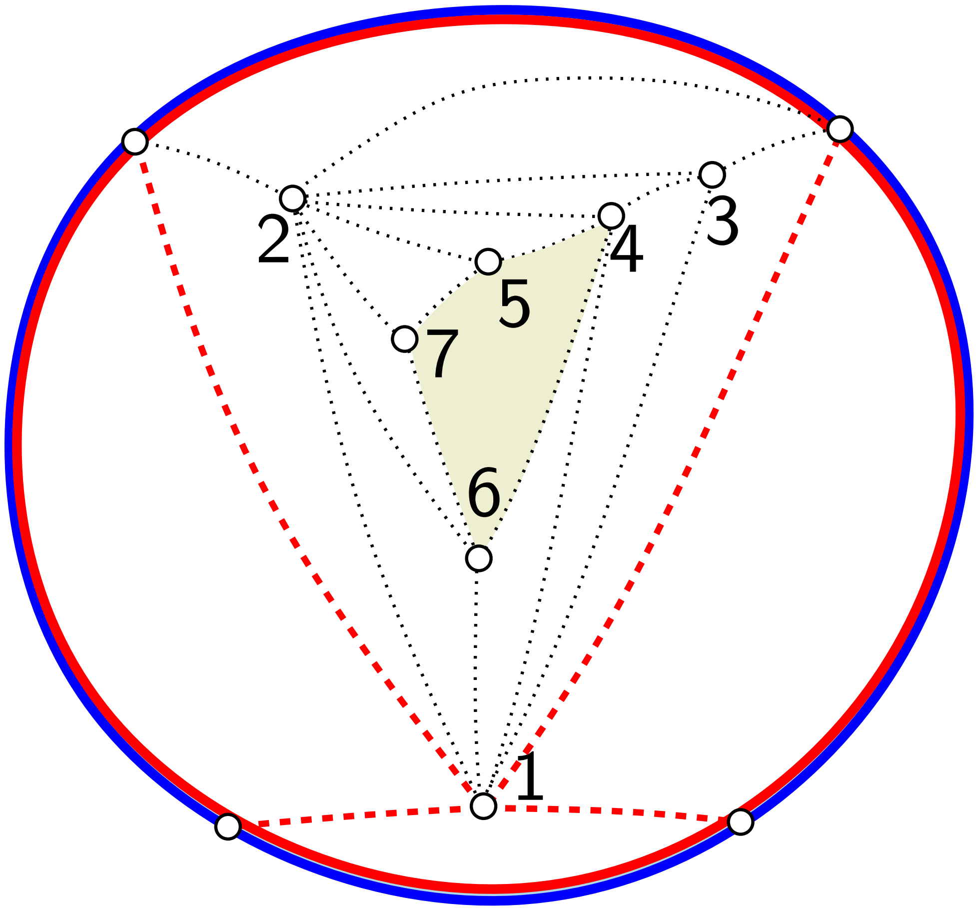
At each step: - insert two vertical strips of width 1
- insert next vertex as in the planar case

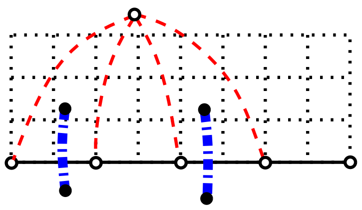
Execution on an example



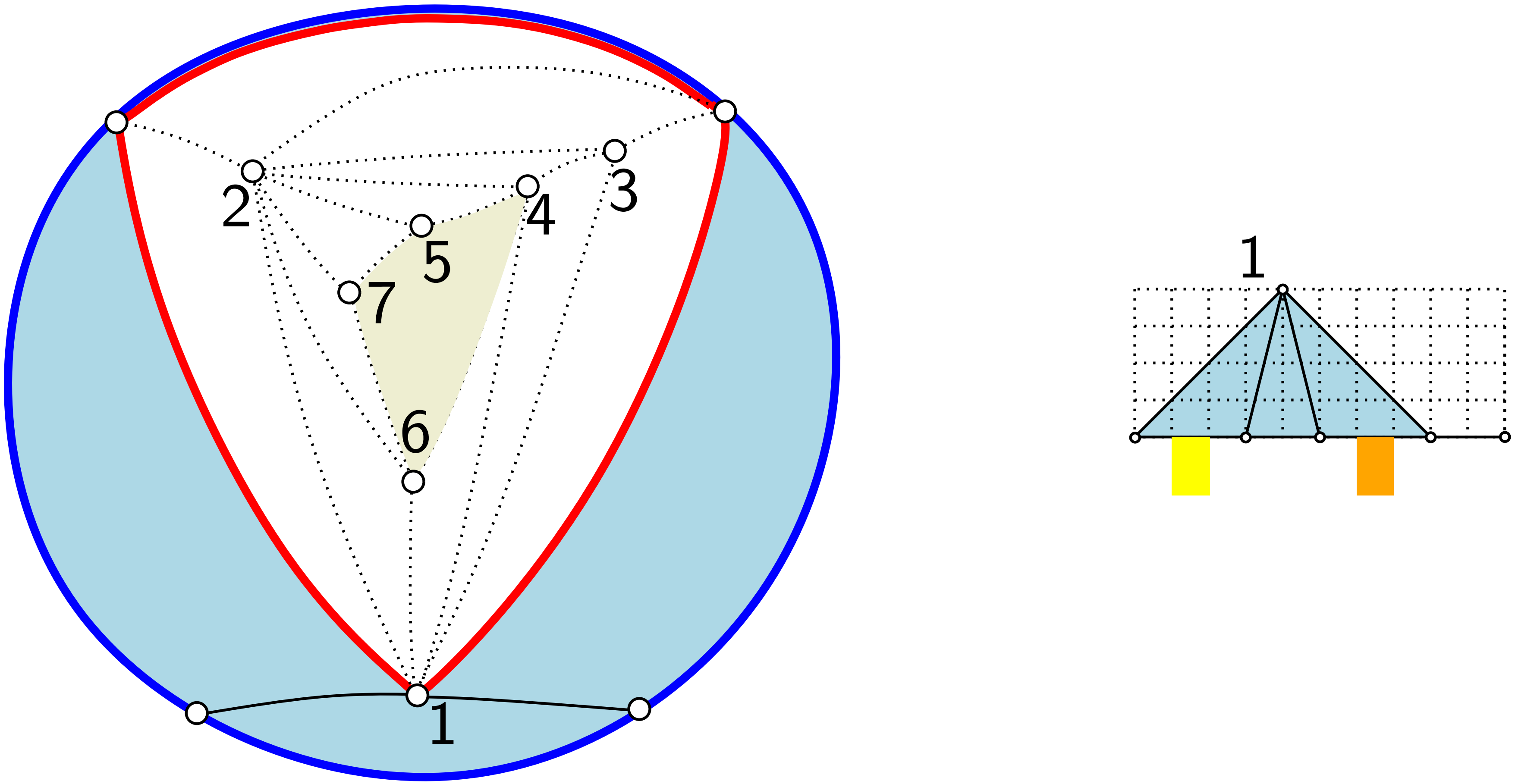


Execution on an example

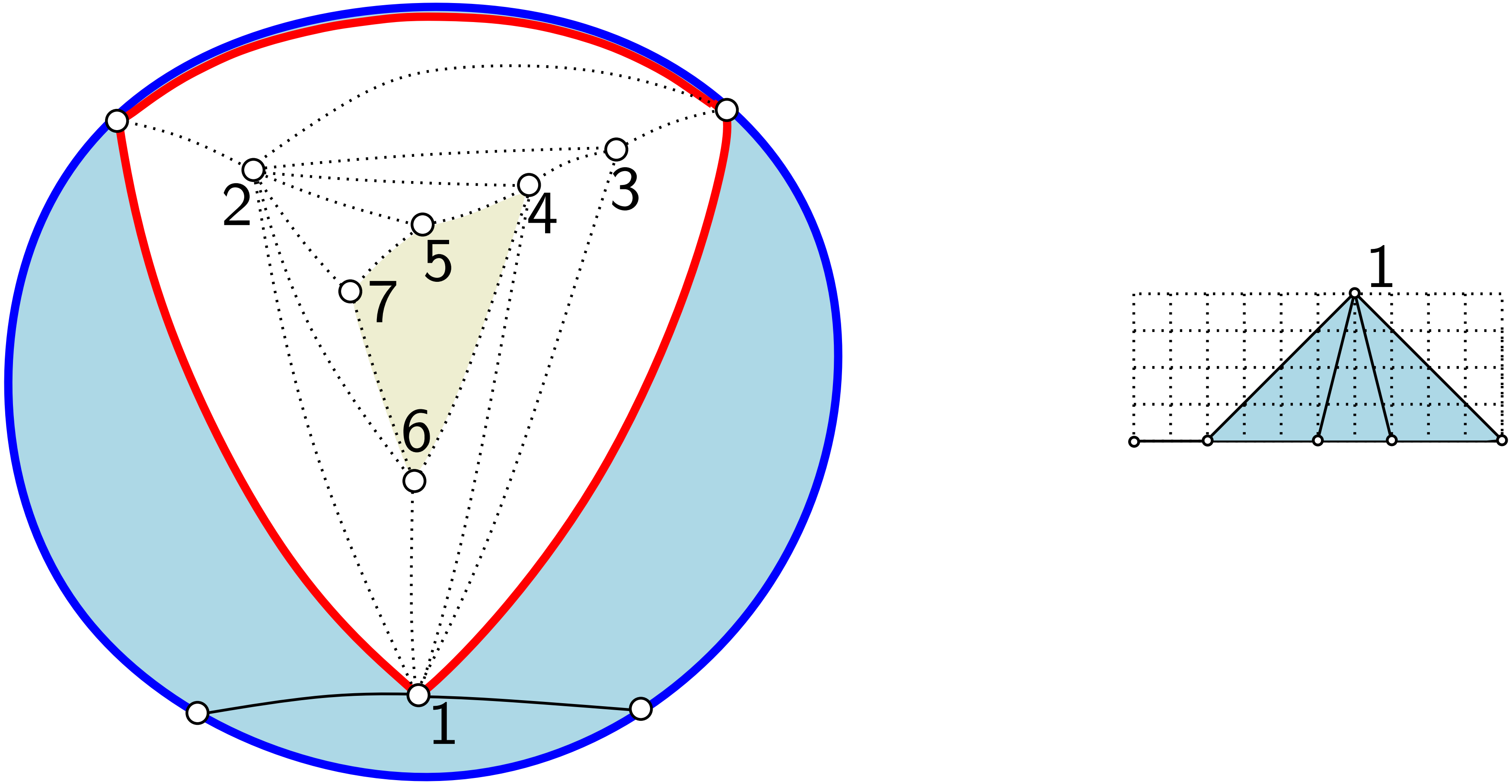




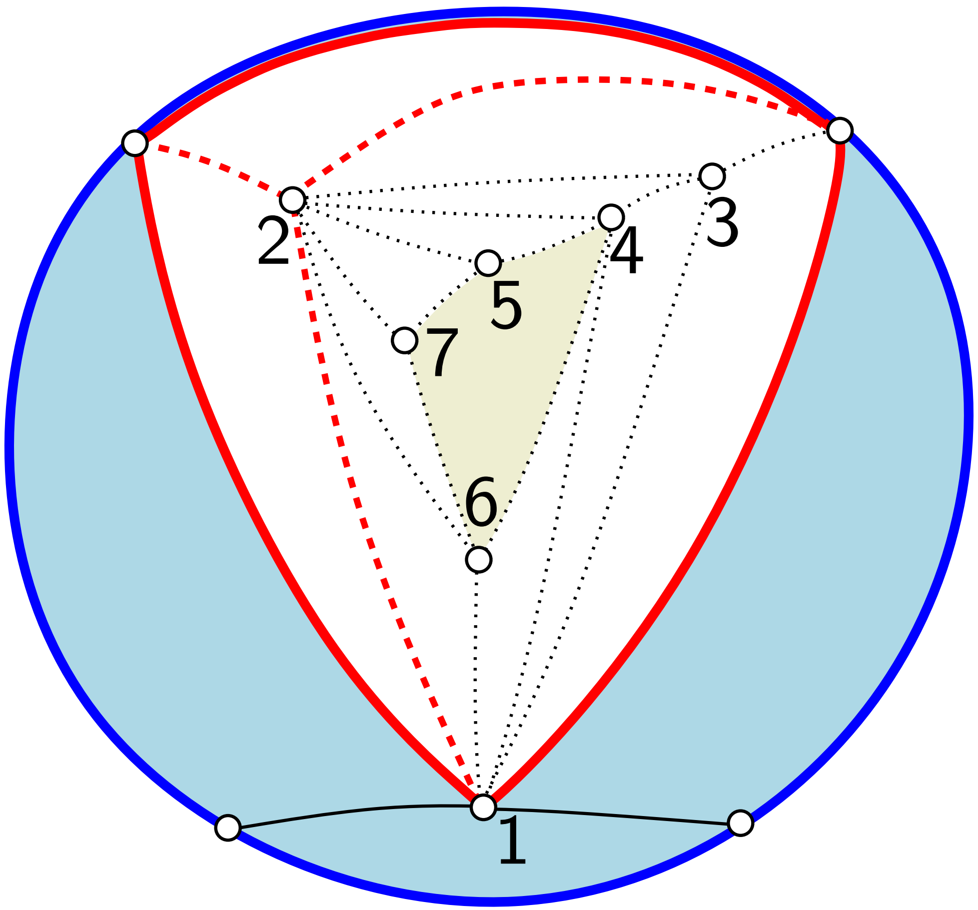
Execution on an example

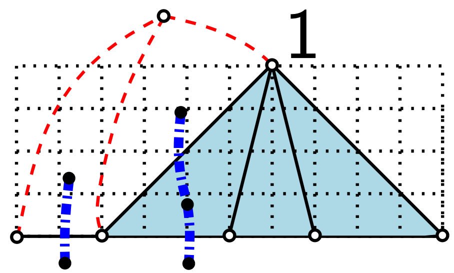


Execution on an example

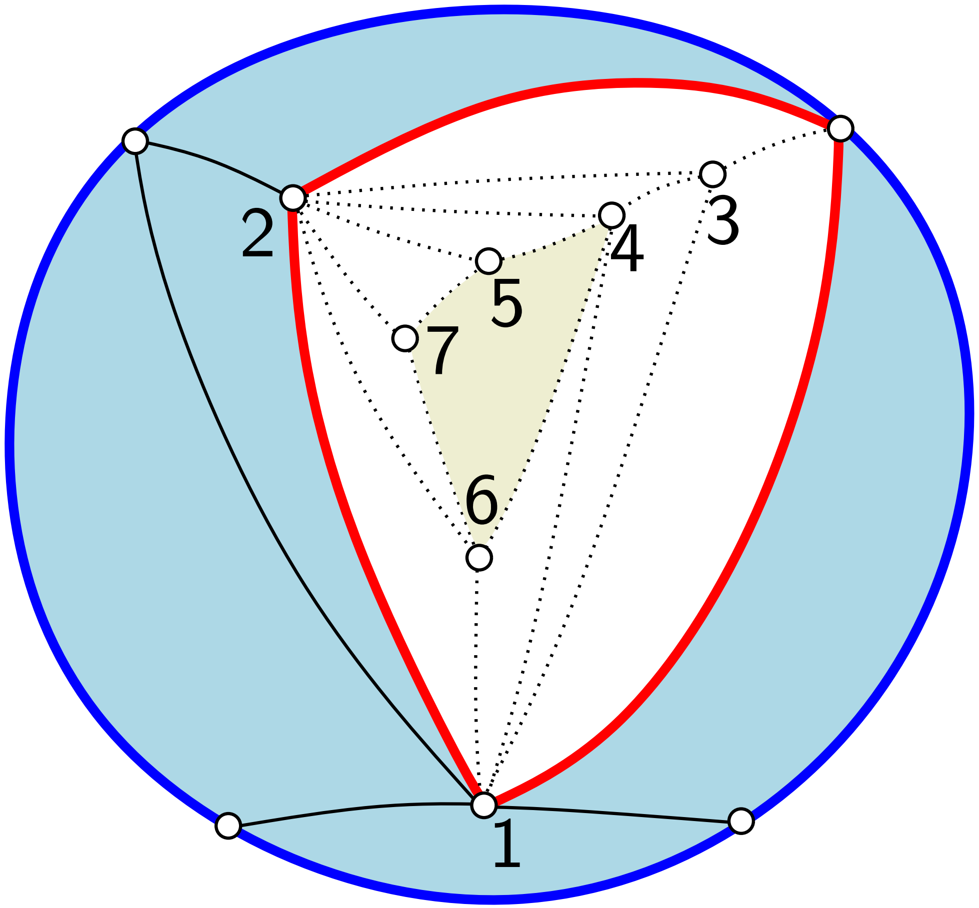


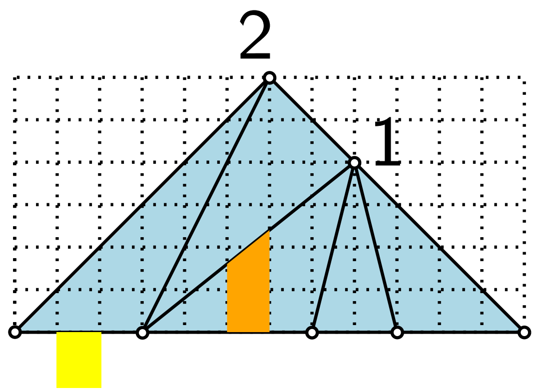
Execution on an example



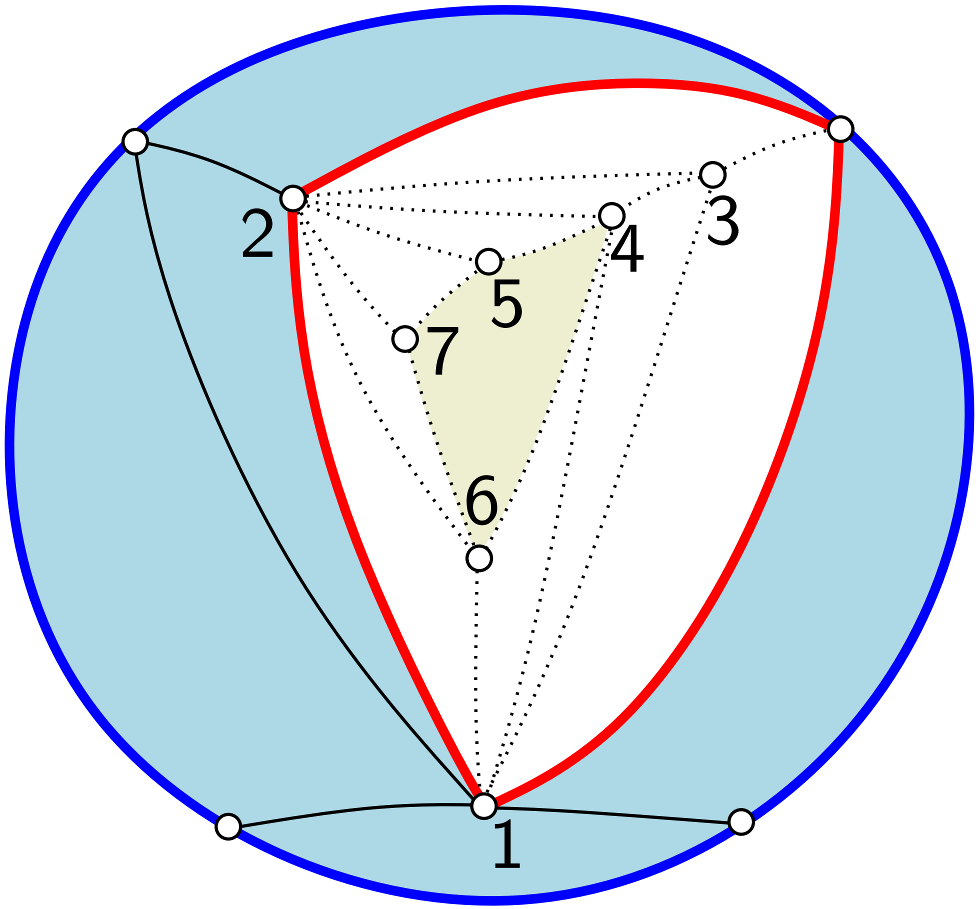


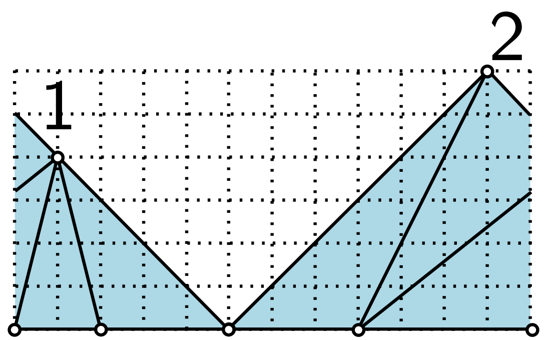
Execution on an example



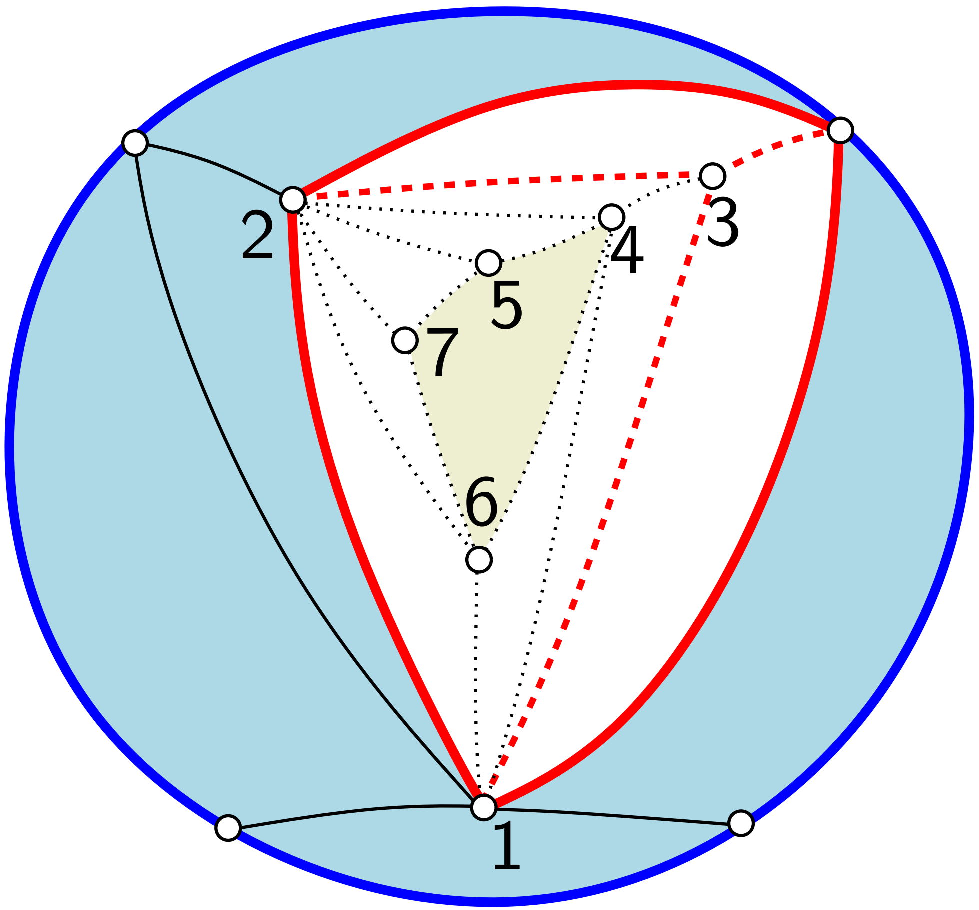


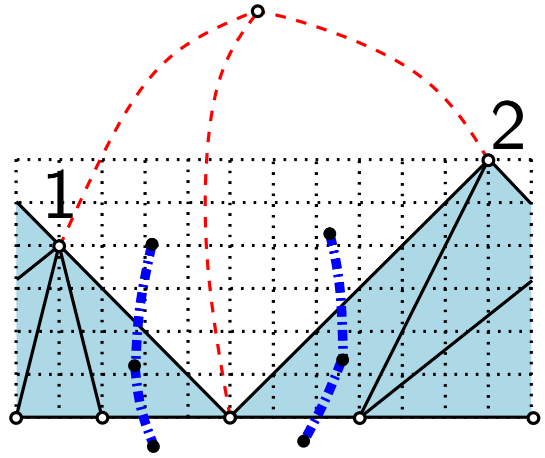
Execution on an example



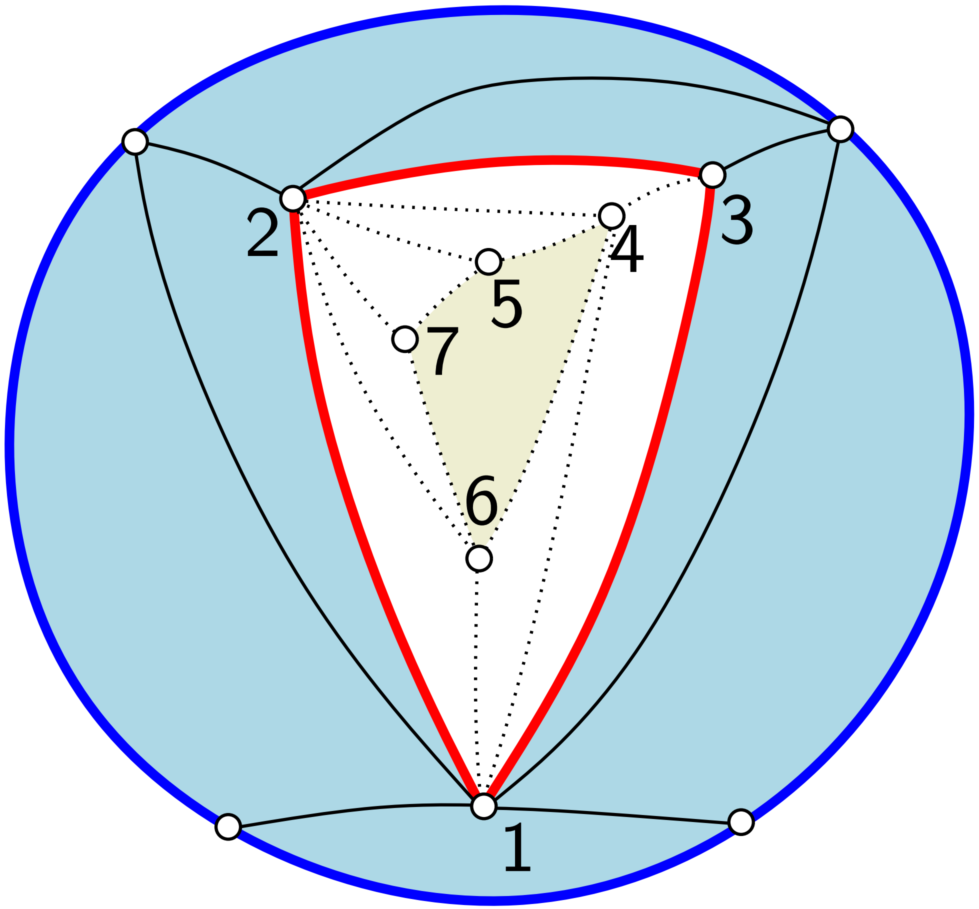


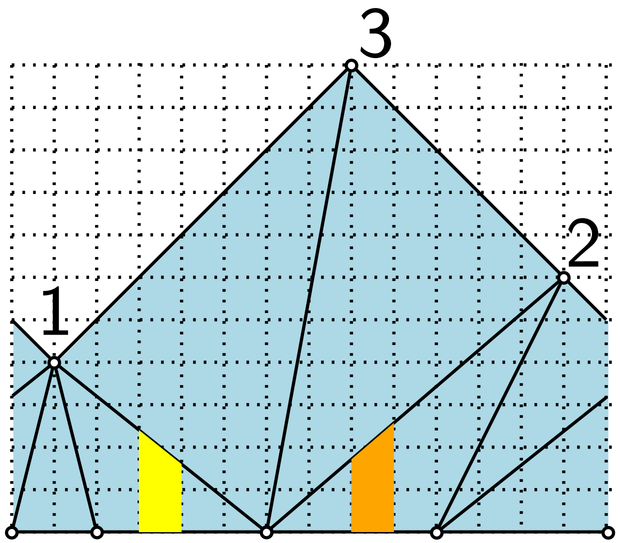
Execution on an example



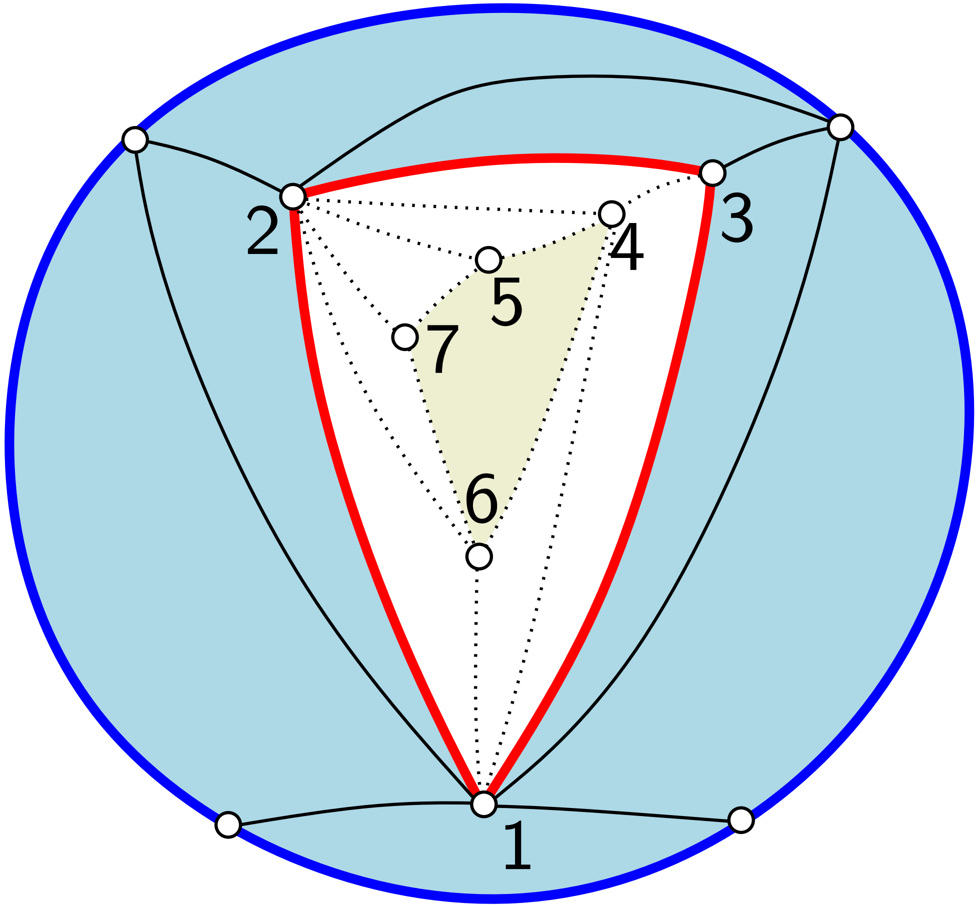


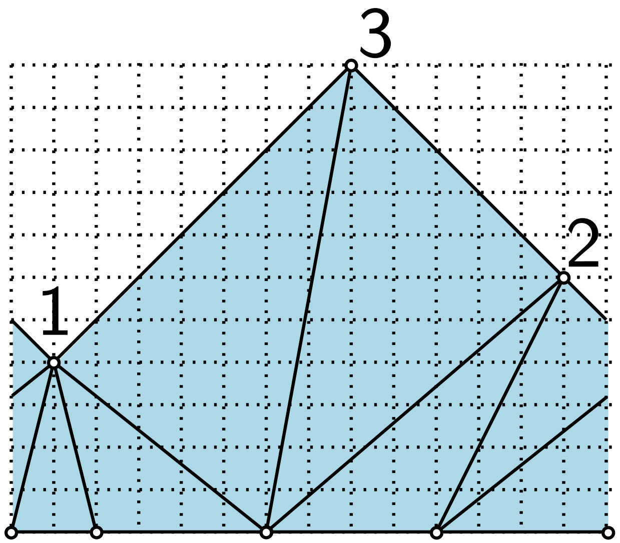
Execution on an example



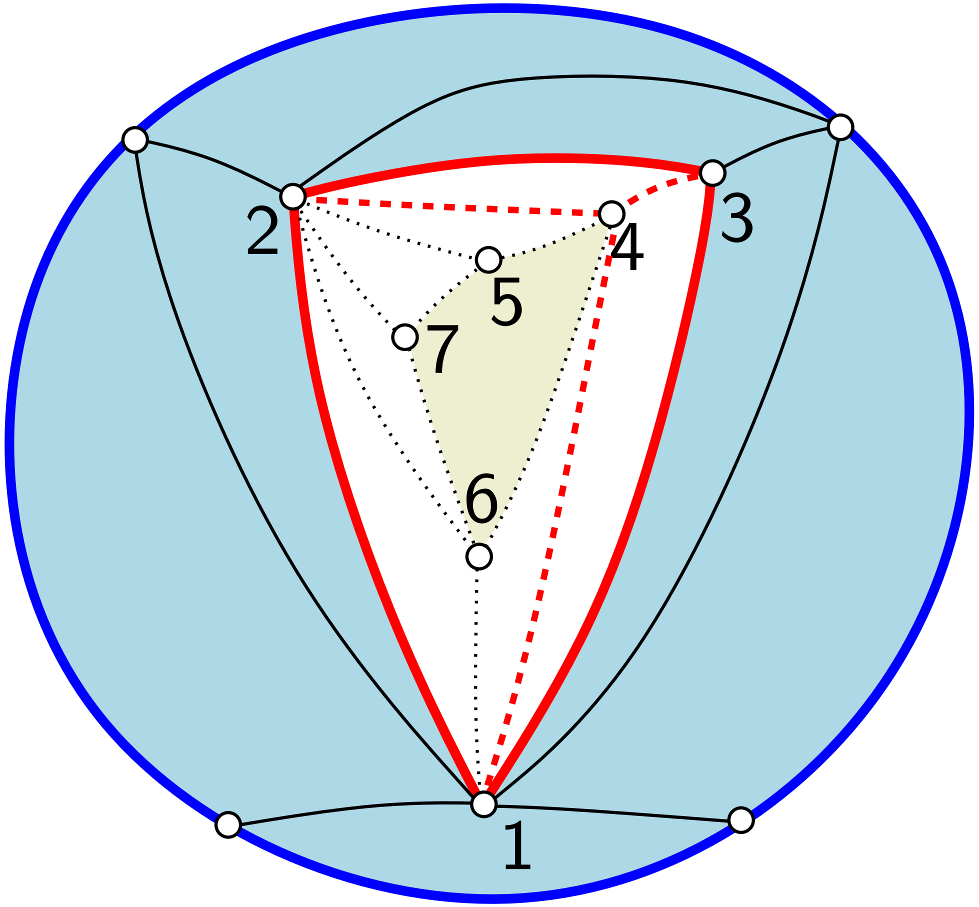


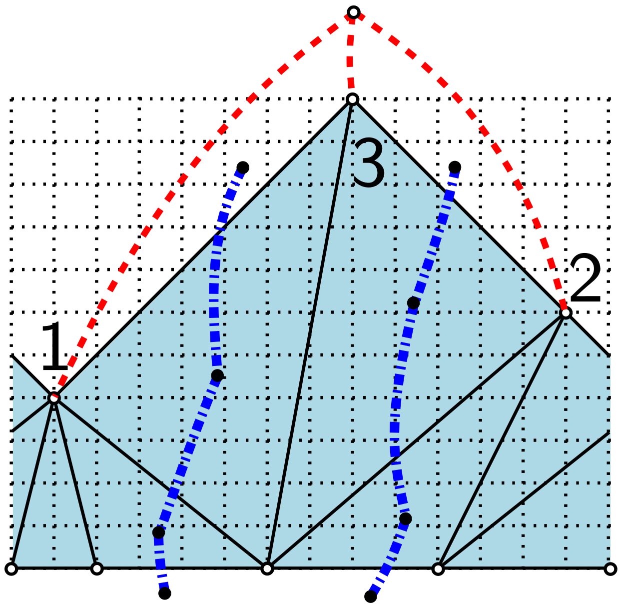
Execution on an example



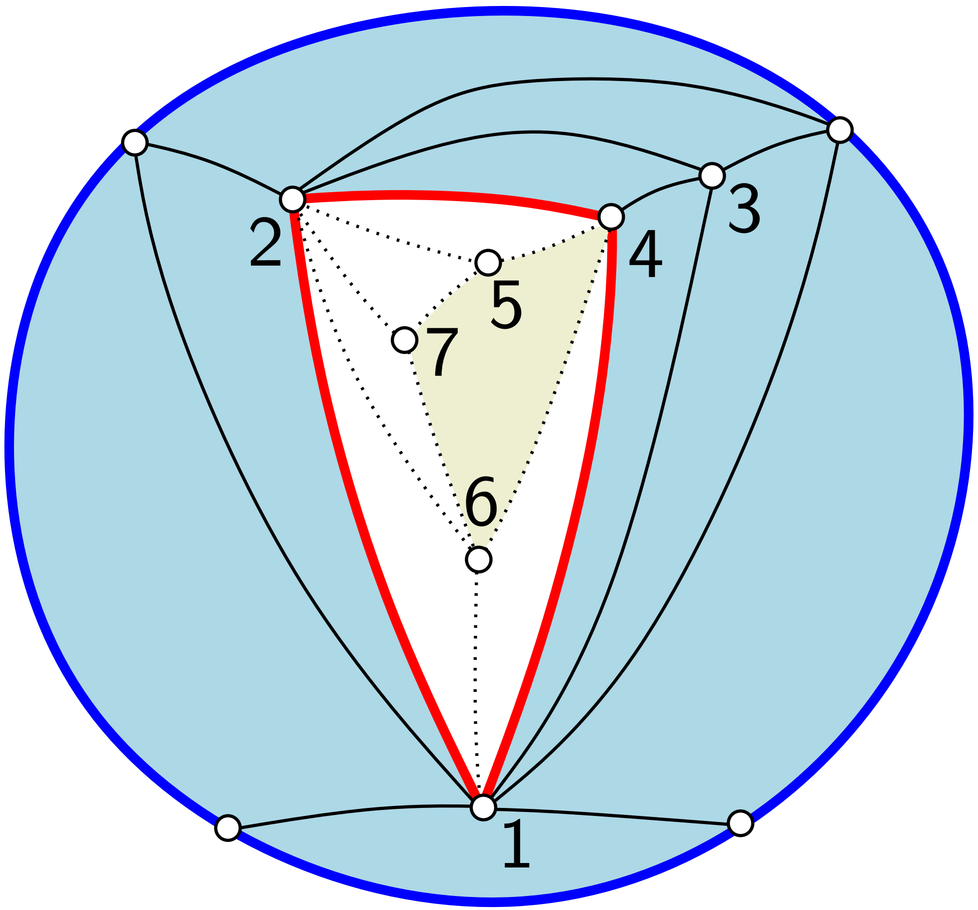


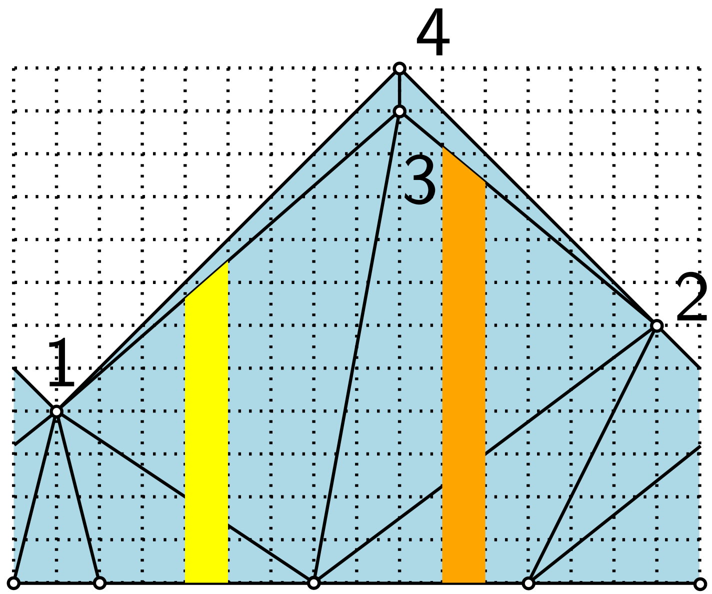
Execution on an example



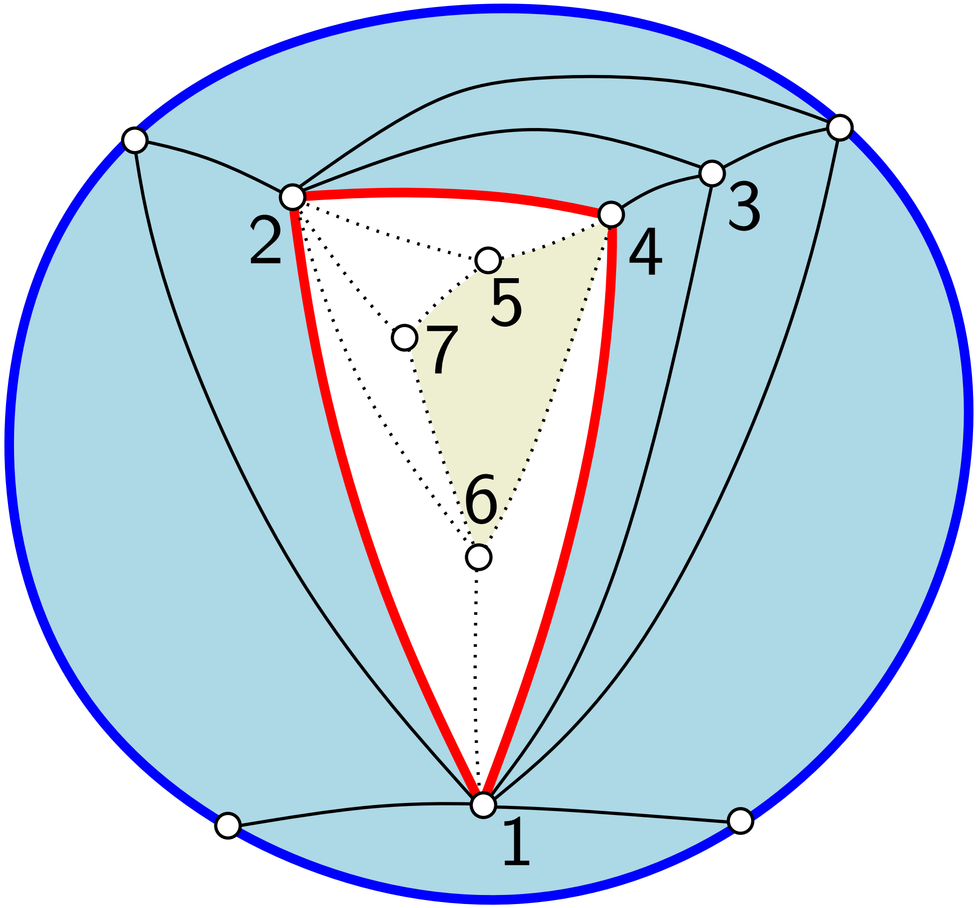


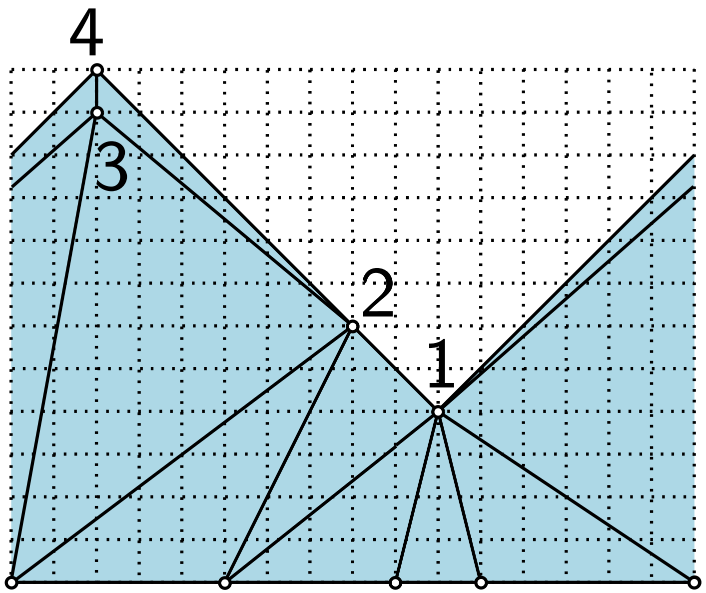
Execution on an example



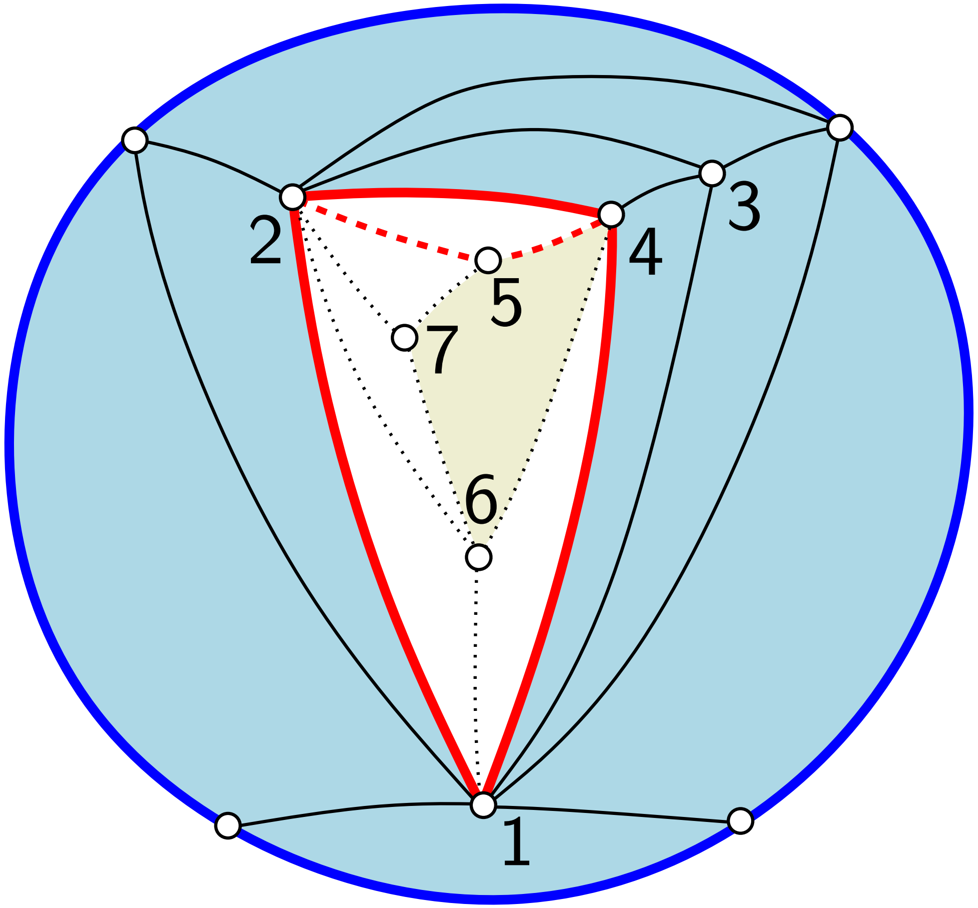


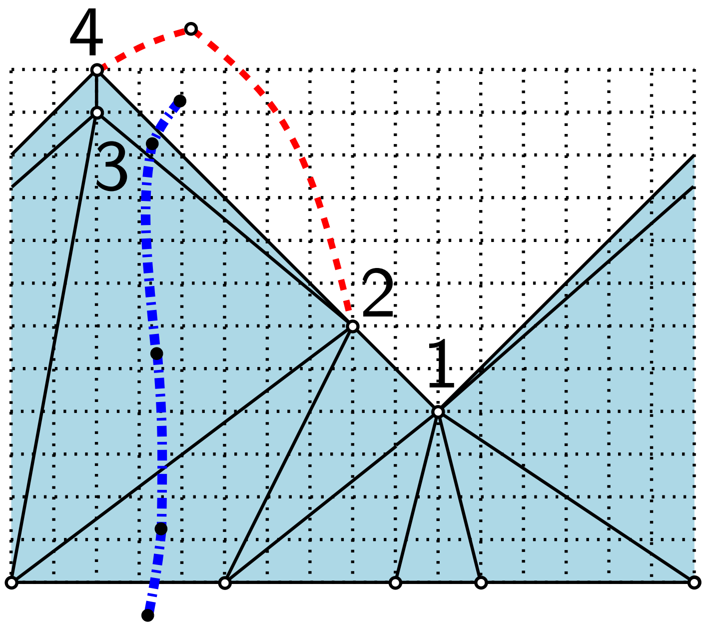
Execution on an example



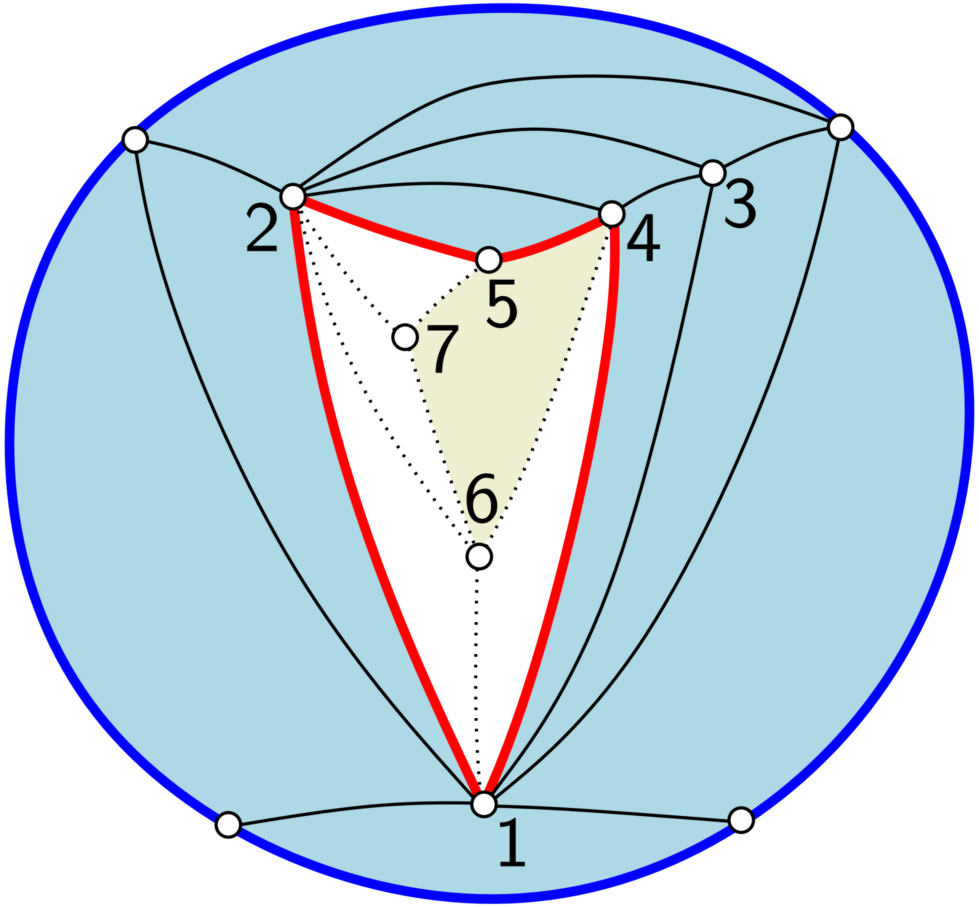


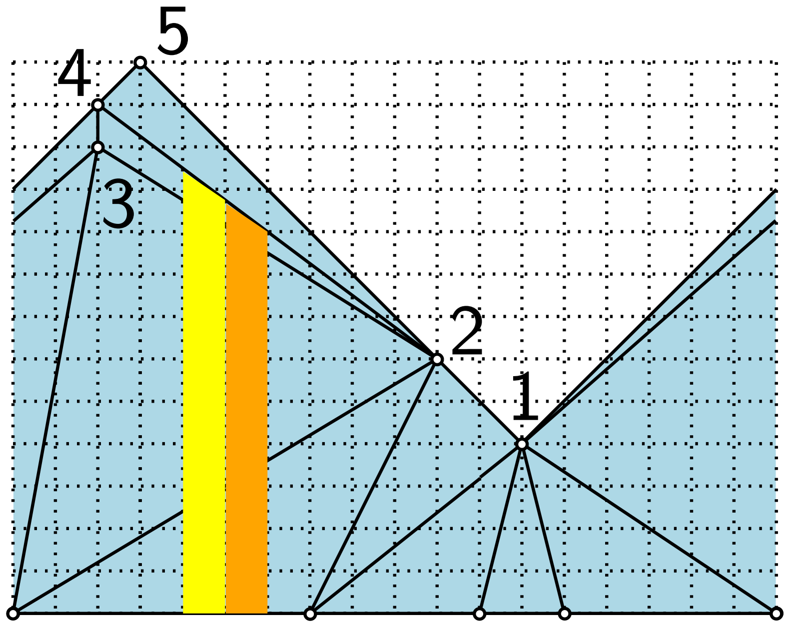
Execution on an example



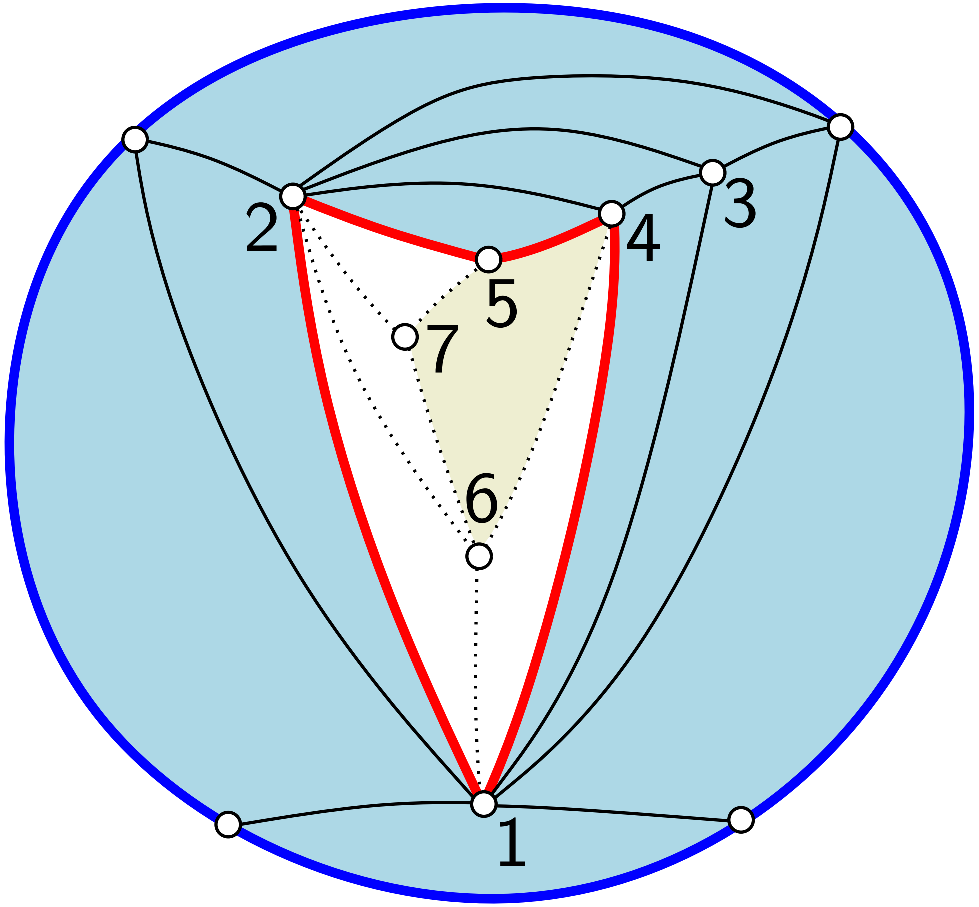


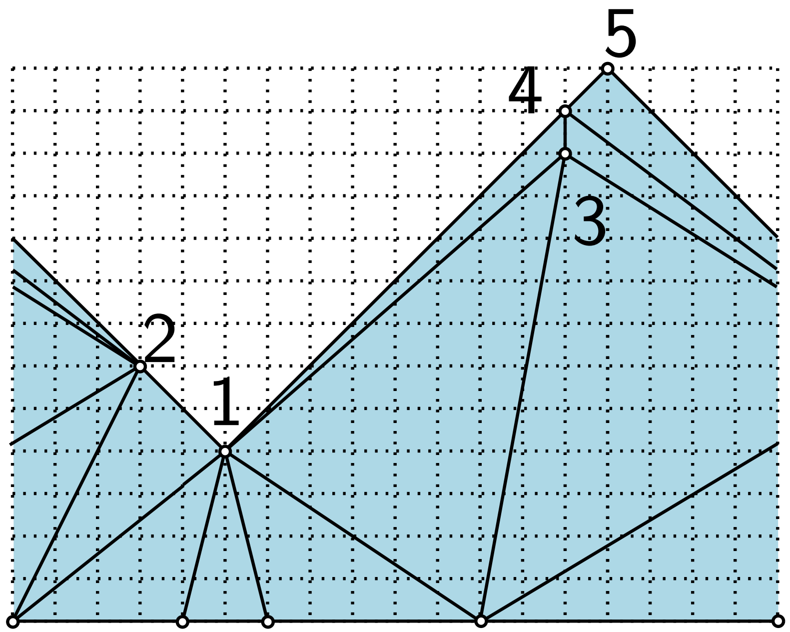
Execution on an example



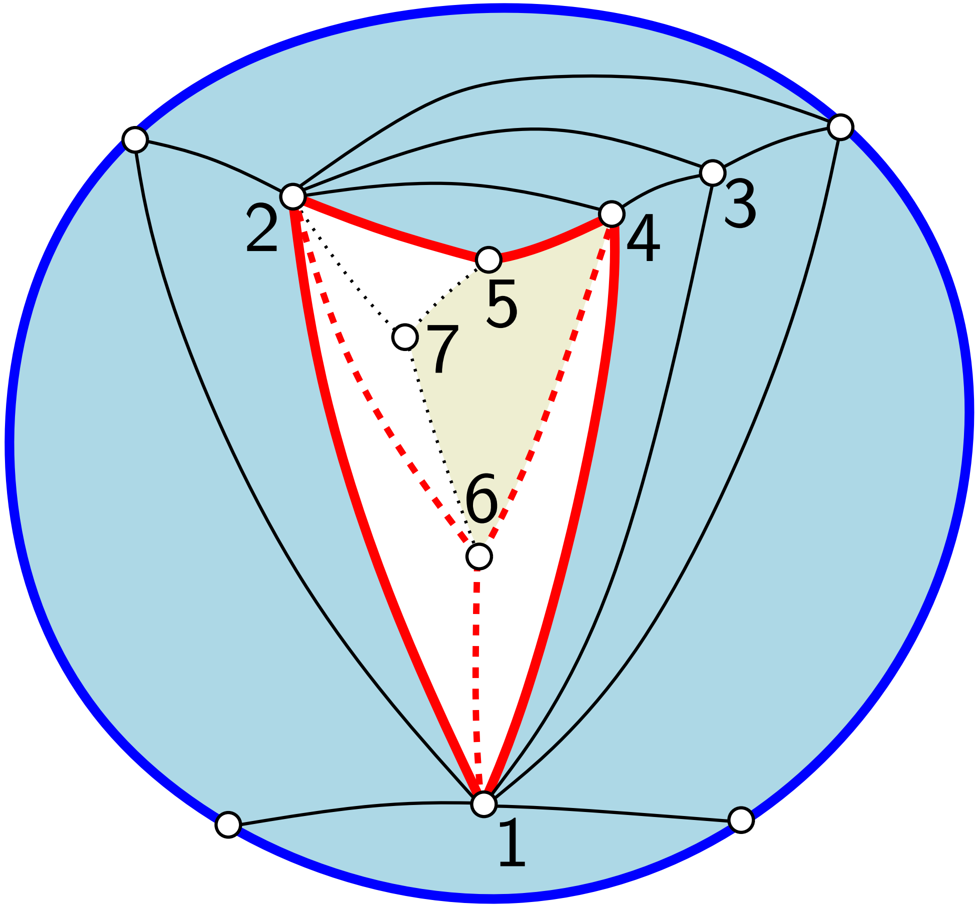


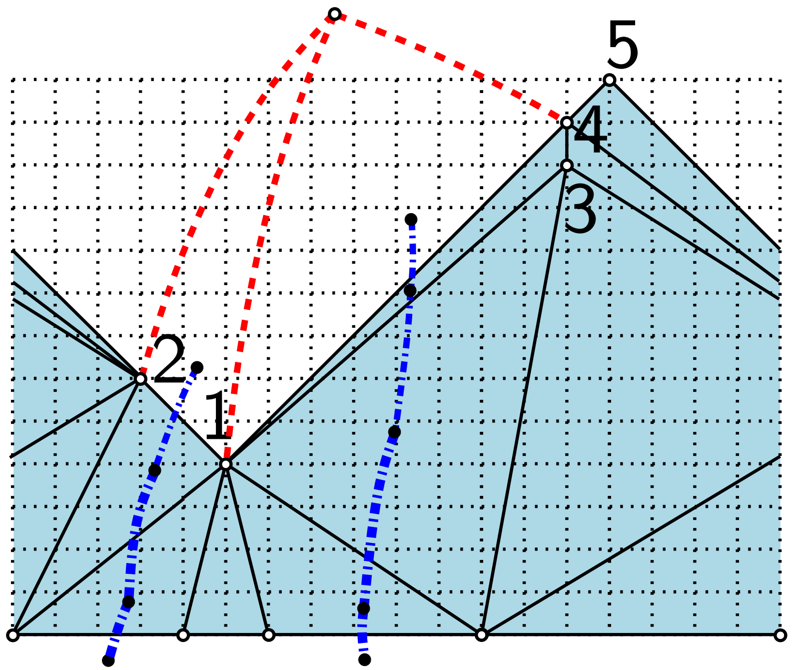
Execution on an example



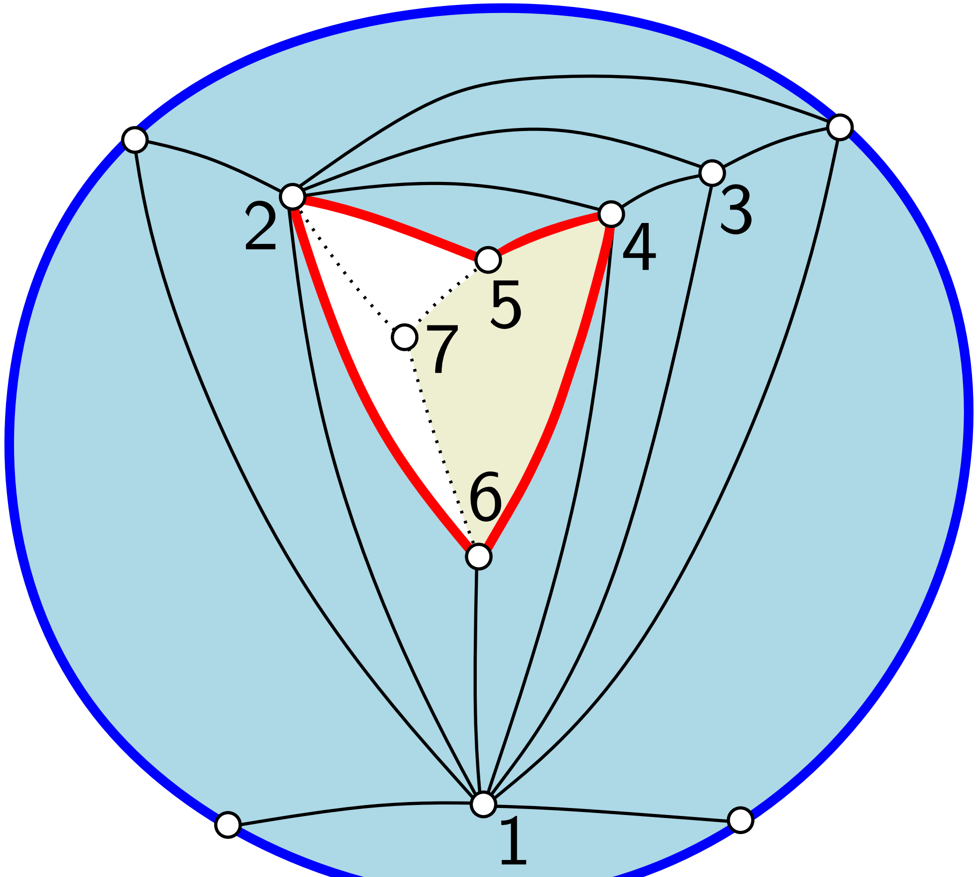


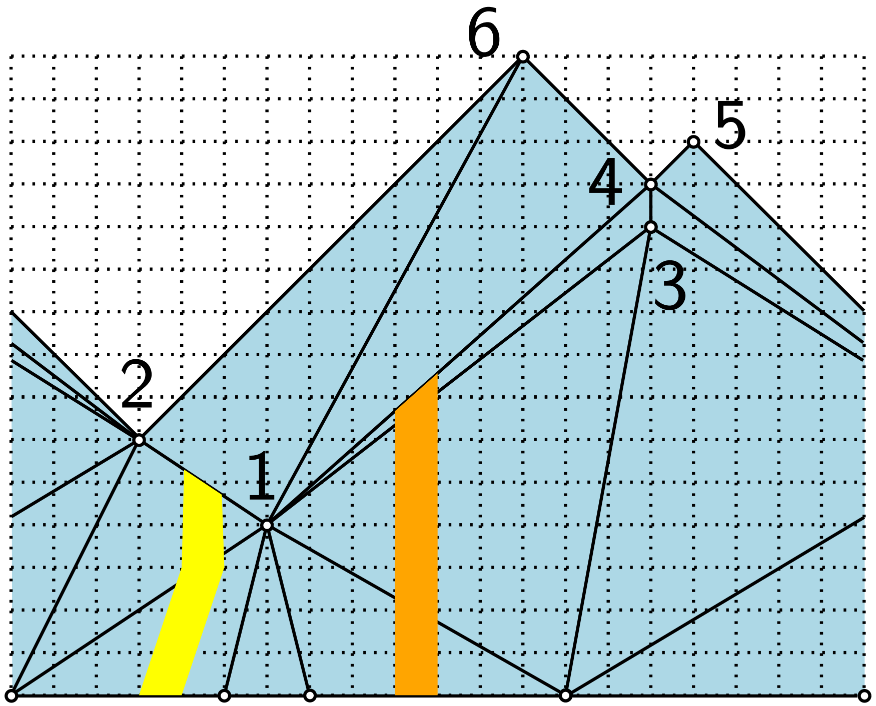
Execution on an example



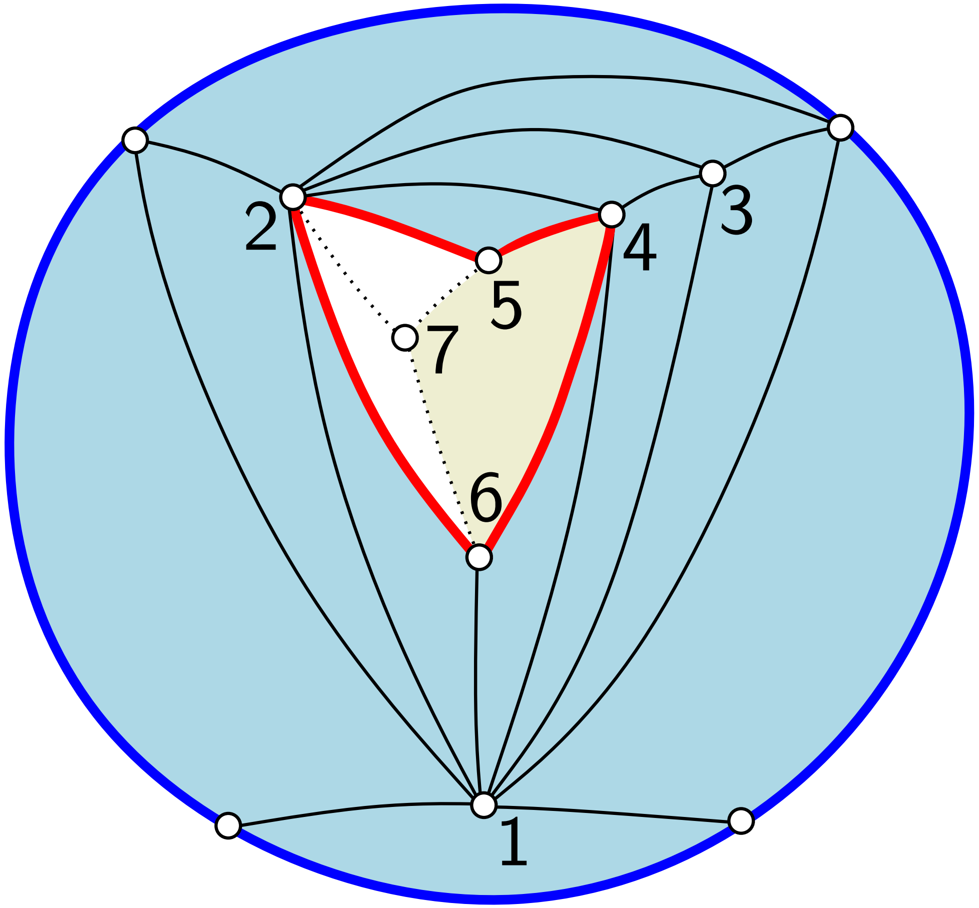


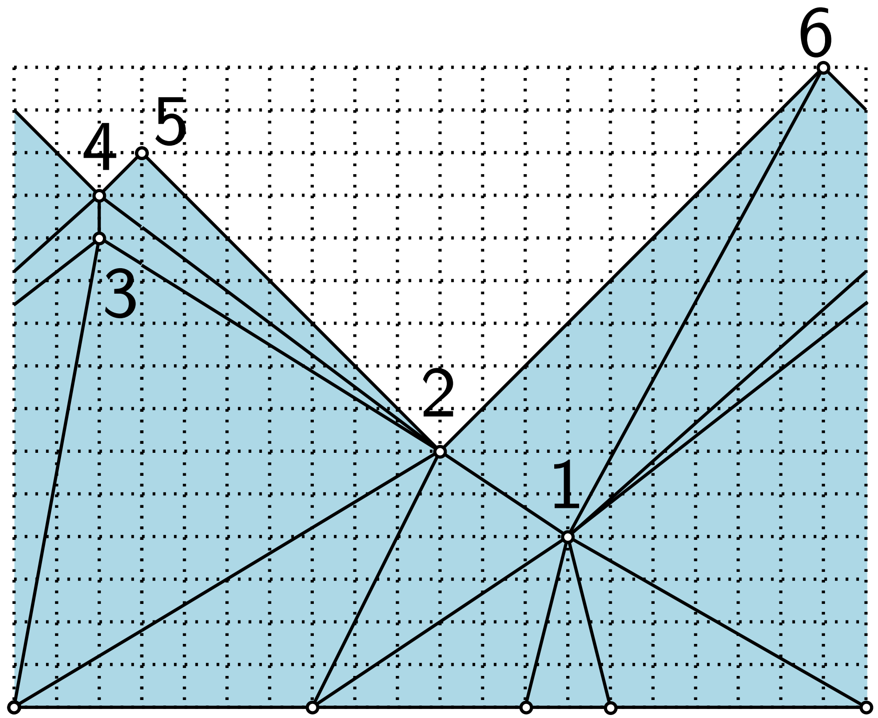
Execution on an example



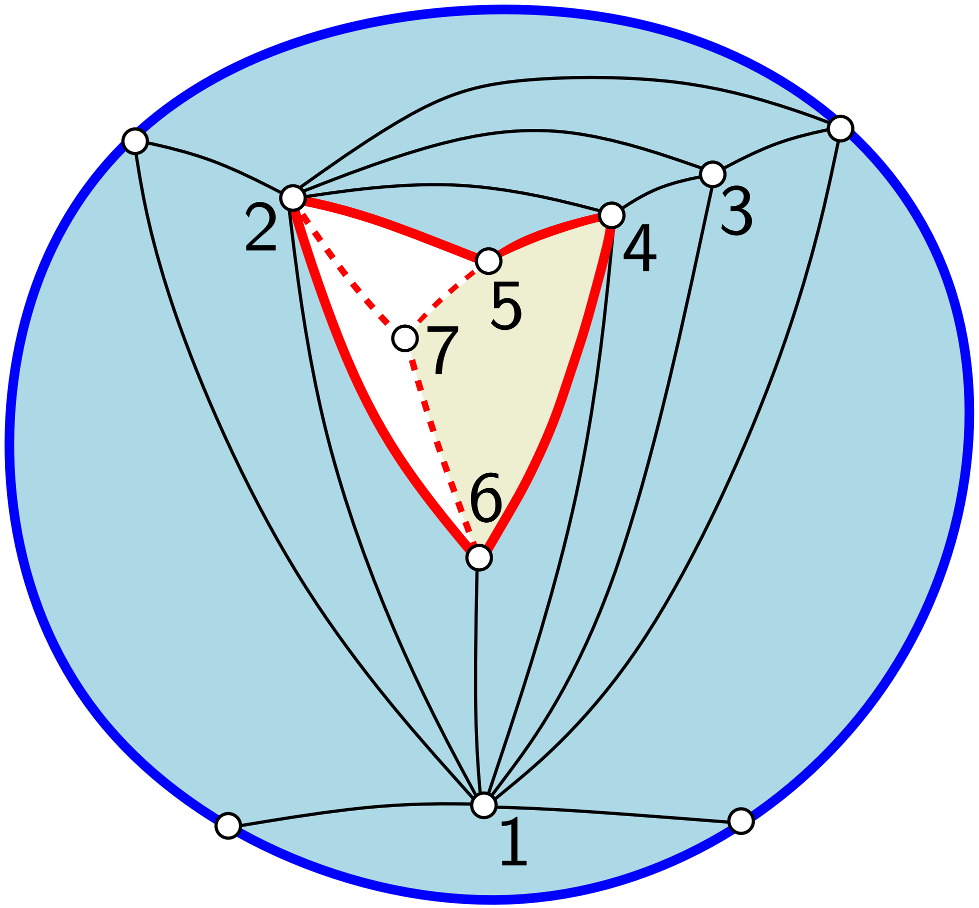


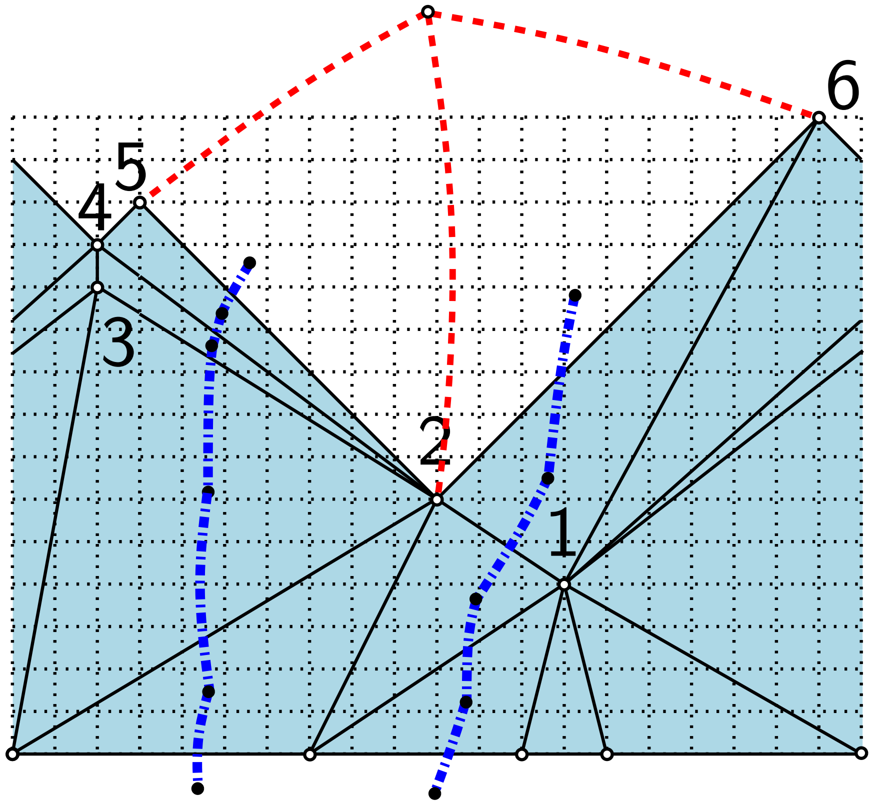
Execution on an example



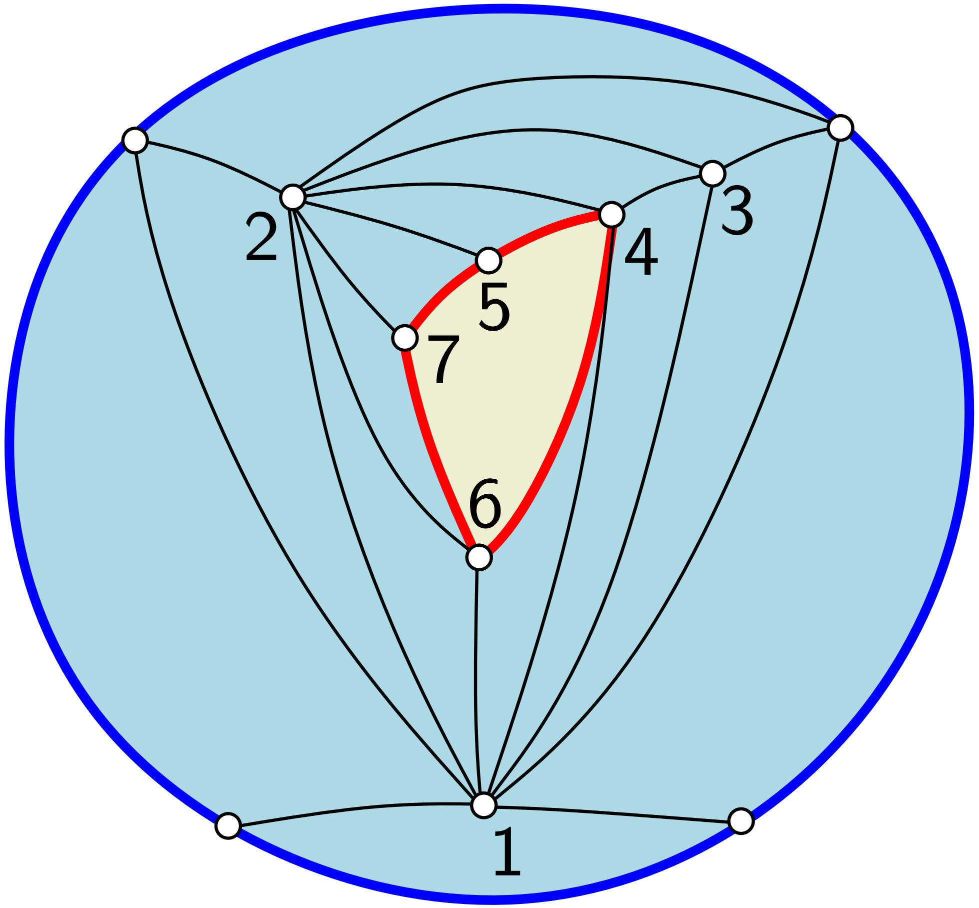


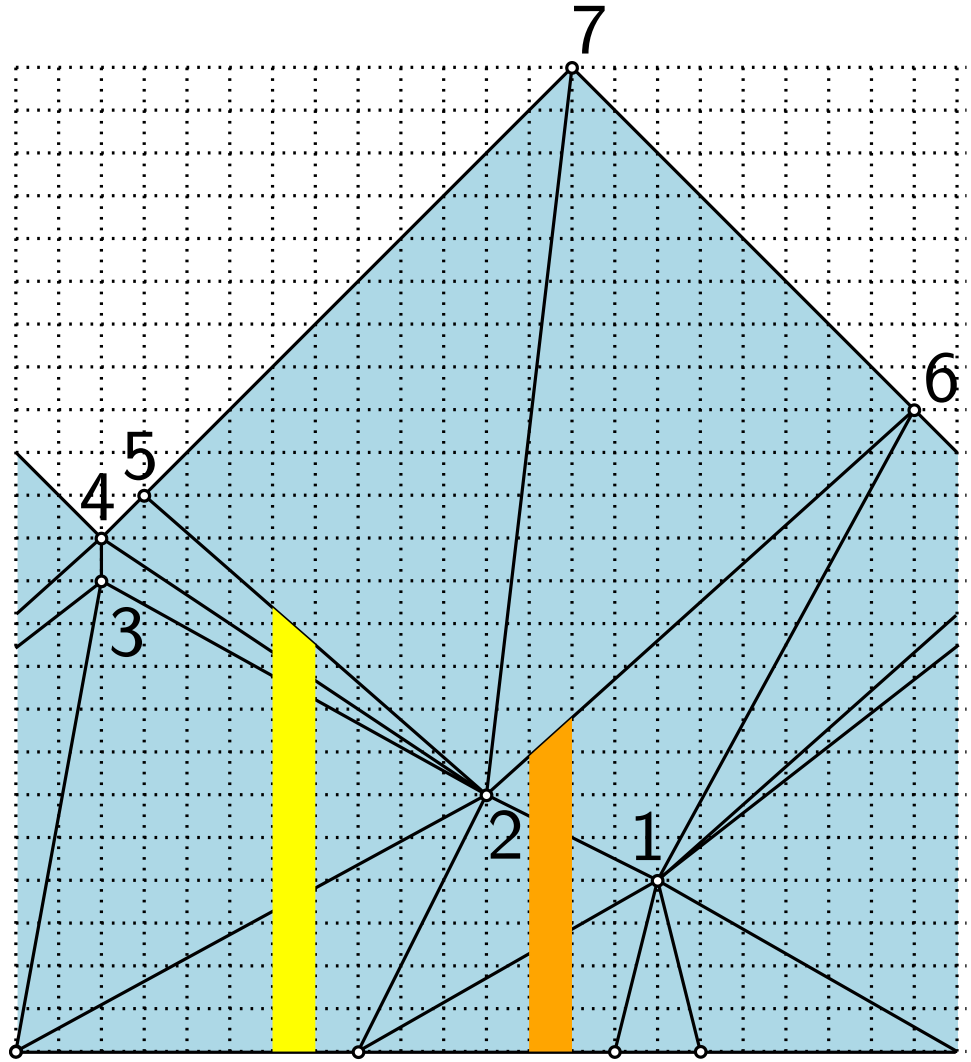
Execution on an example



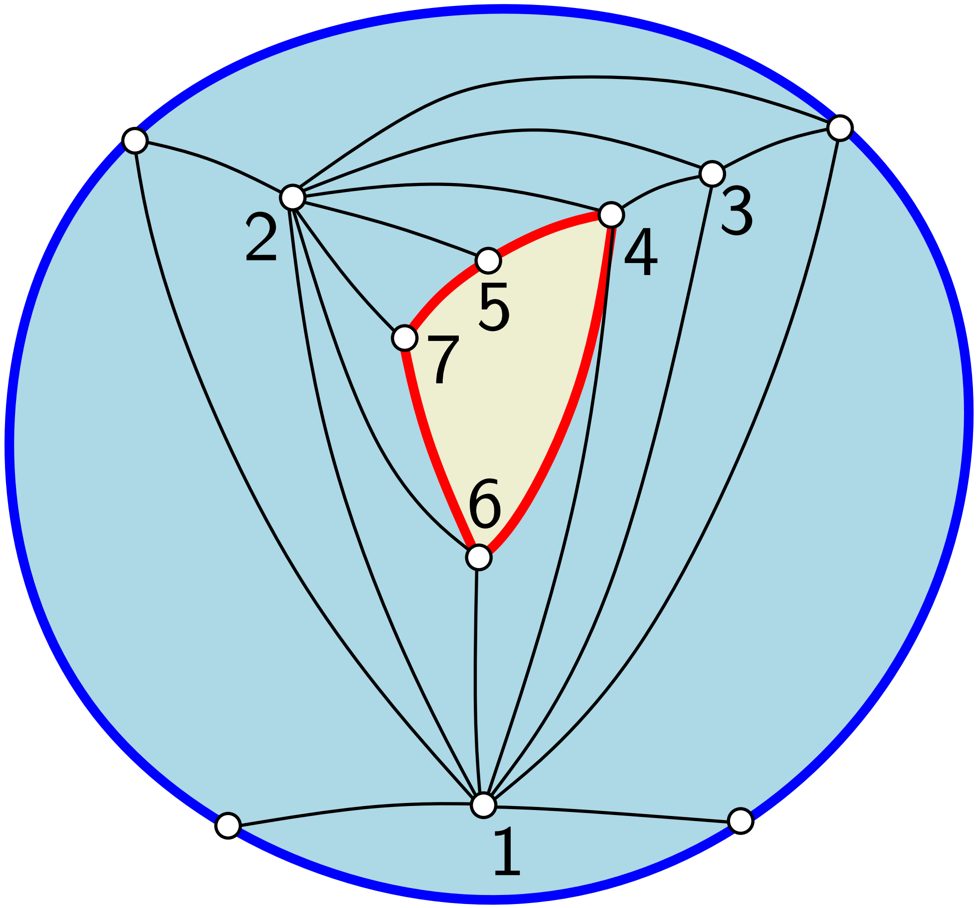


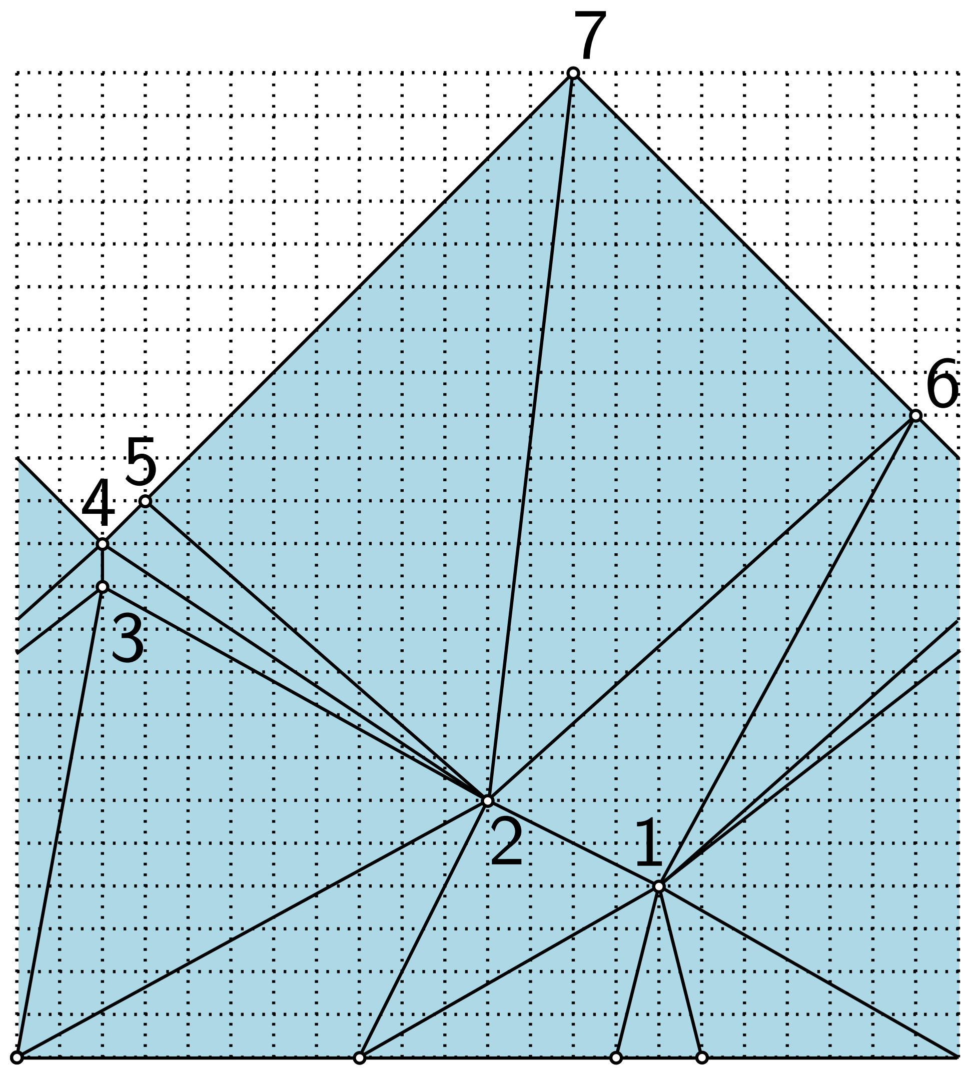
Execution on an example





Execution on an example





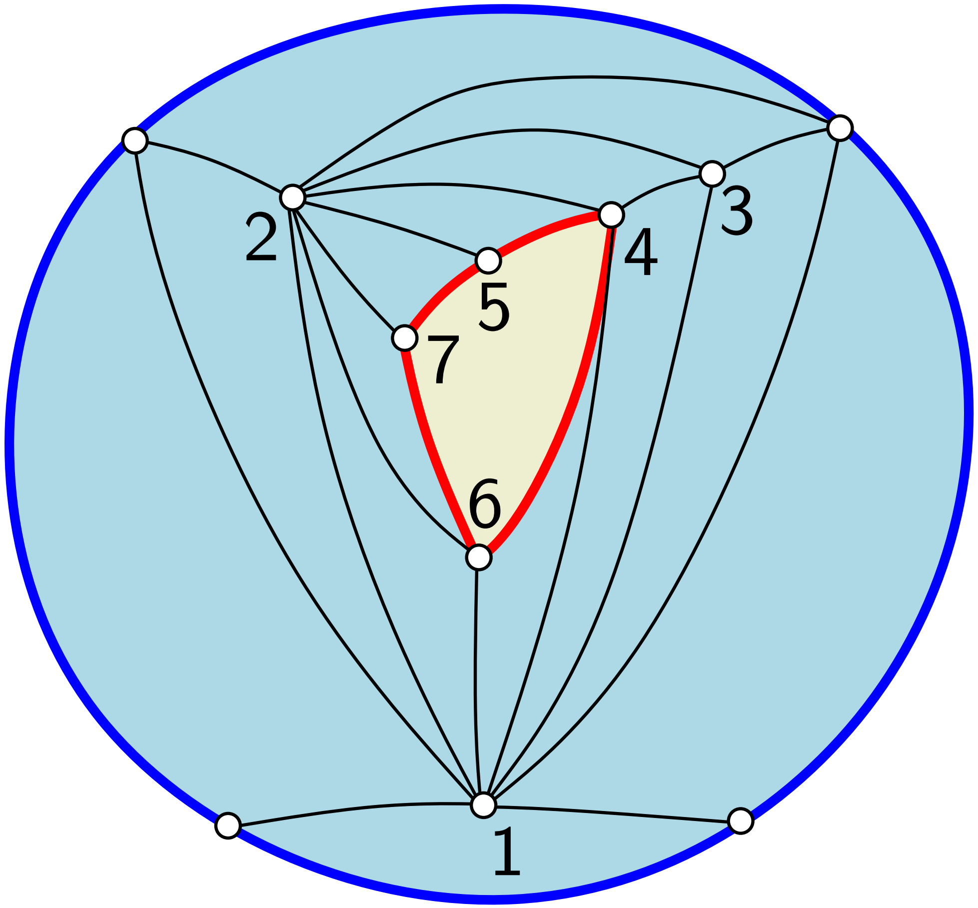
Grid size: $w = 2n$

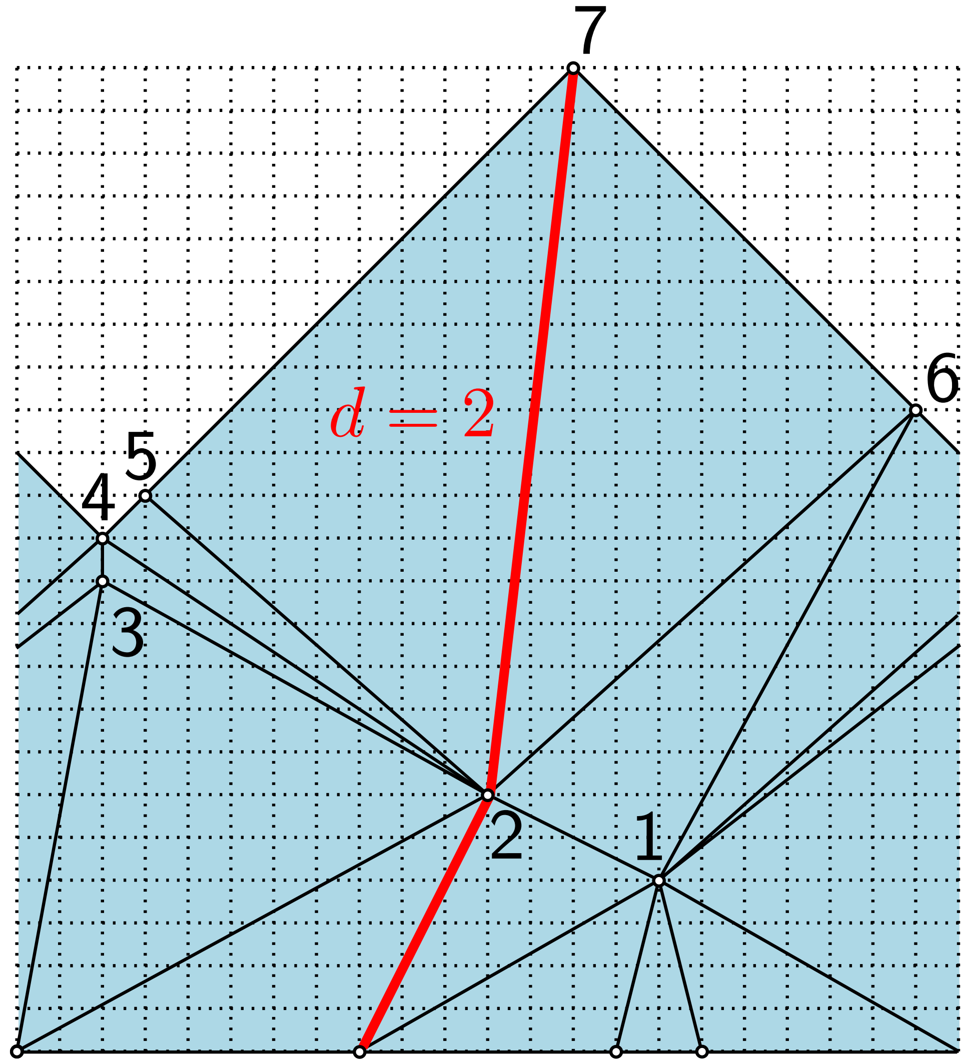
Each edge has vertical extension at most w

$\Rightarrow h \leq n(2d+1)$

with d the graph-distance between the two boundaries

Execution on an example





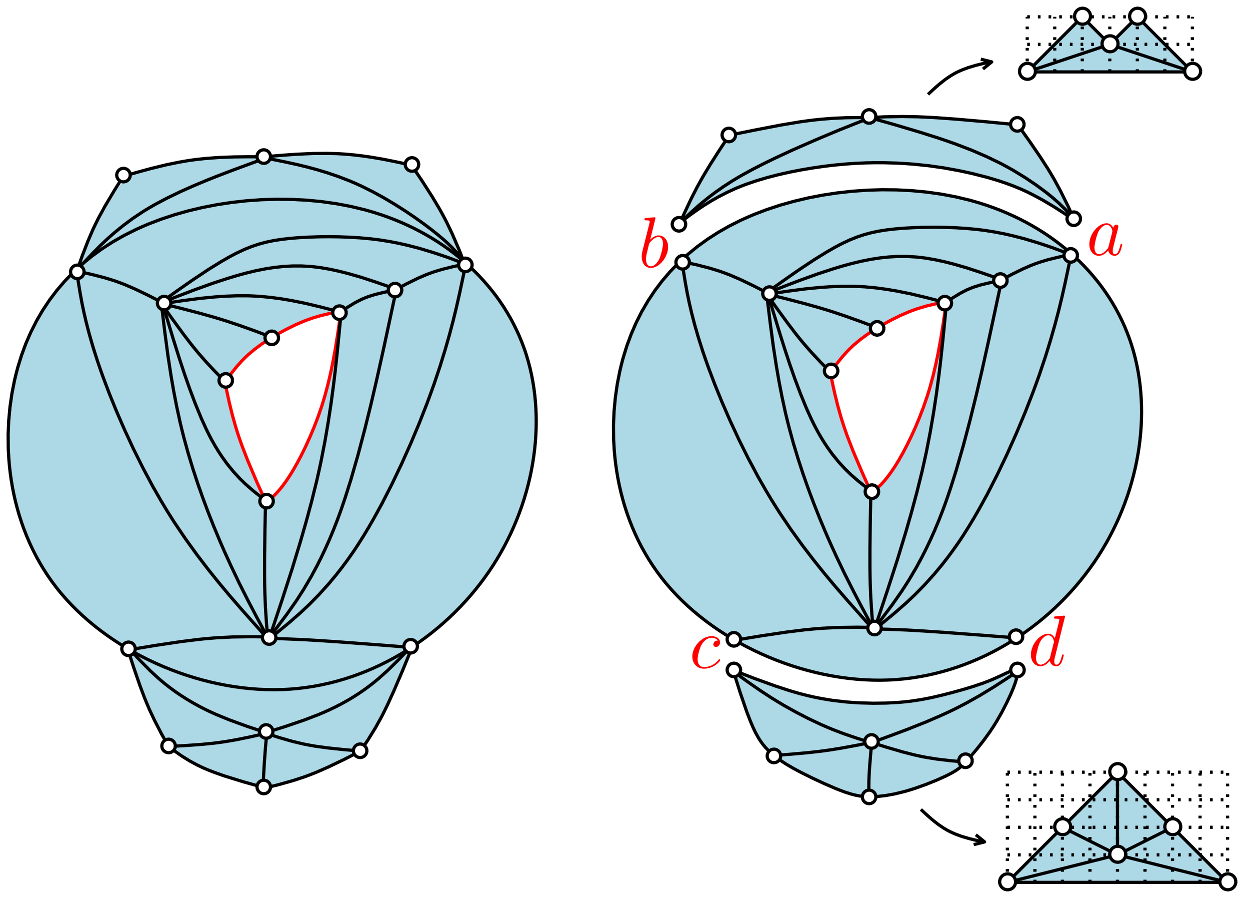
Grid size: $w = 2n$

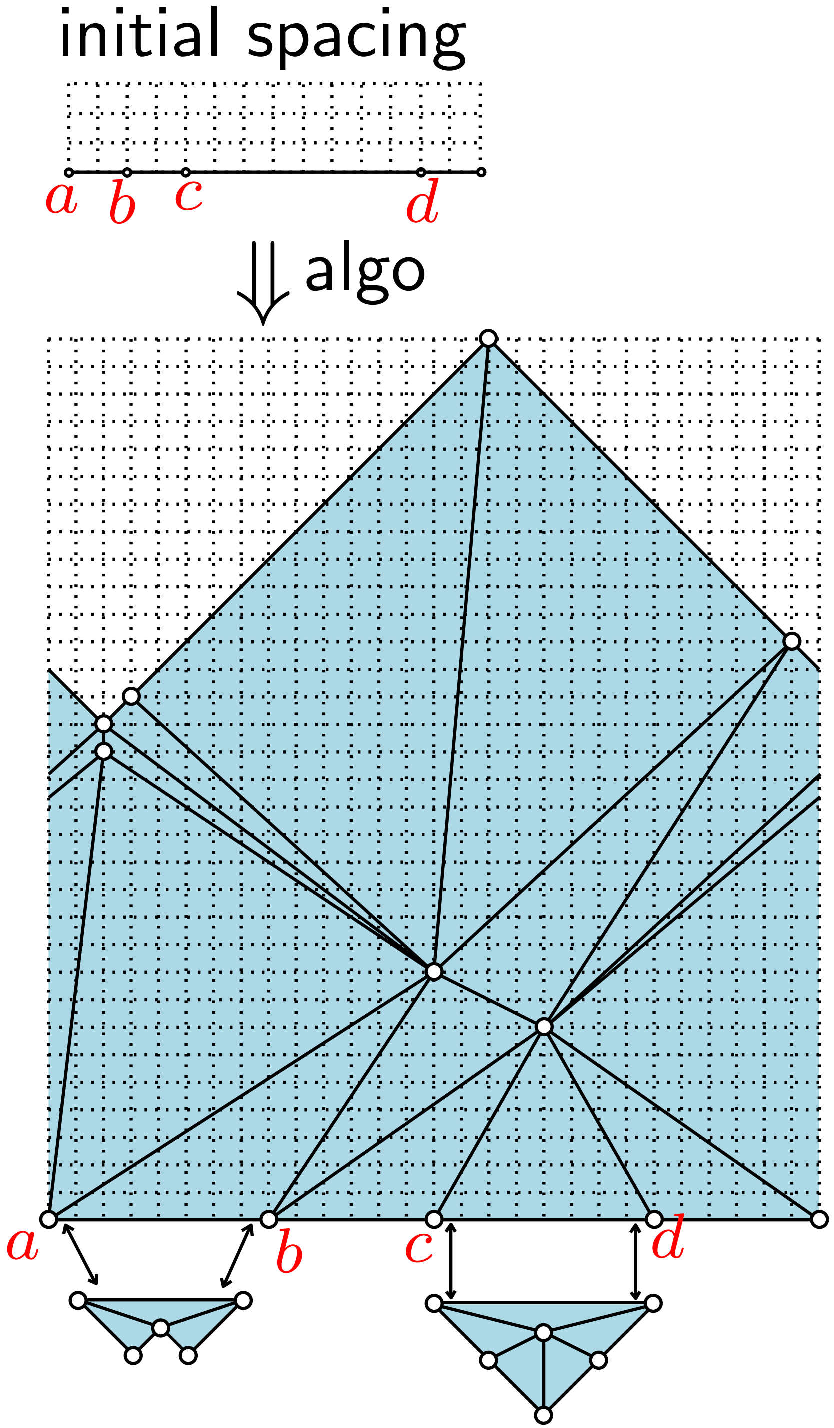
Each edge has vertical extension at most w

$\Rightarrow h \leq n(2d+1)$

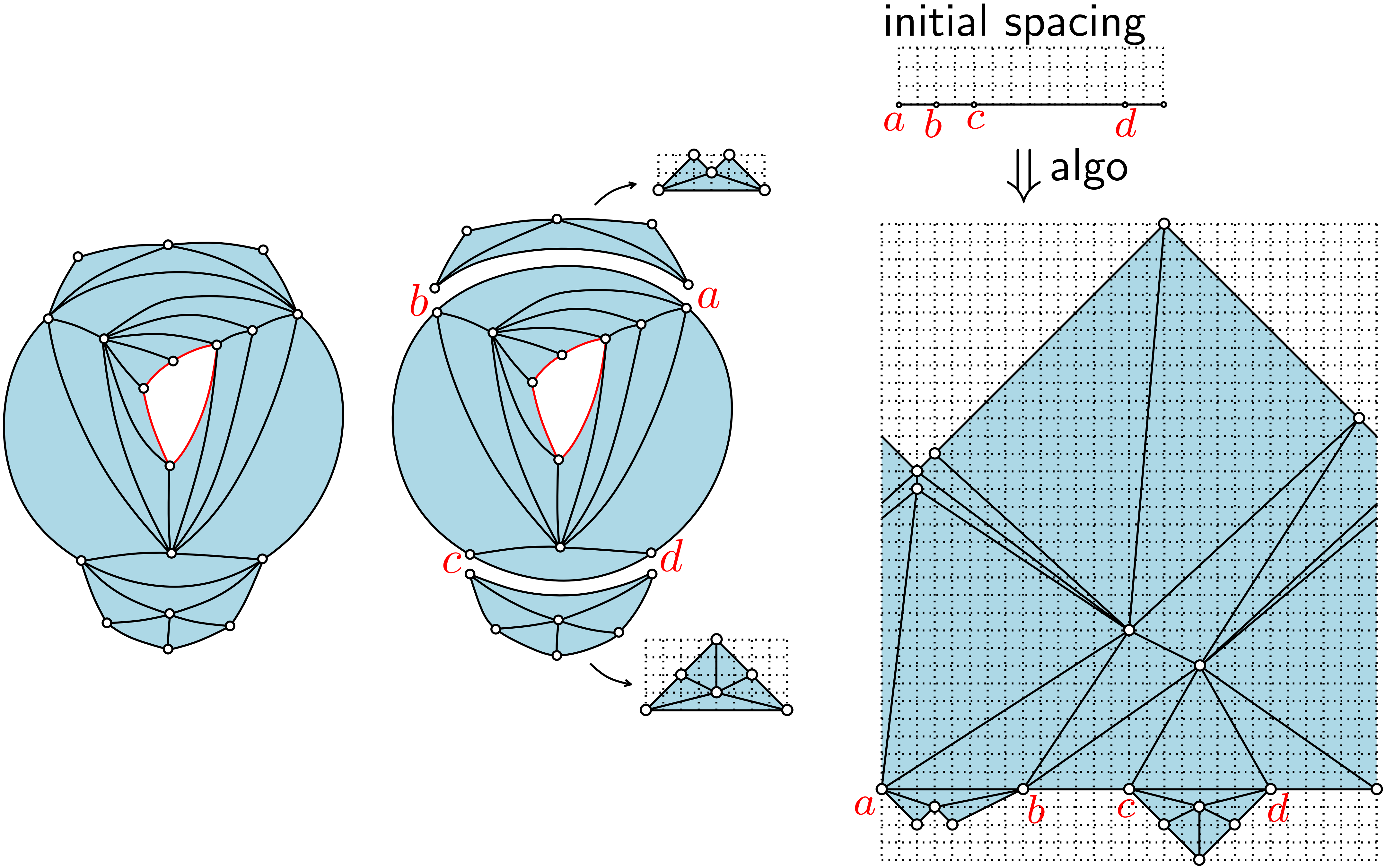
with d the graph-distance between the two boundaries

Dealing with chordal edges at outer cycle





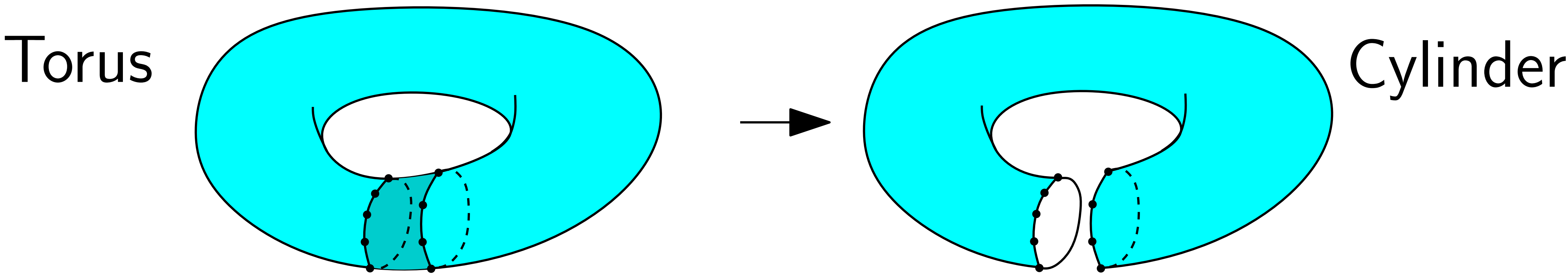
Dealing with chordal edges at outer cycle

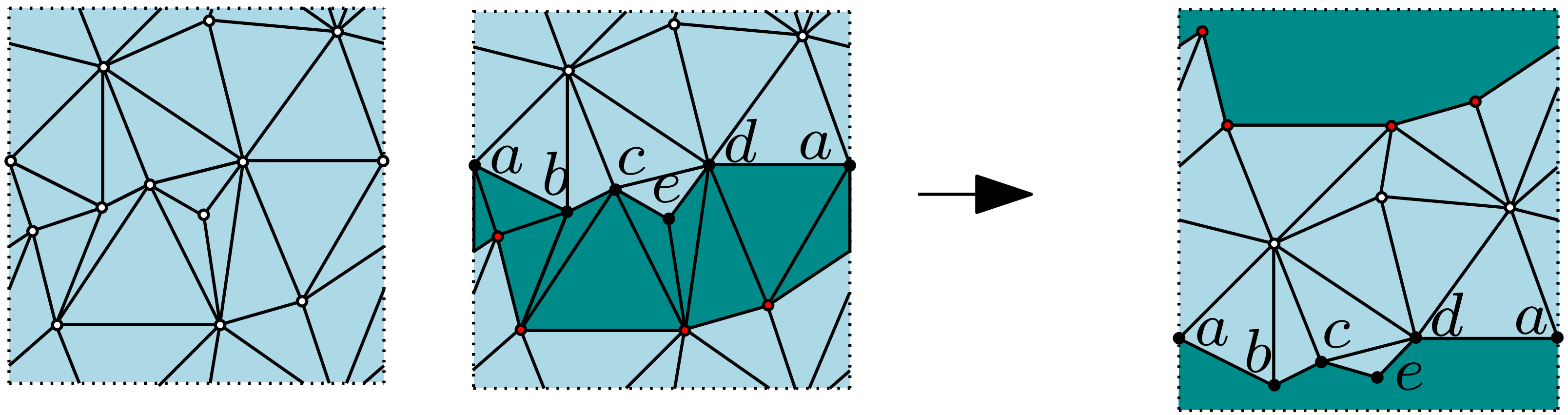


Toroidal triangulations & tambourine

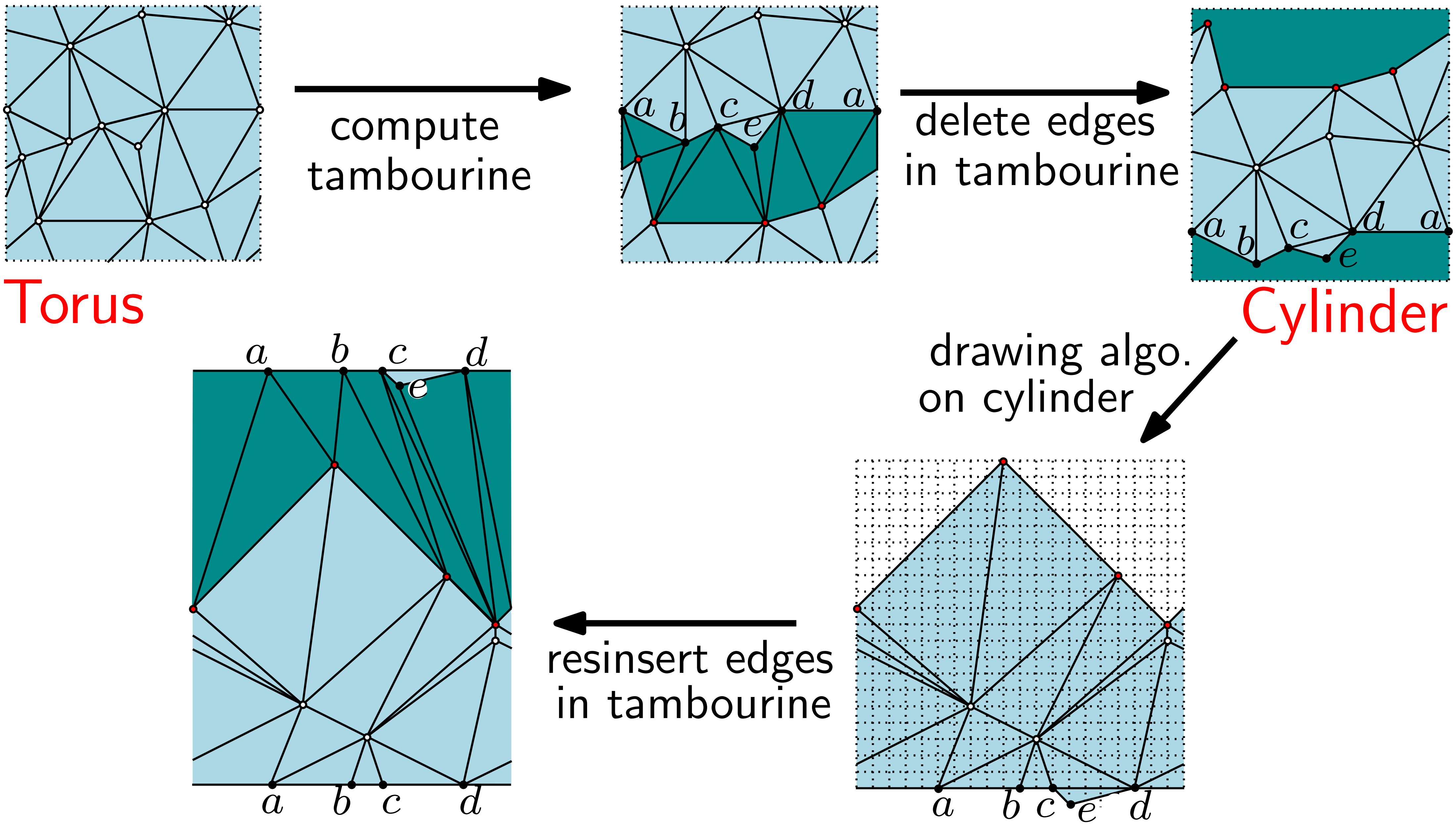
Every toroidal triangulation admits a "tambourine"

[**Bonichon, Gavoille, Labourel'06**]

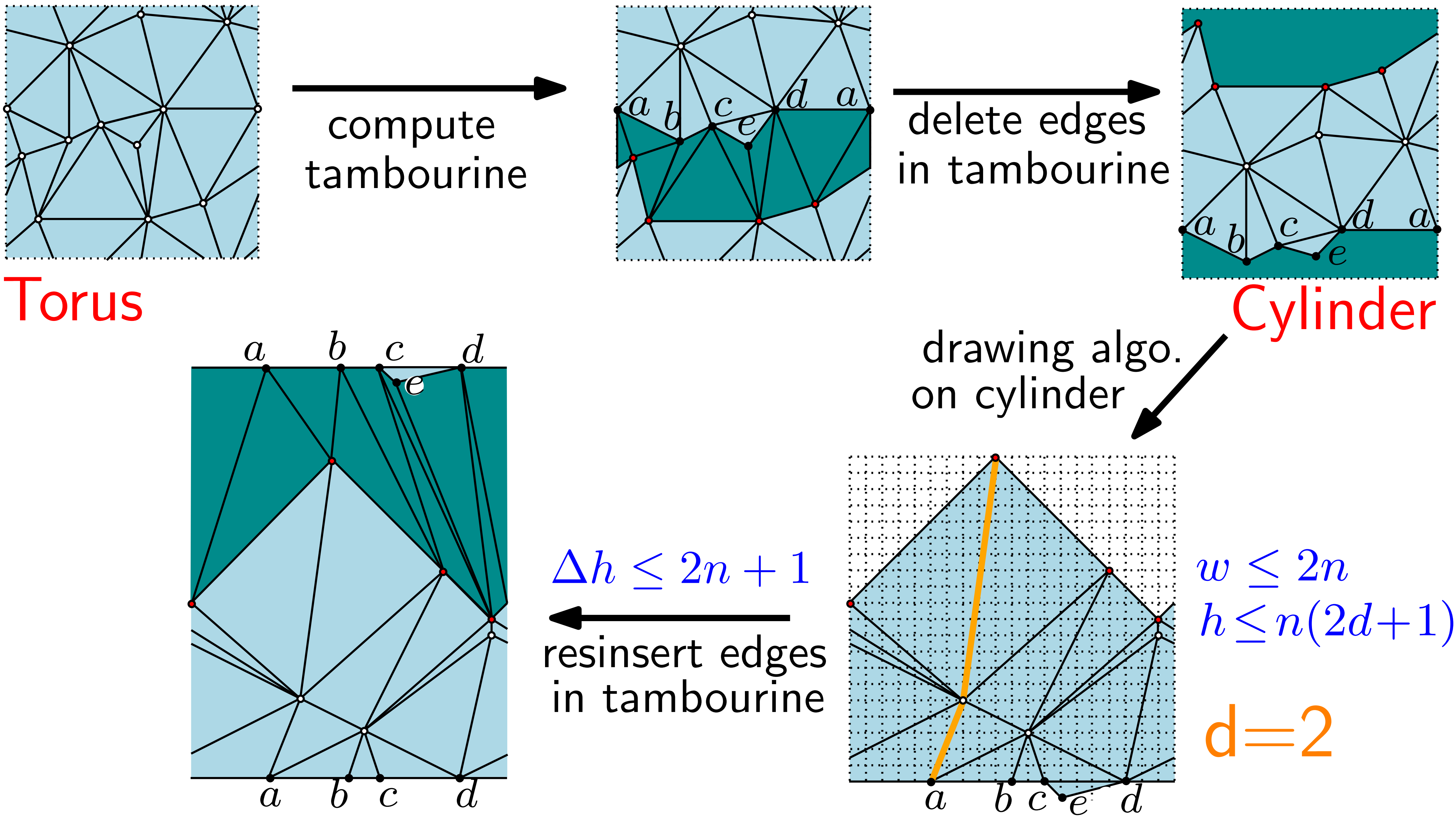




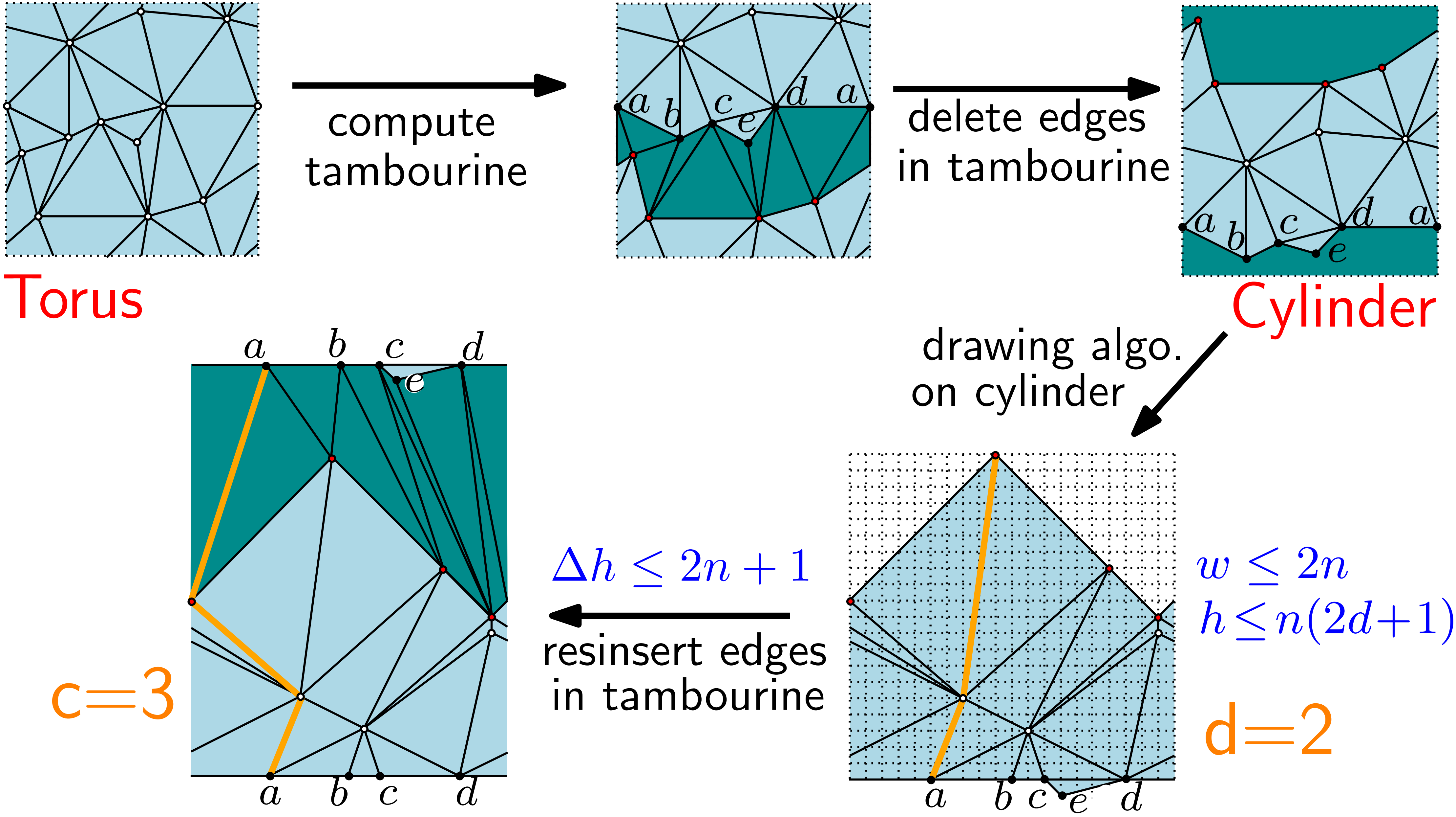
Getting a drawing on the (flat) torus



Getting a drawing on the (flat) torus



Getting a drawing on the (flat) torus

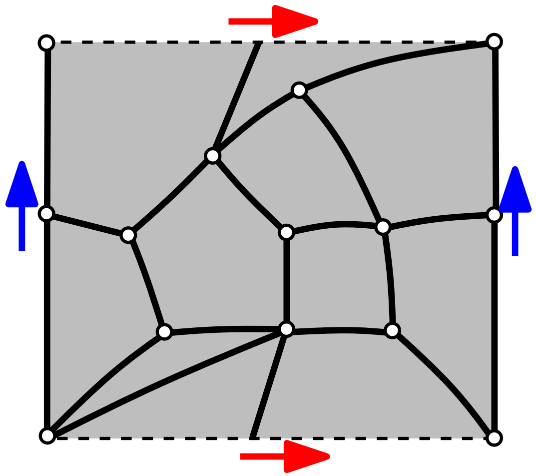


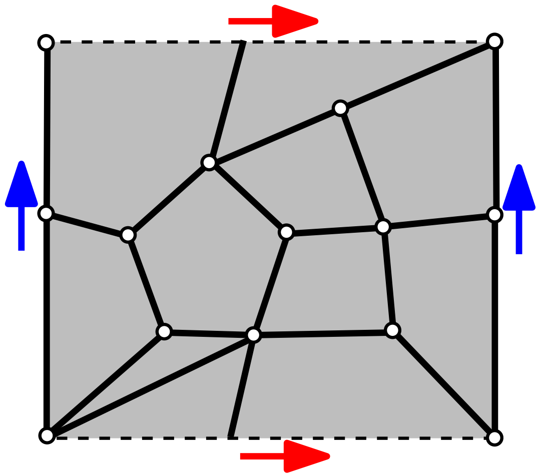
Let c =length shortest non-contractible cycle, $c \leq \sqrt{2n}$ **[Hutchinson, Albert'78]**

Can choose tambourine so that $d < c \Rightarrow$ $\boxed{h = O(n^{3/2})}$

Extensions and perspectives

- Extension to 3-connected maps on cylinder & torus
  (cf Kant's extension of the FPP algorithm in the planar case)

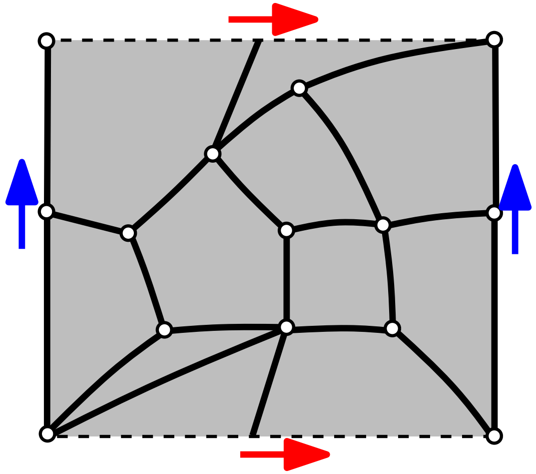


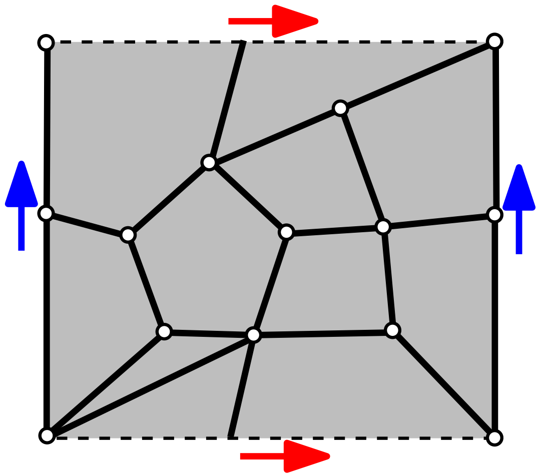


weakly convex periodic
straight-line drawing

Extensions and perspectives

- Extension to 3-connected maps on cylinder & torus
  (cf Kant's extension of the FPP algorithm in the planar case)

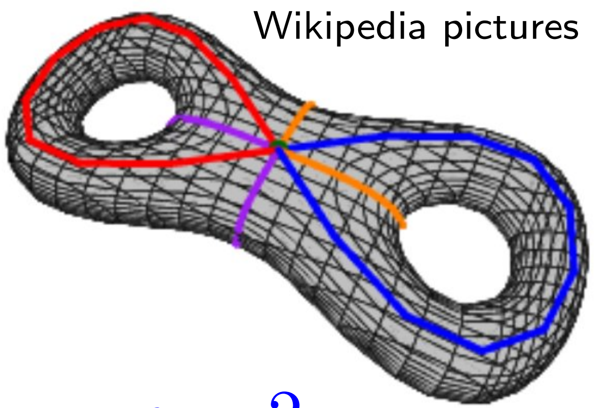




weakly convex periodic straight-line drawing

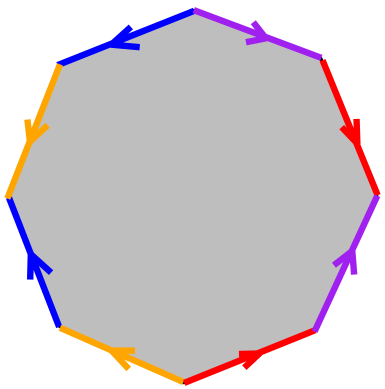
- Extension of our method to higher genus ?

Wikipedia pictures



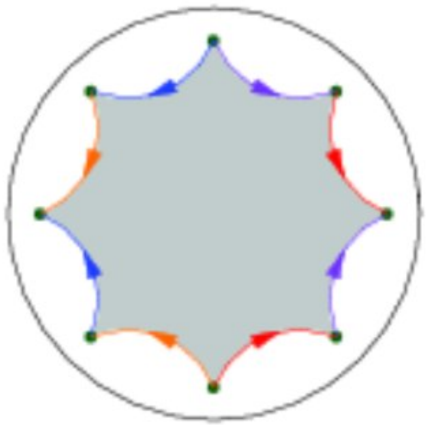
$g = 2$

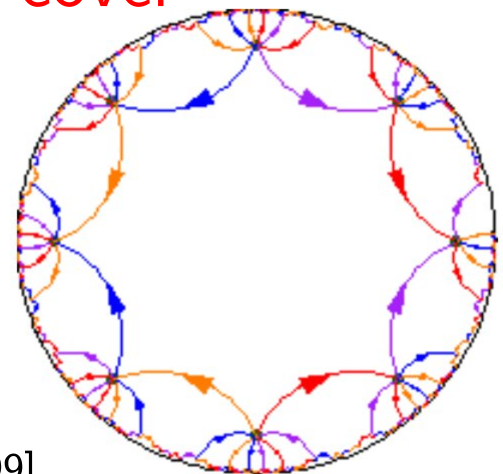
Polygonal scheme



[Duncan, Goodrich, Kobourov'99]
[Chambers, Eppstein, Goodrich, Löffler'00]
drawing in polynomial area

Universal cover





[Mohar'99]
periodic drawing
out of circle packing