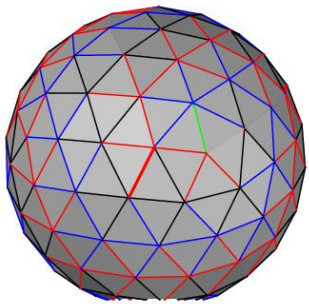


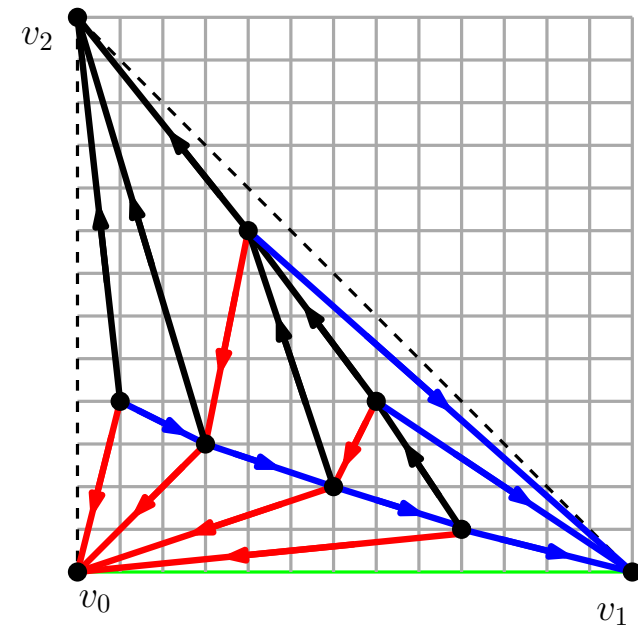
Schnyder woods for higher genus surfaces: from graph encoding to graph drawing



JCB 2014, Labri

Luca Castelli Aleardi

(joint works with O. Devillers, E. Fusy, A. Kostygin, T. Lewiner)



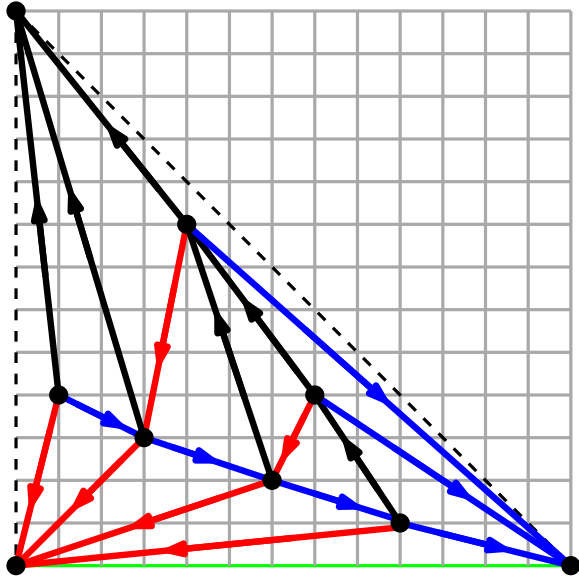
Some facts about planar graphs

("As I have known them")

Some facts about planar graphs

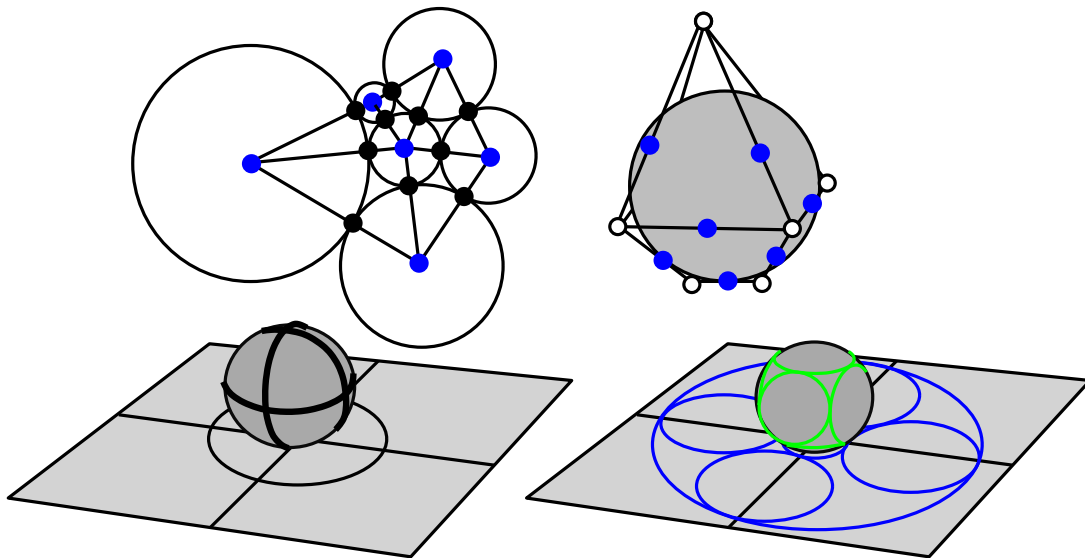
Thm (Schnyder, Trotter, Felsner)

G planar if and only if $\dim(G) \leq 3$



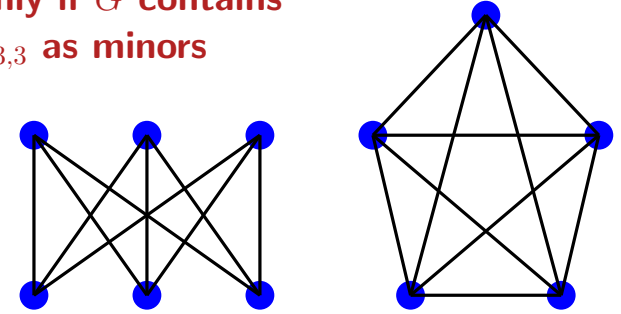
Thm (Koebe-Andreev-Thurston)

Every planar graph with n vertices is isomorphic to the intersection graph of n disks in the plane.



Thm (Kuratowski, excluded minors)

G planar if and only if G contains neither K_5 nor $K_{3,3}$ as minors

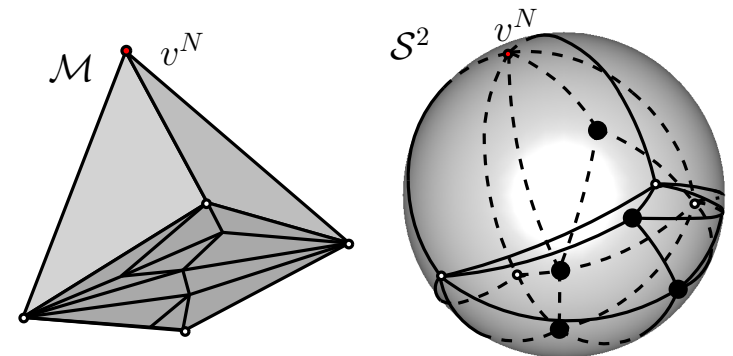


Thm (Y. Colin de Verdière)

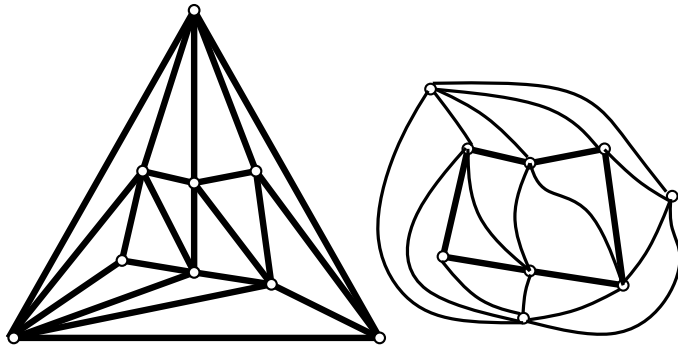
G planar if and only if $\mu(G) \leq 3$

($\mu(G)$ = multiplicity of λ_2 of a generalized laplacian)

$$L_G = \begin{bmatrix} 4 & -1 & \dots & \dots & 0 \\ -1 & 5 & \dots & & \\ \dots & \dots & \dots & & \\ \dots & & & \dots & \\ 0 & \dots & & & 3 \end{bmatrix} \quad L_G[i, k] = \begin{cases} \deg(v_i) \\ -A_G[i, j] \end{cases}$$

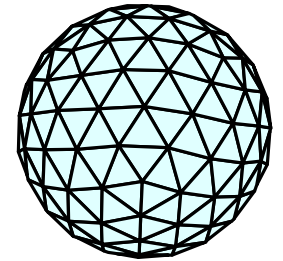


Planar triangulations



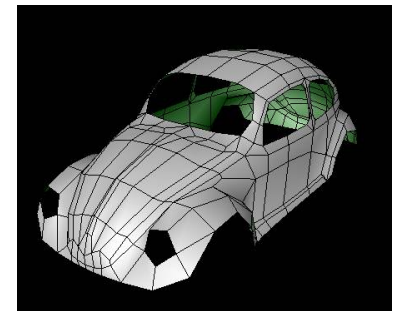
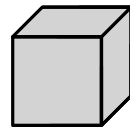
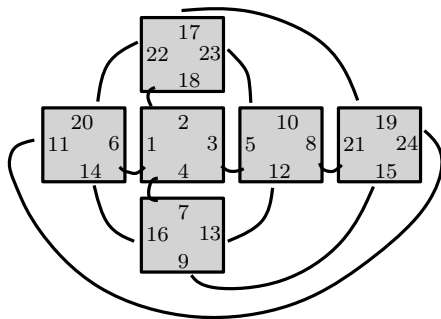
$$n - e + f = 2$$

$$e = 3n - 6$$

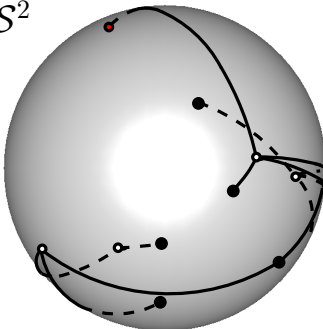


$$\phi = (1, 2, 3, 4)(17, 23, 18, 22)(5, 10, 8, 12)(21, 19, 24, 15) \dots$$

$$\alpha = (2, 18)(4, 7)(12, 13)(9, 15)(14, 16)(10, 23) \dots$$



S^2



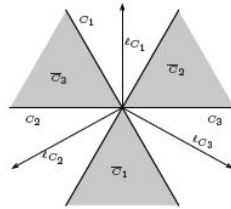
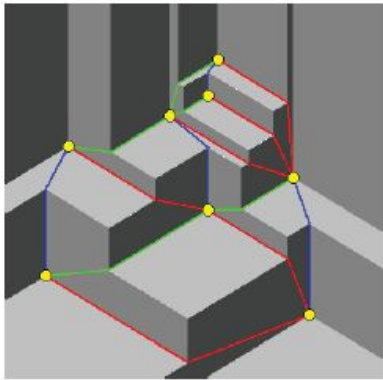
Schnyder woods and canonical orderings: overview of applications

(**graph drawing**, **graph encoding**, succinct representations, compact data structures, exhaustive graph enumeration, bijective counting, greedy drawings, spanners, contact representations, planarity testing, untangling of planar graphs, Steinitz representations of polyhedra, ...)

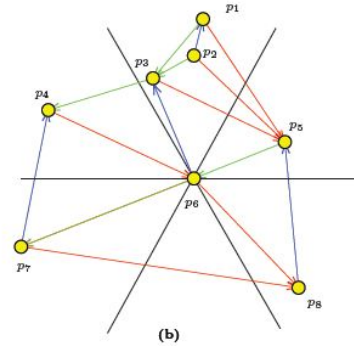
More ("recent") applications

Schnyder woods, TD-Delaunay graphs, orthogonal surfaces and Half- Θ_6 -graphs

[Bonichon et al., WG'10, Icalp '10, ...]



(a)



(b)

Figure 2: A coplanar orthogonal surface with its geodesic embedding.

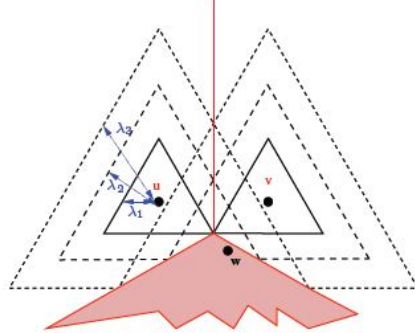
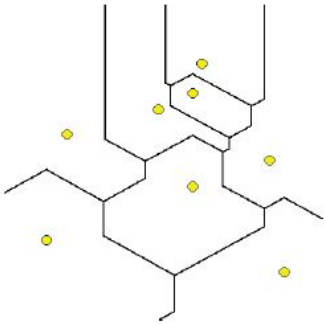
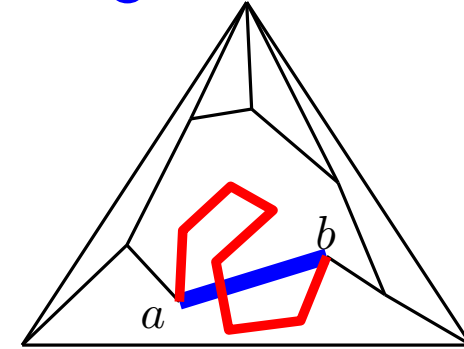
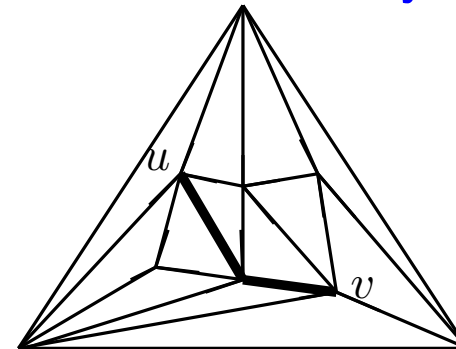


Figure 3: (a) TD-Voronoi diagram. (b) $\lambda_1 < \lambda_2 < \lambda_3$ stand for three triangular distances. Set $\{u, v\}$ is an ambiguous point set, however $\{u, v, w\}$ is non-ambiguous.

Greedy routing



Every planar triangulation admits a **greedy drawing** (Dhandapani, Soda08)

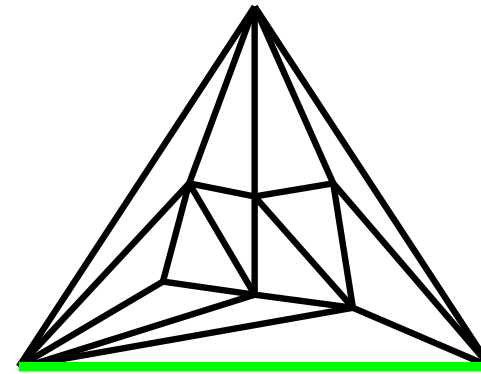
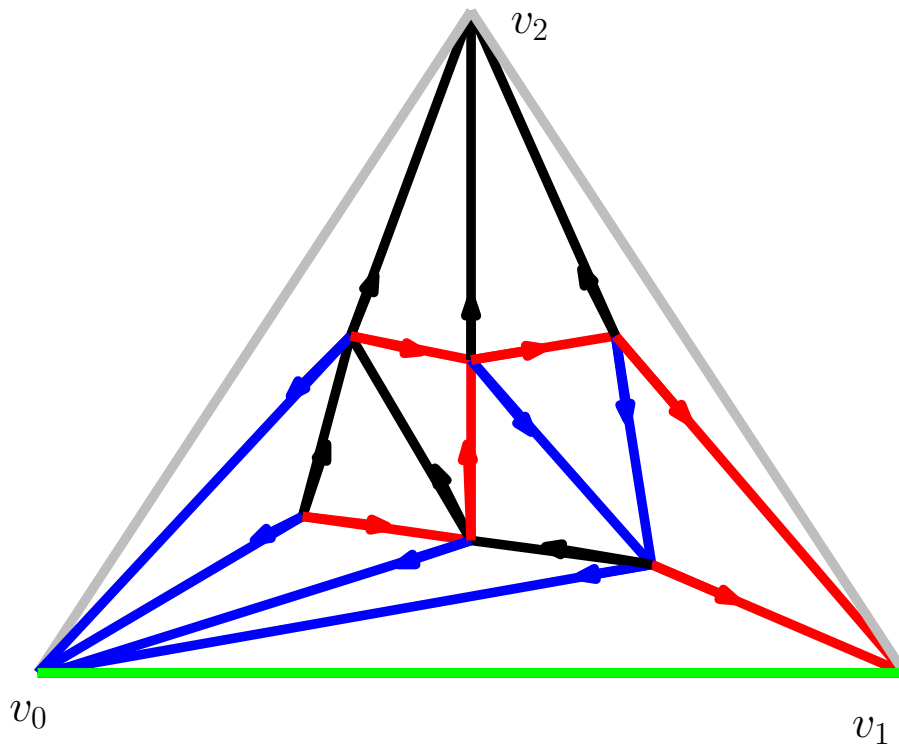
(conjectured by Papadimitriou and Ratajczak for 3-connected planar graphs)

Schnyder woods

(the definition)

Schnyder woods: (planar) definition

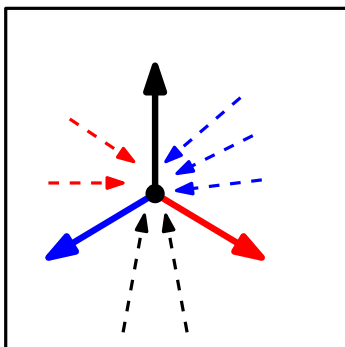
[Schnyder '90]



rooted triangulation on
 n nodes

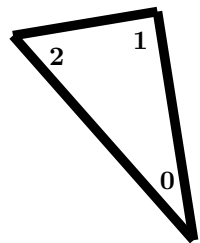
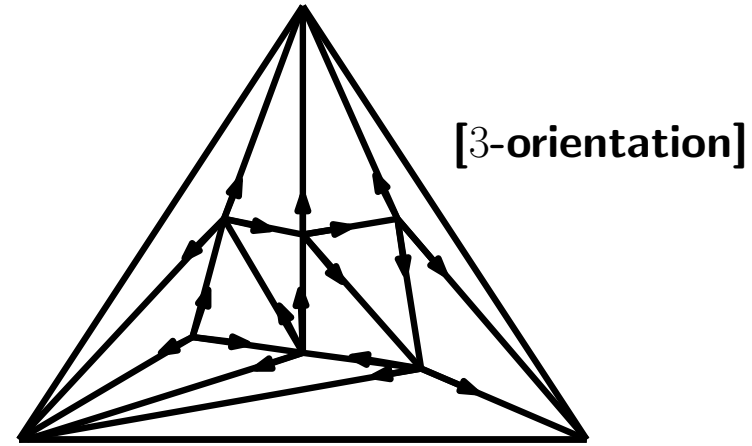
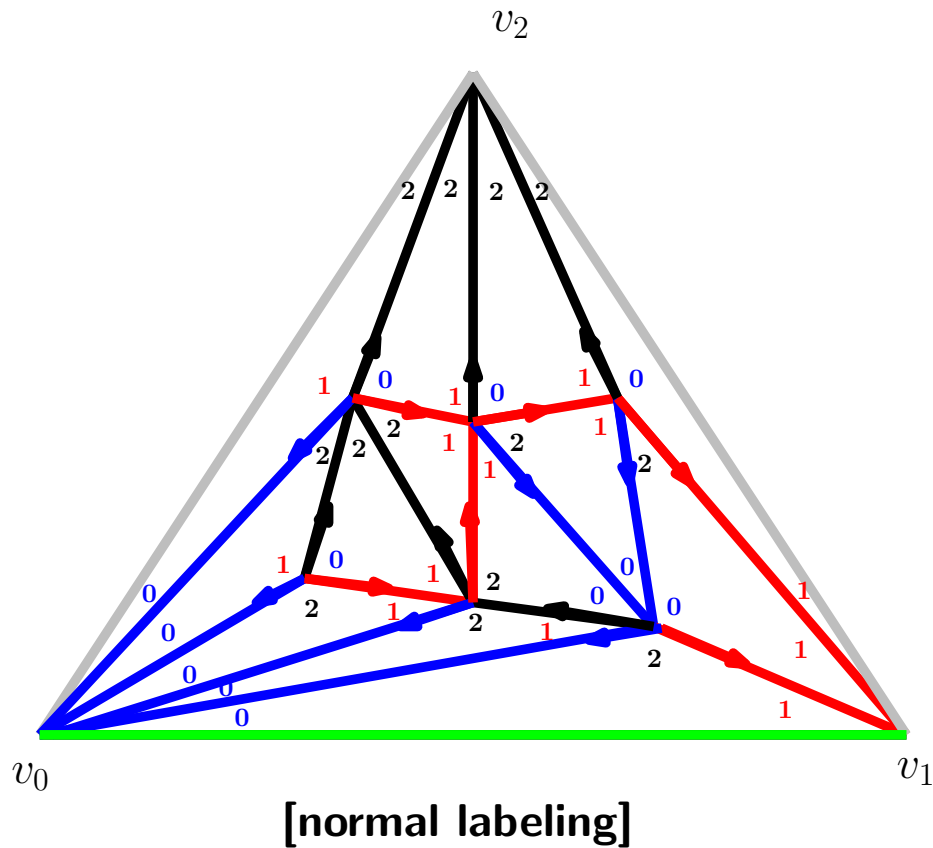
A Schnyder wood of a (rooted) planar triangulation is partition of all inner edges into three sets T_0 , T_1 and T_2 such that

i) edge are colored and oriented in such a way that each inner nodes has exacty one outgoing edge of each color

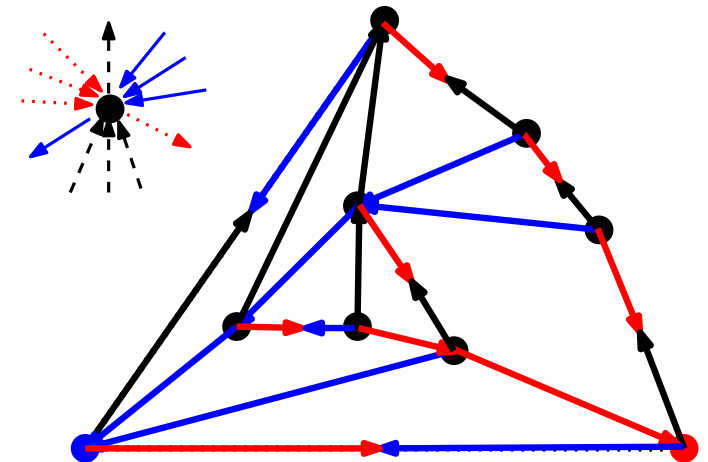


ii) colors and orientations around each inner node must respect the local Schnyder condition

Schnyder woods: equivalent formulation

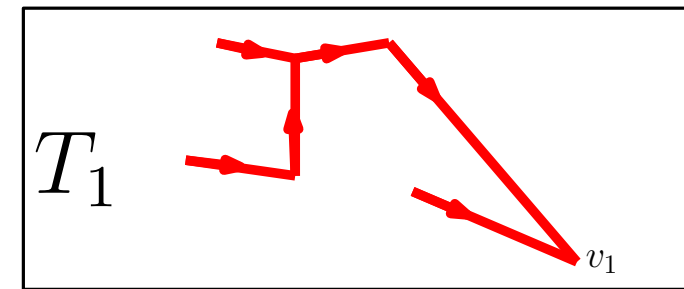
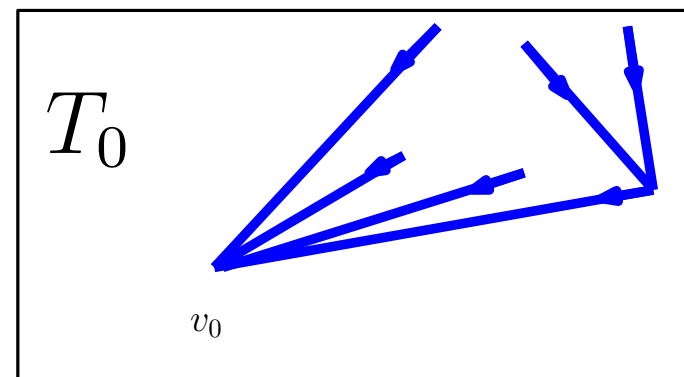
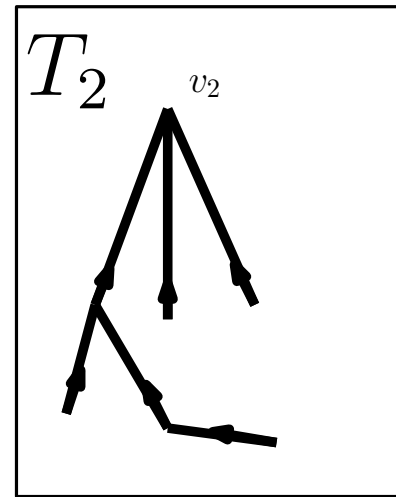
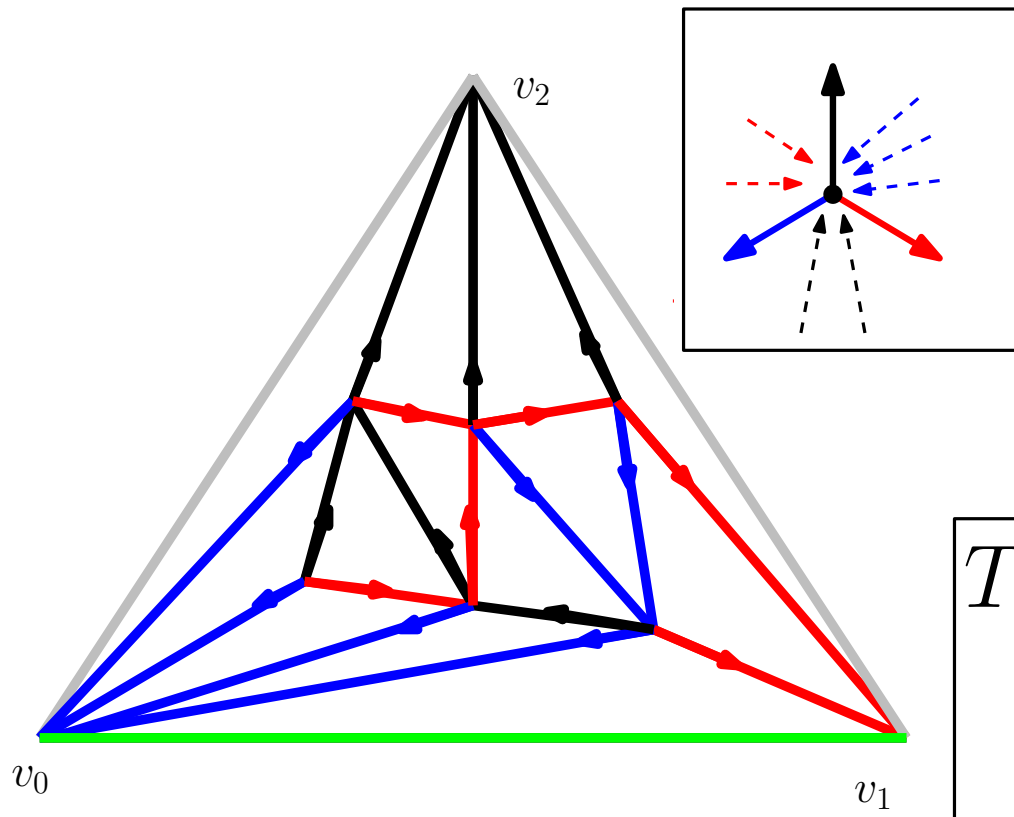


3-connected graphs [Felsner]



Schnyder woods: spanning property

[Schnyder '90]



Theorem

The three sets T_0 , T_1 , T_2 are spanning trees of the inner vertices of \mathcal{T} (each rooted at vertex v_i)

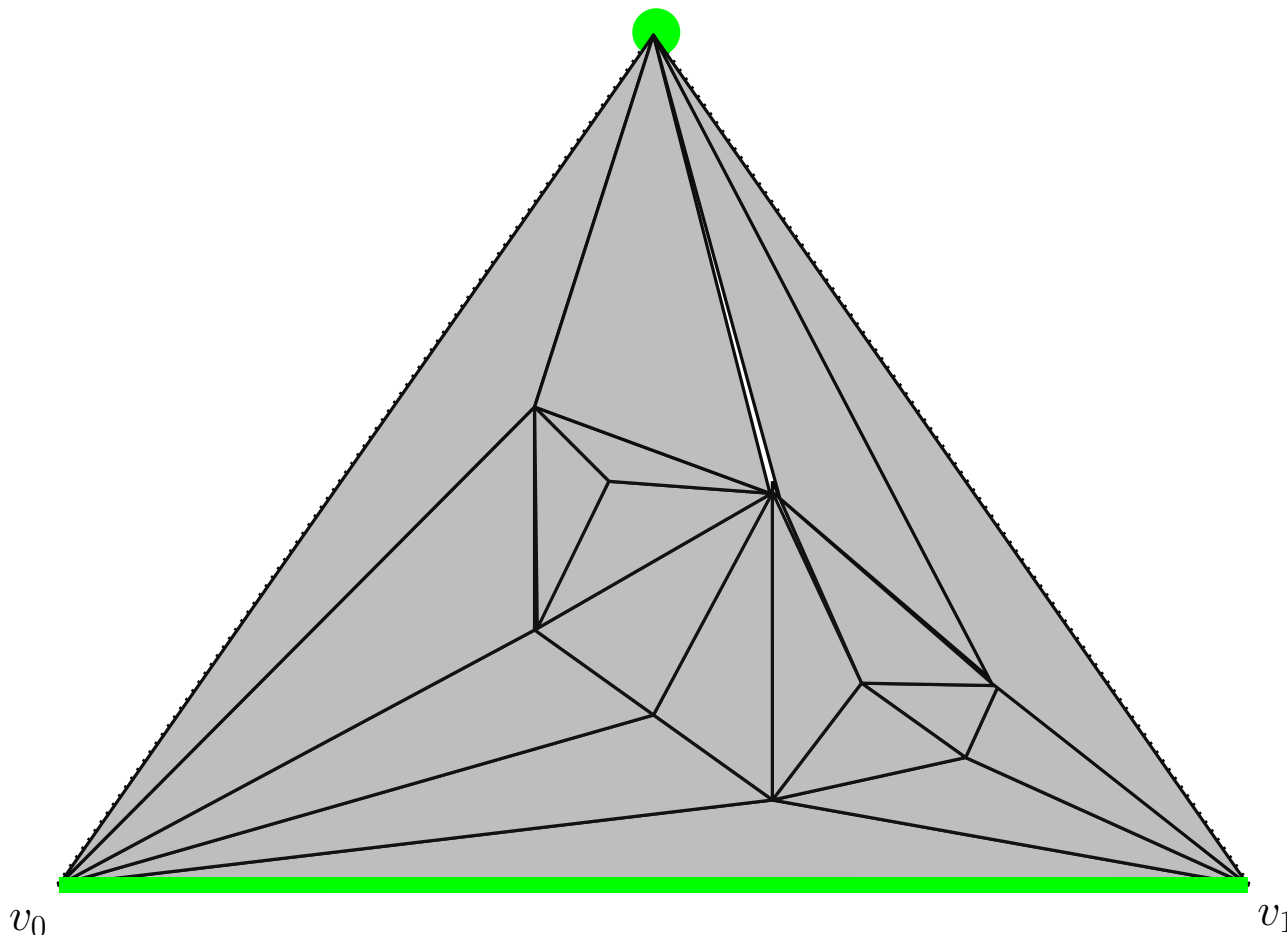
Schnyder woods: existence (algorithm I)

[incremental vertex shelling, Brehm's thesis]

The traversal starts from the root face

Theorem

Every planar triangulation admits a Schnyder wood, which can be computed in linear time.



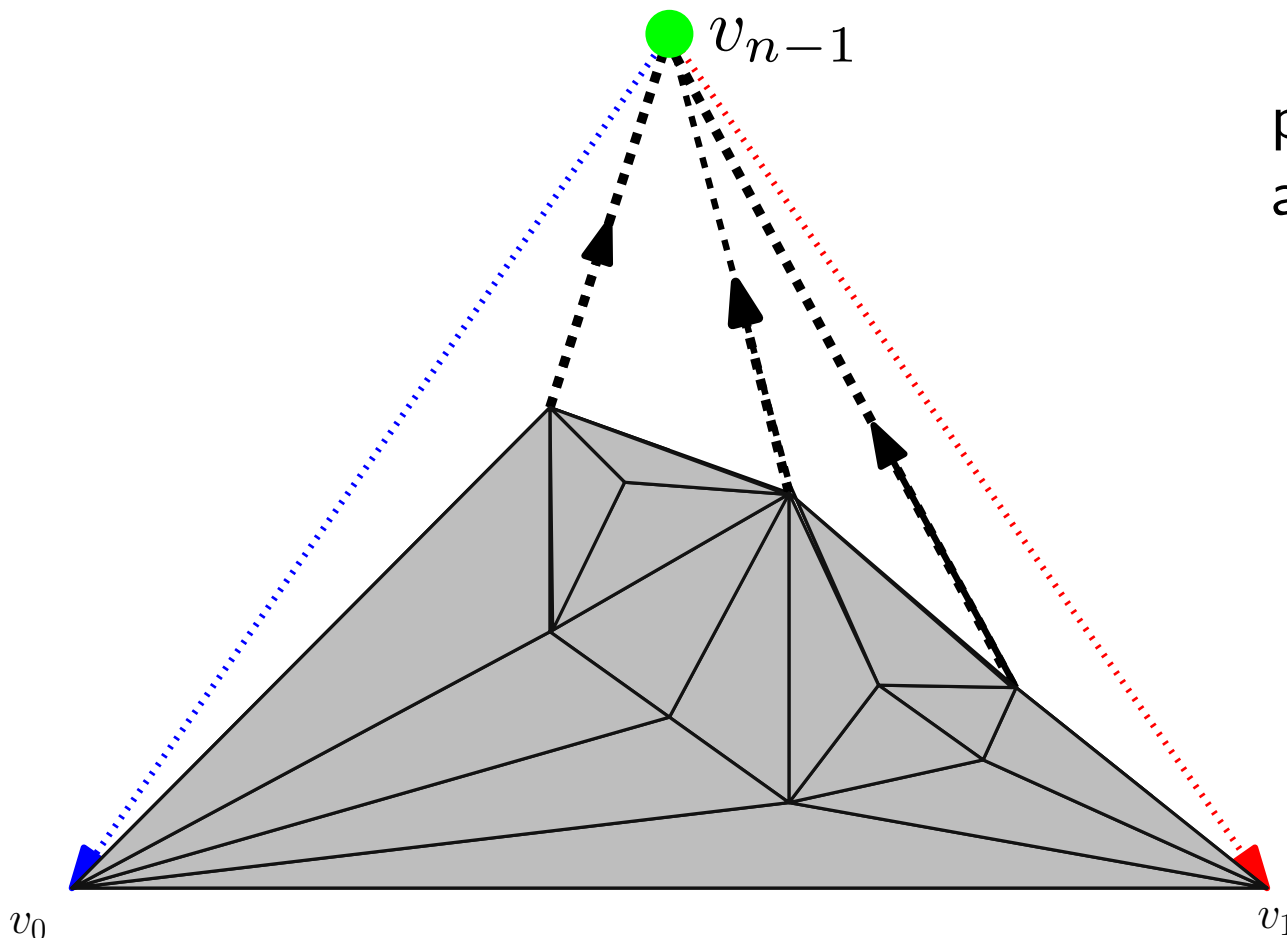
Schnyder woods: existence (algorithm I)

[incremental vertex shelling, Brehm's thesis]

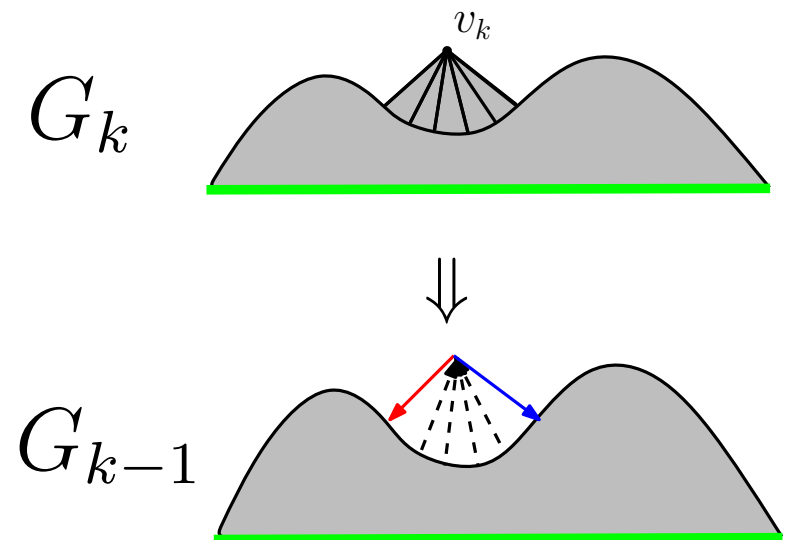
The traversal starts from the root face

Theorem

Every planar triangulation admits a Schnyder wood, which can be computed in linear time.



perform a vertex conquest at each step



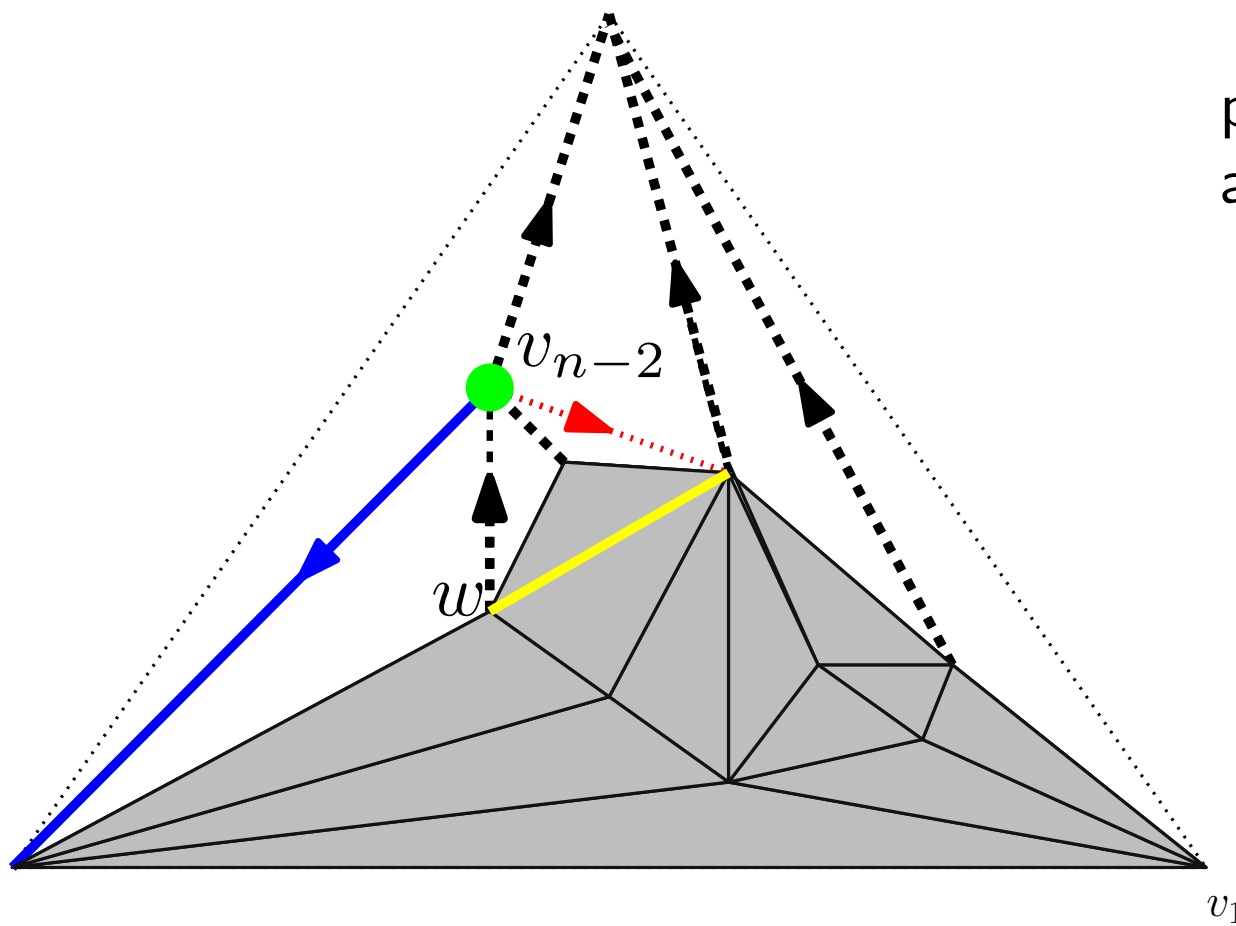
Schnyder woods: existence (algorithm I)

[incremental vertex shelling, Brehm's thesis]

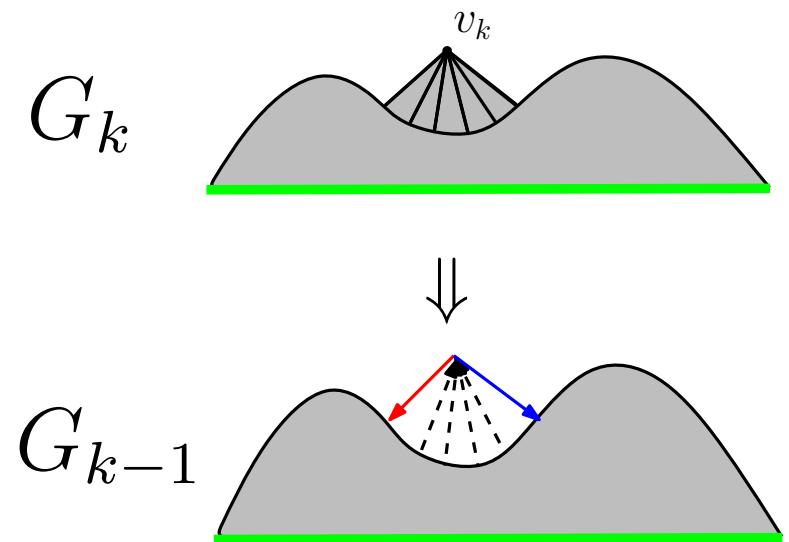
The traversal starts from the root face

Theorem

Every planar triangulation admits a Schnyder wood, which can be computed in linear time.



perform a vertex conquest at each step



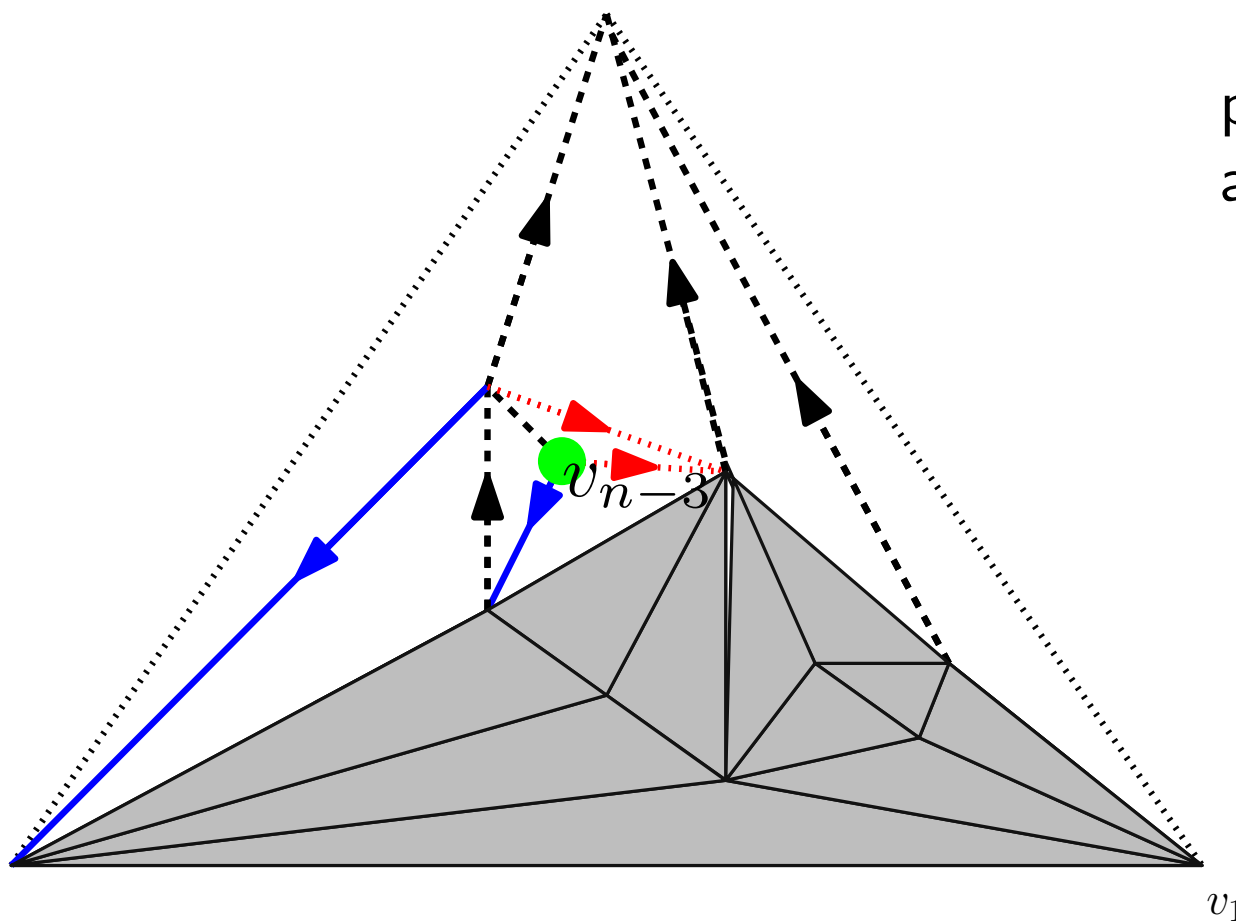
Schnyder woods: existence (algorithm I)

[incremental vertex shelling, Brehm's thesis]

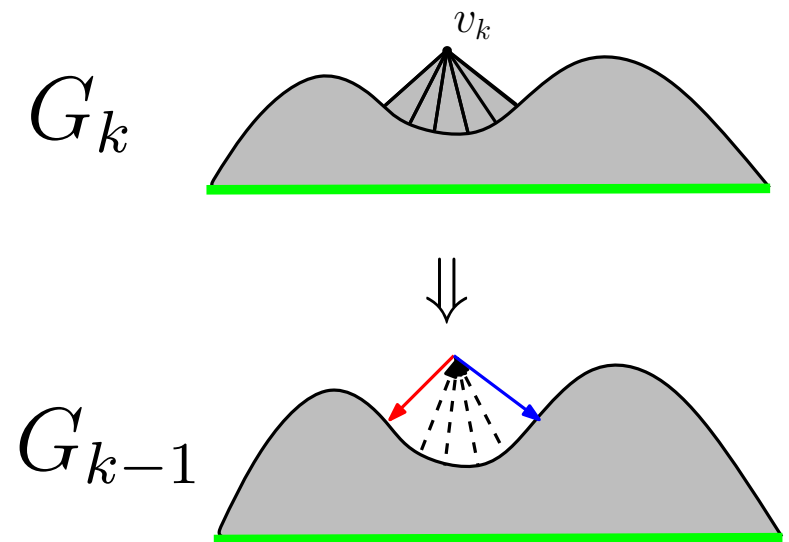
The traversal starts from the root face

Theorem

Every planar triangulation admits a Schnyder wood, which can be computed in linear time.



perform a vertex conquest at each step



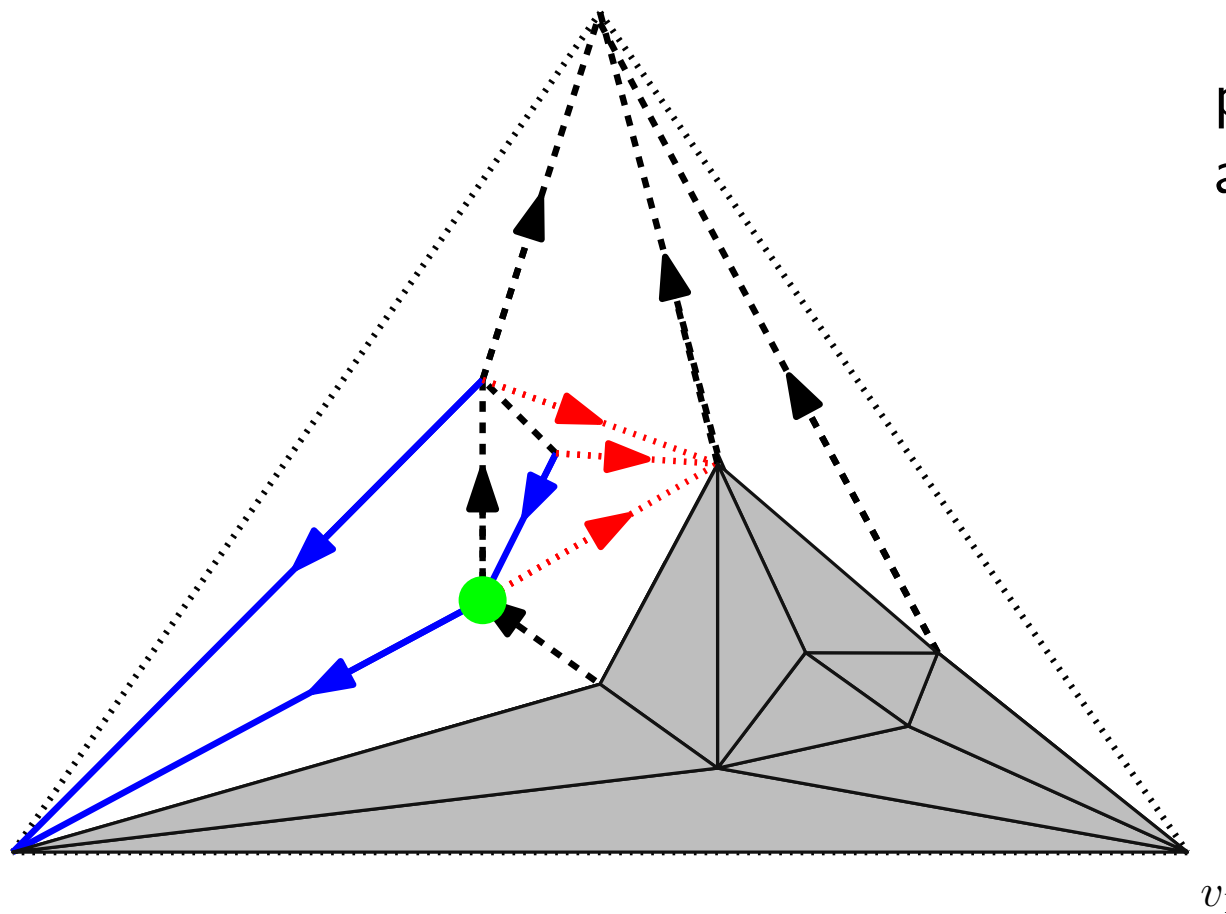
Schnyder woods: existence (algorithm I)

[incremental vertex shelling, Brehm's thesis]

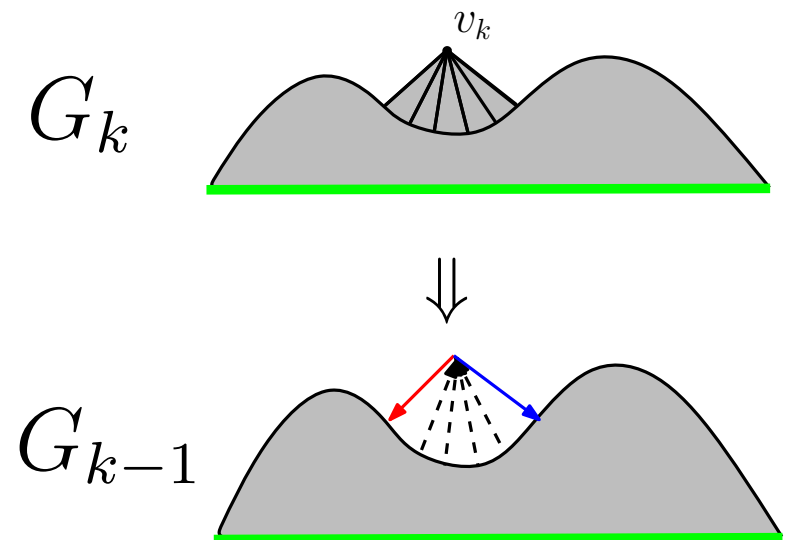
The traversal starts from the root face

Theorem

Every planar triangulation admits a Schnyder wood, which can be computed in linear time.



perform a vertex conquest at each step



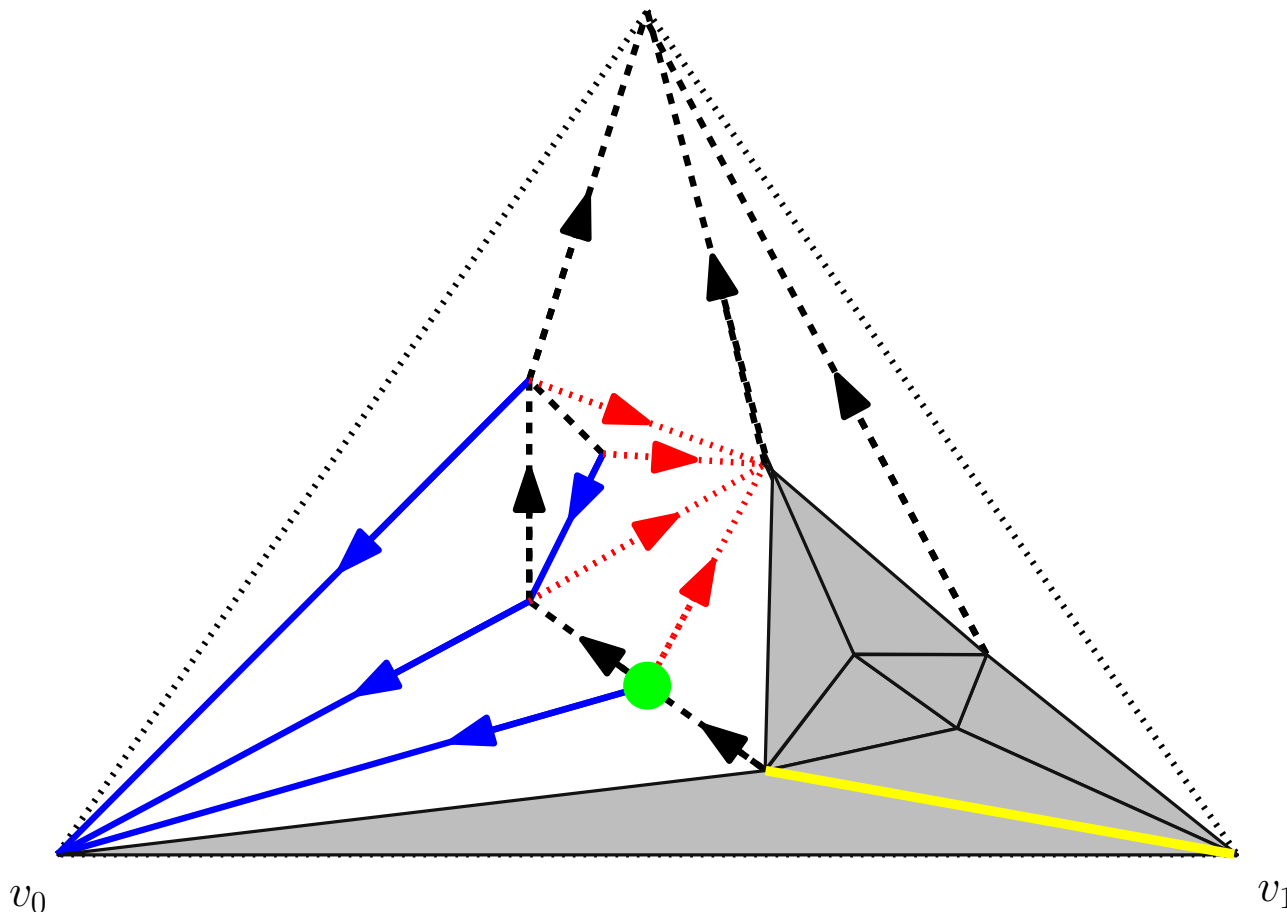
Schnyder woods: existence (algorithm I)

[incremental vertex shelling, Brehm's thesis]

The traversal starts from the root face

Theorem

Every planar triangulation admits a Schnyder wood, which can be computed in linear time.



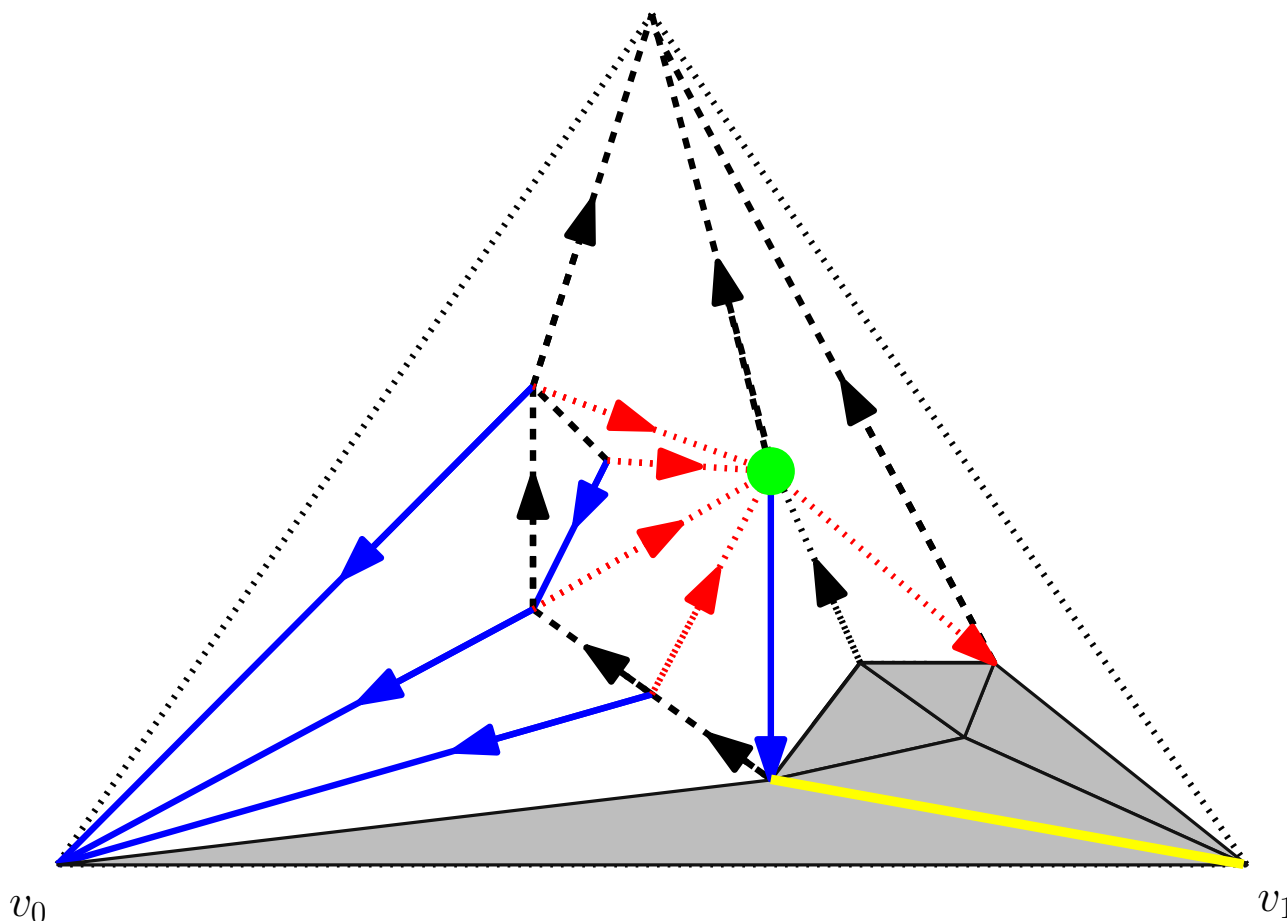
Schnyder woods: existence (algorithm I)

[incremental vertex shelling, Brehm's thesis]

The traversal starts from the root face

Theorem

Every planar triangulation admits a Schnyder wood, which can be computed in linear time.



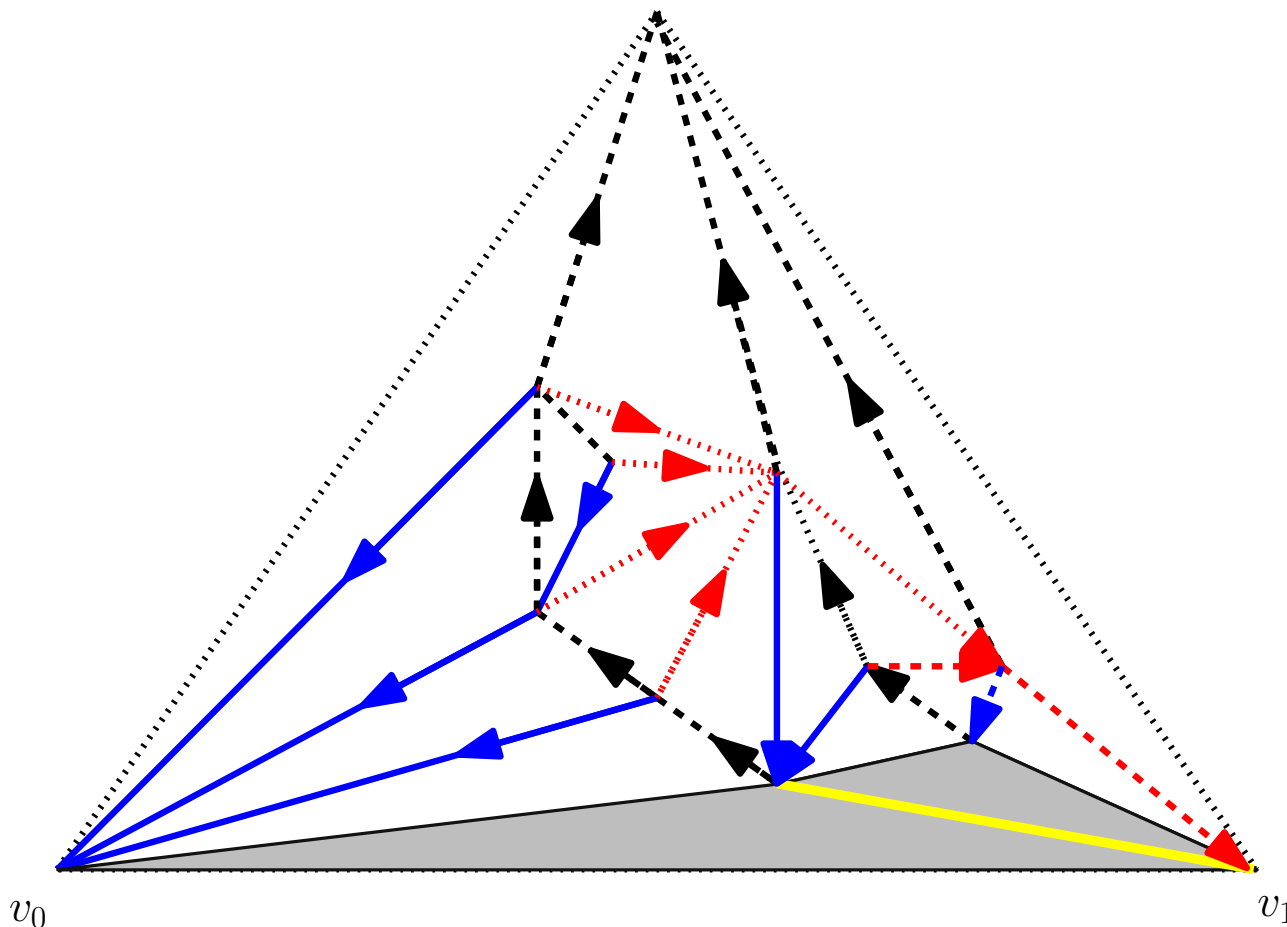
Schnyder woods: existence (algorithm I)

[incremental vertex shelling, Brehm's thesis]

The traversal starts from the root face

Theorem

Every planar triangulation admits a Schnyder wood, which can be computed in linear time.



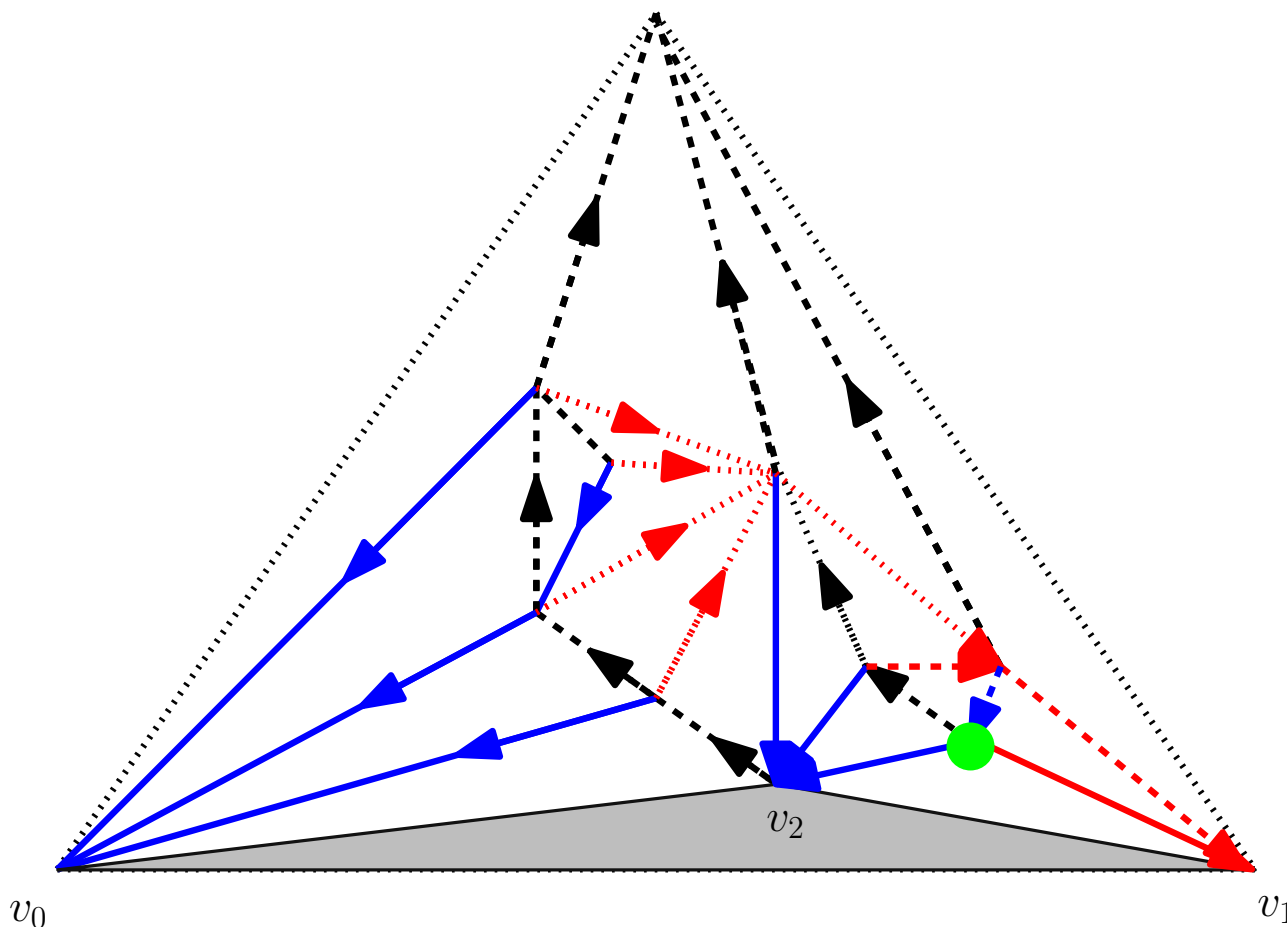
Schnyder woods: existence (algorithm I)

[incremental vertex shelling, Brehm's thesis]

The traversal starts from the root face

Theorem

Every planar triangulation admits a Schnyder wood, which can be computed in linear time.



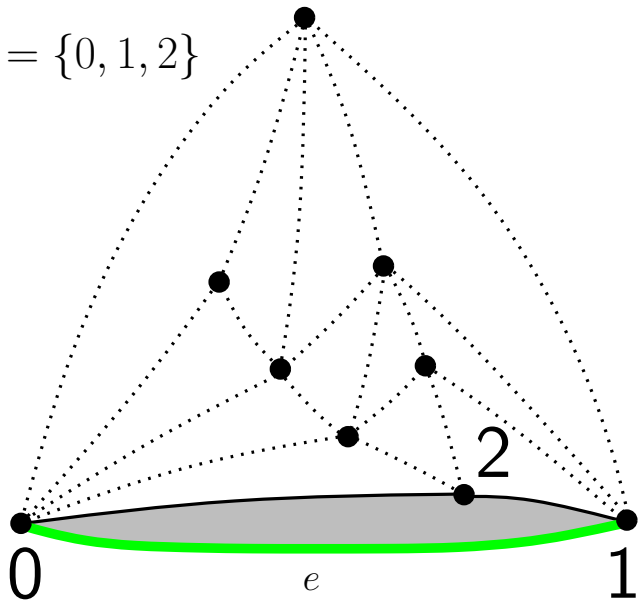
Canonical orderings

(the definition)

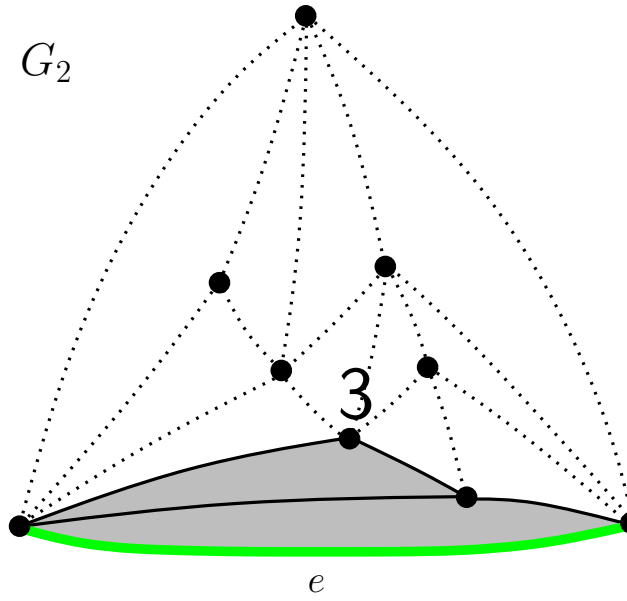
Canonical orderings: definition

[de Fraysseix Pach Pollack]

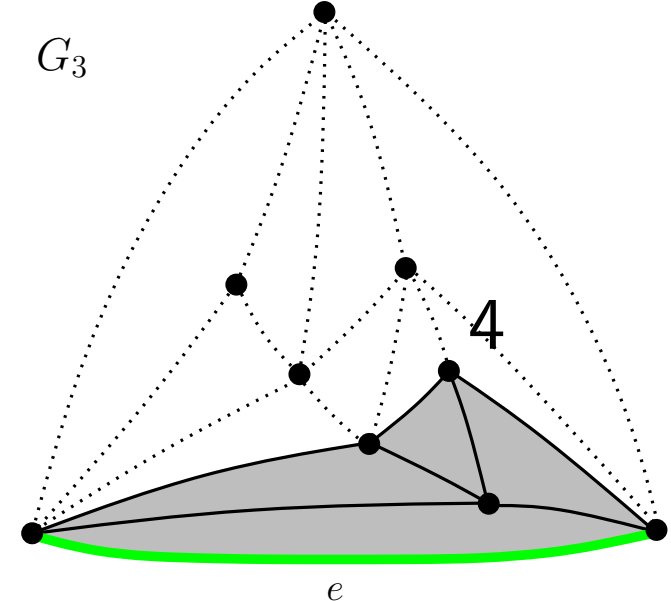
$G_1 = \{0, 1, 2\}$



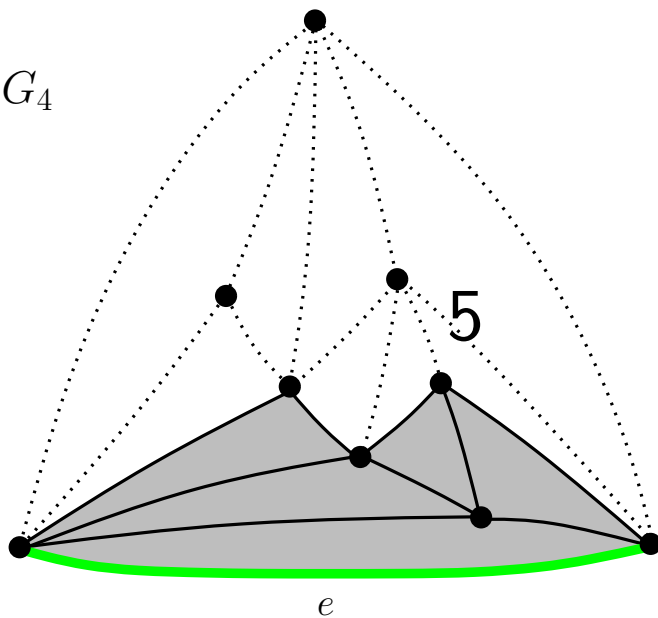
G_2



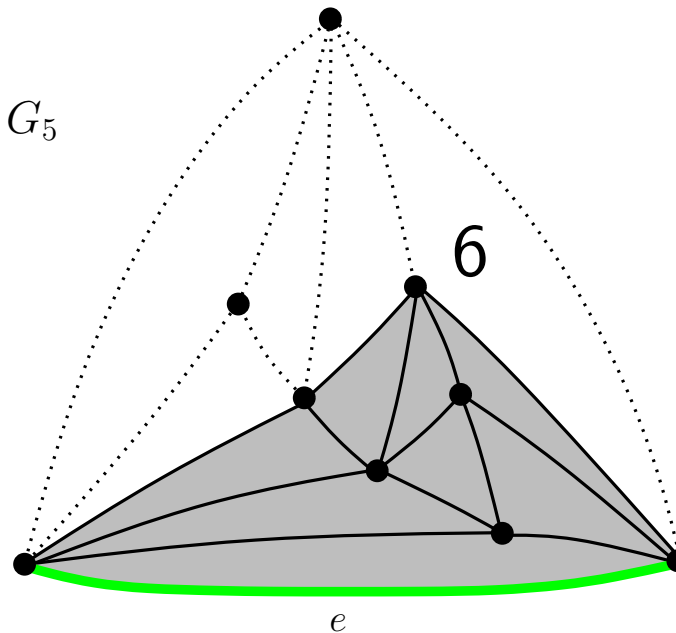
G_3



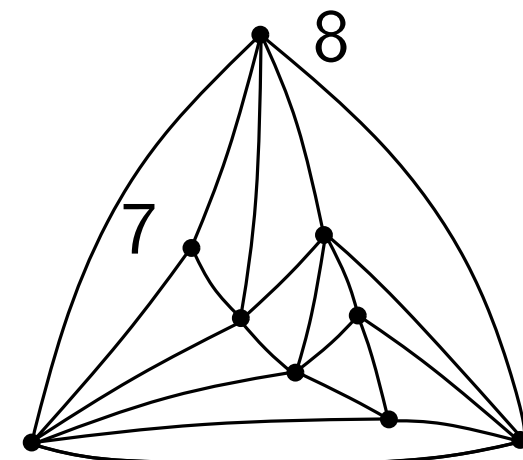
G_4



G_5



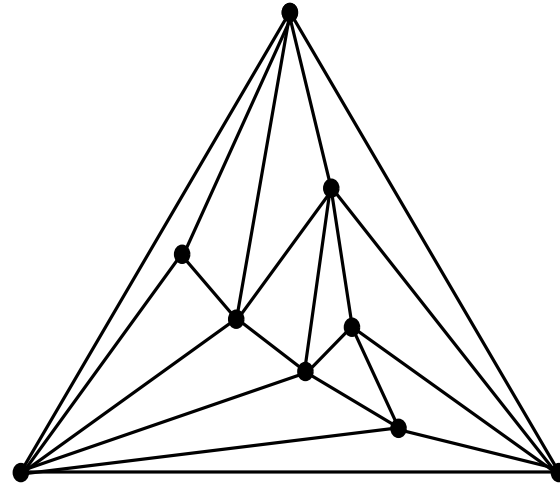
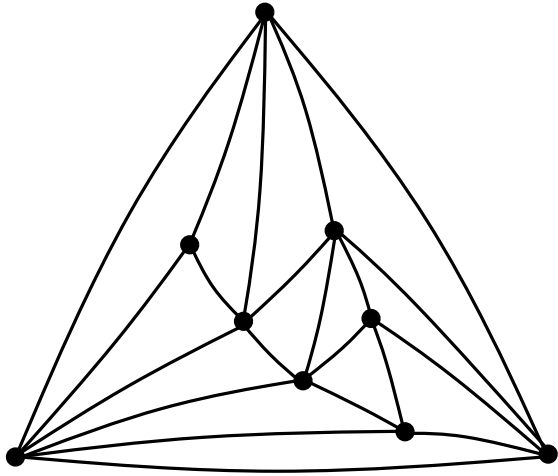
$G = G_7$



Planar straight-line drawings

(of planar graphs)

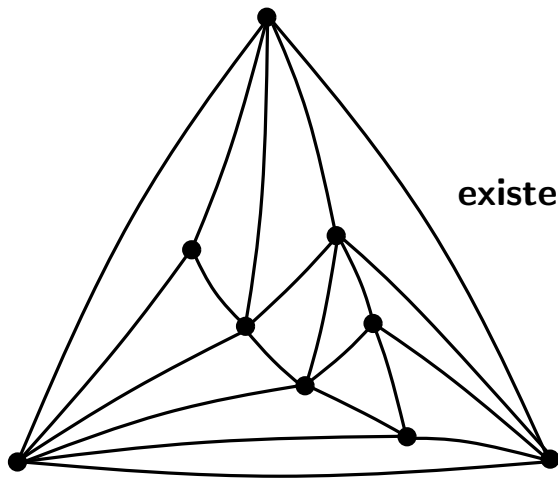
Planar straight-line drawings



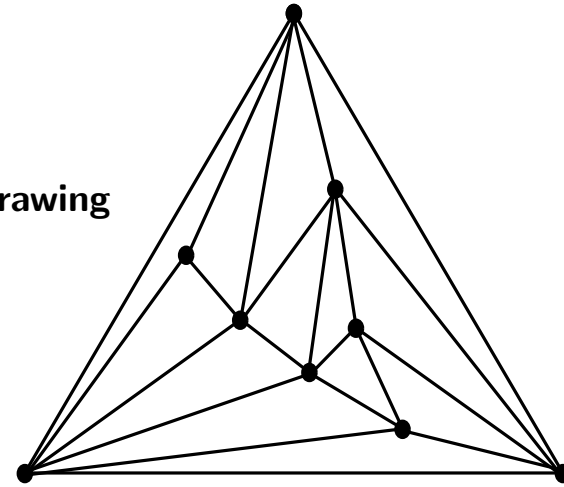
[Wagner'36]

[Fary'48]

Planar straight-line drawings



existence of straight-line drawing

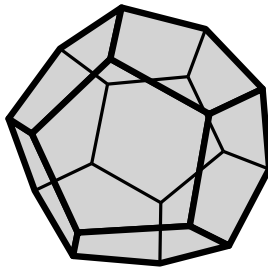
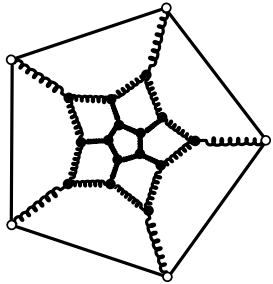


[Wagner'36]

[Fary'48]

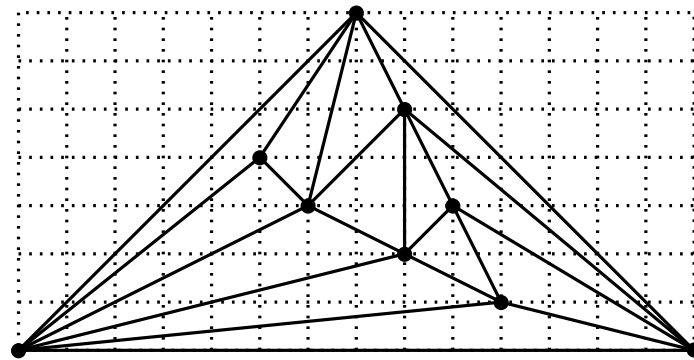
[Stein'51]

Classical algorithms:



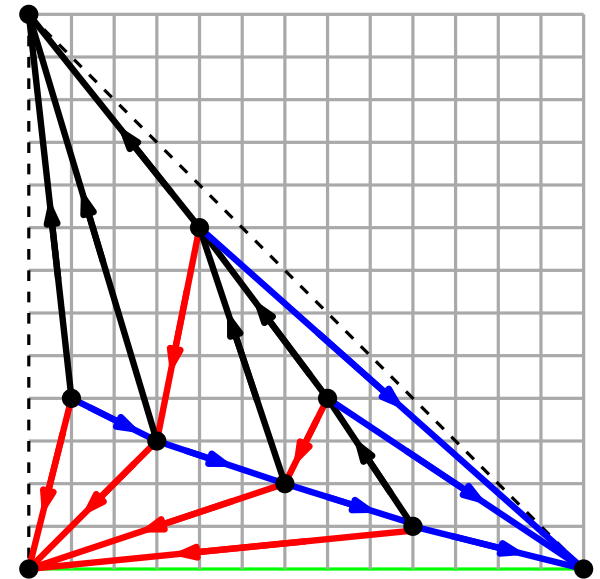
[Tutte'63]

spring-embedding



[De Fraysseix, Pach, Pollack 89]

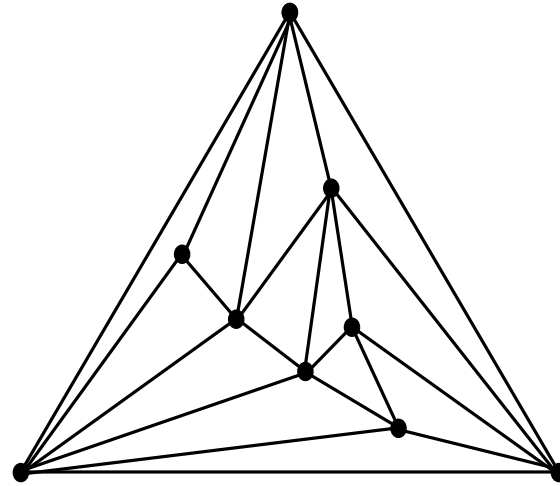
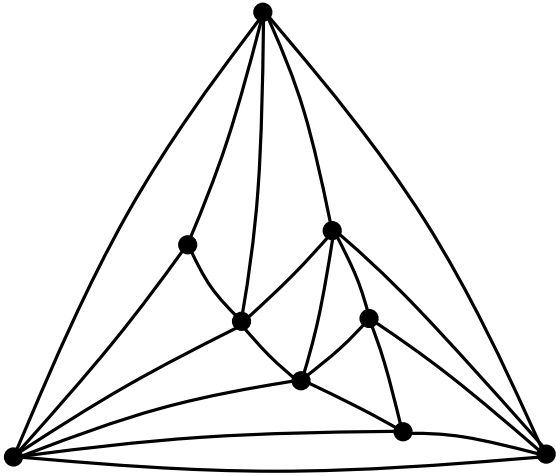
incremental (**Shift-algorithm**)



[Schnyder'90]

face-counting principle

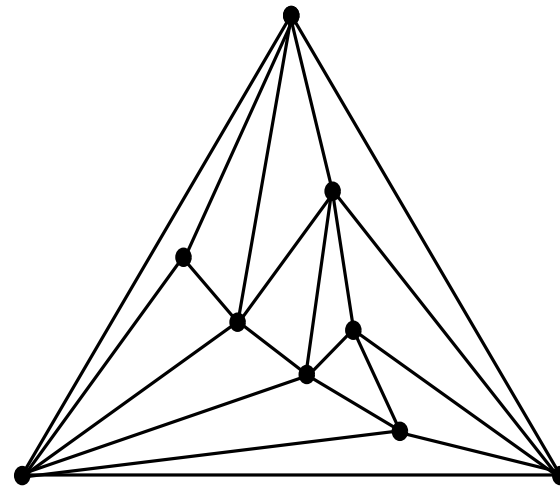
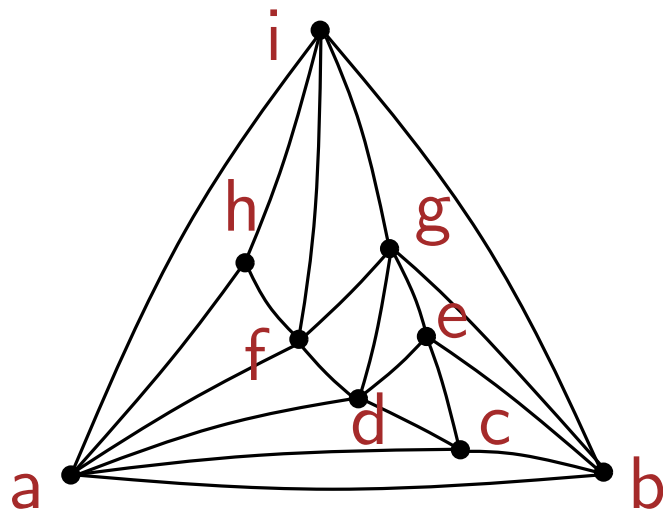
Planar straight-line drawings



[Wagner'36]

[Fary'48]

Planar straight-line grid drawings



[Wagner'36]

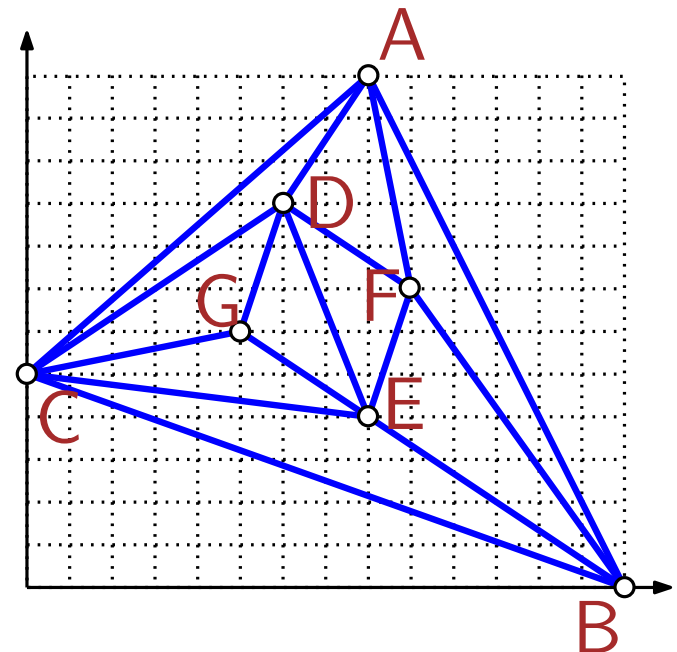
[Fary'48]

Input of the problem
set of triangle faces

Output

geometric coordinates of vertices

(a, b, c)	(d, e, g)	(i, g, b)
(a, c, d)	(e, b, g)	(i, b, a)
(d, c, e)	(a, f, h)	
(c, b, e)	(a, h, i)	
(a, d, f)	(i, h, f)	
(f, d, g)	(i, f, g)	

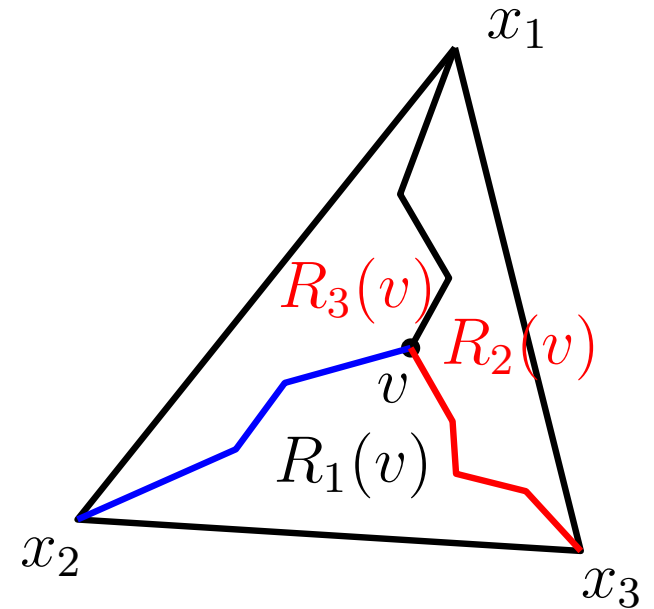
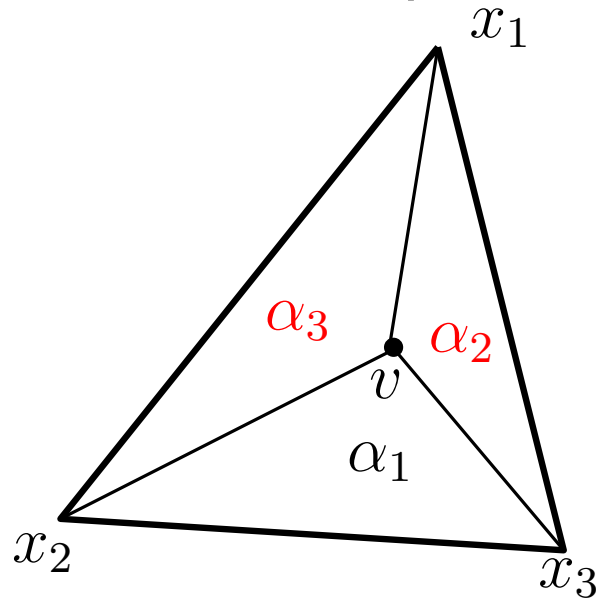


Face counting algorithm

(Schnyder algorithm, 1990)

Face counting algorithm

Geometric interpretation



$$v = \alpha_1 x_1 + \alpha_2 x_2 + \alpha_3 x_3$$

where α_i is the normalized area

$$v = \frac{|R_1(v)|}{|T|} x_1 + \frac{|R_2(v)|}{|T|} x_2 + \frac{|R_3(v)|}{|T|} x_3$$

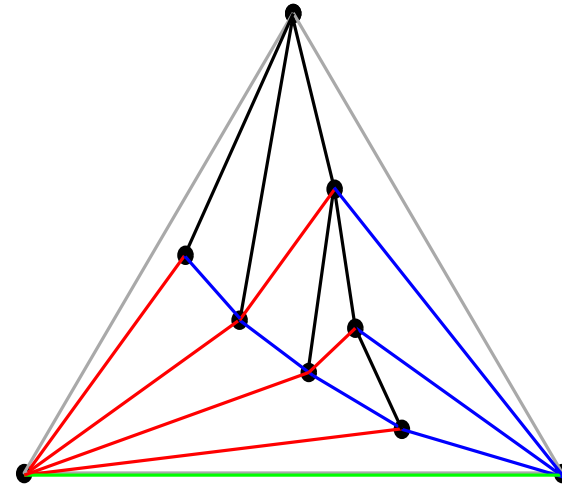
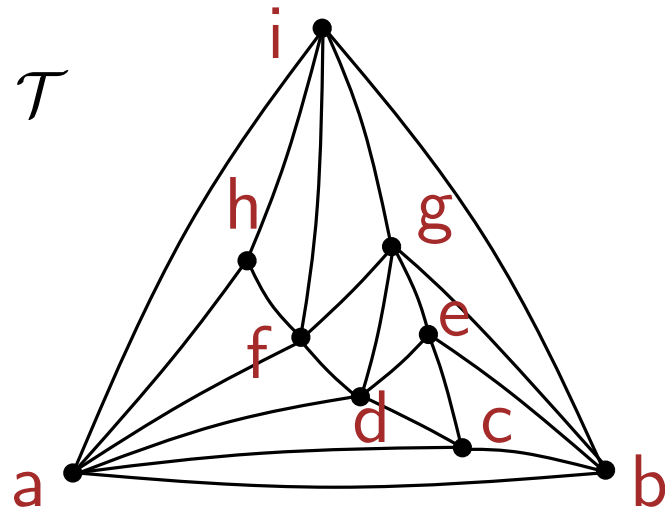
where $|R_i(v)|$ is the number of triangles

Theorem (Schnyder, Soda '90)

For a triangulation \mathcal{T} having n vertices, we can draw it on a grid of size $(2n - 5) \times (2n - 5)$, by setting $x_1 = (2n - 5, 0)$, $x_2 = (0, 0)$ and $x_3 = (0, 2n - 5)$.

Face counting algorithm

Input: \mathcal{T}



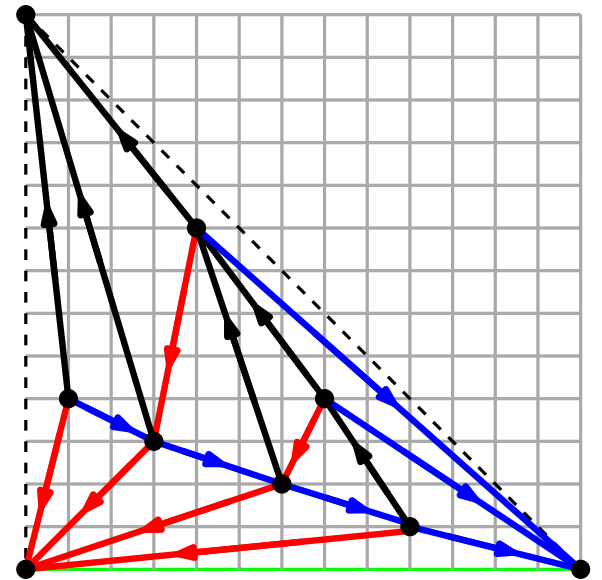
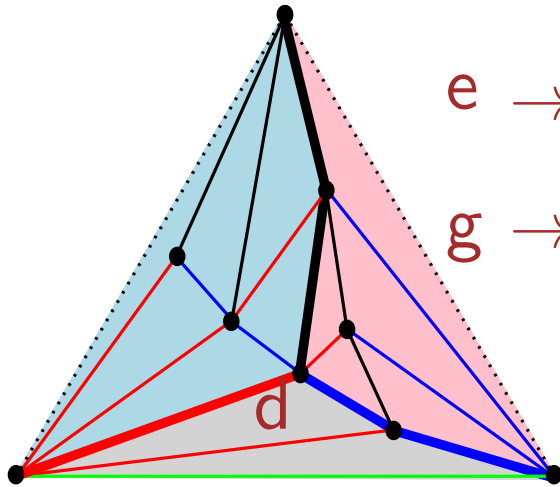
\mathcal{T} endowed with a Schnyder wood

$$a \rightarrow (0, 0) \quad b \rightarrow (0, 1) \quad i \rightarrow (1, 0)$$

$$c \rightarrow \left(\frac{9}{13}, \frac{1}{13}\right) \quad d \rightarrow \left(\frac{5}{13}, \frac{6}{13}\right)$$

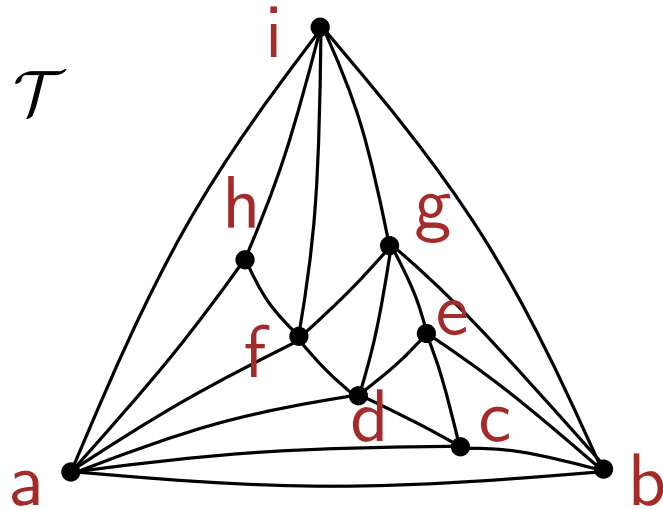
$$e \rightarrow \left(\frac{7}{13}, \frac{4}{13}\right) \quad f \rightarrow \left(\frac{3}{13}, \frac{3}{13}\right)$$

$$g \rightarrow \left(\frac{4}{13}, \frac{8}{13}\right) \quad h \rightarrow \left(\frac{1}{13}, \frac{4}{13}\right)$$

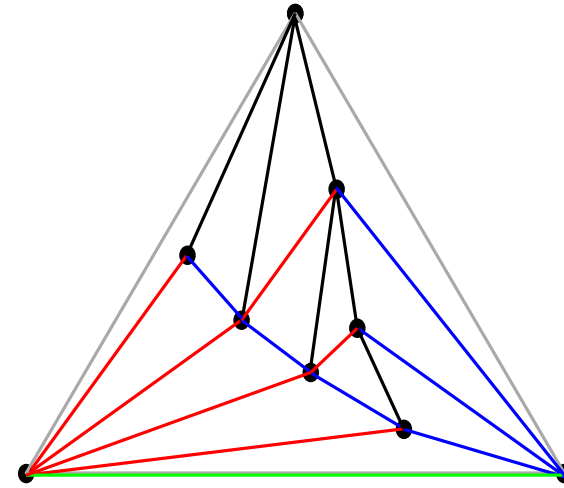


Face counting algorithm: proof (sketch)

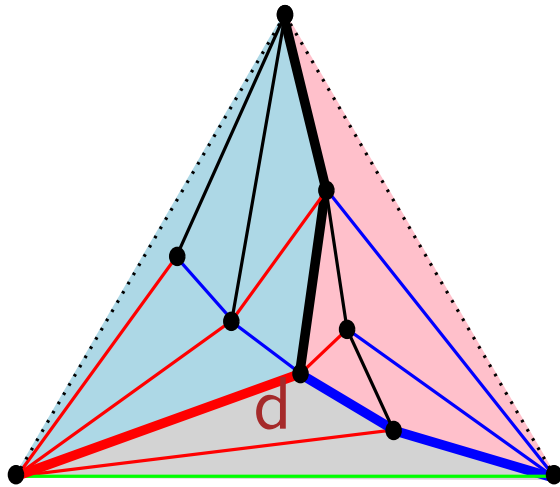
Input: \mathcal{T}



\Rightarrow



\mathcal{T} endowed with a Schnyder wood



$$a \rightarrow (13, 0, 0)$$

$$b \rightarrow (0, 13, 0)$$

$$c \rightarrow (9, 3, 1)$$

$$d \rightarrow (5, 6, 2)$$

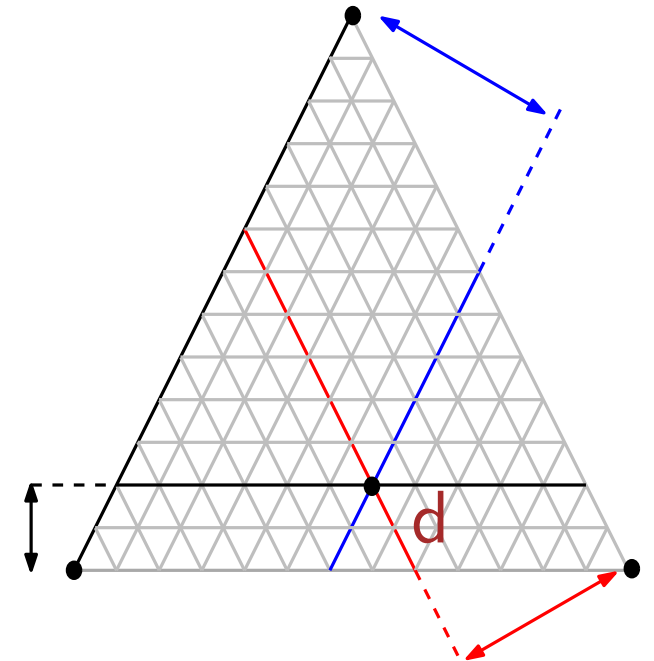
$$e \rightarrow (2, 7, 4)$$

$$f \rightarrow (7, 3, 3)$$

$$g \rightarrow (1, 4, 8)$$

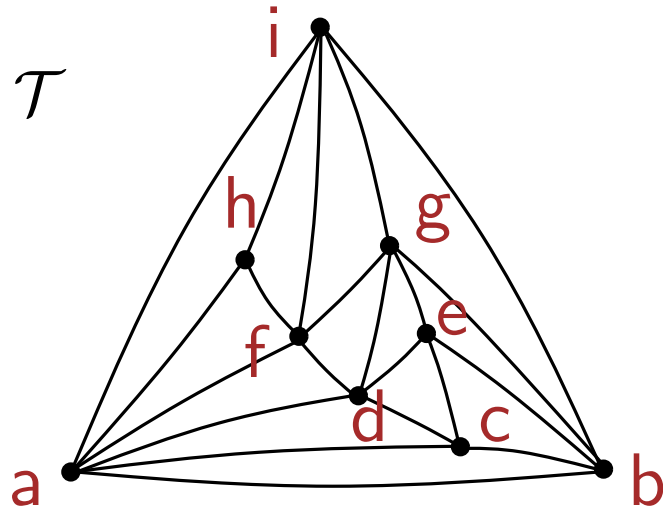
$$h \rightarrow (8, 1, 4)$$

$$i \rightarrow (0, 0, 13)$$

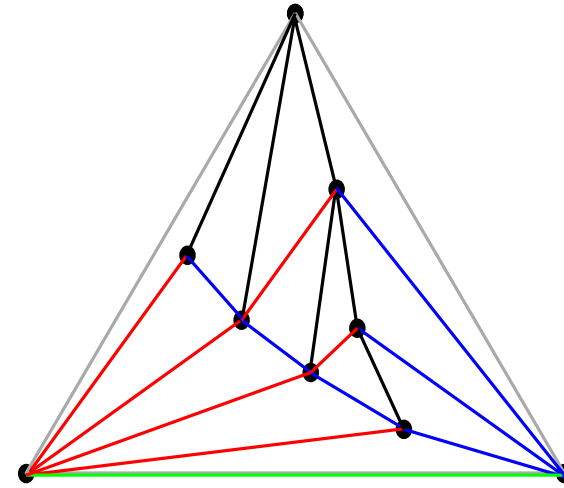


Face counting algorithm: proof (sketch)

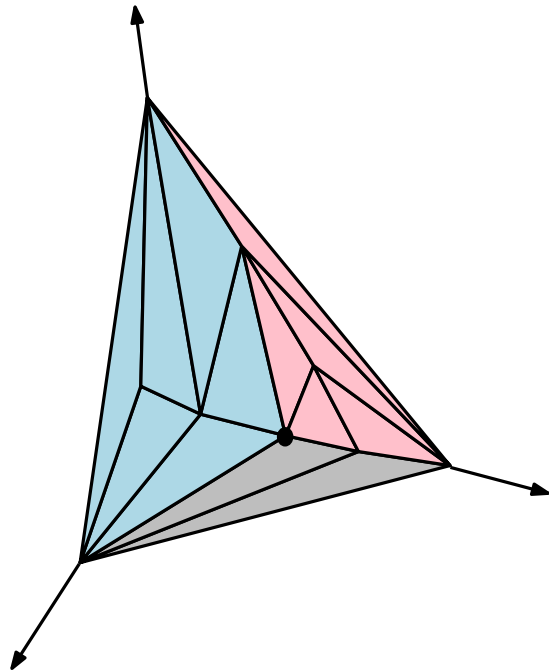
Input: \mathcal{T}



\Rightarrow



\mathcal{T} endowed with a Schnyder wood



$$a \rightarrow (13, 0, 0)$$

$$b \rightarrow (0, 13, 0)$$

$$c \rightarrow (9, 3, 1)$$

$$d \rightarrow (5, 6, 2)$$

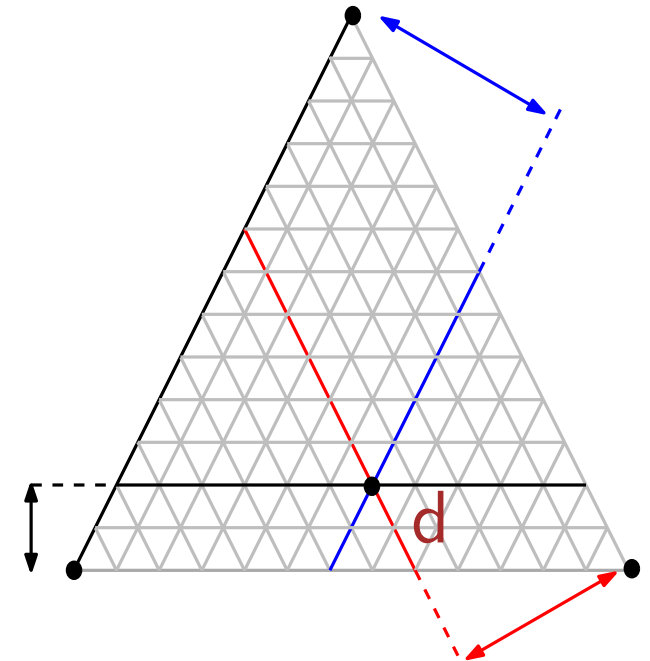
$$e \rightarrow (2, 7, 4)$$

$$f \rightarrow (7, 3, 3)$$

$$g \rightarrow (1, 4, 8)$$

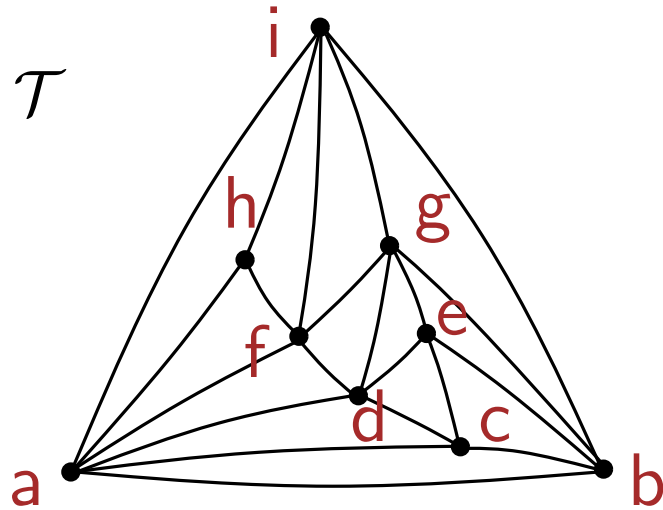
$$h \rightarrow (8, 1, 4)$$

$$i \rightarrow (0, 0, 13)$$

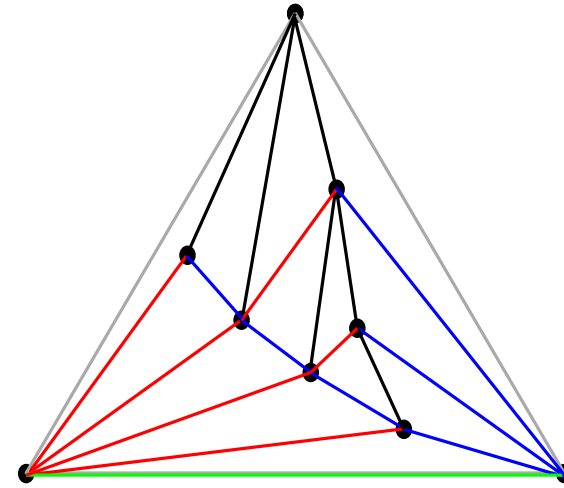


Face counting algorithm: proof (sketch)

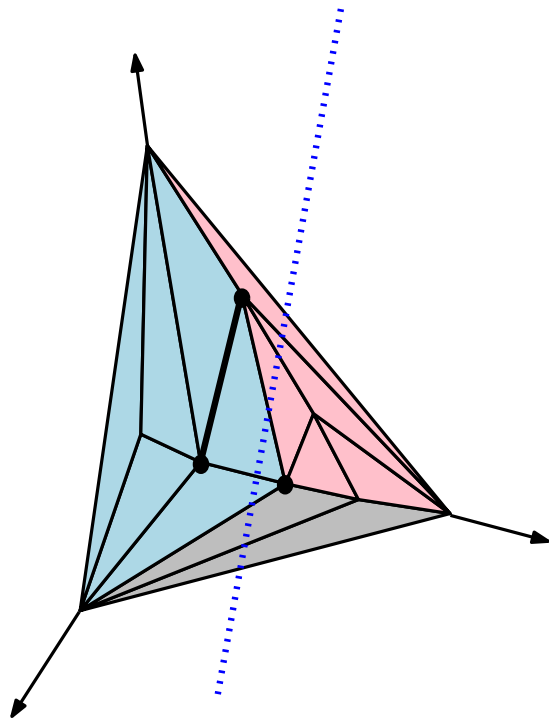
Input: \mathcal{T}



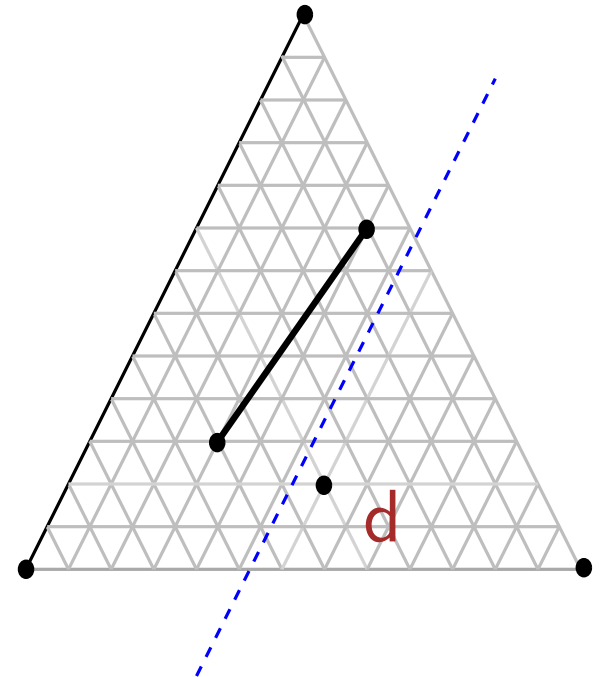
\Rightarrow



\mathcal{T} endowed with a Schnyder wood



- a** \rightarrow (13, 0, 0)
- b** \rightarrow (0, 13, 0)
- c** \rightarrow (9, 3, 1)
- d** \rightarrow (5, 6, 2)
- e** \rightarrow (2, 7, 4)
- f** \rightarrow (7, 3, 3)
- g** \rightarrow (1, 4, 8)
- h** \rightarrow (8, 1, 4)
- i** \rightarrow (0, 0, 13)

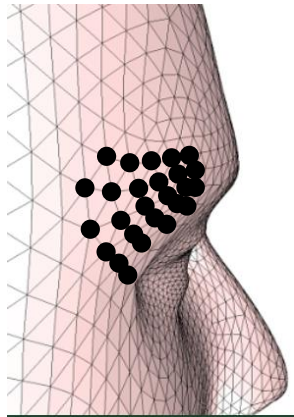


Graph encoding

(practical) motivation

Geometric v.s combinatorial information

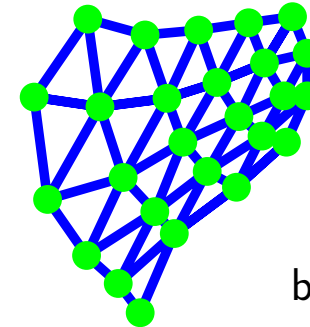
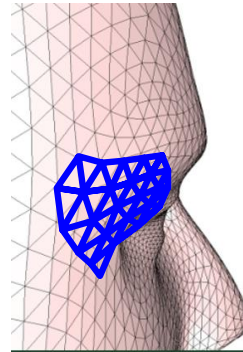
Geometry



vertex coordinates

between 30 et 96 bits/vertex

"Connectivity": the underlying triangulation



adjacency relations
between triangles, vertices

vertex	1 reference to a triangle
triangle	3 references to vertices 3 references to triangles

$13n \log n$ or $416n$ bits

$$\#\{\text{triangulations}\} = \frac{2(4n+1)!}{(3n+2)!(n+1)!} \approx \frac{16}{27} \sqrt{\frac{3}{2\pi}} n^{-5/2} \left(\frac{256}{27}\right)^n$$

$$\Rightarrow \text{entropy} = \log_2 \frac{256}{27} \approx 3.24 \text{ bpv.}$$

David statue (Stanford's Digital
Michelangelo Project, 2000)

2 billions polygons

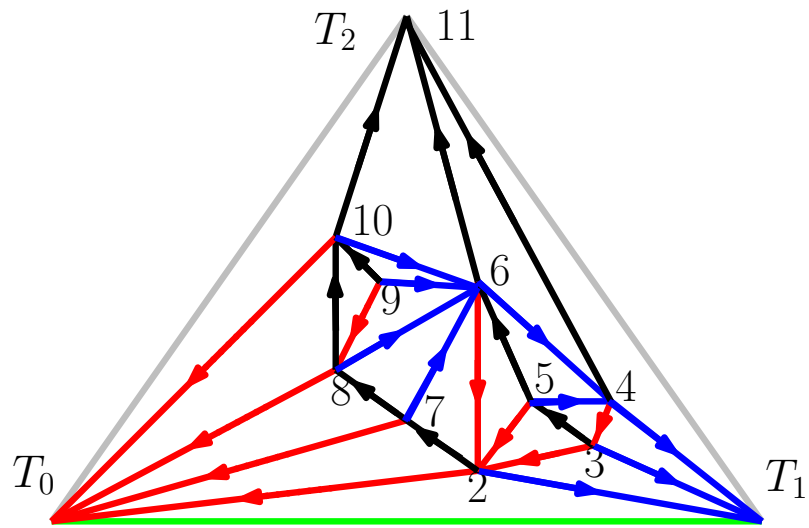
32 Giga bytes (without compression)

No existing algorithm nor data
structure for dealing with the
entire model



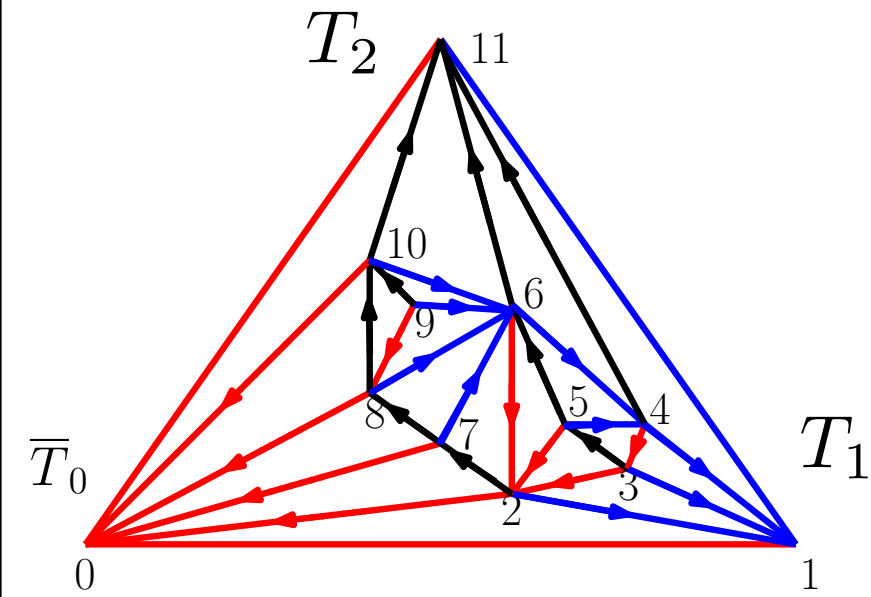
A more efficient encoding

Canonical orderings - Schnyder woods (He, Kao, Lu '99)



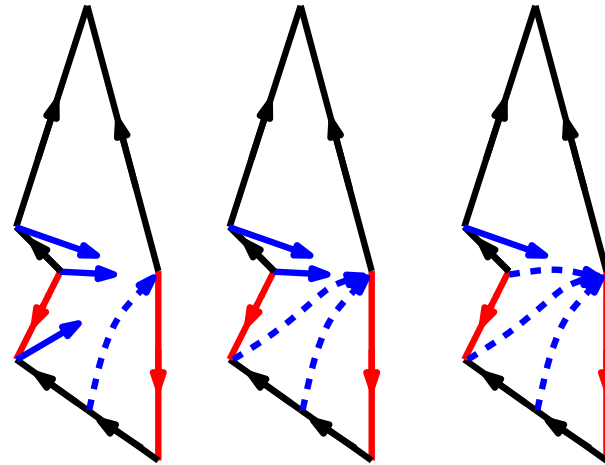
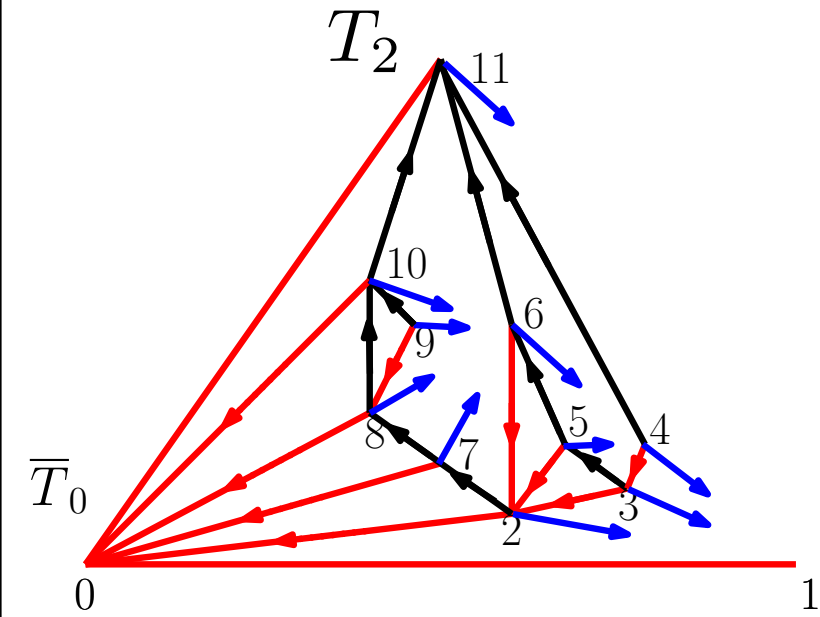
A more efficient encoding

Canonical orderings - Schnyder woods (He, Kao, Lu '99)



A more efficient encoding

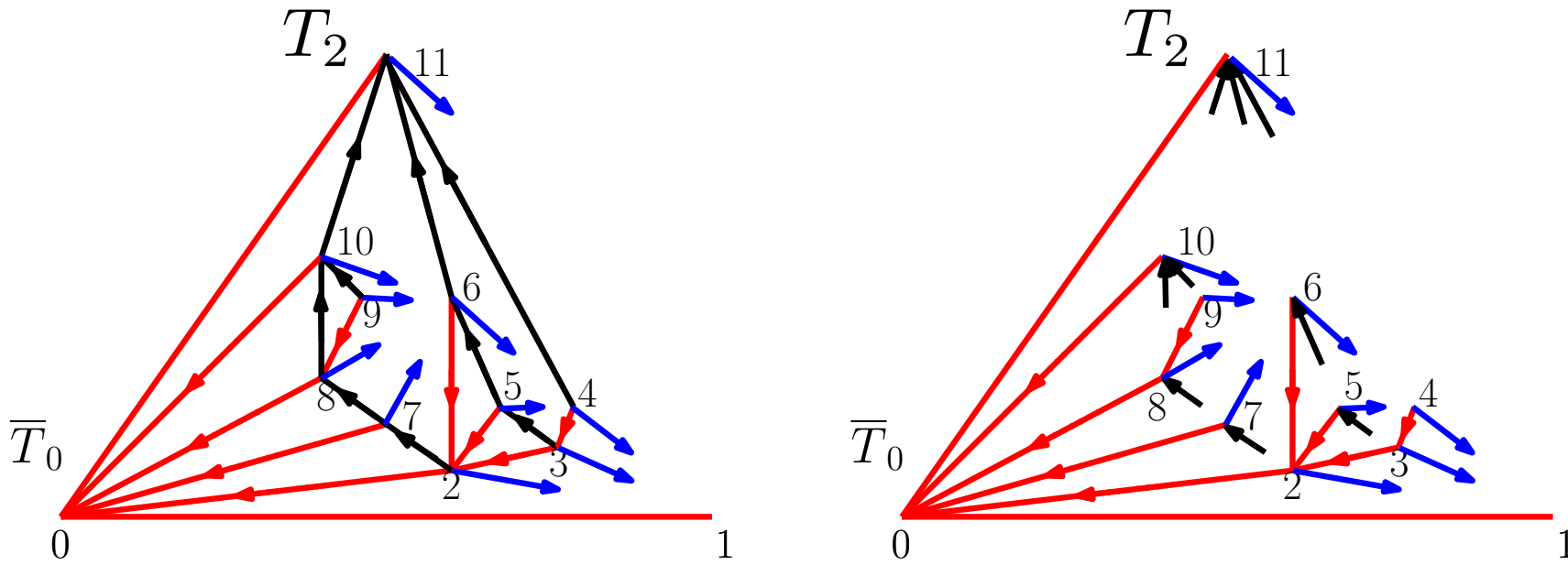
Canonical orderings - Schnyder woods (He, Kao, Lu '99)



T_1 is redundant: reconstruct from T_0, T_2

A more efficient encoding

Canonical orderings - Schnyder woods (He, Kao, Lu '99)

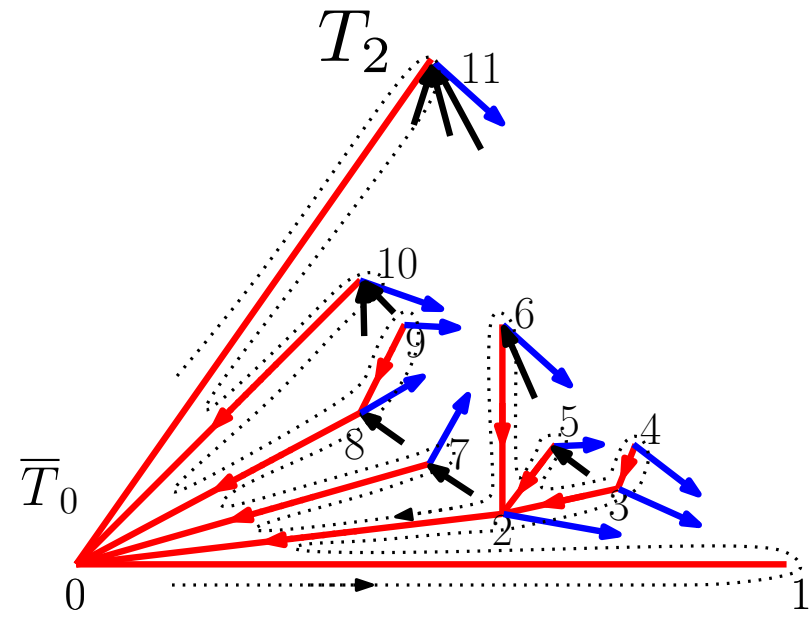
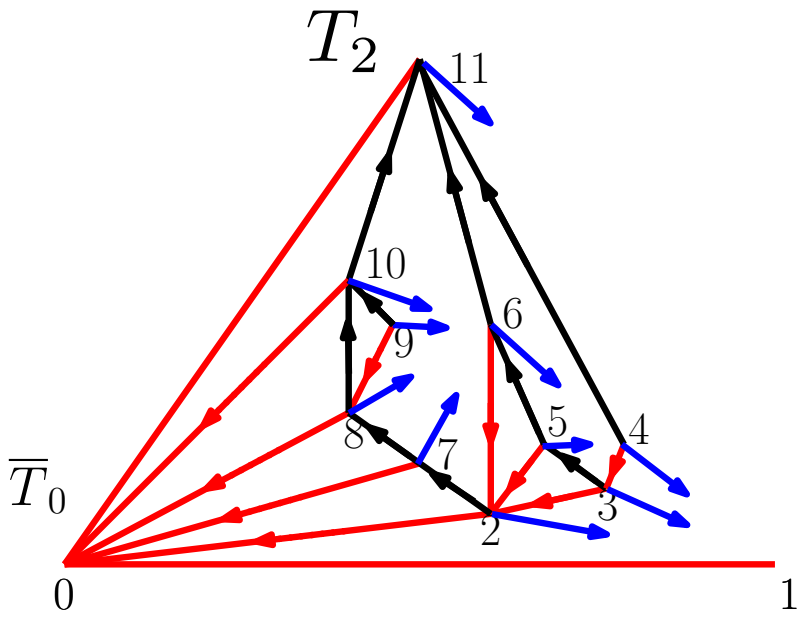


T_1 is redundant: reconstruct from T_0, T_2

T_2 can be reconstructed from T_0 and the number of ingoing edges (for each node)

A more efficient encoding

Canonical orderings - Schnyder woods (He, Kao, Lu '99)
 $4n$ bits (for triangulations)



\bar{T}_0 $() (((()) () ()) () (()) () ()$

$2(n - 1)$ symbols = $2(n - 1)$ bits

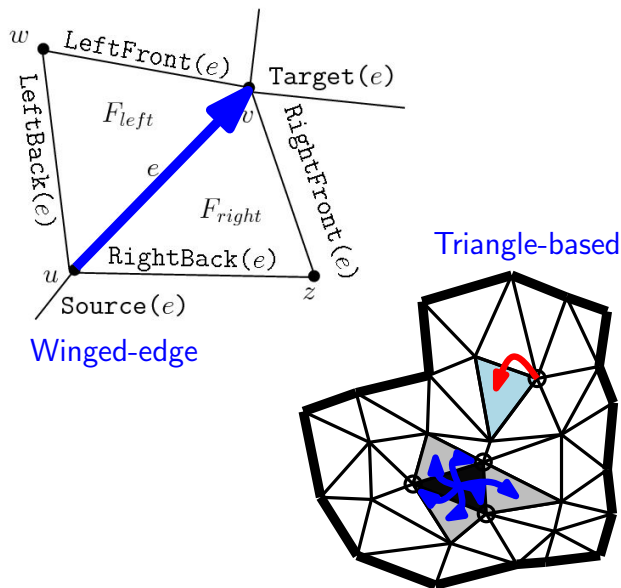
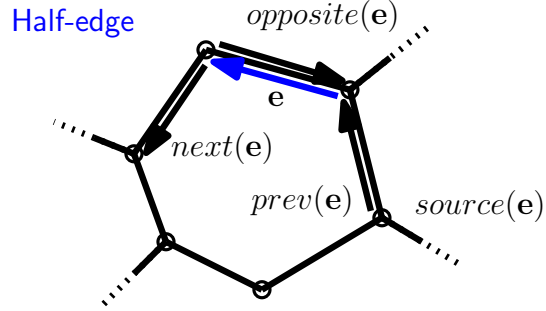
\bar{T}_2 00000101010100110111

$(n - 1) + (n - 3) = 2n - 4$ bits

Compact (practical) mesh data structures

(non compact) data structures

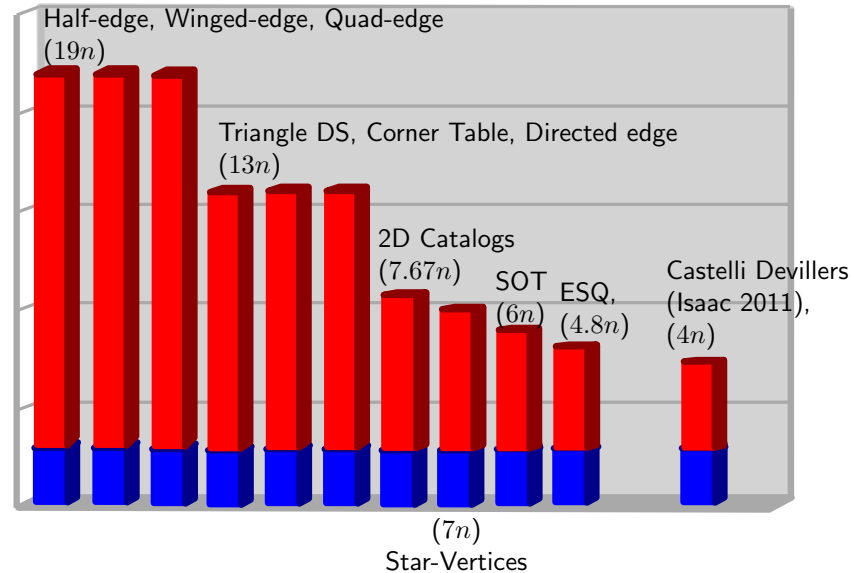
compact data structures



Data Structure	size	navigation time	vertex access	dynamic
Half-edge/Winged-edge/Quad-edge	$18n + n$	$O(1)$	$O(1)$	yes
Triangle based DS / Corner Table	$12n + n$	$O(1)$	$O(1)$	yes
Directed edge (Campagna et al. '99)	$12n + n$	$O(1)$	$O(1)$	yes
2D Catalogs (Castelli Aleardi et al., '06)	$7.67n$	$O(1)$	$O(1)$	yes
Star vertices (Kallmann et al. '02)	$7n$	$O(d)$	$O(1)$	no
TRIPOD (Snoeyink, Speckmann, '99)	$6n$	$O(1)$	$O(d)$	no
SOT (Gurung et al. 2010)	$6n$	$O(1)$	$O(d)$	no
SQUAD (Gurung et al. 2011)	$(4 + \epsilon)n$	$O(1)$	$O(d)$	no
ESQ (Castelli Aleardi, Devillers, Rossignac'12)	$4.8n$	$O(1)$	$O(d)$	yes
Castelli Aleardi and Devillers (2011)	$4n$ (or $6n$)	$O(1)$	$O(d)$ (or $O(1)$)	no
LR (Gurung et al. 2011)	$(2 + \delta)n$	$O(1)$	$O(d)$	no

ϵ between 0.09 and 0.3

δ about 0.8 and 0.3



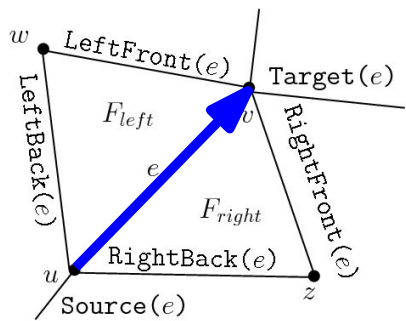
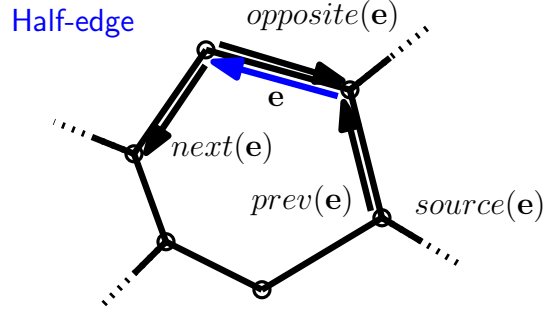
Compact (practical) mesh data structures

(non compact) data structures

compact data structures

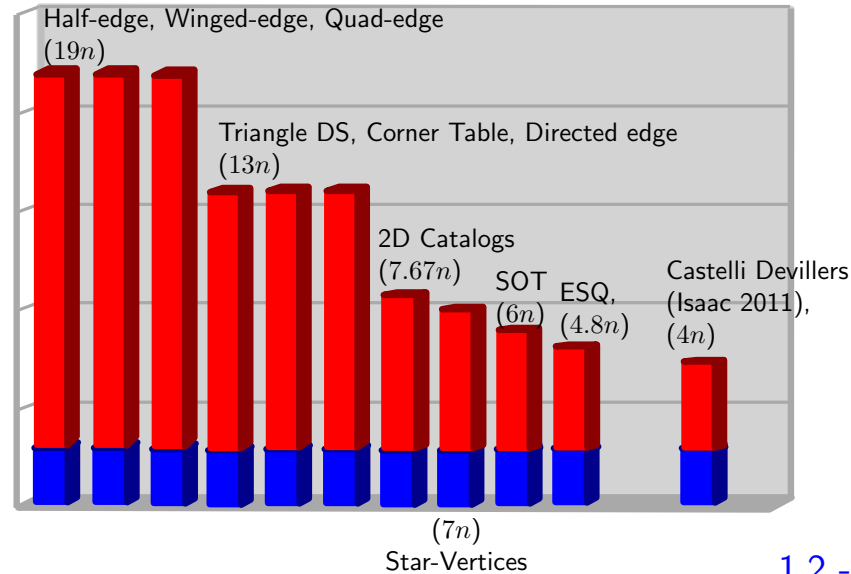
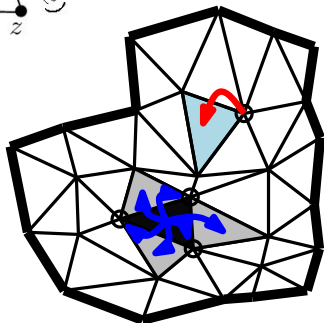
Data Structure	size	navigation time	vertex access	dynamic
Half-edge/Winged-edge/Quad-edge	$18n + n$	$O(1)$	$O(1)$	yes
Triangle based DS / Corner Table	$12n + n$	$O(1)$	$O(1)$	yes
Directed edge (Campagna et al. '99)	$12n + n$	$O(1)$	$O(1)$	yes
2D Catalogs (Castelli Aleardi et al., '06)	$7.67n$	$O(1)$	$O(1)$	yes
Star vertices (Kallmann et al. '02)	$7n$	$O(d)$	$O(1)$	no
TRIPOD (Snoeyink, Speckmann, '99)	$6n$	$O(1)$	$O(d)$	no
SOT (Gurung et al. 2010)	$6n$	$O(1)$	$O(d)$	no
SQUAD (Gurung et al. 2011)	$(4 + \epsilon)n$	$O(1)$	$O(d)$	no
ESQ (Castelli Aleardi, Devillers, Rossignac'12)	$4.8n$	$O(1)$	$O(d)$	yes
Castelli Aleardi and Devillers (2011)	$4n$ (or $6n$)	$O(1)$	$O(d)$ (or $O(1)$)	no
LR (Gurung et al. 2011)	$(2 + \delta)n$	$O(1)$	$O(d)$	no

ϵ between 0.09 and 0.3
 δ about 0.8 and 0.3

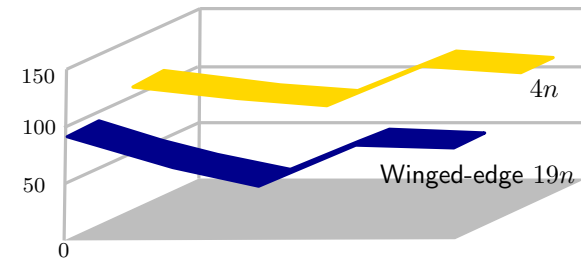


Triangle-based

Winged-edge



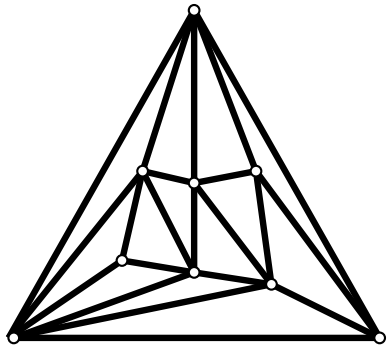
(timings are expressed in nanoseconds/vertex)
vertex degree (only topological navigation)



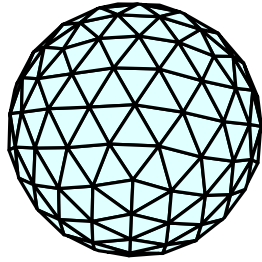
1.2 - 1.9 times slower than Winged-edge
(experimental evaluation)

Graphs on surfaces

Graphs on surfaces

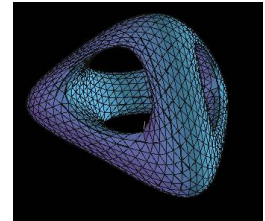
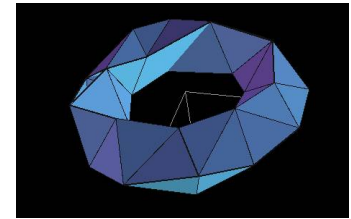


$$e = 3n - 6$$



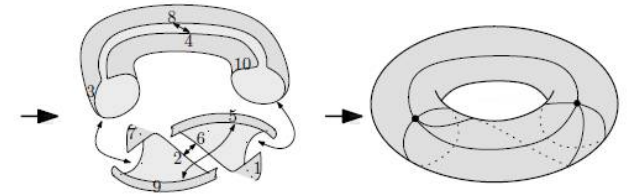
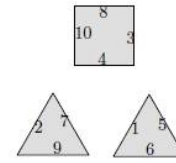
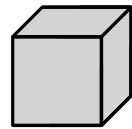
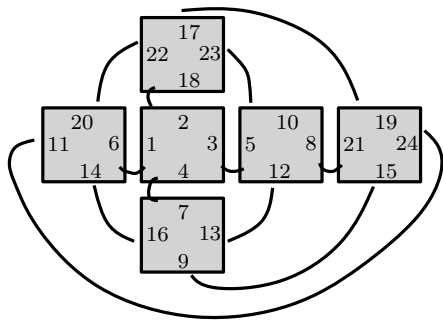
$$n - e + f = 2 - 2g$$

$$g = 1 \quad e = 3n$$

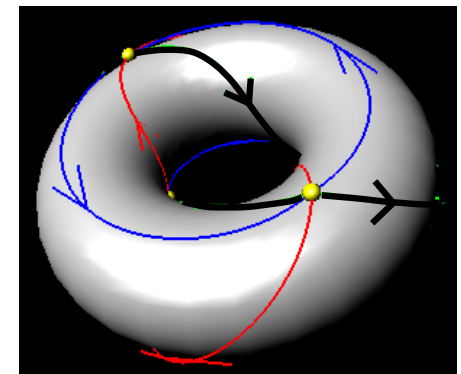
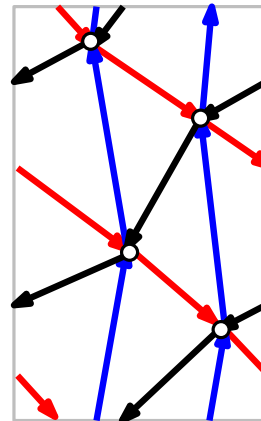
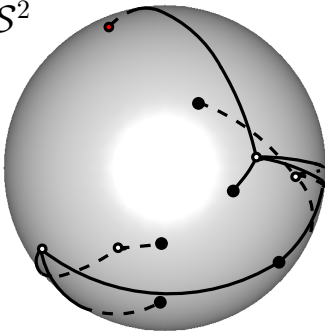


$$\phi = (1, 2, 3, 4)(17, 23, 18, 22)(5, 10, 8, 12)(21, 19, 24, 15) \dots$$

$$\alpha = (2, 18)(4, 7)(12, 13)(9, 15)(14, 16)(10, 23) \dots$$

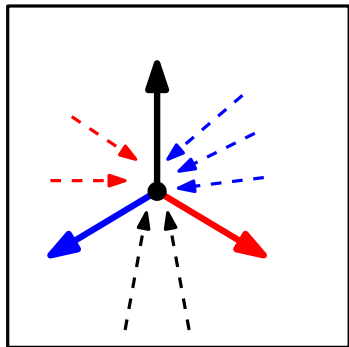
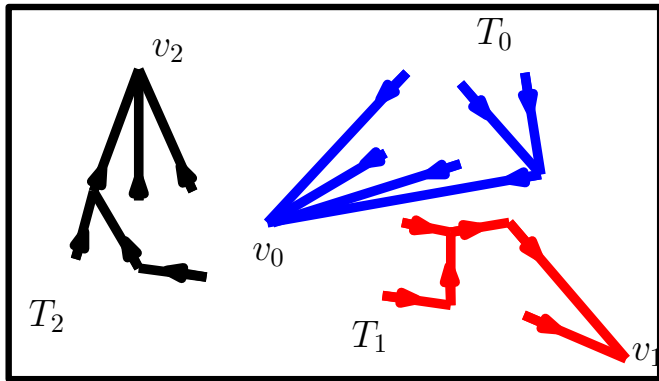
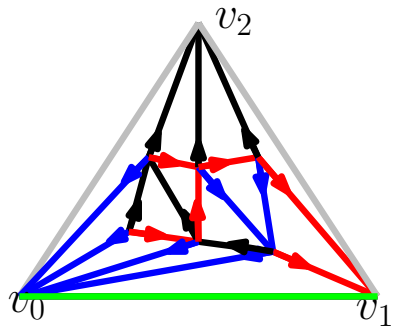


S^2



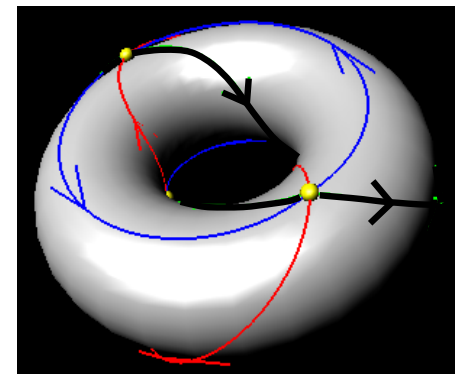
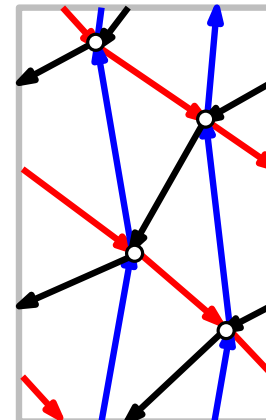
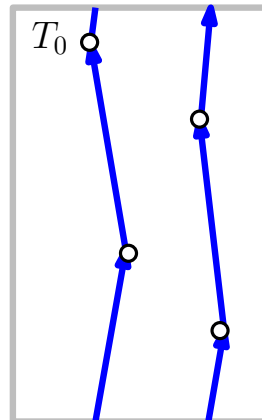
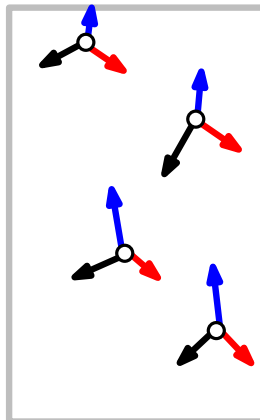
what can we to extend to higher genus?

$$e = 3n - 6$$



[Goncalves Lévêque, DCG'14]

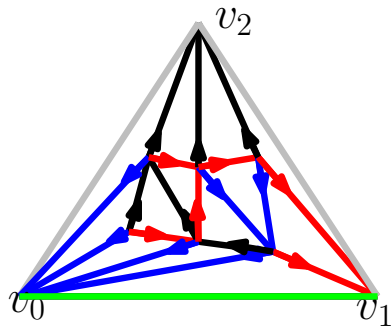
$$g = 1 \quad e = 3n$$



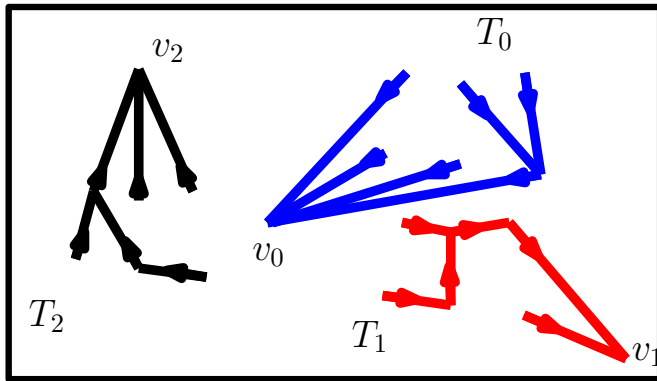
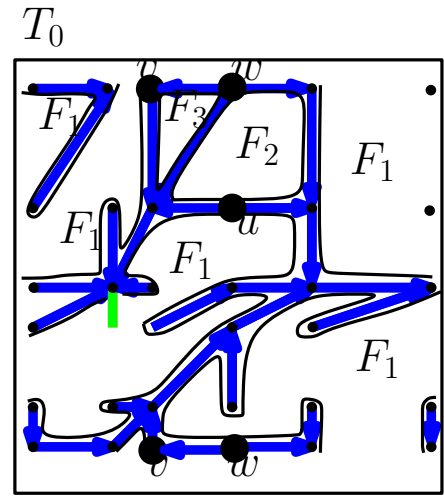
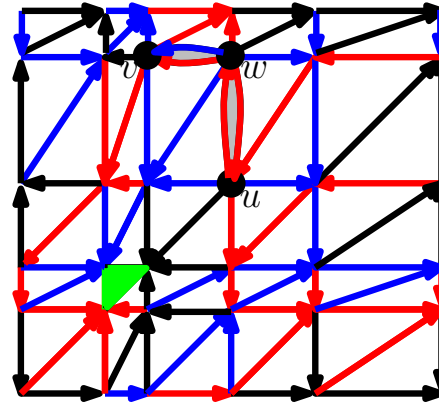
what can we to extend to higher genus?

$$e = 3n - 6$$

[Castelli-Aleardi Fusy Lewiner, SoCG'08]

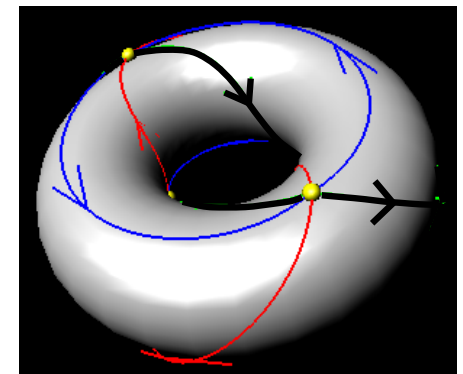
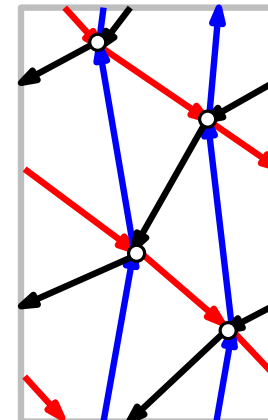
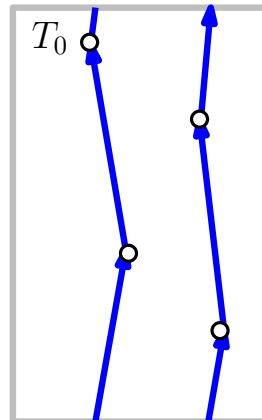
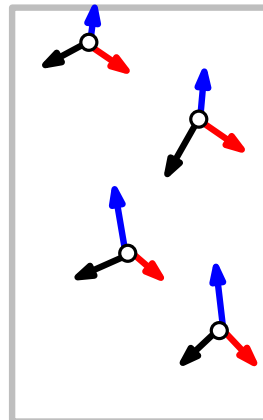
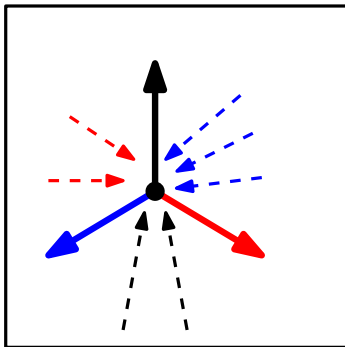


$$\mathcal{E} = \{(u, w), (v, w)\}$$



[Goncalves Lévêque, DCG'14]

$$g = 1 \quad e = 3n$$



Schnyder woods and higher genus surfaces

(several possible generalizations)

(pioneeristic) toroidal tree decomposition

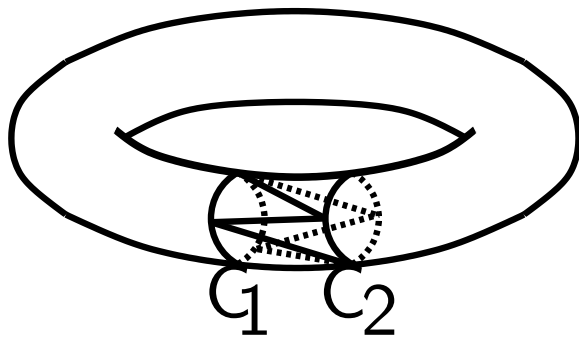
[Bonichon Gavoille Labourel, 2005]



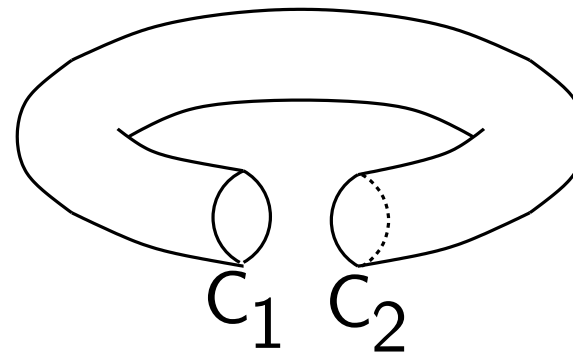
the "tambourine"
solution

Compute a pair of adjacent non contractible cycles

Graph G



Graph H



Tambourine
T



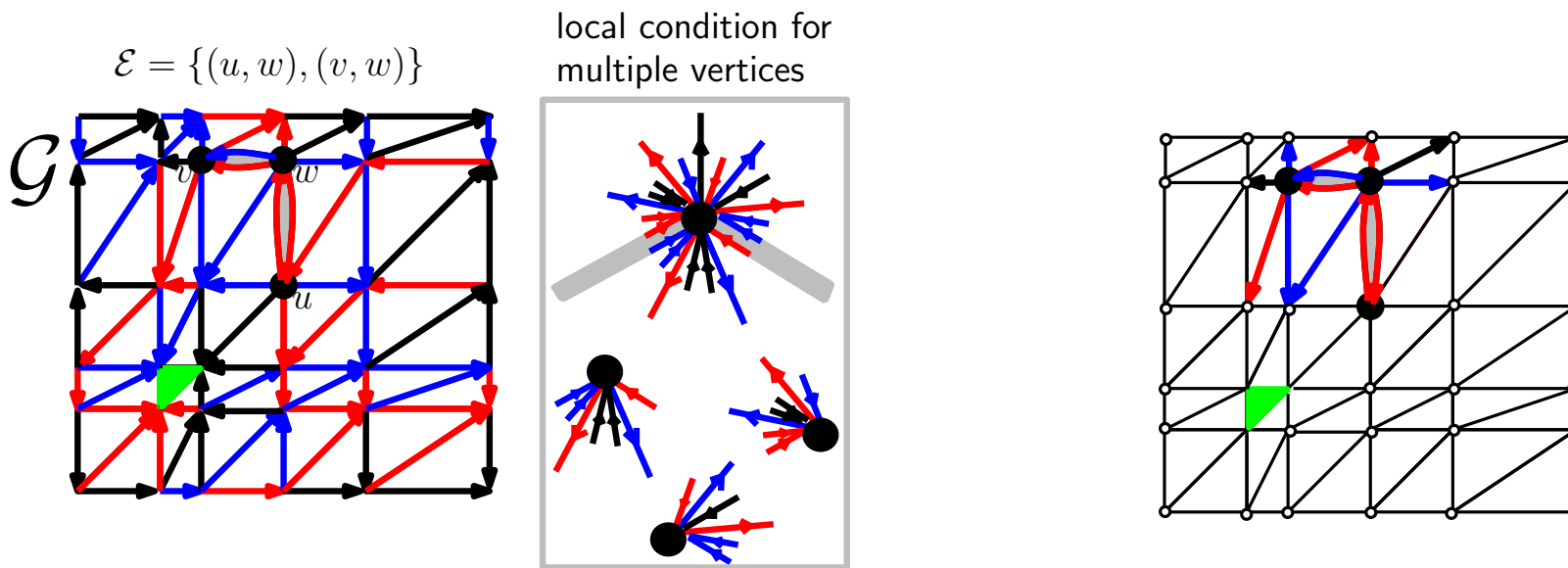
Result:

Inconvenients:

- valid only for toroidal triangulations (genus 1)
- potentially large number of vertices (on C_1 and C_2) not satisfying the local condition
- shortest non trivial cycles are "hard" to compute

Definition 1: genus g Schnyder woods

[Castelli-Aleardi Fusy Lewiner, SoCG'08]



Def: partition of all "inner" edges into four sets

$$T_0, T_1, T_2 \text{ and } \mathcal{E}$$

such that

almost all vertices have outgoing degree 3

all edges in T_0, T_1 and T_2 have one color/orientation

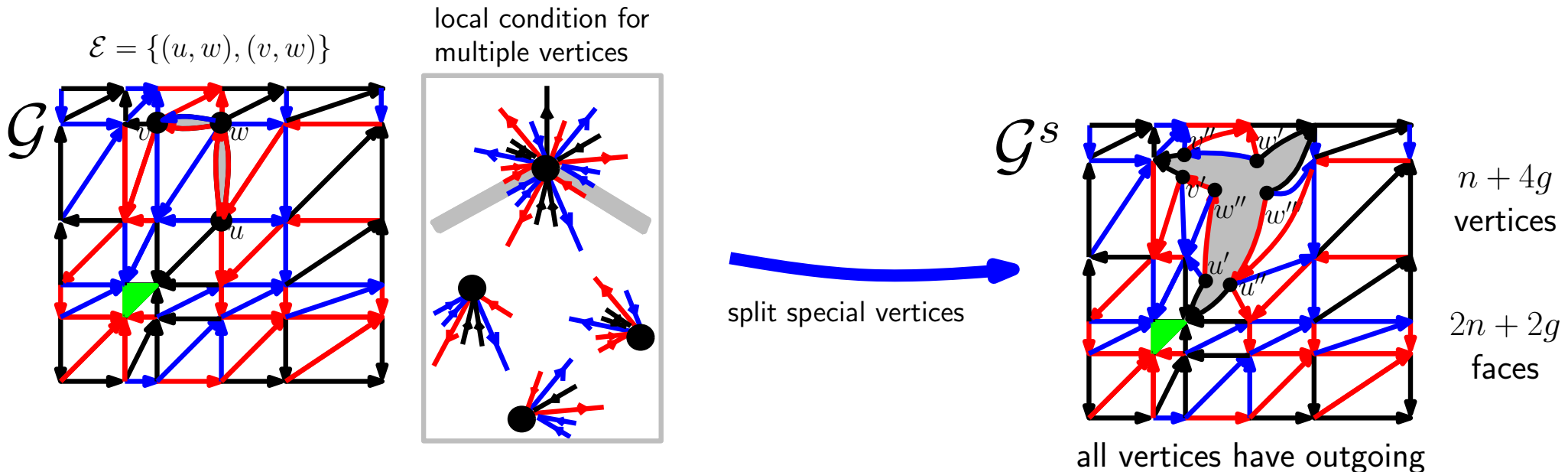
at most $4g$ special vertices (outdegree > 3)

the set \mathcal{E} contains at most $2g$ edges (multiple edges)

new local conditions around special vertices

Definition 1: genus g Schnyder woods

[Castelli-Aleardi Fusy Lewiner, SoCG'08]



Def: partition of all "inner" edges into four sets

$$T_0, T_1, T_2 \text{ and } \mathcal{E}$$

such that

almost all vertices have outgoing degree 3

all edges in T_0, T_1 and T_2 have one color/orientation

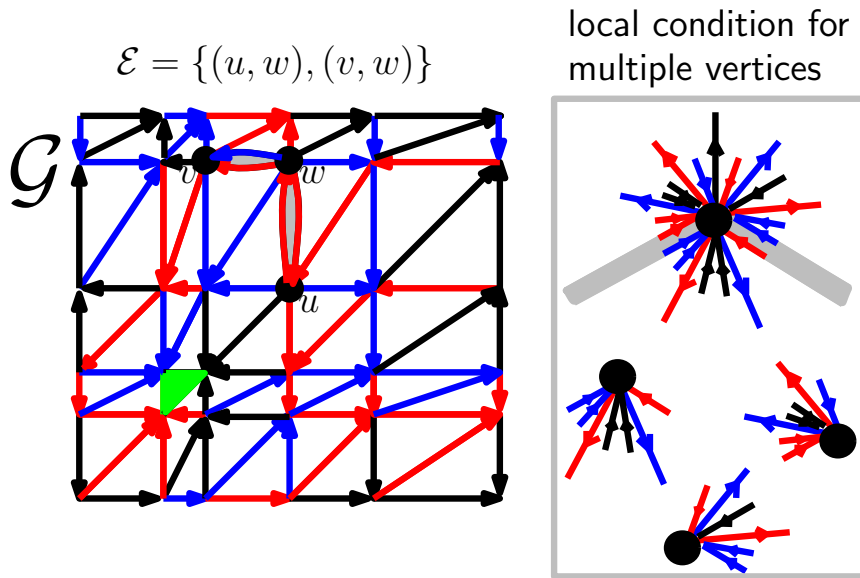
at most $4g$ special vertices (outdegree > 3)

the set \mathcal{E} contains at most $2g$ edges (multiple edges)

new local conditions around special vertices

Genus g Schnyder woods: spanning property

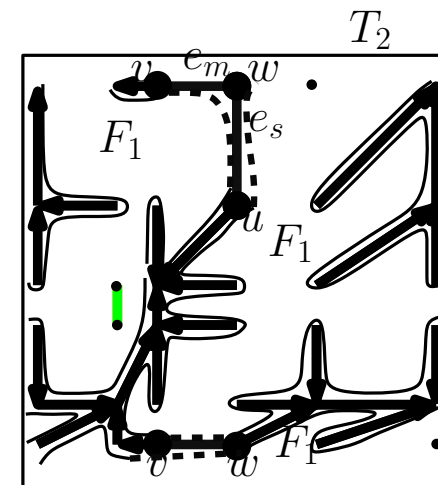
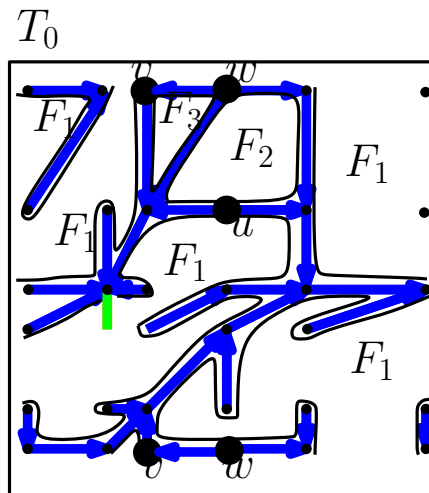
[Castelli-Aleardi Fusy Lewiner, SoCG'08]



Theorem

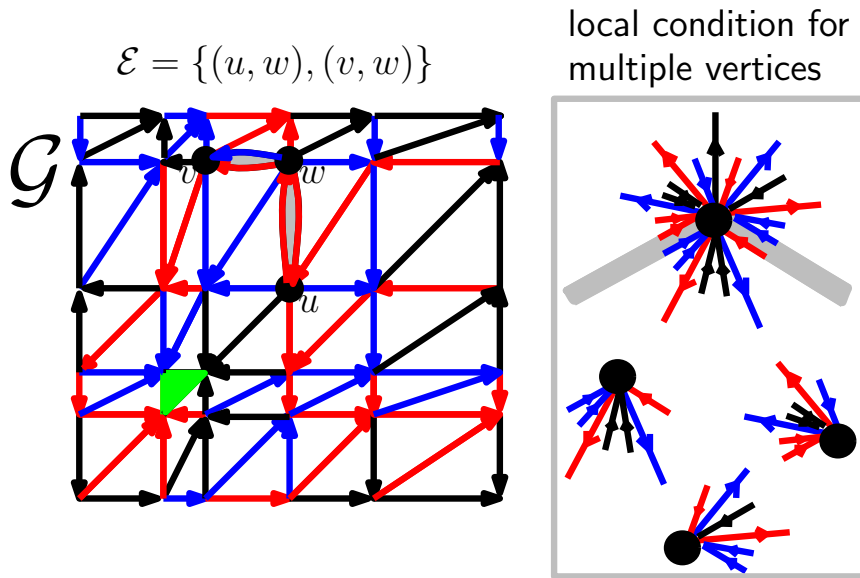
The three sets of edges T_0 and T_1 (red and blue edges), as well as the set $T_2 \cup \mathcal{E}$ (black edges and special edges) are maps of genus g satisfying:

- T_0, T_1 are maps with at most $1 + 2g$ faces;
- $T_2 \cup \mathcal{E}$ is a 1 face map (a g -tree)



Genus g Schnyder woods: application

[Castelli-Aleardi Fusy Lewiner, SoCG'08]



Corollary

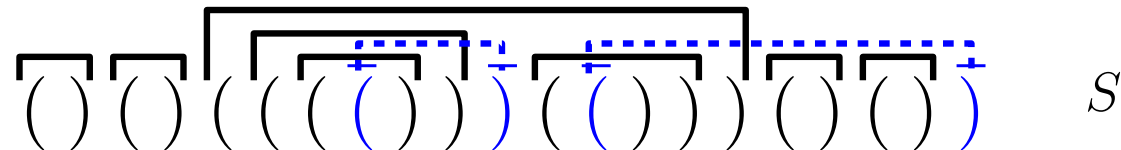
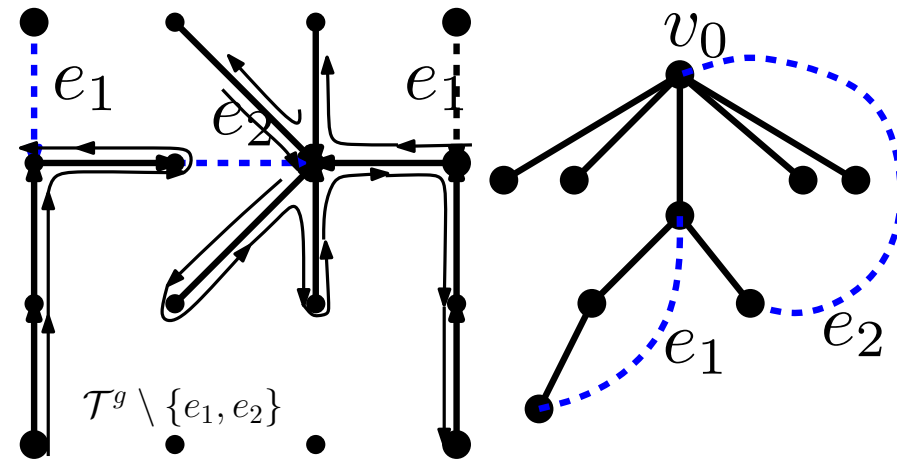
A triangulation of genus g having n vertices can be encoded with $4n + O(g \log n)$ bits

Encode map $T_2 \cup \mathcal{E}$: a tree plus $2g$ edges:
 $2n + O(g \log n)$ bits

Mark special vertices: $O(g \log n)$ bits

Store outgoing edges incident to special edges: $O(g \log n)$ bits

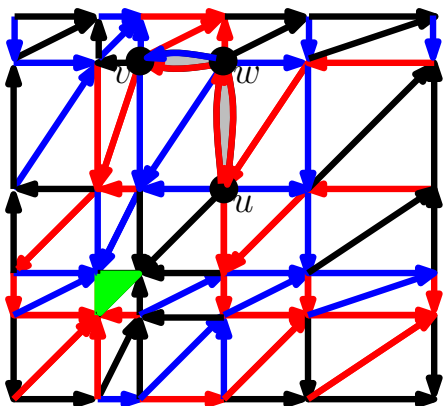
For each node in $T_2 \cup \mathcal{E}$ store the number of ingoing edges of color 0:
 $2n + O(g \log n)$ bits



Genus g Schnyder woods: existence

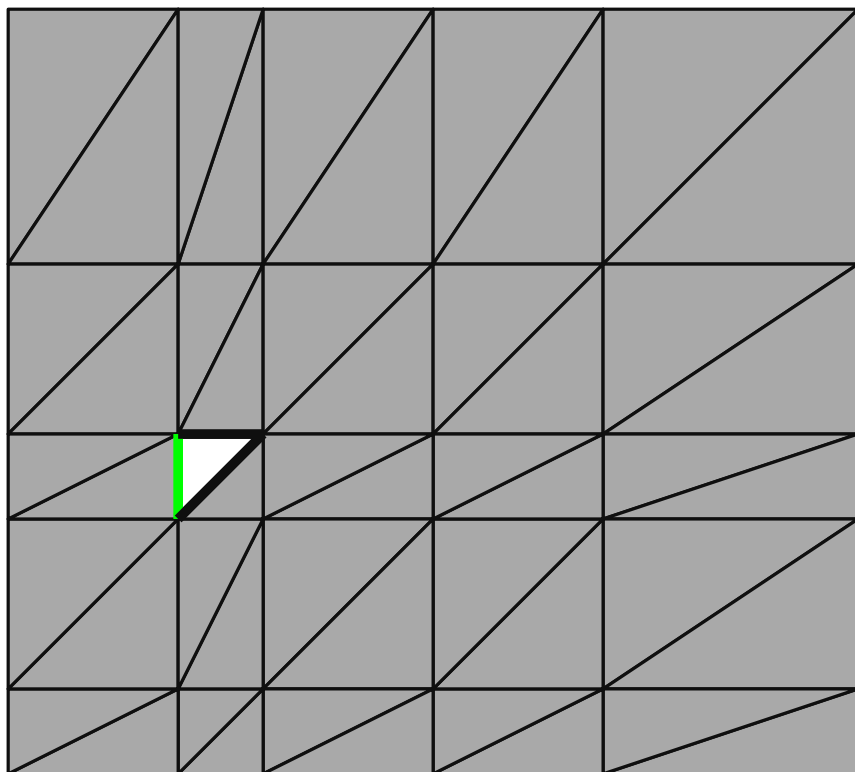
[Castelli-Aleardi Fusy Lewiner, SoCG'08]

$$\mathcal{E} = \{(u, w), (v, w)\}$$

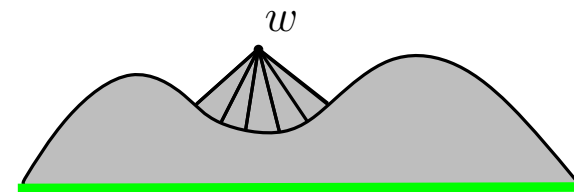


Incremental algorithm

Perform a vertex conquest (as far as you can)

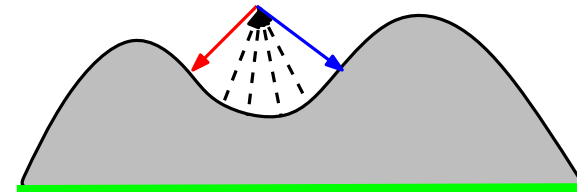


G_k

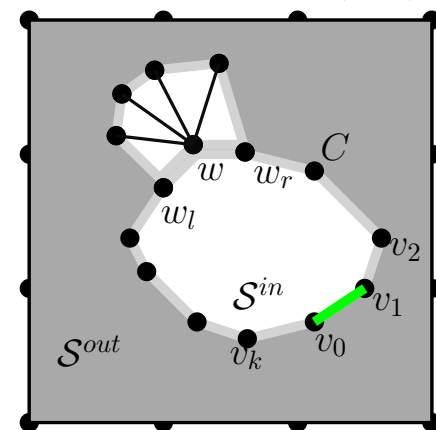


\Downarrow conquer(w)

G_{k-1}



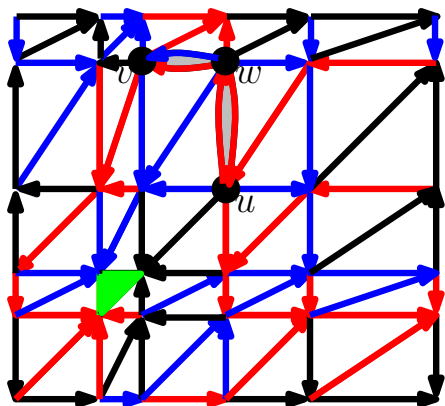
conquer(w)



Genus g Schnyder woods: existence

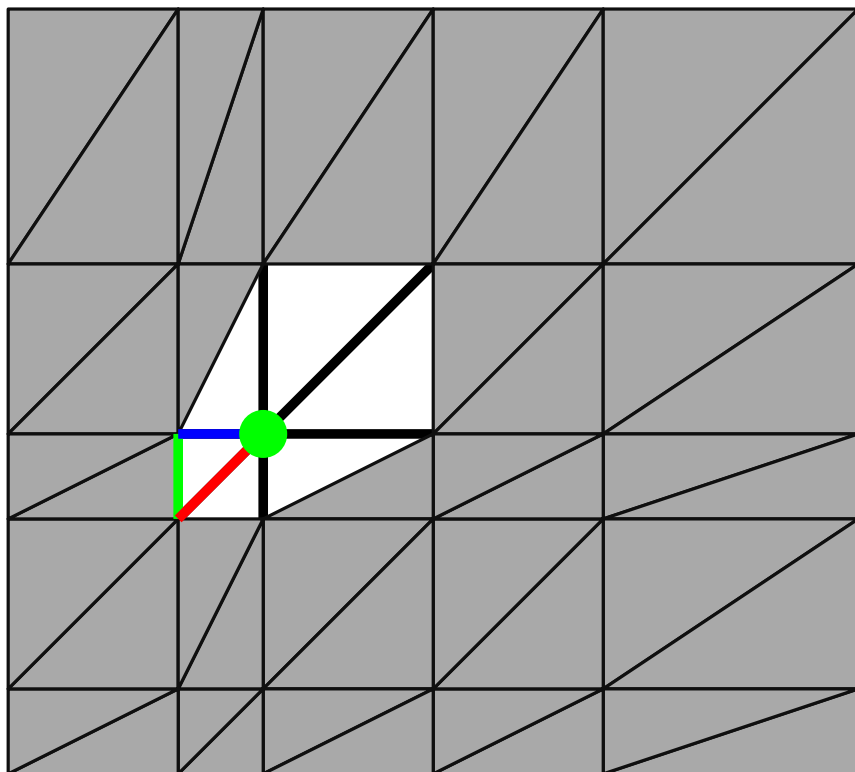
[Castelli-Aleardi Fusy Lewiner, SoCG'08]

$$\mathcal{E} = \{(u, w), (v, w)\}$$

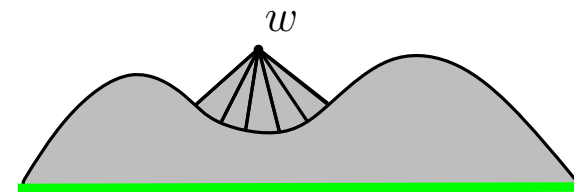


Incremental algorithm

Perform a vertex conquest (as far as you can)

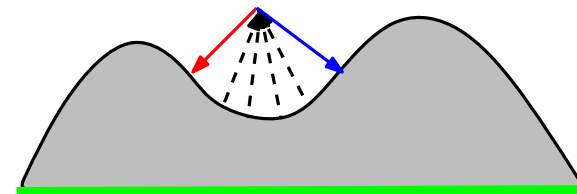


G_k

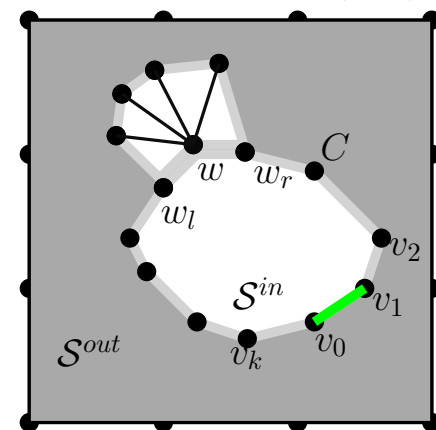


\Downarrow conquer(w)

G_{k-1}



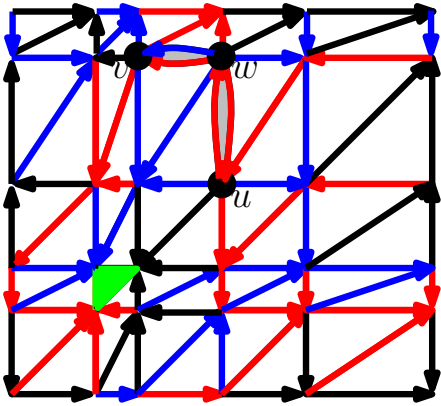
conquer(w)



Genus g Schnyder woods: existence

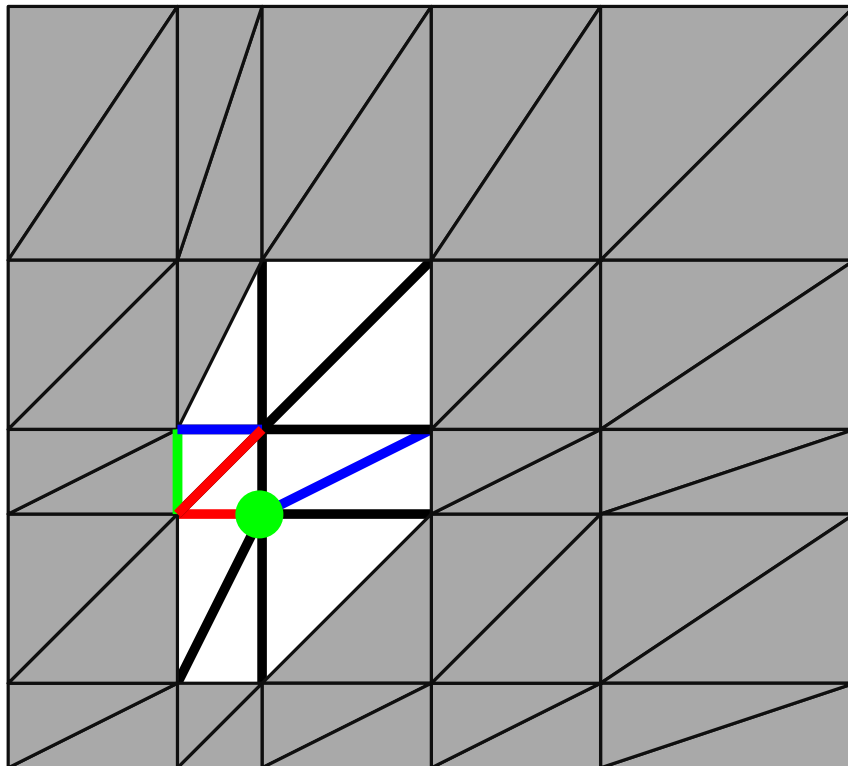
[Castelli-Aleardi Fusy Lewiner, SoCG'08]

$$\mathcal{E} = \{(u, w), (v, w)\}$$

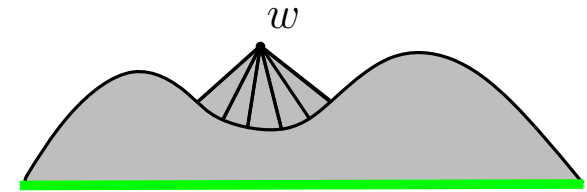


Incremental algorithm

Perform a vertex conquest (as far as you can)

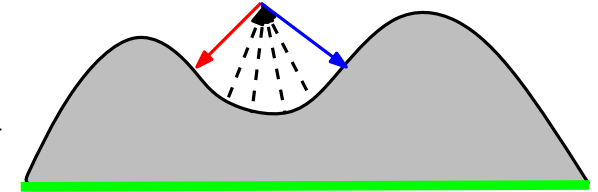


G_k

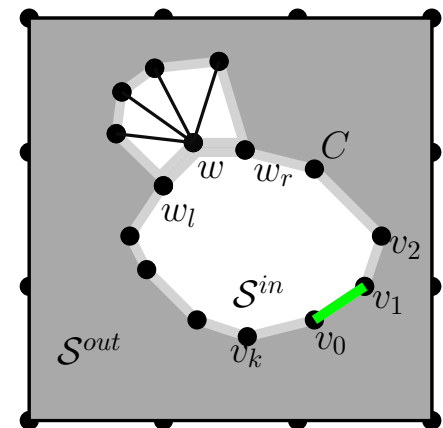


\Downarrow conquer(w)

G_{k-1}



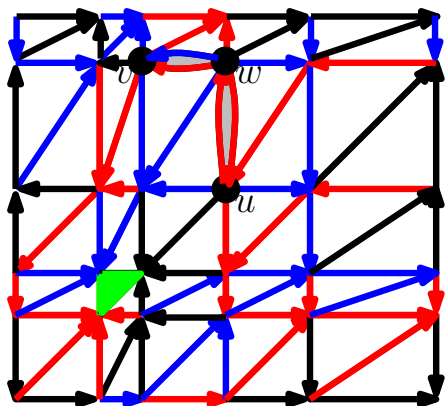
conquer(w)



Genus g Schnyder woods: existence

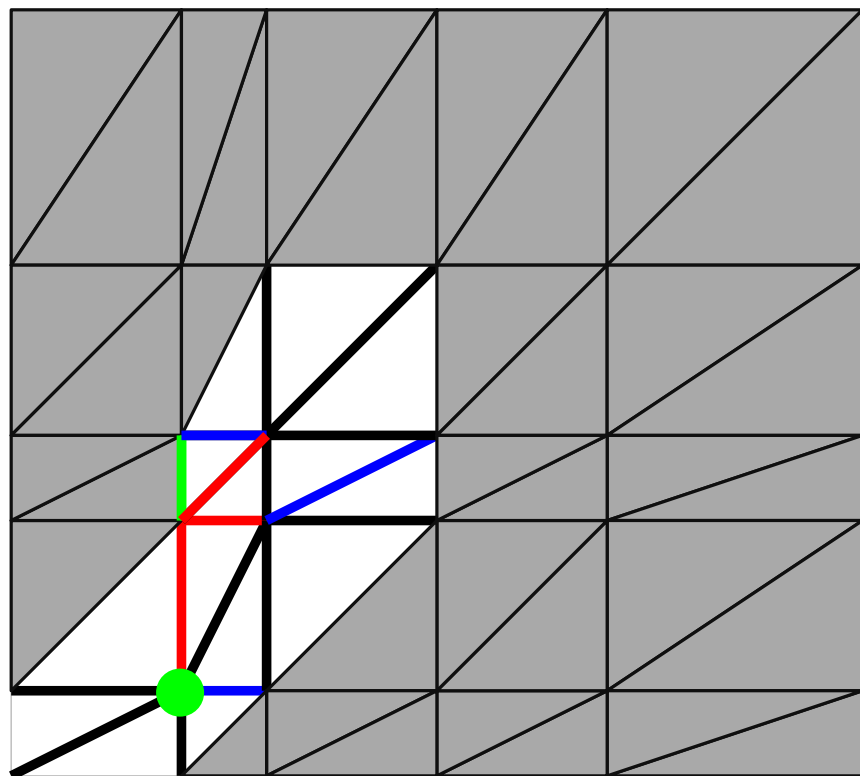
[Castelli-Aleardi Fusy Lewiner, SoCG'08]

$$\mathcal{E} = \{(u, w), (v, w)\}$$

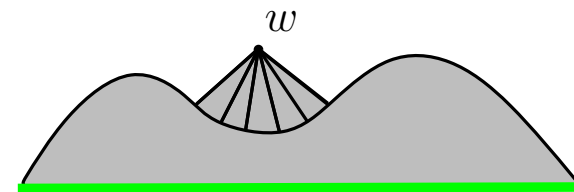


Incremental algorithm

Perform a vertex conquest (as far as you can)

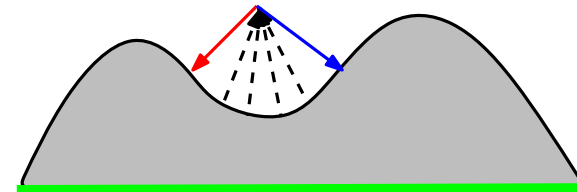


G_k

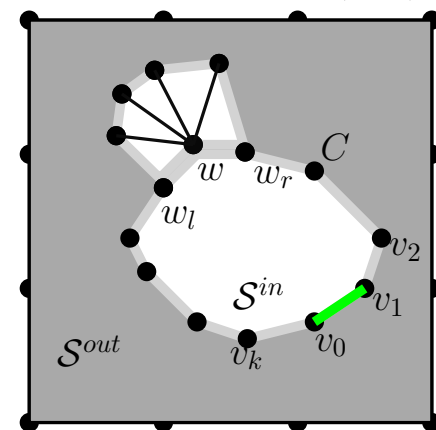


\Downarrow conquer(w)

G_{k-1}



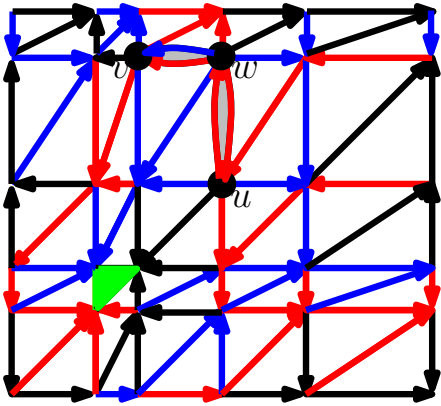
conquer(w)



Genus g Schnyder woods: existence

[Castelli-Aleardi Fusy Lewiner, SoCG'08]

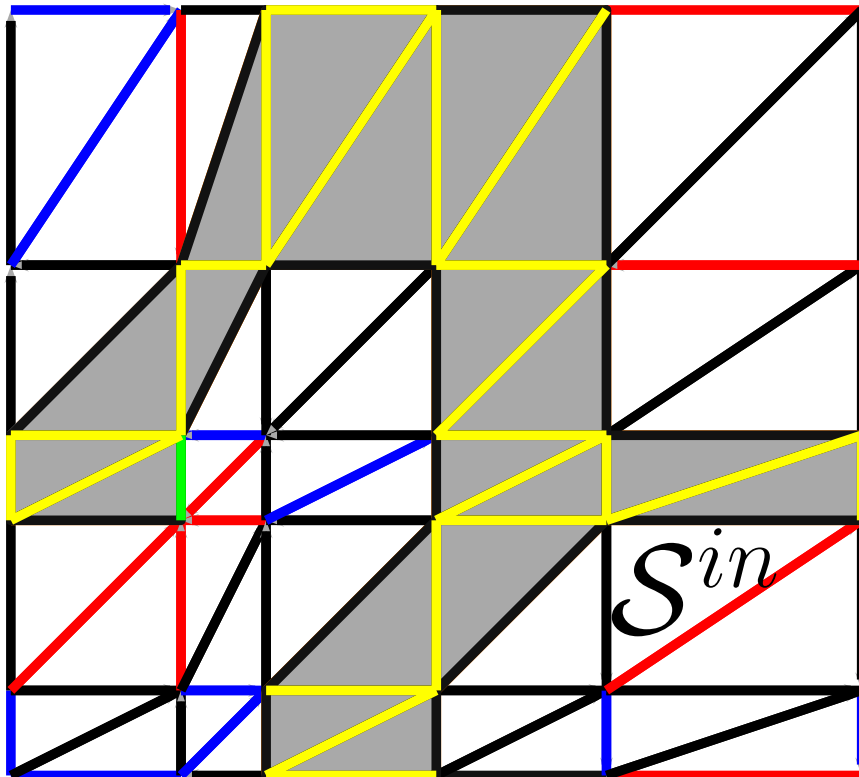
$$\mathcal{E} = \{(u, w), (v, w)\}$$



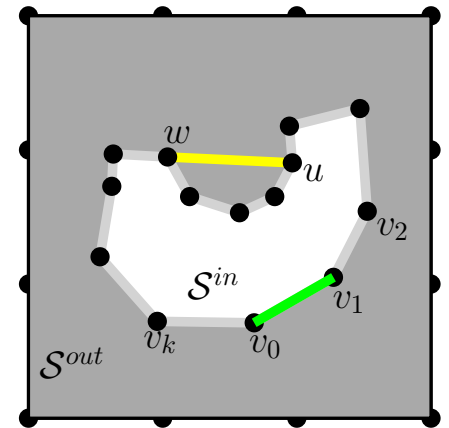
Incremental algorithm

Perform a vertex conquest (as far as you can)
when you get stuck

\mathcal{S}^{in} is a topological disk



chordal edge (u, w)

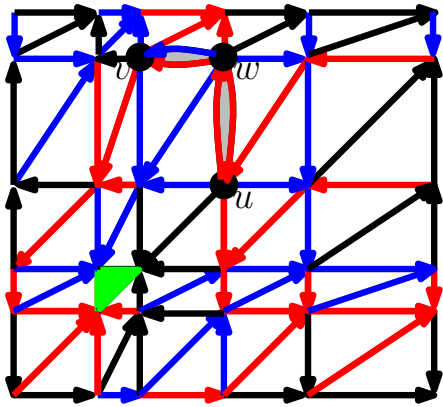


No more free vertices

Genus g Schnyder woods: existence

[Castelli-Aleardi Fusy Lewiner, SoCG'08]

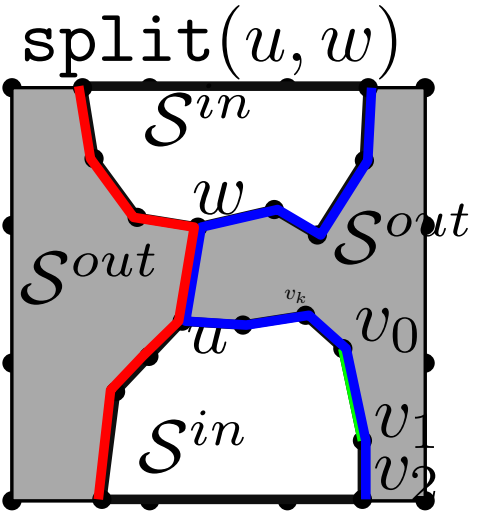
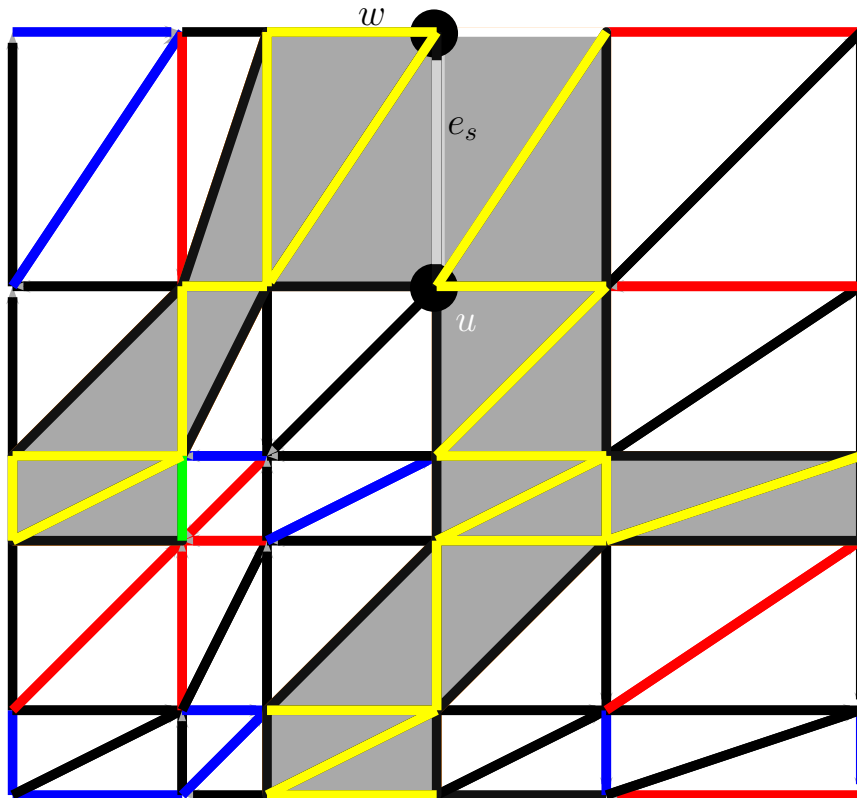
$$\mathcal{E} = \{(u, w), (v, w)\}$$



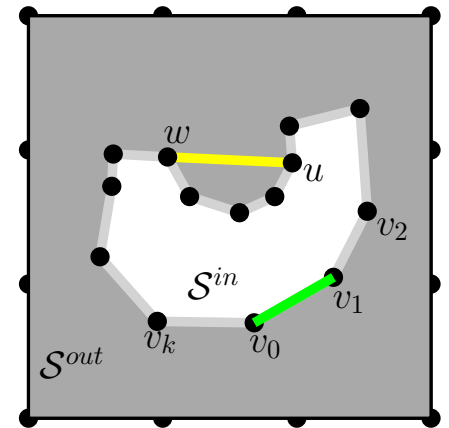
Incremental algorithm

Perform a vertex conquest (as far as you can)
when you get stuck
perform edge split

\mathcal{S}^{in} is a topological disk



chordal edge (u, w)

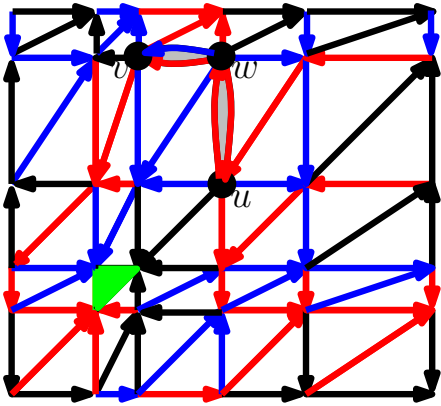


Now there are free vertices

Genus g Schnyder woods: existence

[Castelli-Aleardi Fusy Lewiner, SoCG'08]

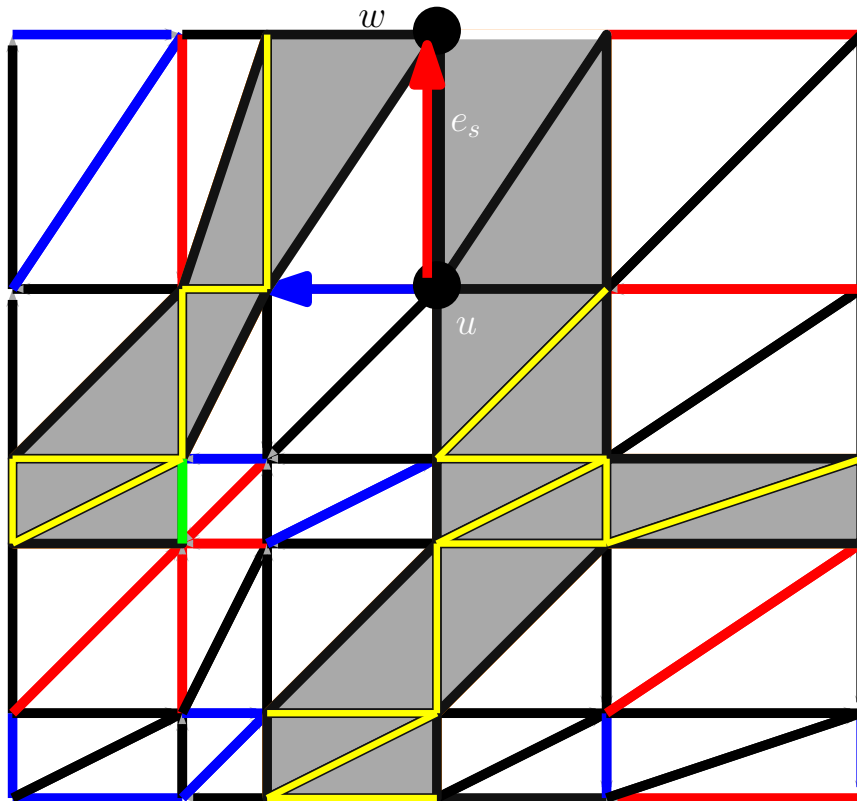
$$\mathcal{E} = \{(u, w), (v, w)\}$$



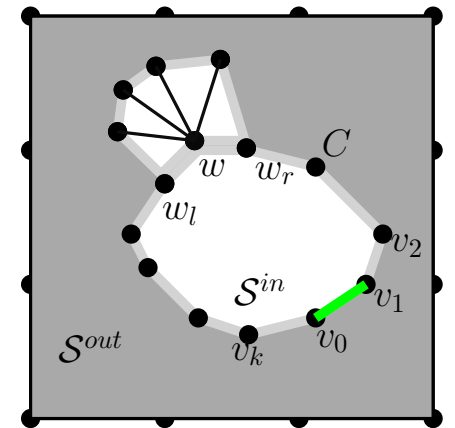
Incremental algorithm

- Perform a vertex conquest (as far as you can)
- when you get stuck
 - perform edge split
- Perform a vertex conquest (as far as you can)

\mathcal{S}^{in} is a topological disk



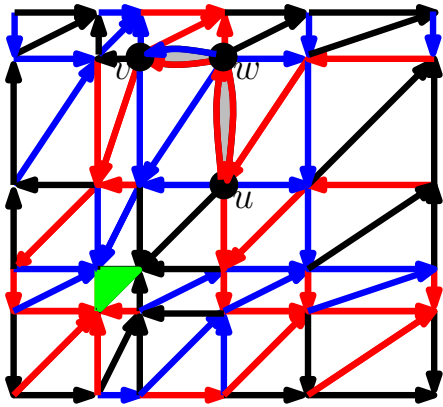
conquer(w)



Genus g Schnyder woods: existence

[Castelli-Aleardi Fusy Lewiner, SoCG'08]

$$\mathcal{E} = \{(u, w), (v, w)\}$$

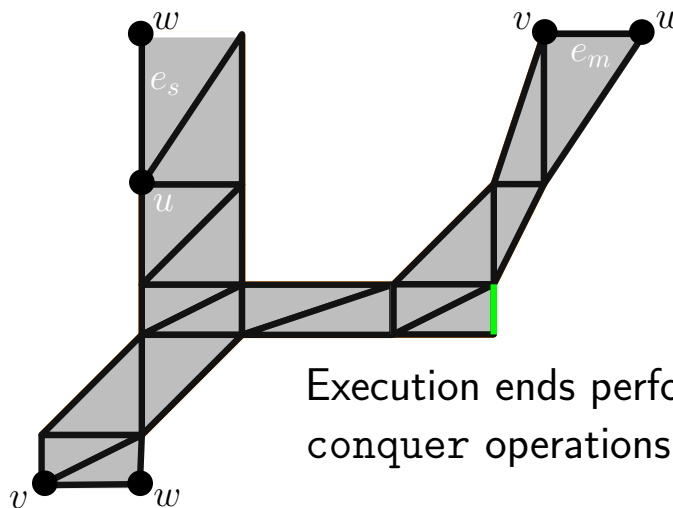
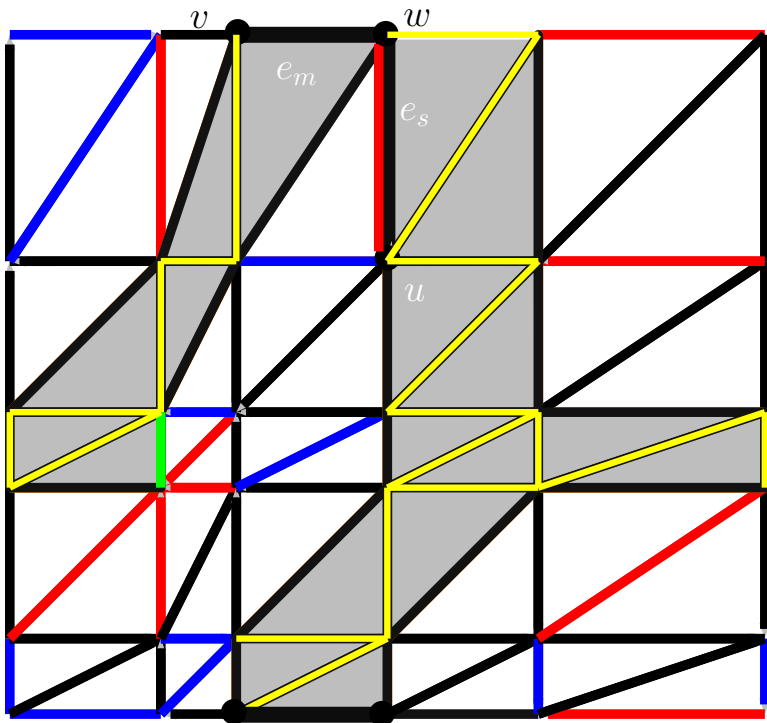
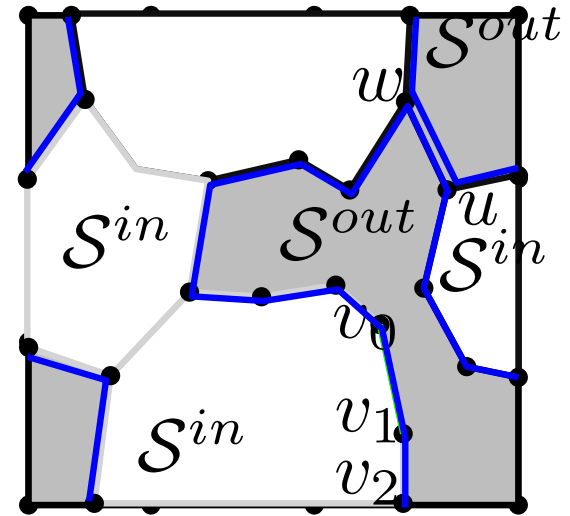


Incremental algorithm

- Perform a vertex conquest (as far as you can)
- when you get stuck
 - perform edge split
- Perform a vertex conquest (as far as you can)
- perform edge split

\mathcal{S}^{in} is a topological disk

$\text{merge}(u, w)$



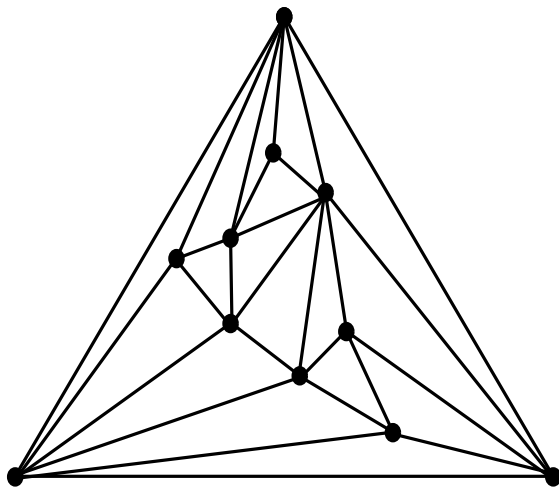
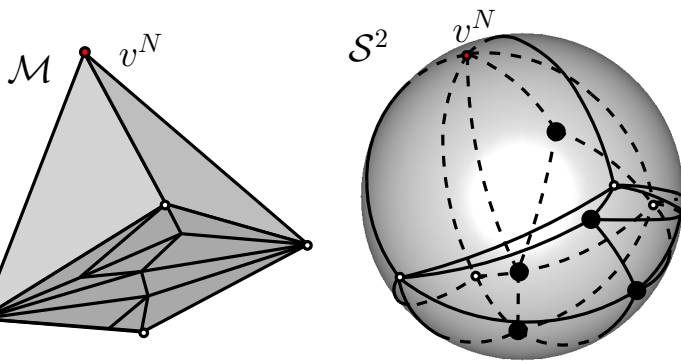
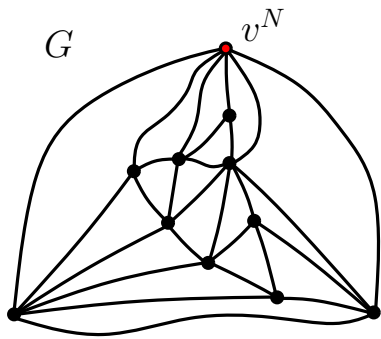
Execution ends performing a sequence of conquer operations

Periodic straight-line drawings

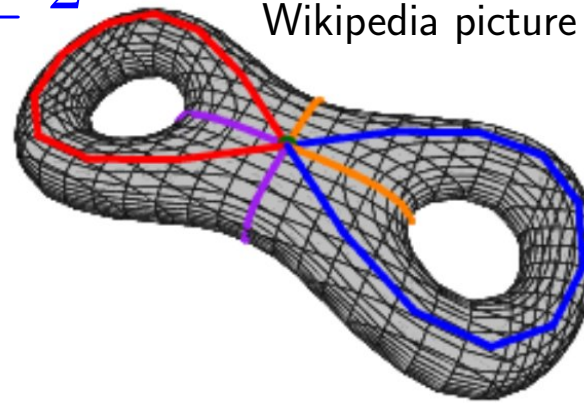
(of higher genus graphs)

Drawing higher genus graphs

$g = 0$

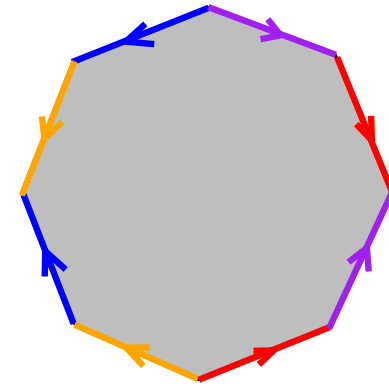


$g \geq 2$



Wikipedia picture

Polygonal scheme

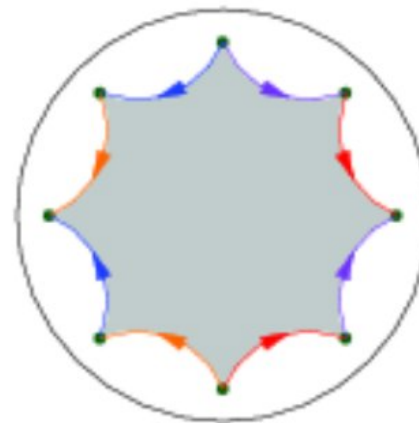


drawing in polynomial area

[Duncan, Goodrich, Kobourov, GD'09]

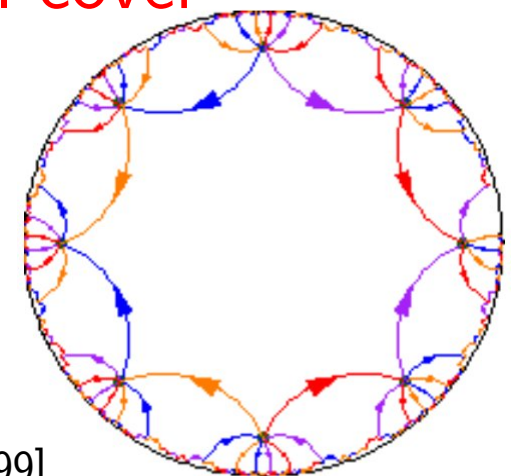
[Chambers, Eppstein, Goodrich, Löffler, GD'10]

Universal cover



[Mohar'99]

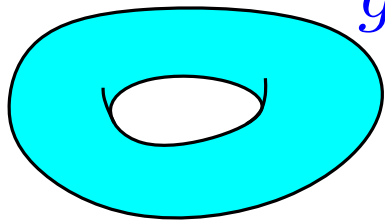
periodic drawing



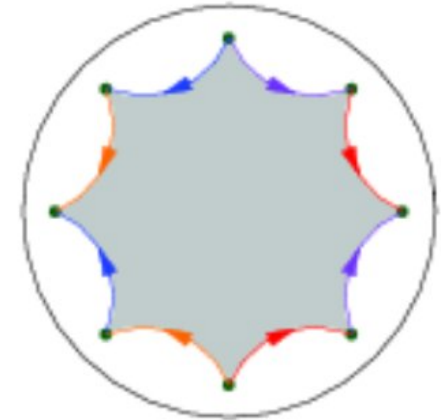
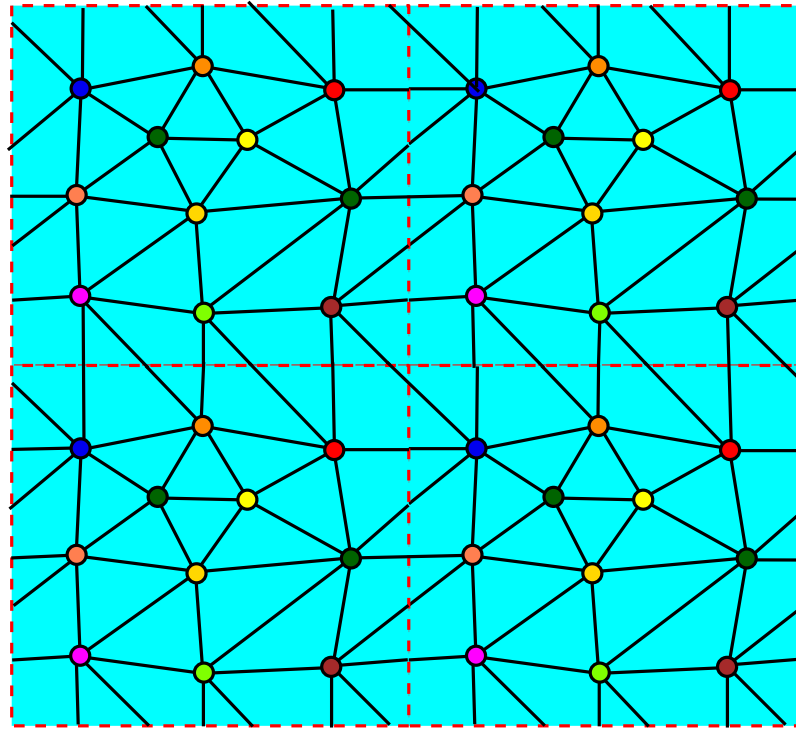
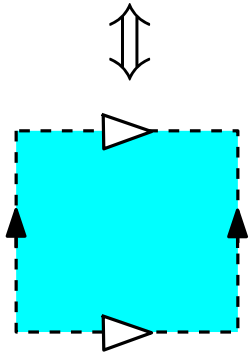
out of circle packing

Drawing toroidal graphs

On the torus

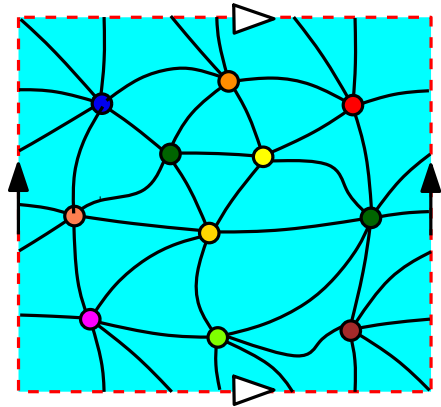
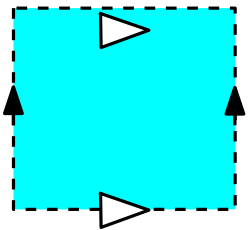
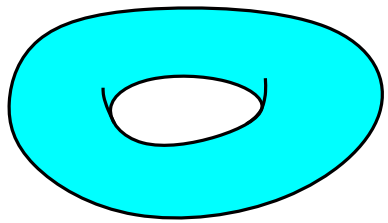


$g = 1$

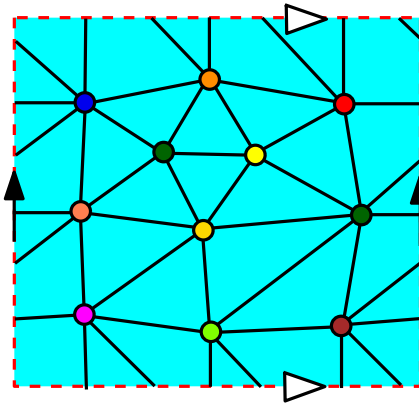


Periodic straight-line drawings

On the torus



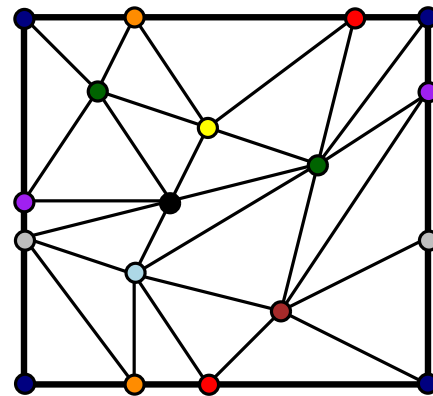
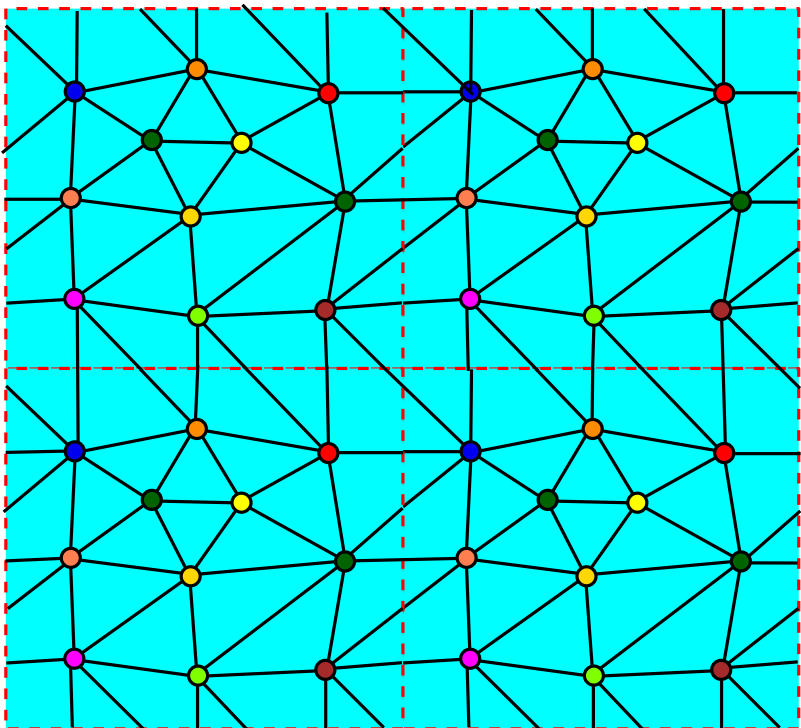
drawing on the flat torus



straight-line drawing
x-periodic and
y-periodic drawing

[Castelli Devillers Fusy, GD'12]
 $O(n \times n^{\frac{3}{2}})$ **grid**

[Goncalves Lévêque, DCG]
 $O(n^2 \times n^2)$ **grid**

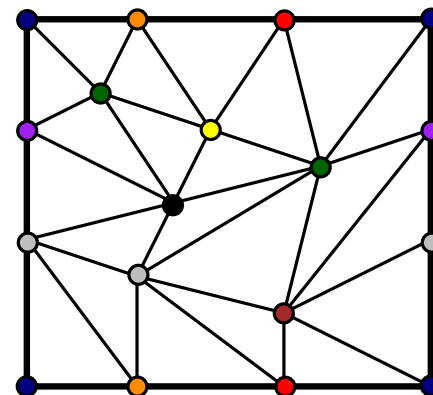


straight-line frame
not *x*-periodic
not *y*-periodic

[Chambers et al., GD'10]

[Duncan et al., GD'09]

$O(n \times n^2)$ **grid**

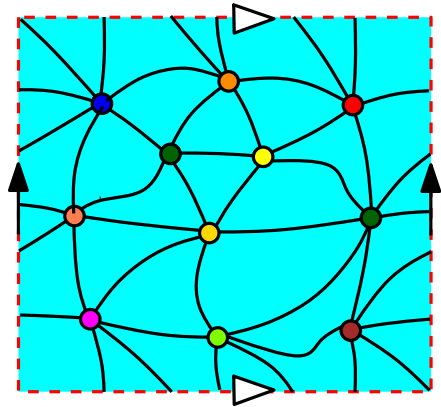
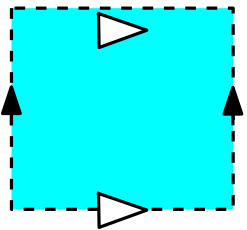
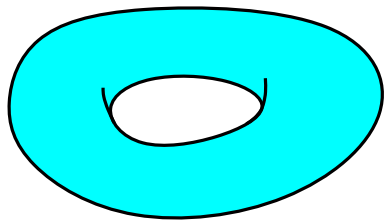


straight-line frame
x-periodic and
y-periodic drawing

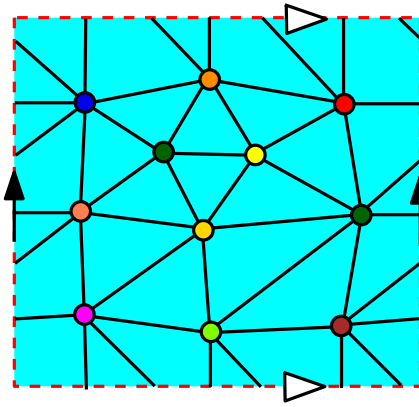
[Castelli Fusy Kostygin, Latin'14]

Periodic straight-line drawings

On the torus



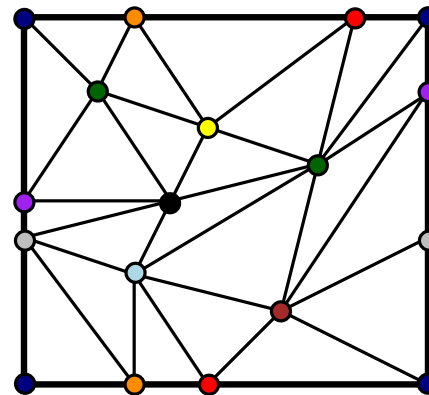
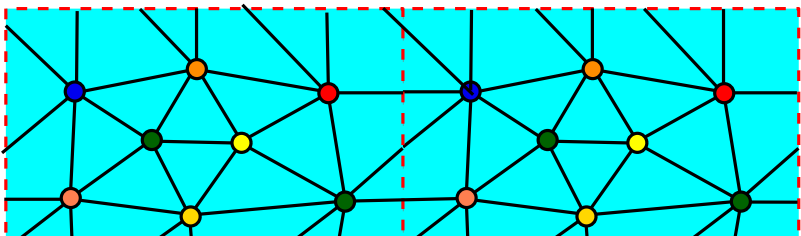
drawing on the flat torus



straight-line drawing
x-periodic and
y-periodic drawing

[Castelli Devillers Fusy, GD'12]
 $O(n \times n^{\frac{3}{2}})$ **grid**

[Goncalves Lévêque, DCG]
 $O(n^2 \times n^2)$ **grid**

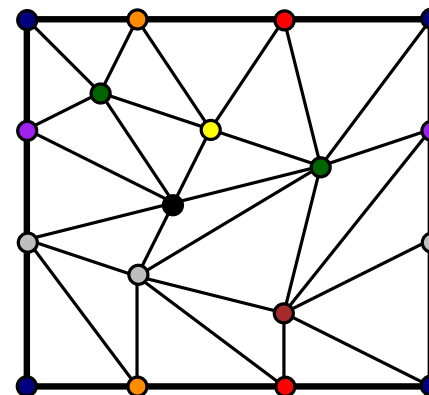


straight-line frame
not *x*-periodic
not *y*-periodic

[Chambers et al., GD'10]

[Duncan et al., GD'09]

$O(n \times n^2)$ **grid**



straight-line frame
x-periodic and
y-periodic drawing

[Castelli Fusy Kostygin, Latin'14]

$O(n^4 \times n^4)$ **grid**

A shift-algorithm for the torus

1. Recall algorithm of

2. Extend to the cylinder

3. Get toroidal drawings

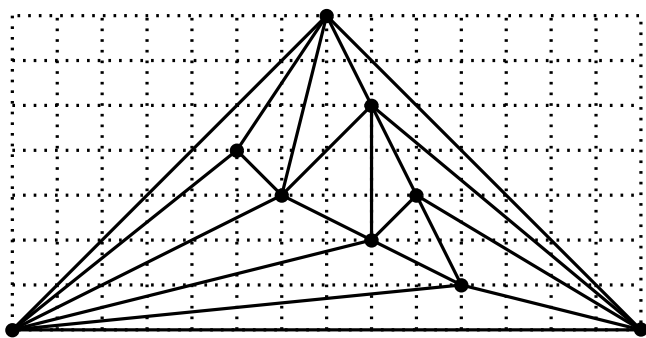
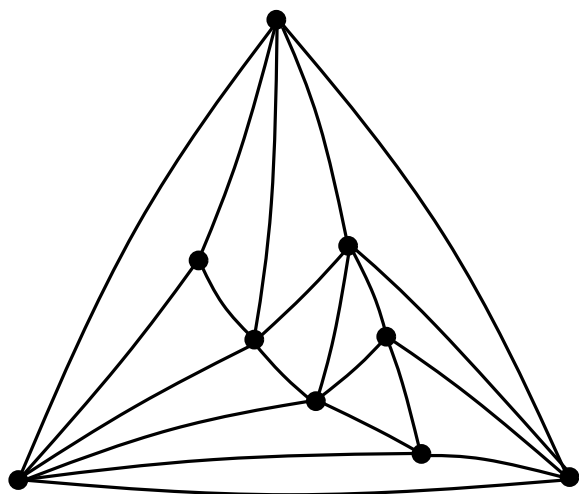
[De Fraysseix et al'89]

[Castelli Aleardi Fusy Devillers 2012]

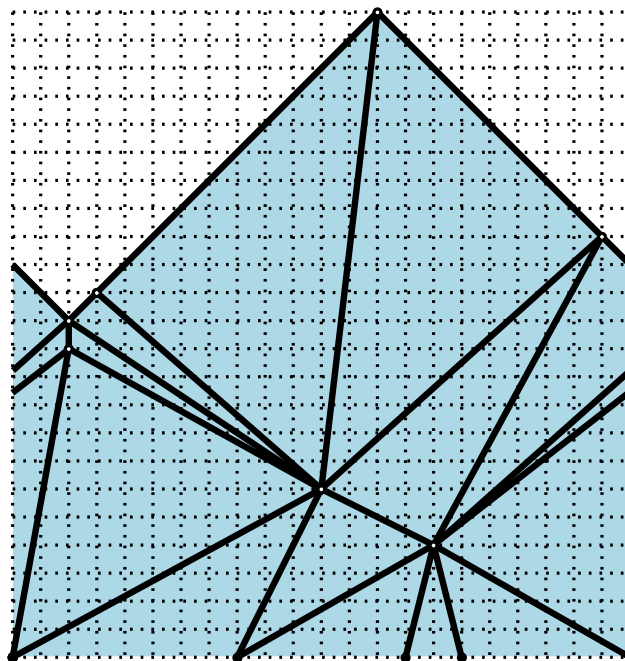
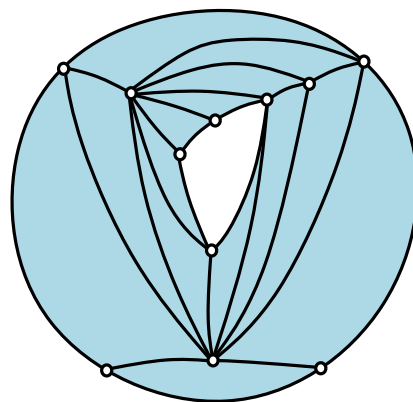
Plane

Cylinder

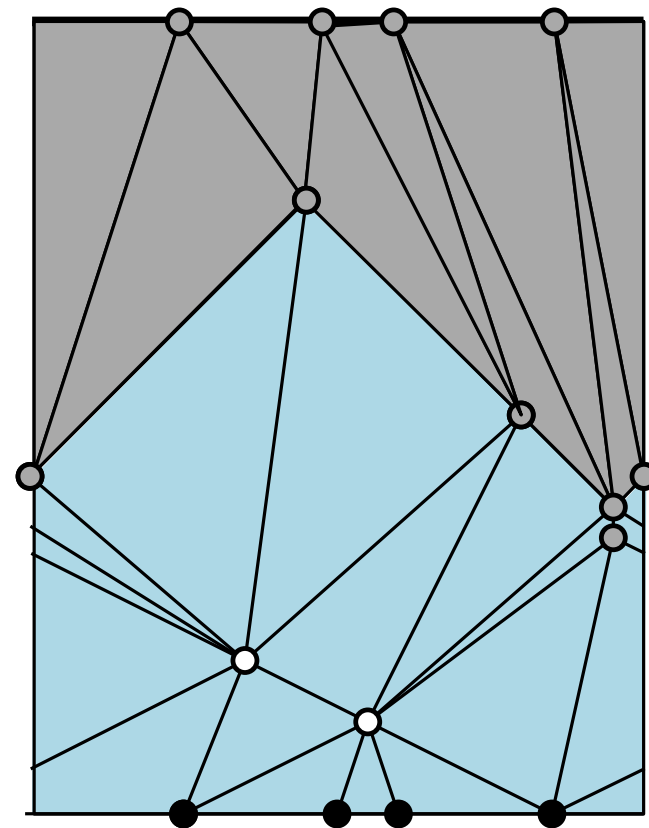
Torus



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



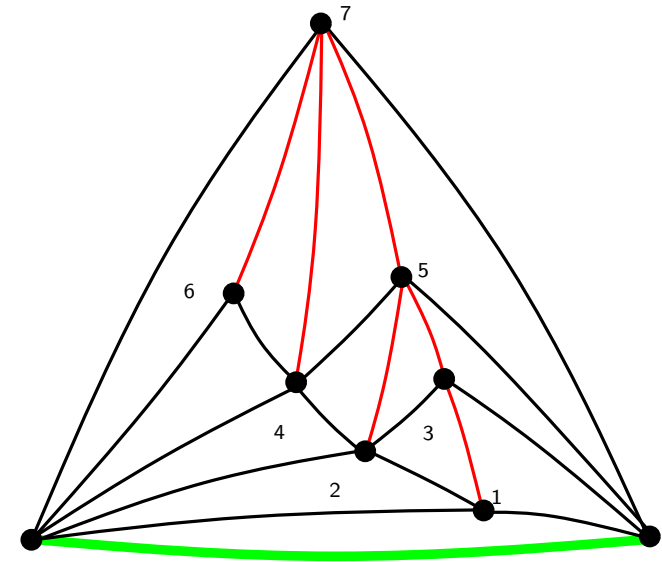
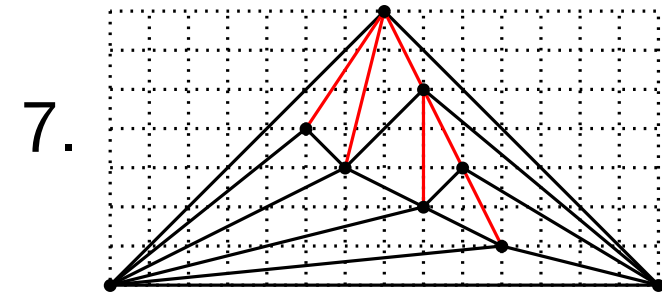
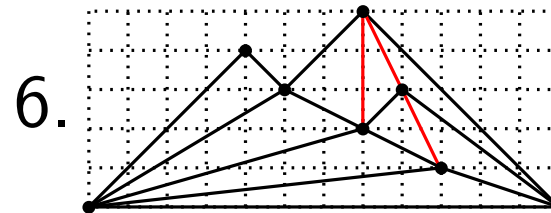
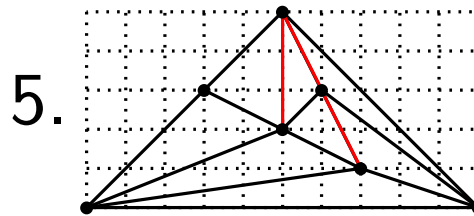
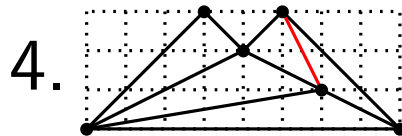
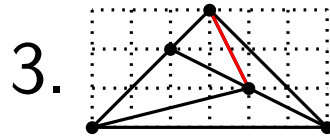
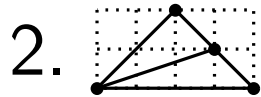
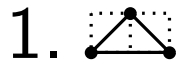
Grid $\leq 2n \times n(2d+1)$



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

Incremental drawing algorithm

[de Fraysseix, Pollack, Pach'89]

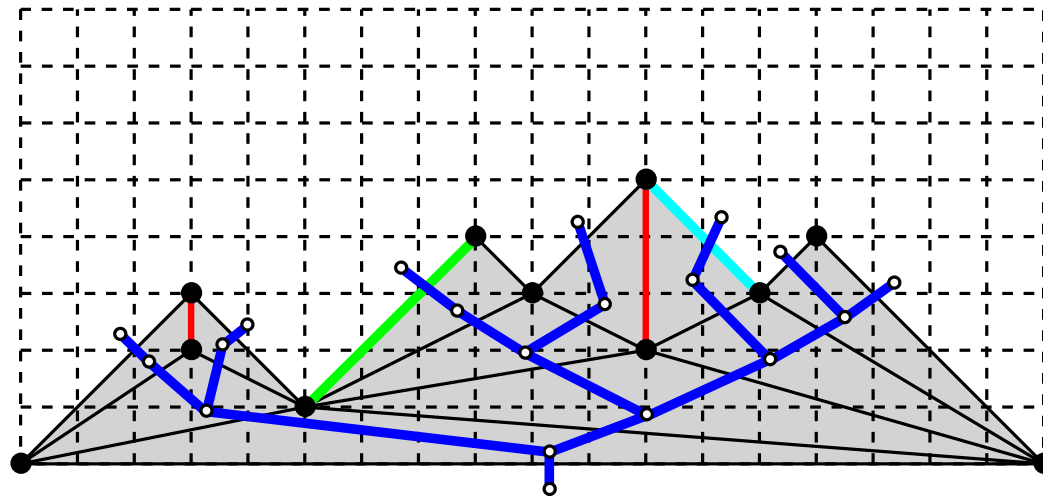
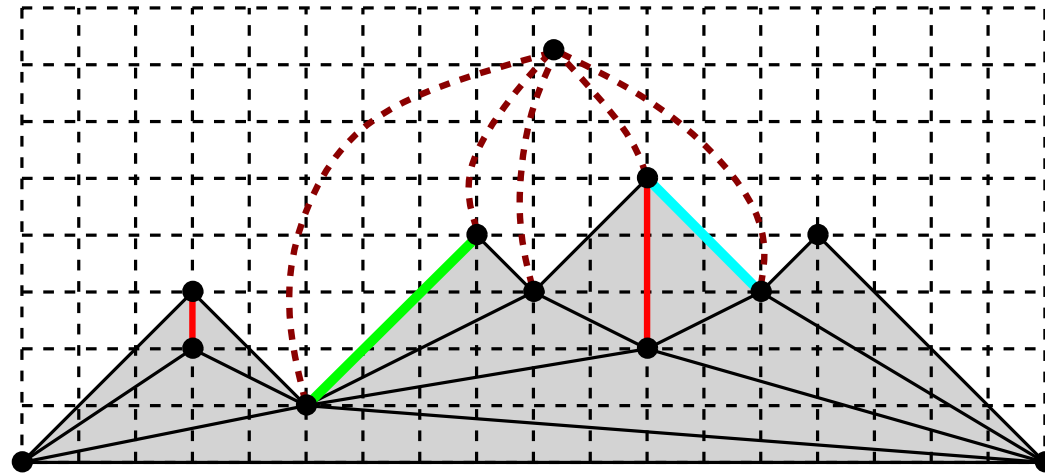


Grid size of G_k : $2k \times k$

Reformulation of the shift-step

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

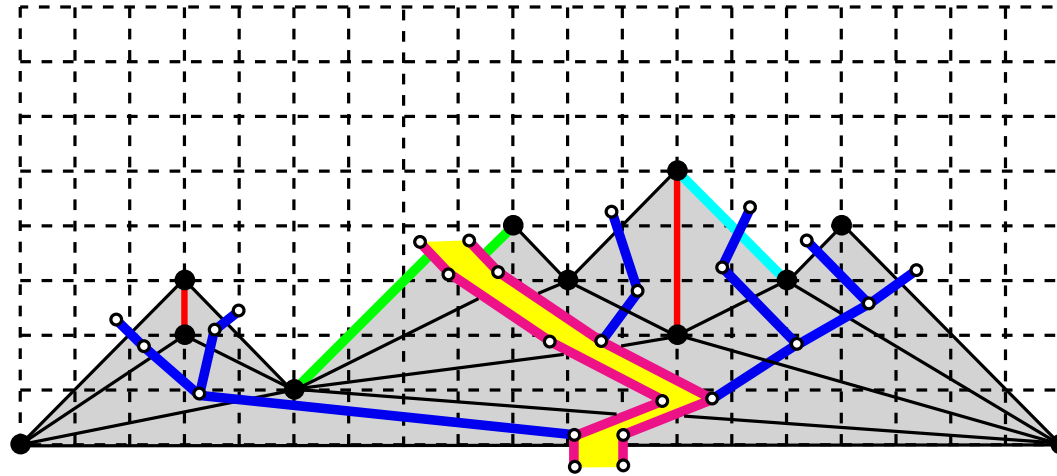
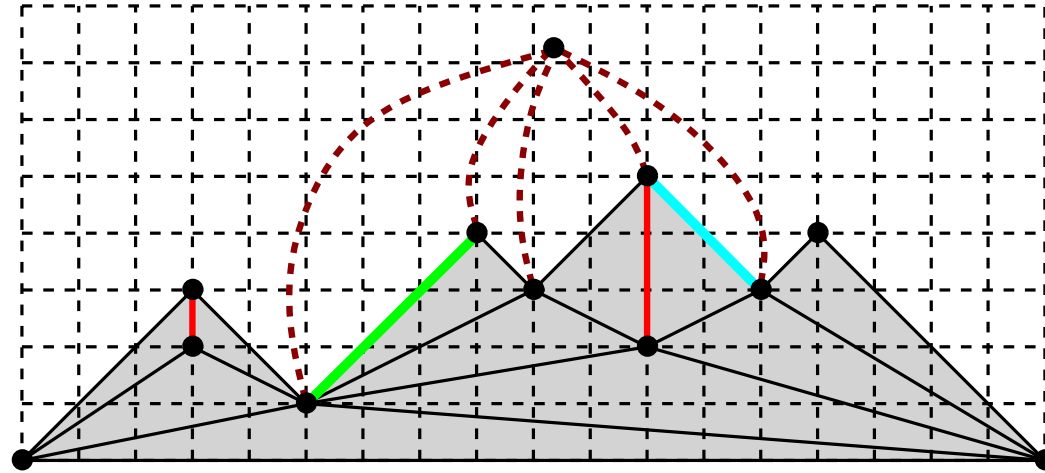
G_{k-1}



Reformulation of the shift-step

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

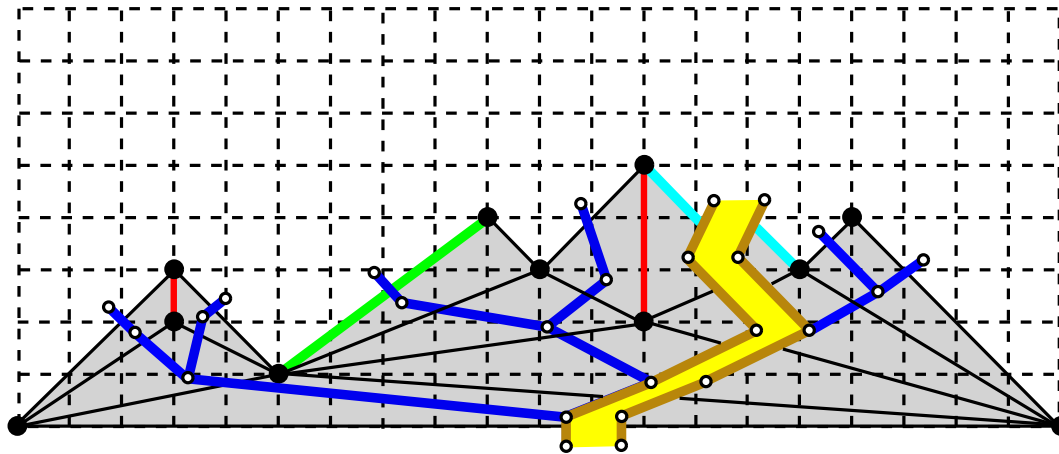
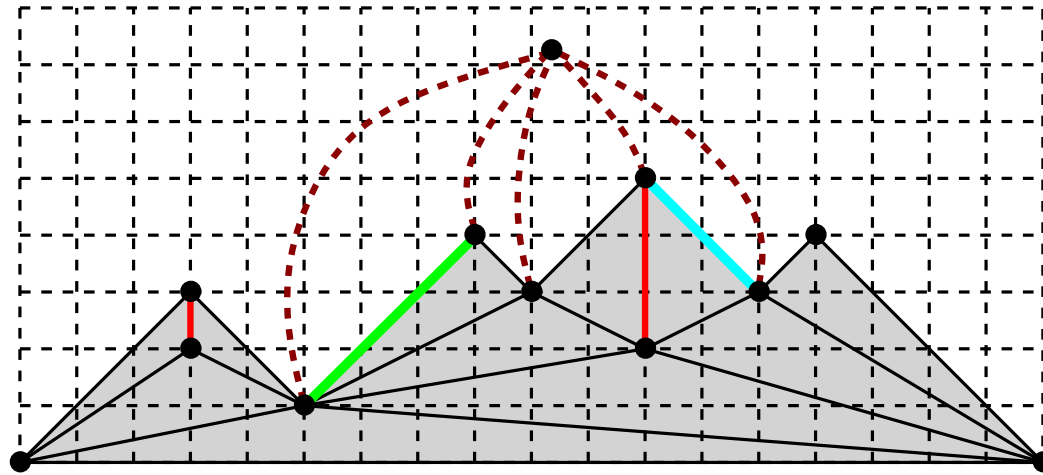
G_{k-1}



Reformulation of the shift-step

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

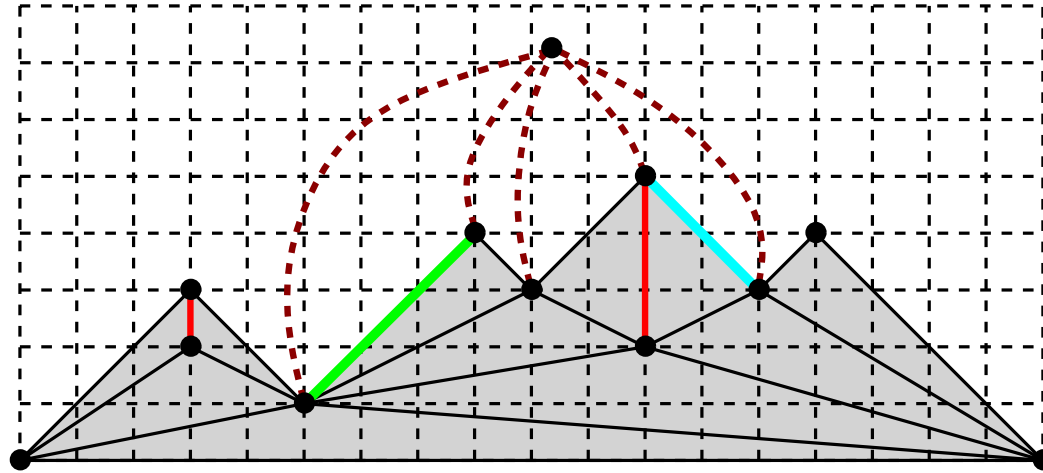
G_{k-1}



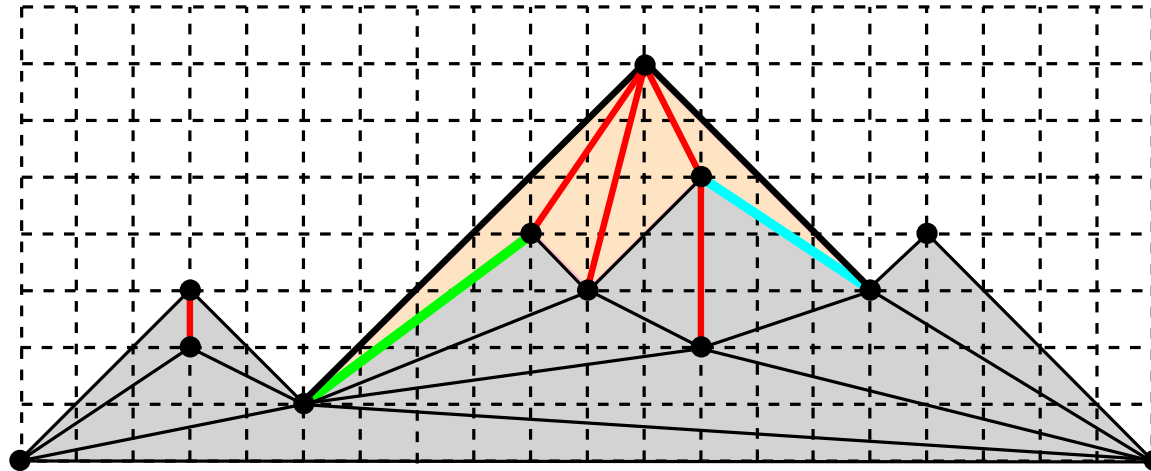
Reformulation of the shift-step

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

G_{k-1}

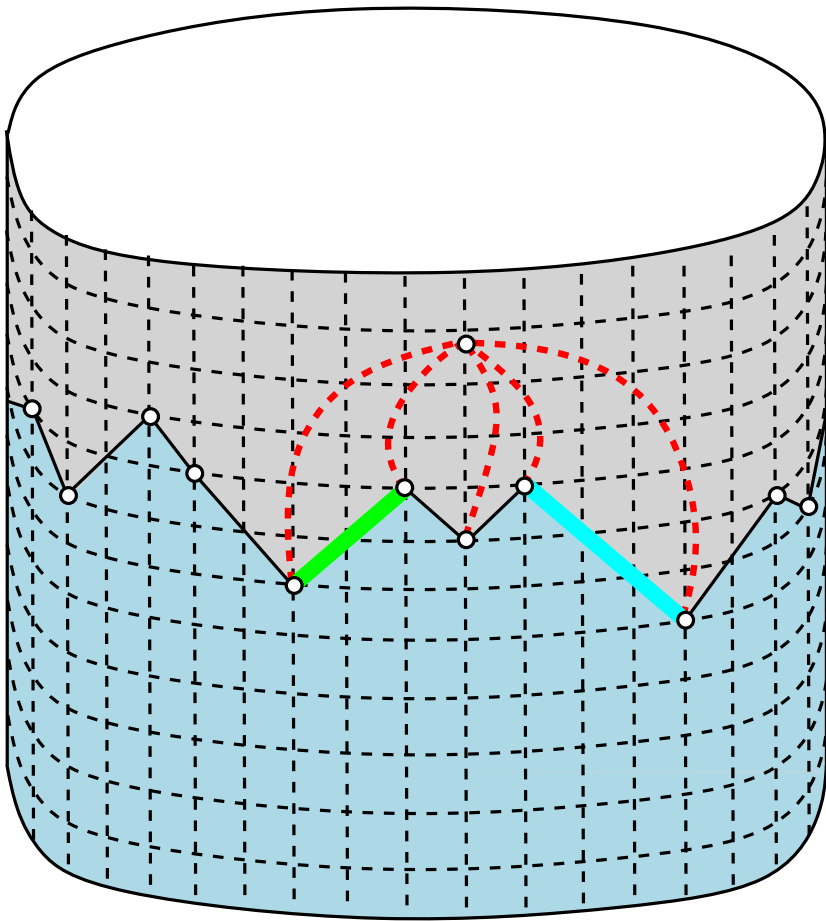


G_k



Extension to the cylinder: drawing algorithm

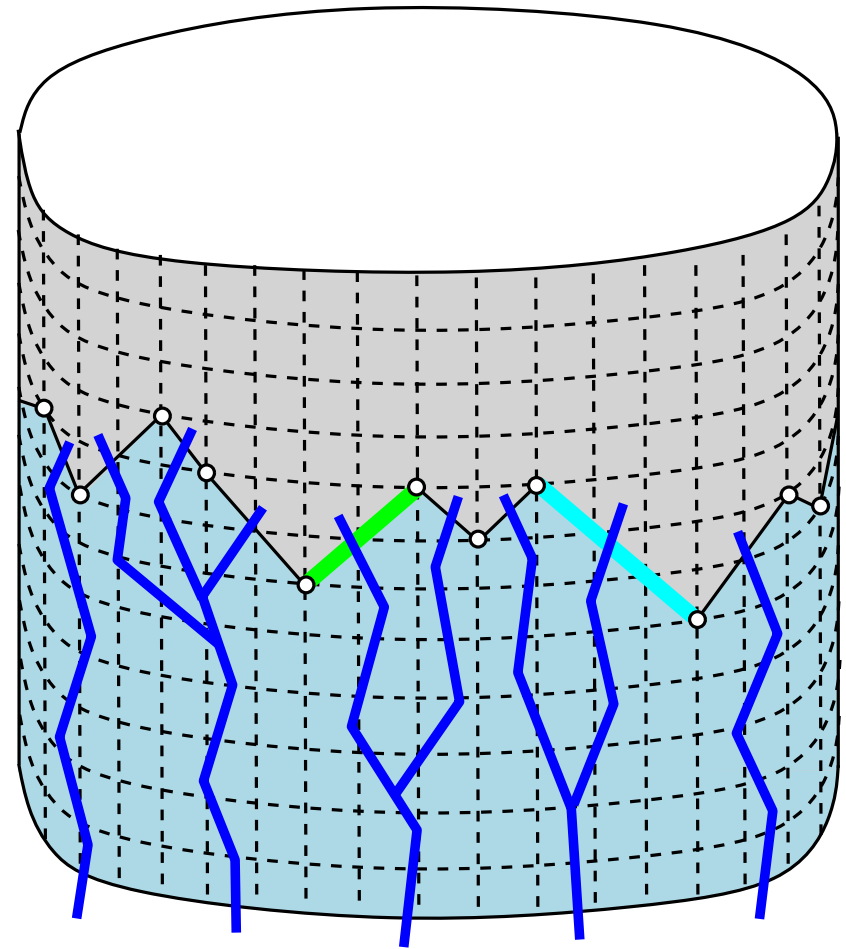
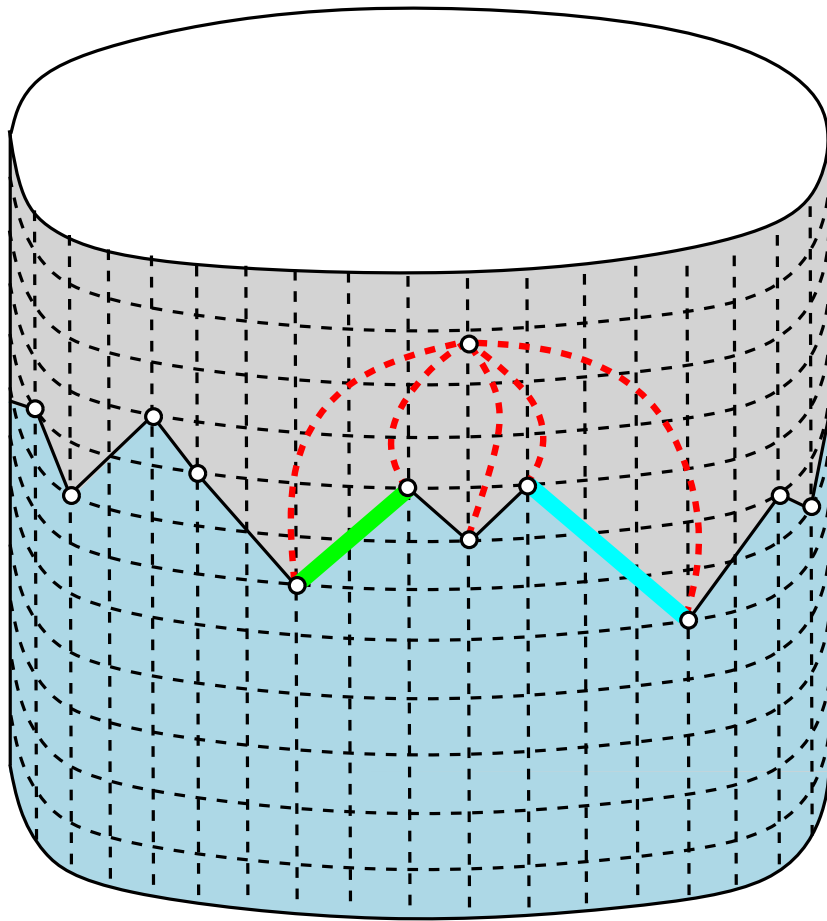
G_{k-1}



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

Extension to the cylinder: drawing algorithm

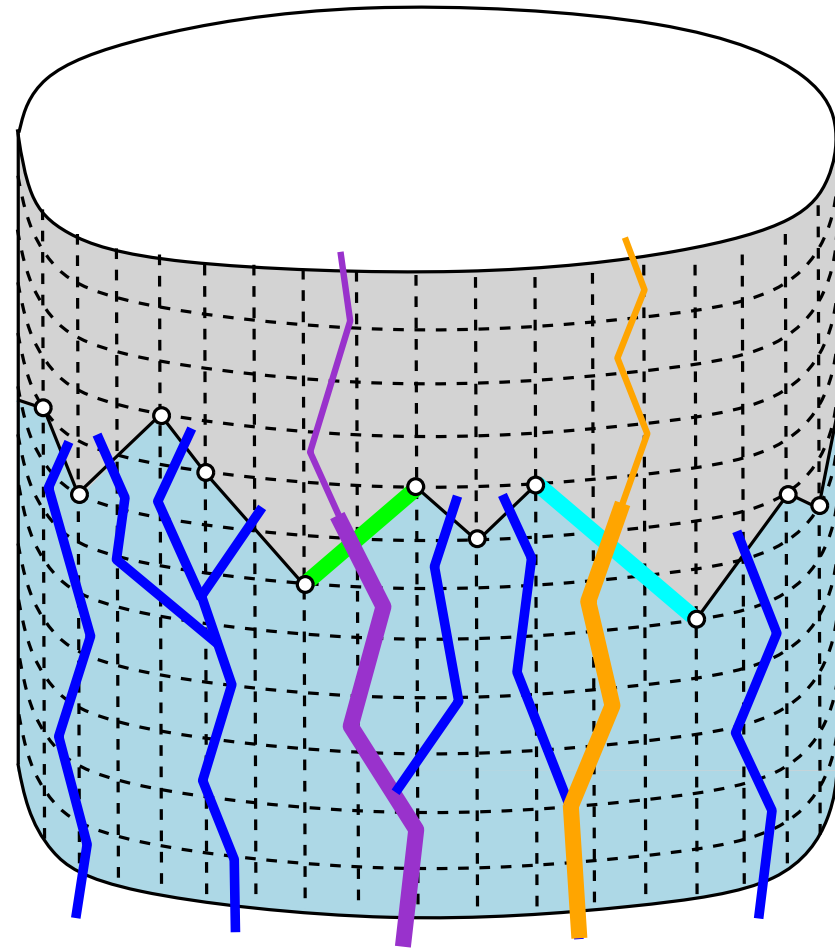
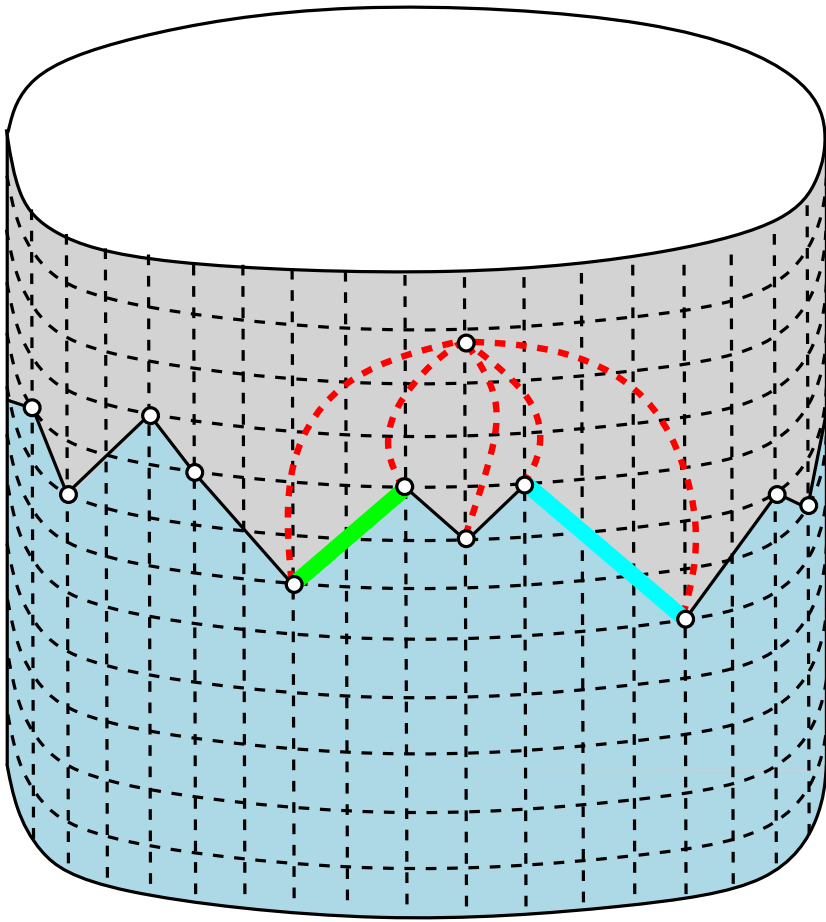
G_{k-1}



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

Extension to the cylinder: drawing algorithm

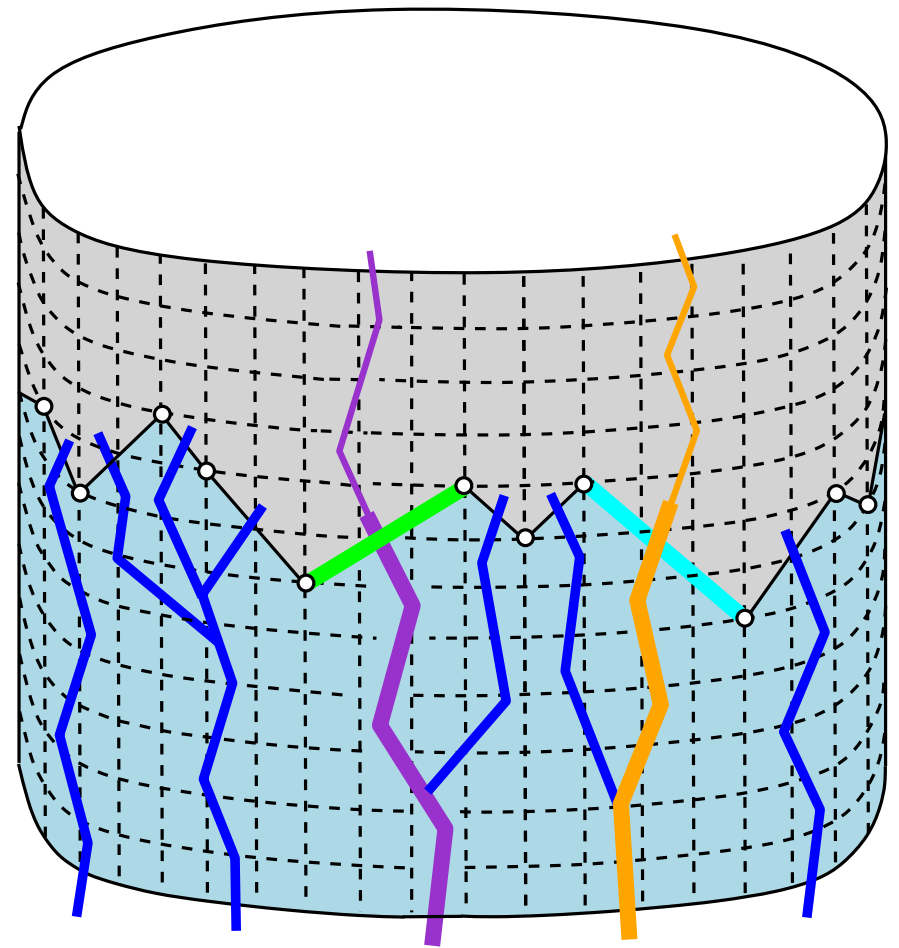
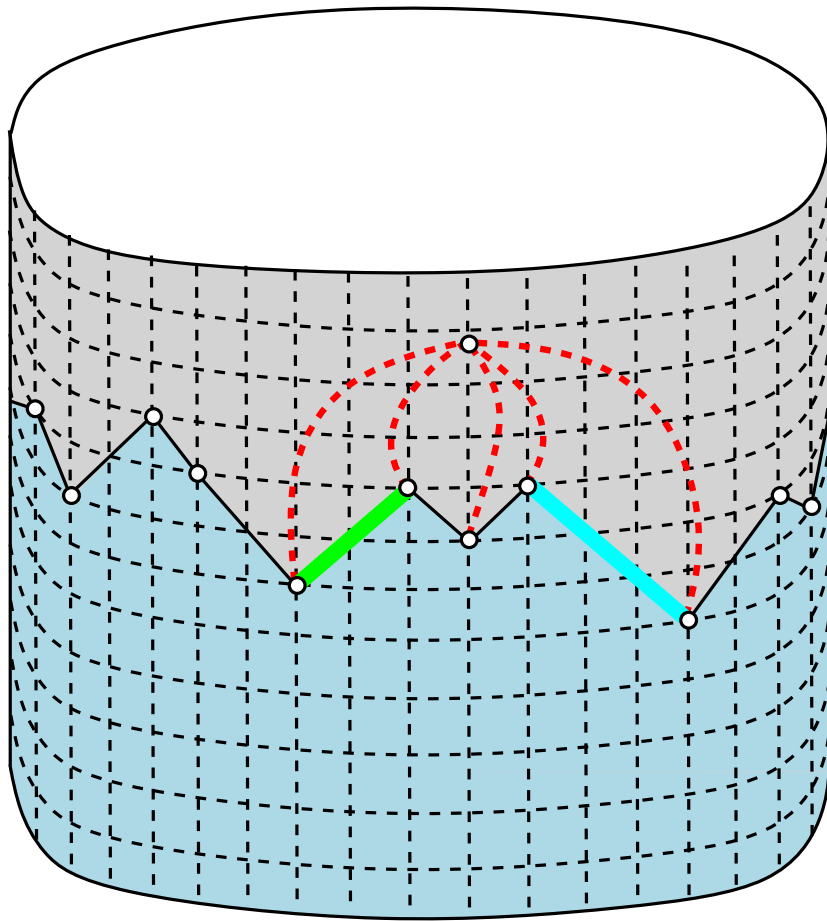
G_{k-1}



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

Extension to the cylinder: drawing algorithm

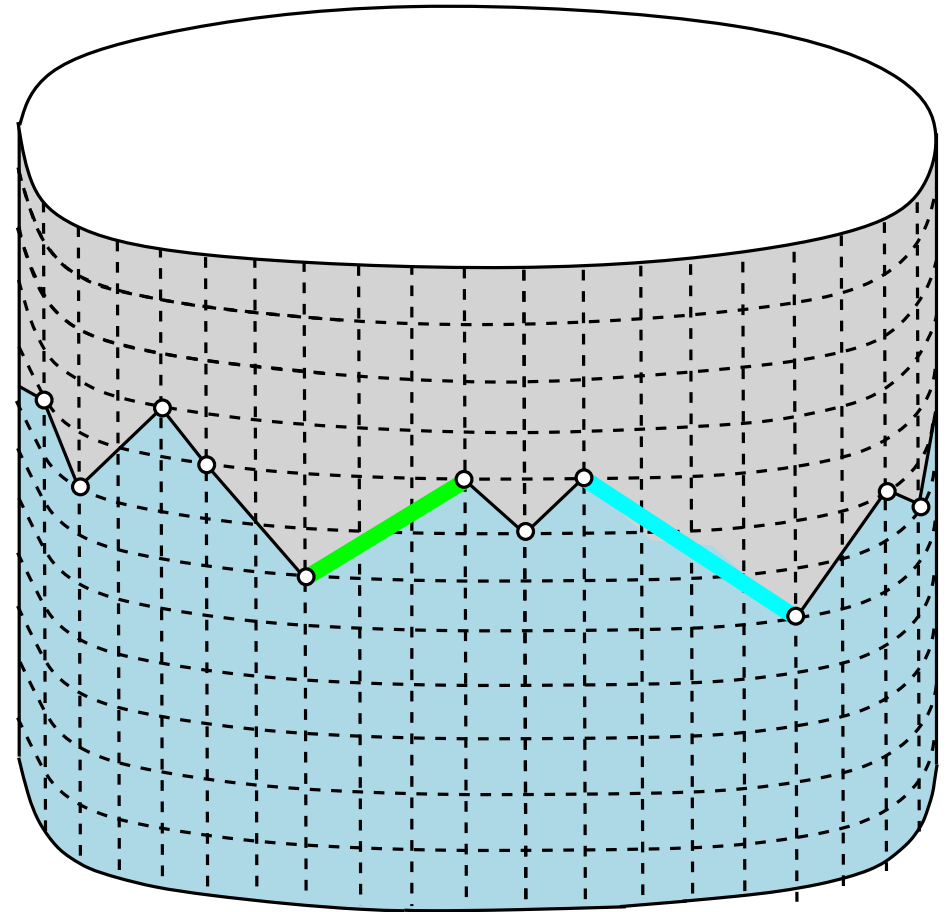
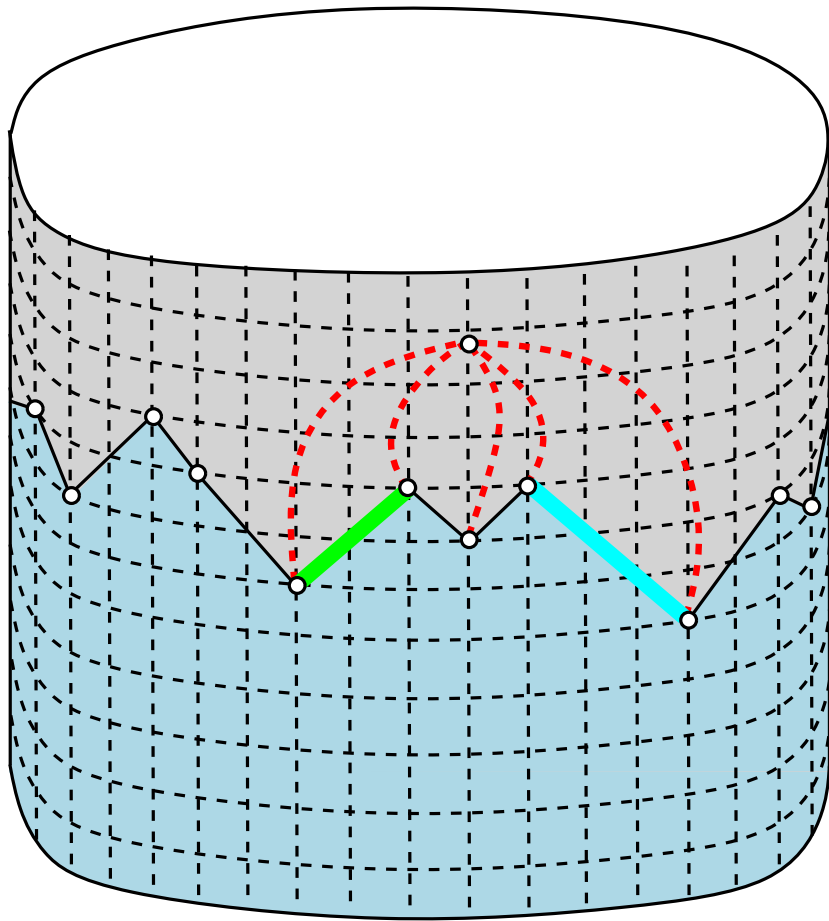
G_{k-1}



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

Extension to the cylinder: drawing algorithm

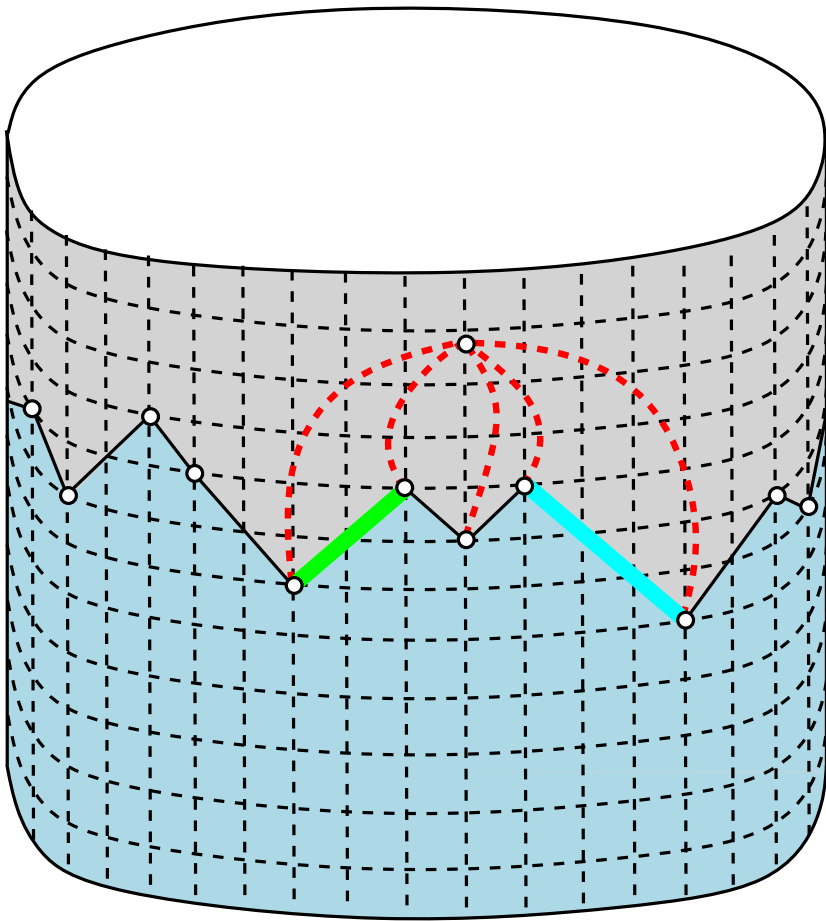
G_{k-1}



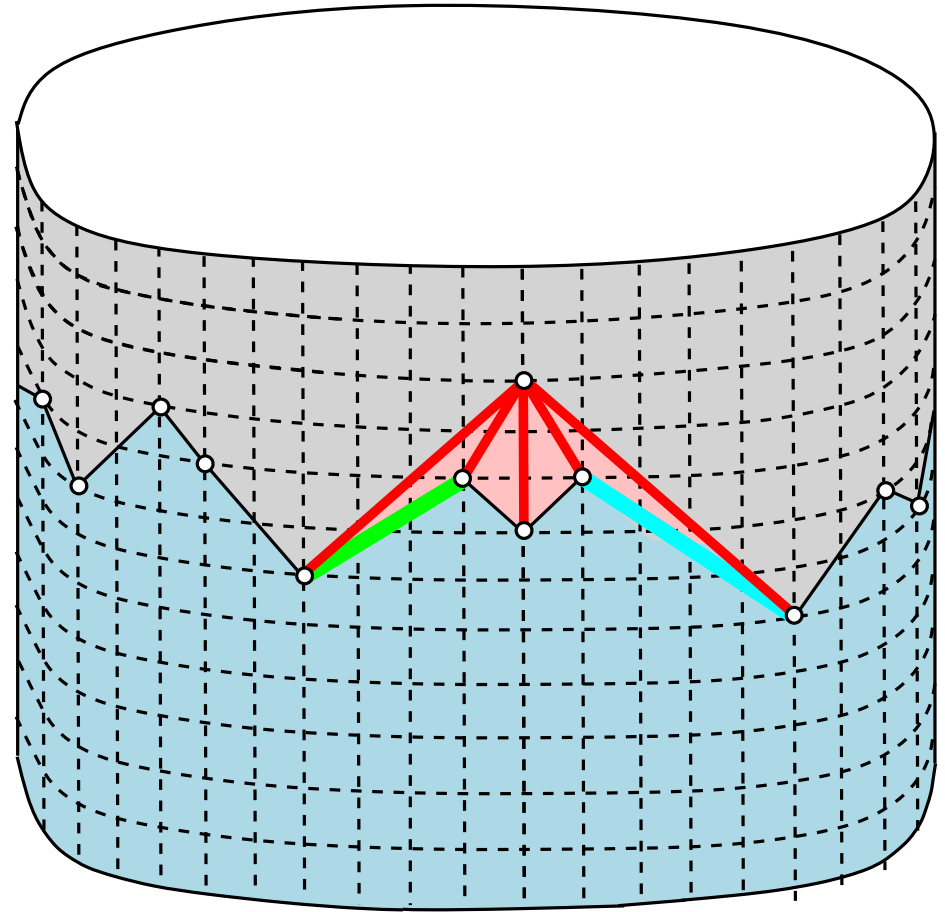
- At each step:
- insert two vertical strips of width 1
 - insert the next vertex as in the planar case

Extension to the cylinder: drawing algorithm

G_{k-1}



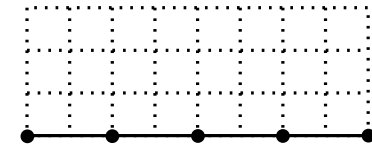
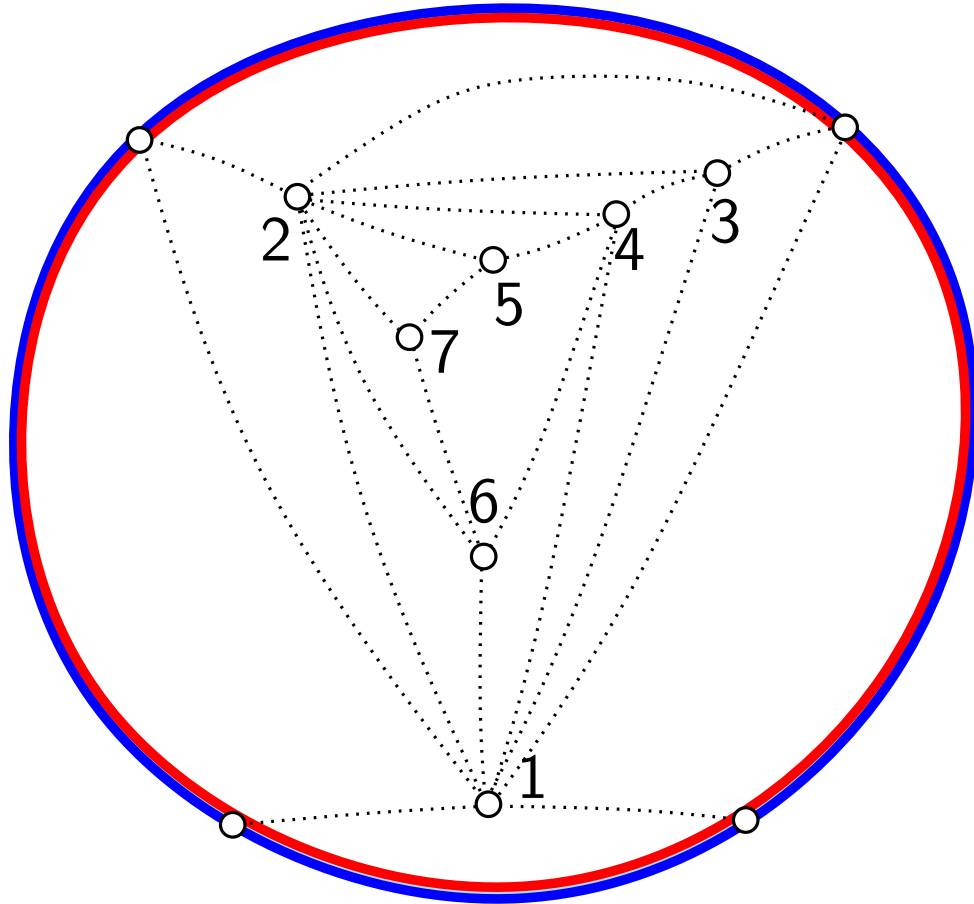
G_k



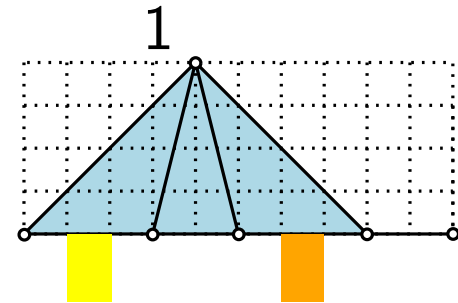
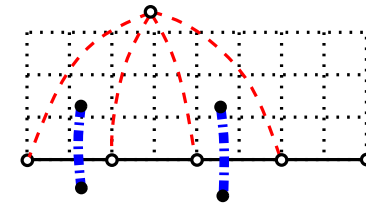
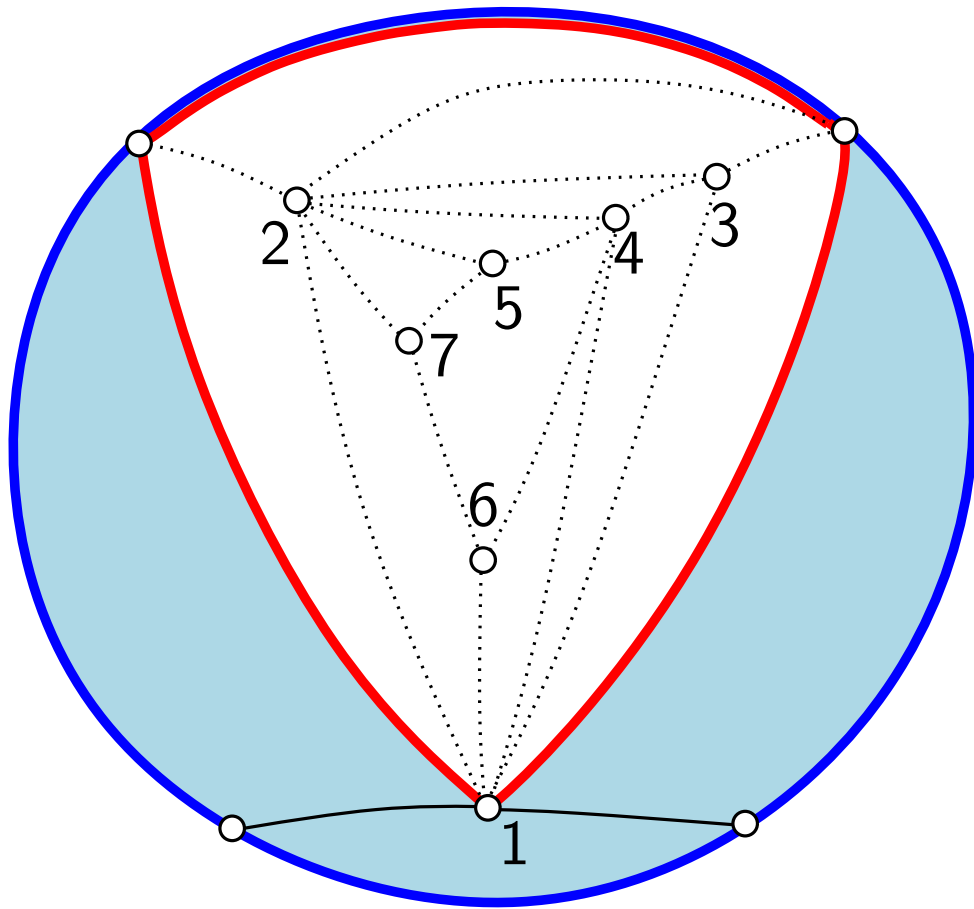
At each step:

- insert two vertical strips of width 1
- insert the next vertex as in the planar case

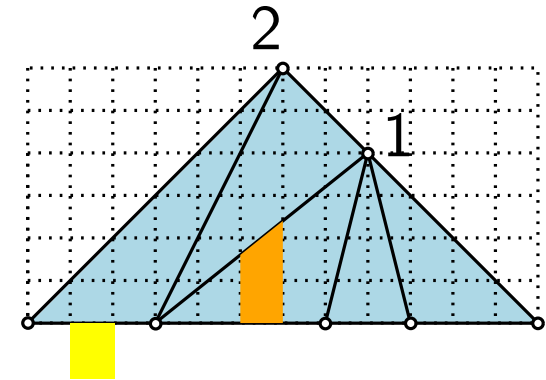
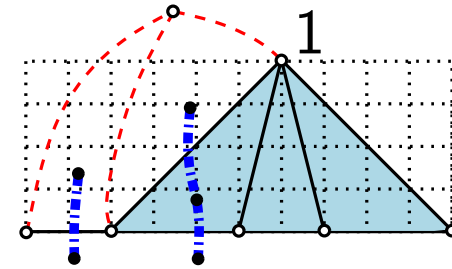
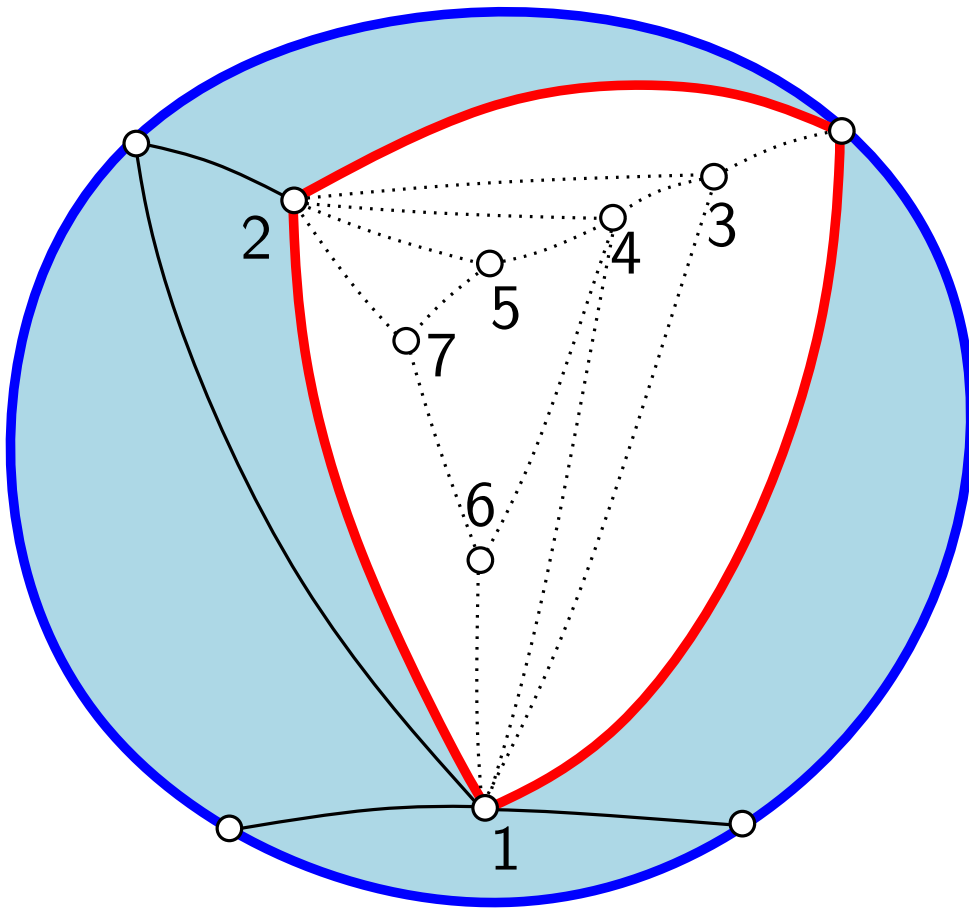
Extension to the cylinder: drawing algorithm



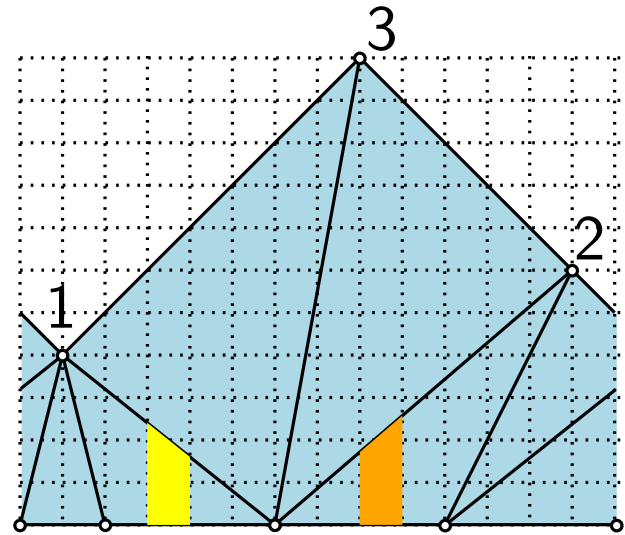
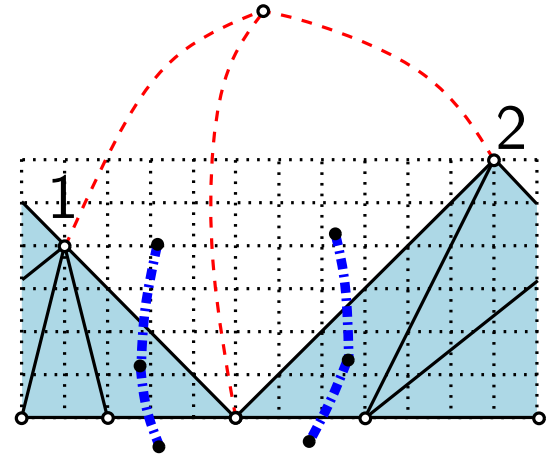
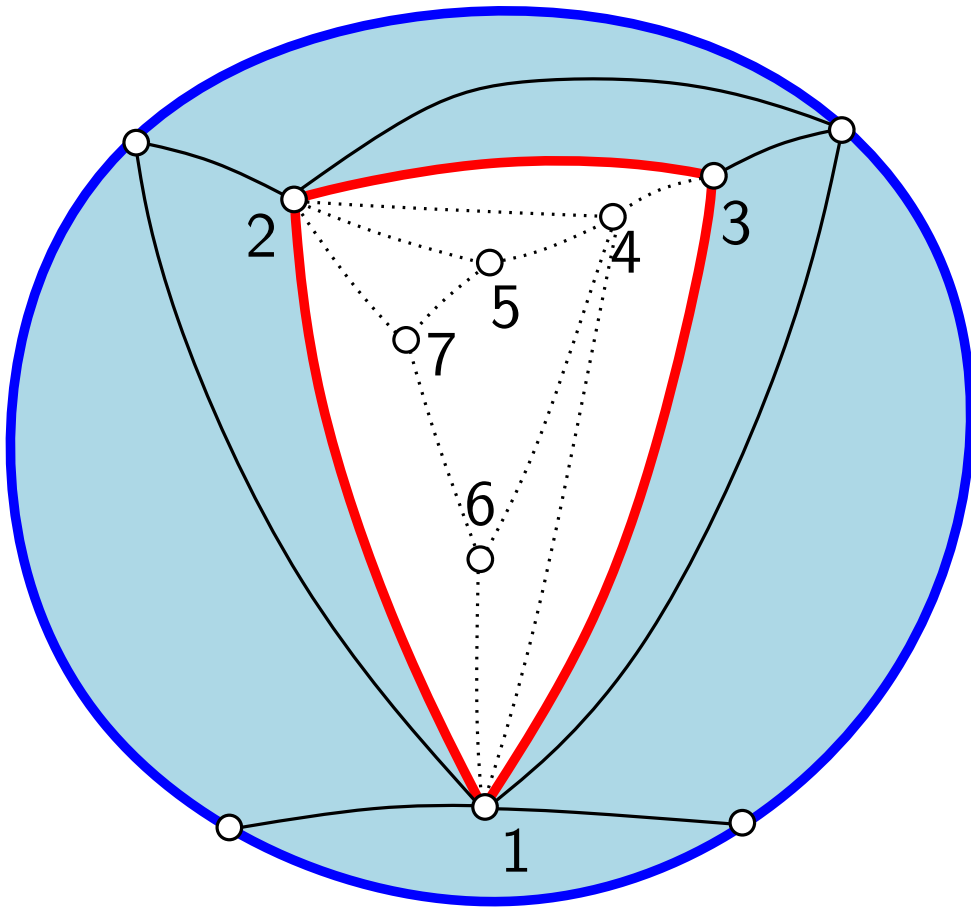
Extension to the cylinder: drawing algorithm



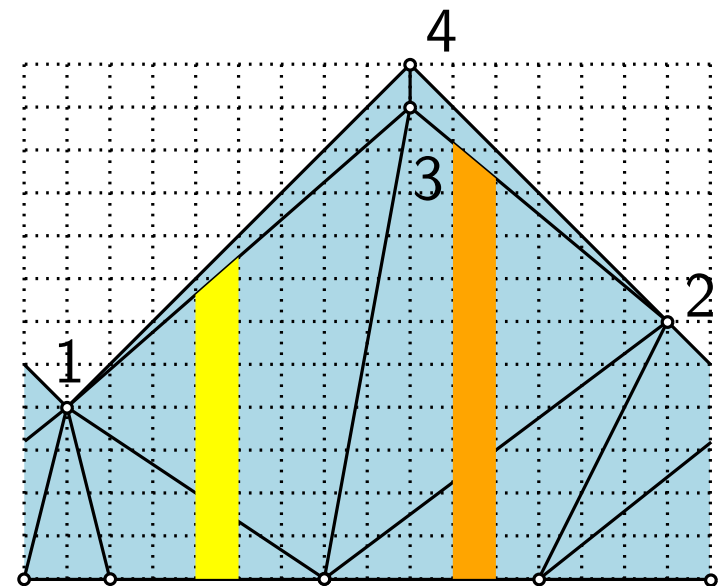
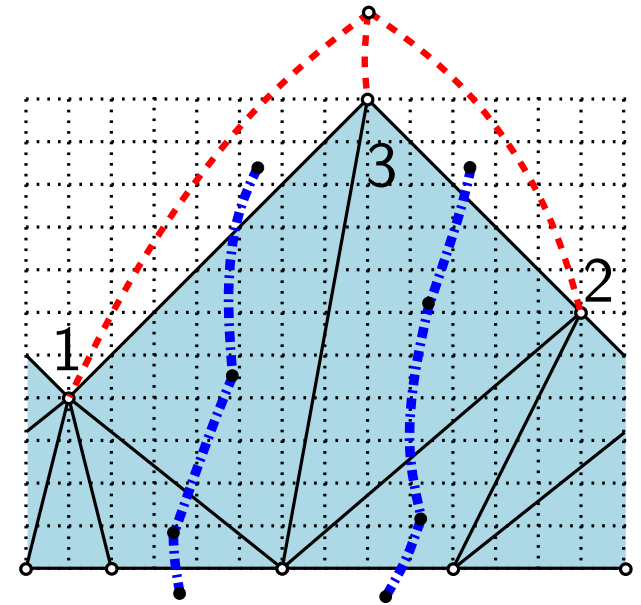
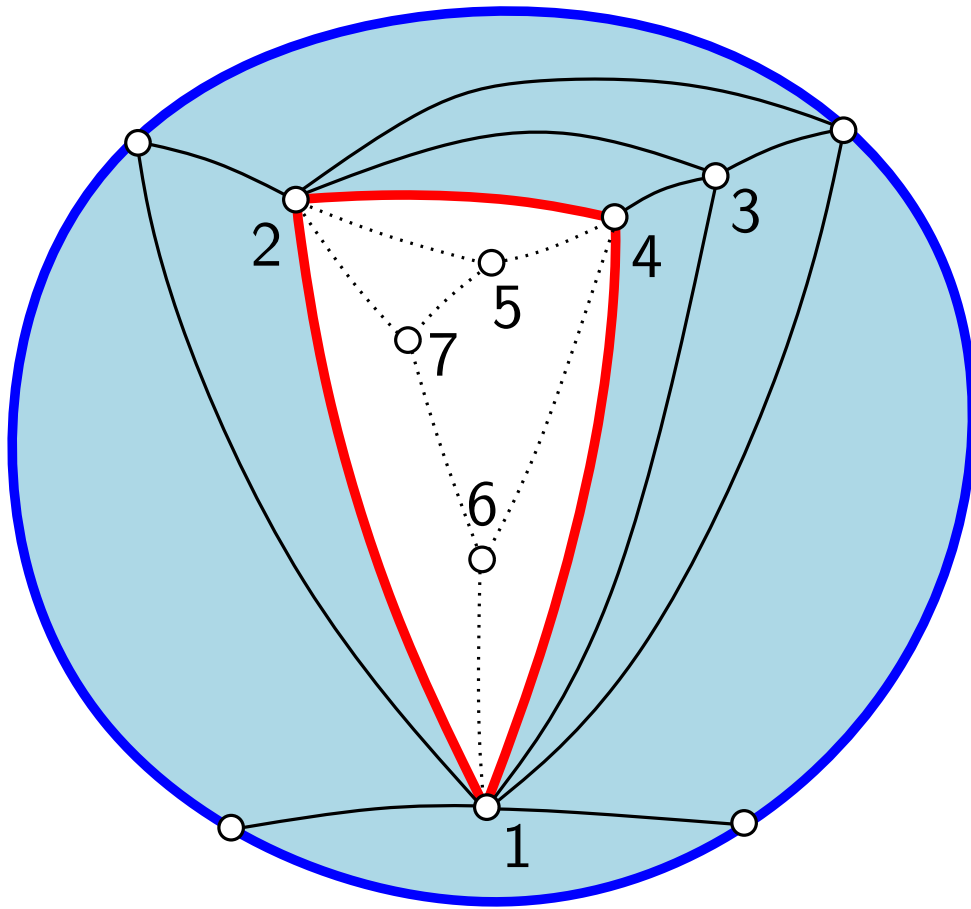
Extension to the cylinder: drawing algorithm



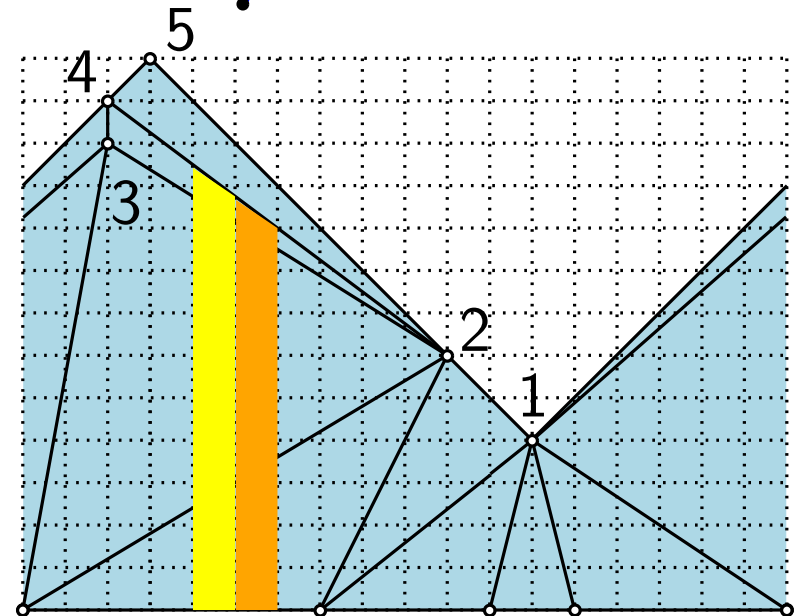
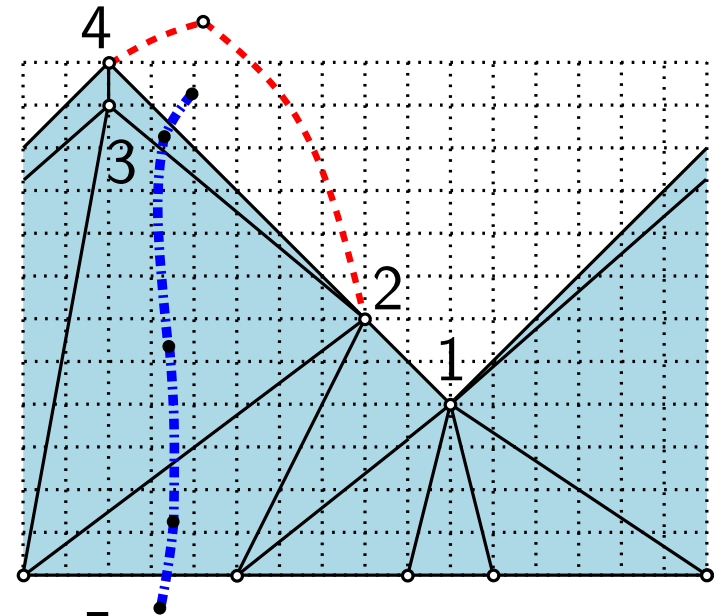
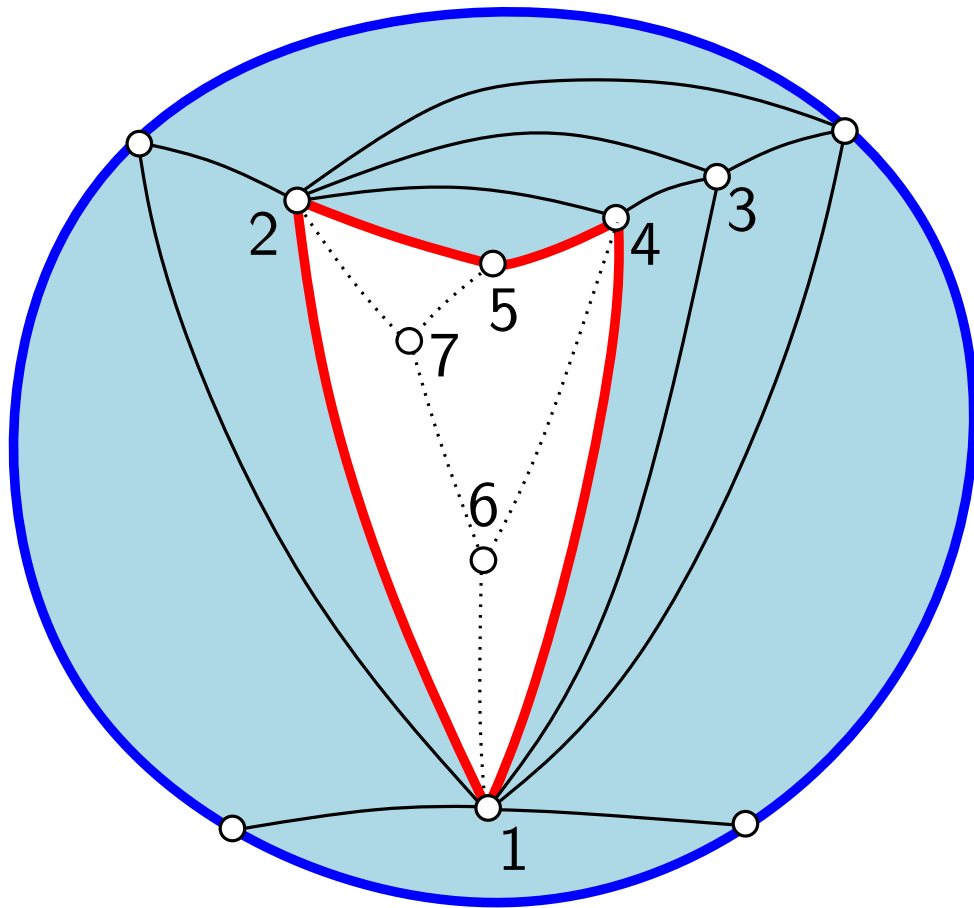
Extension to the cylinder: drawing algorithm



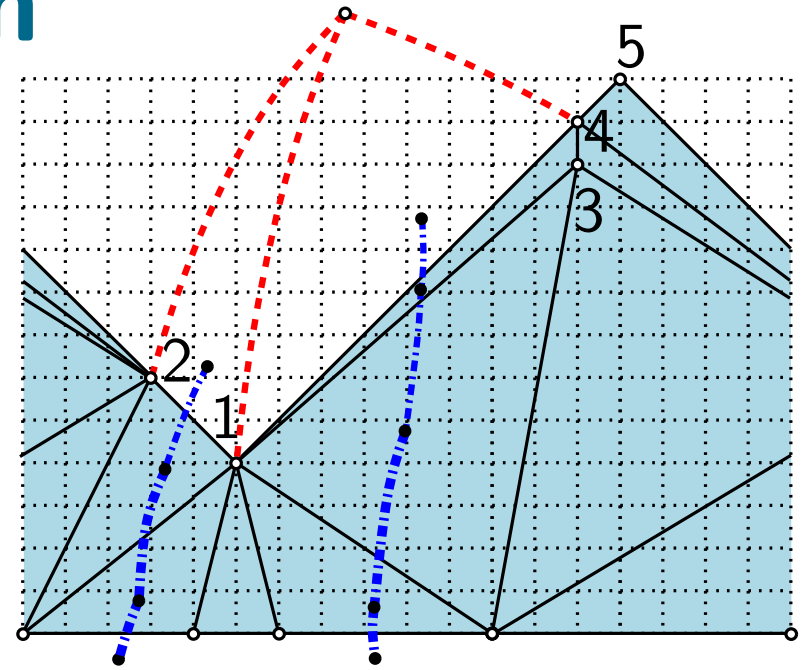
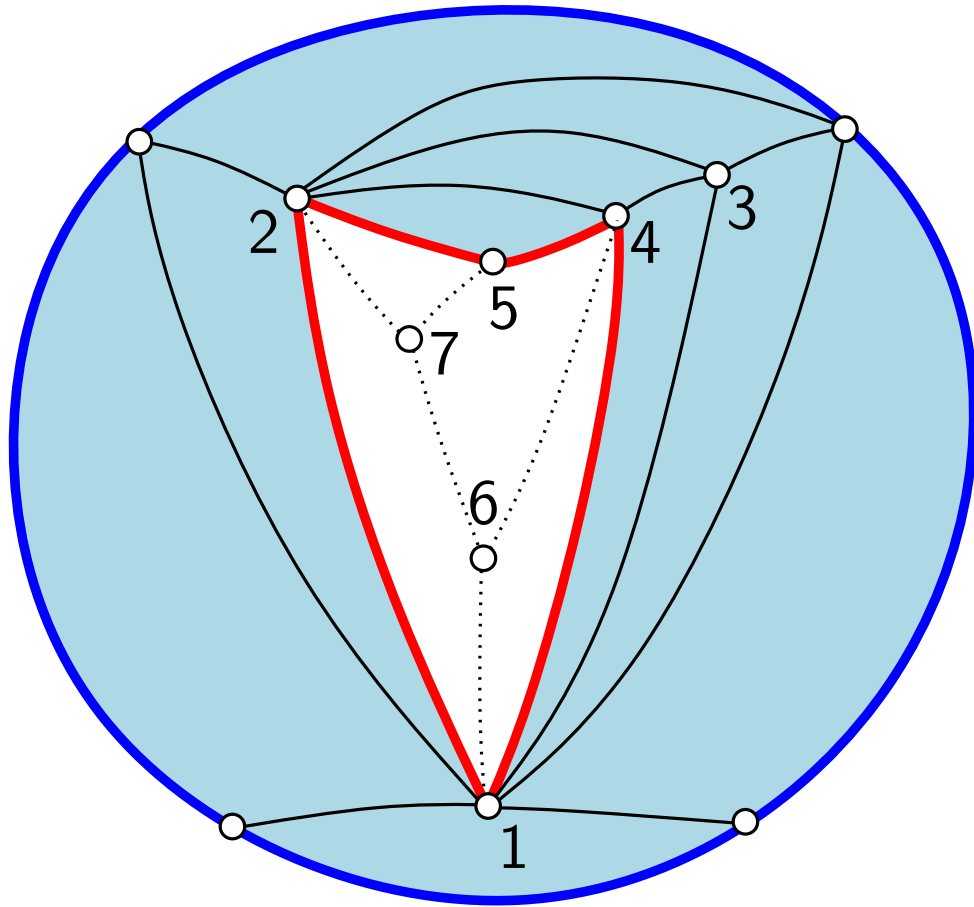
Extension to the cylinder: drawing algorithm



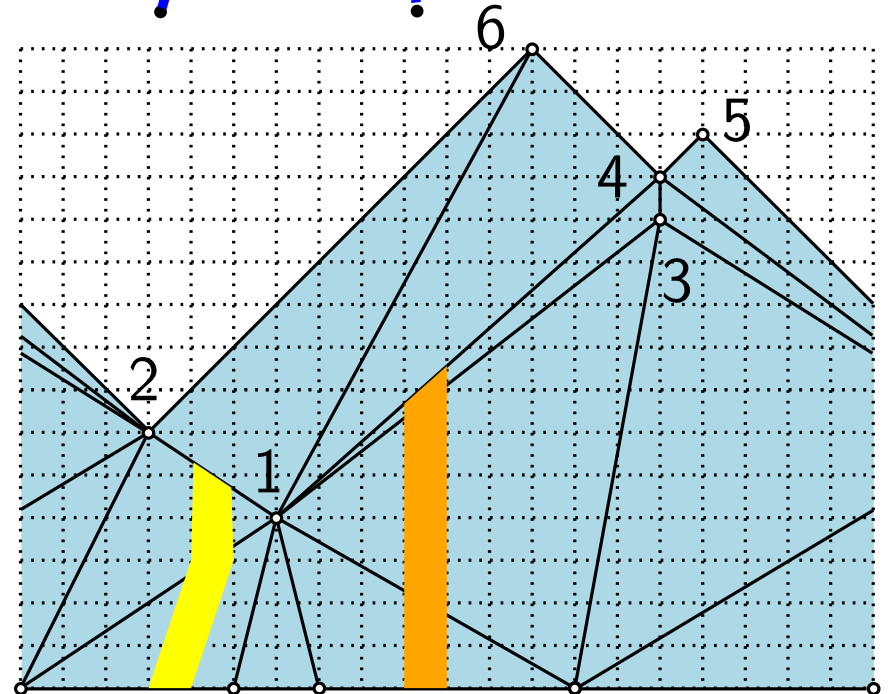
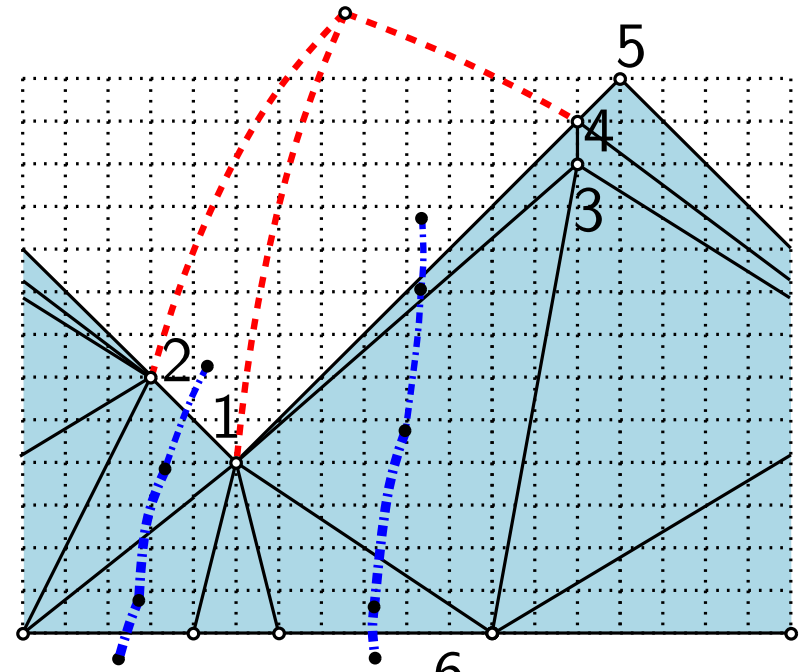
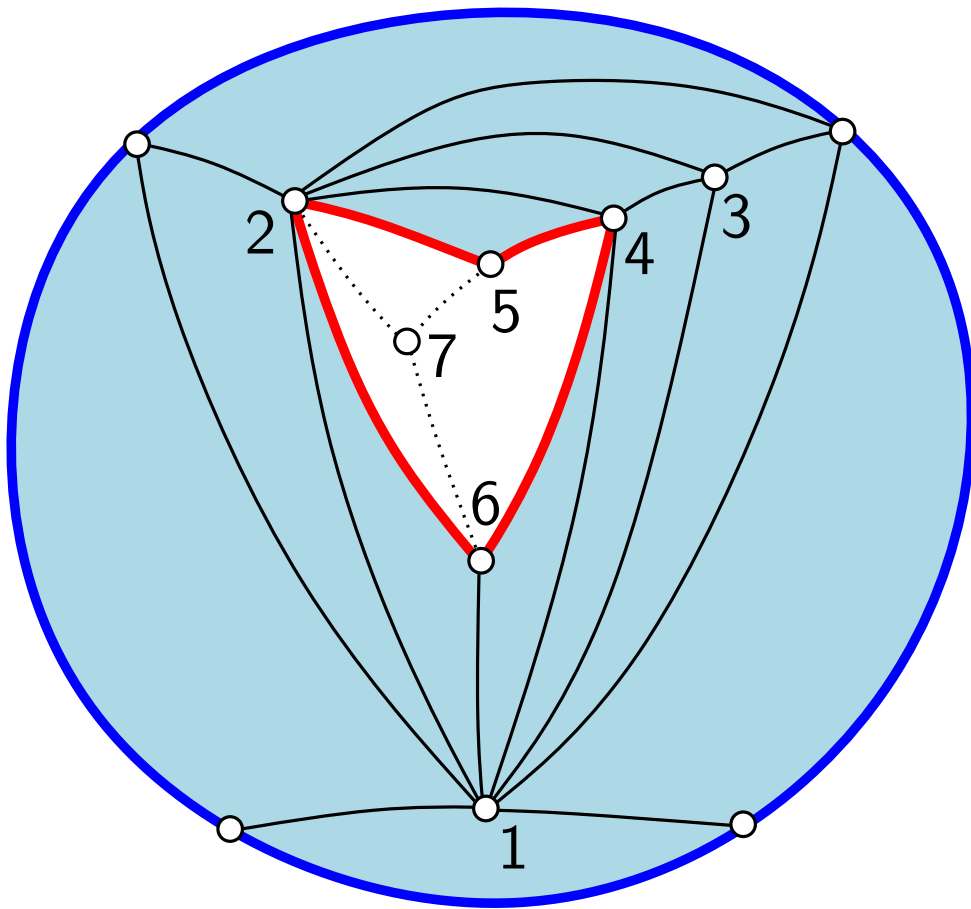
Extension to the cylinder: drawing algorithm



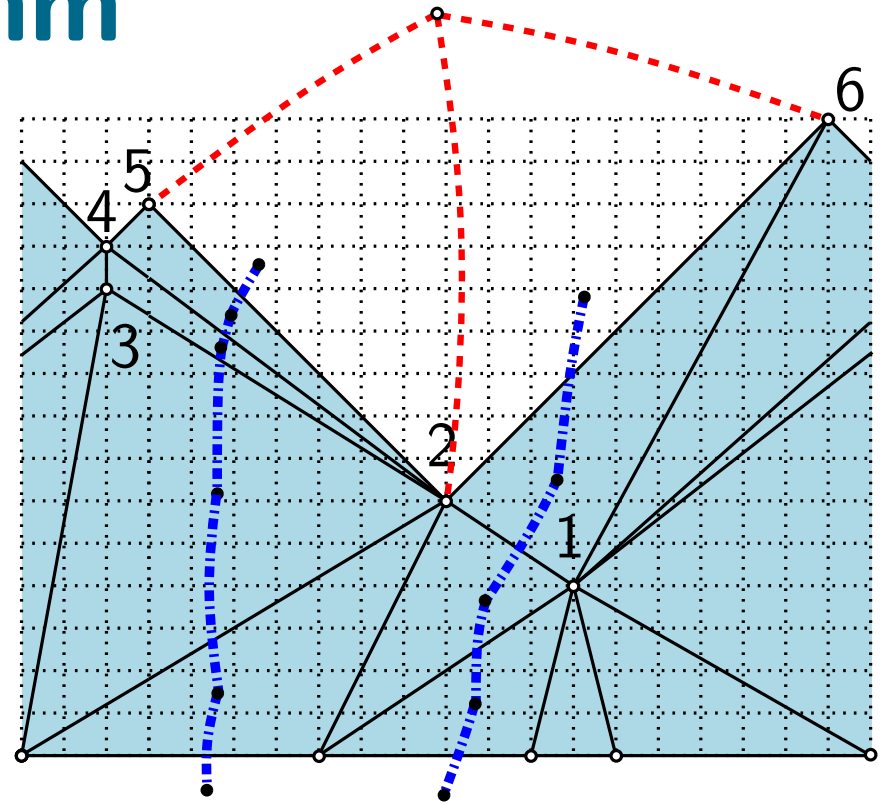
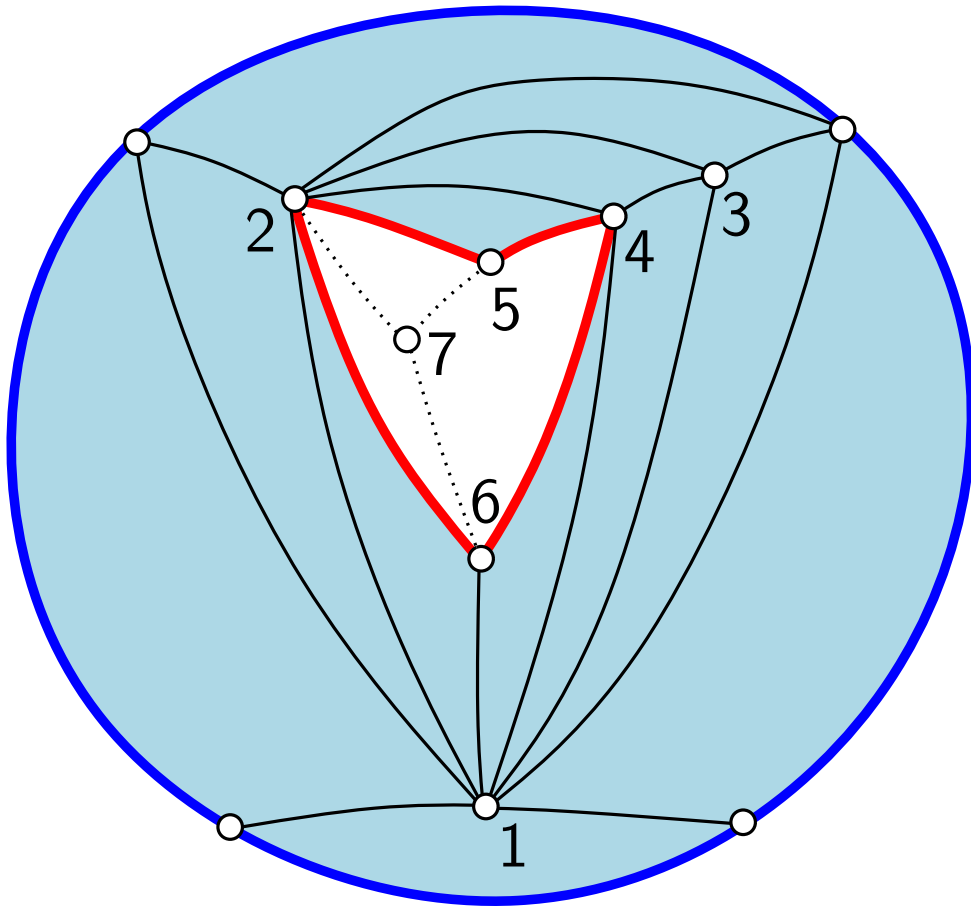
Extension to the cylinder: drawing algorithm



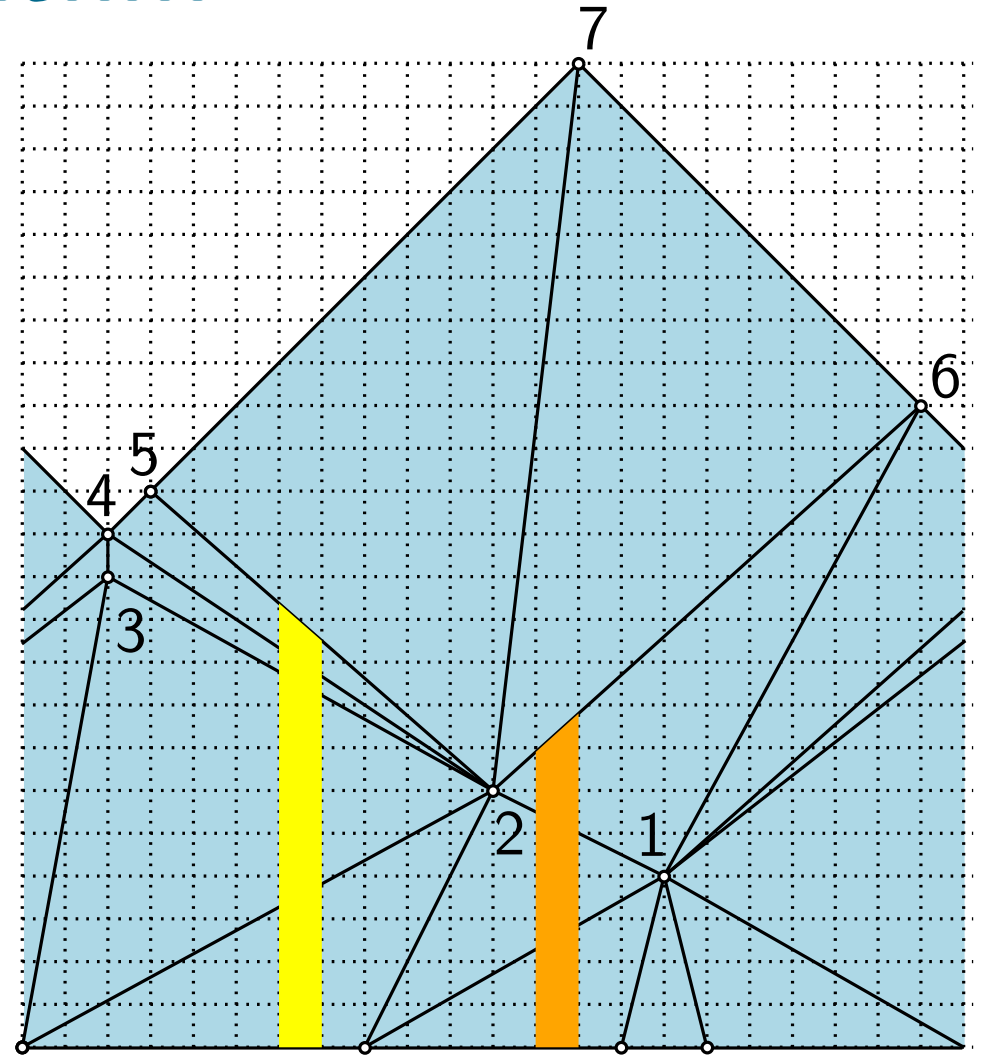
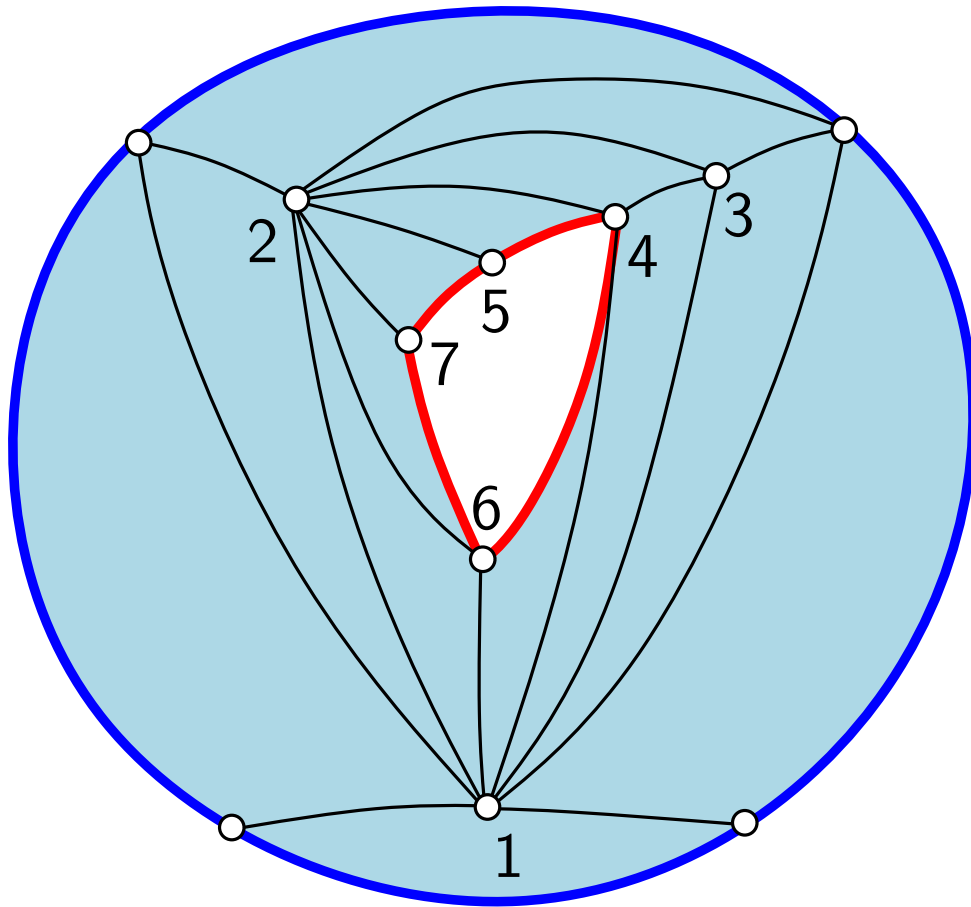
Extension to the cylinder: drawing algorithm



Extension to the cylinder: drawing algorithm



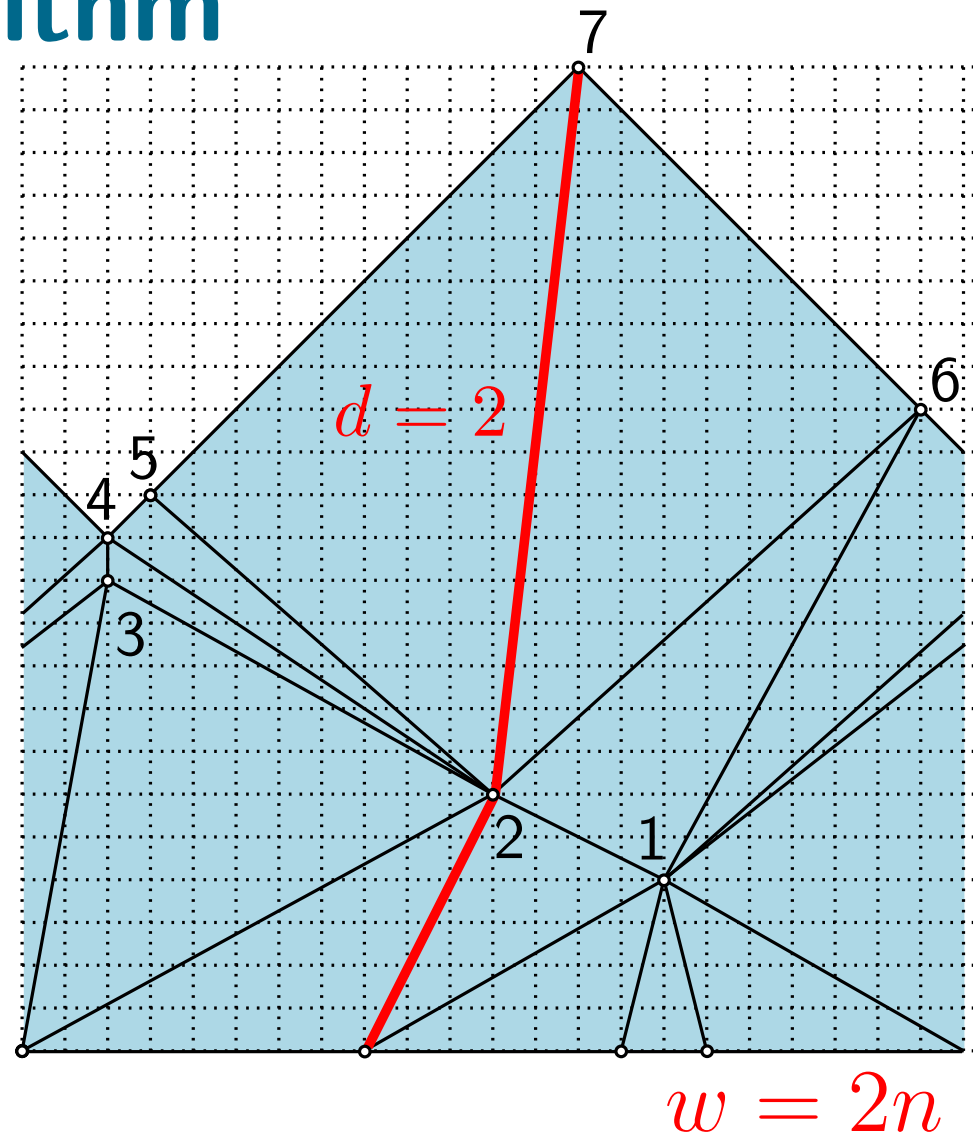
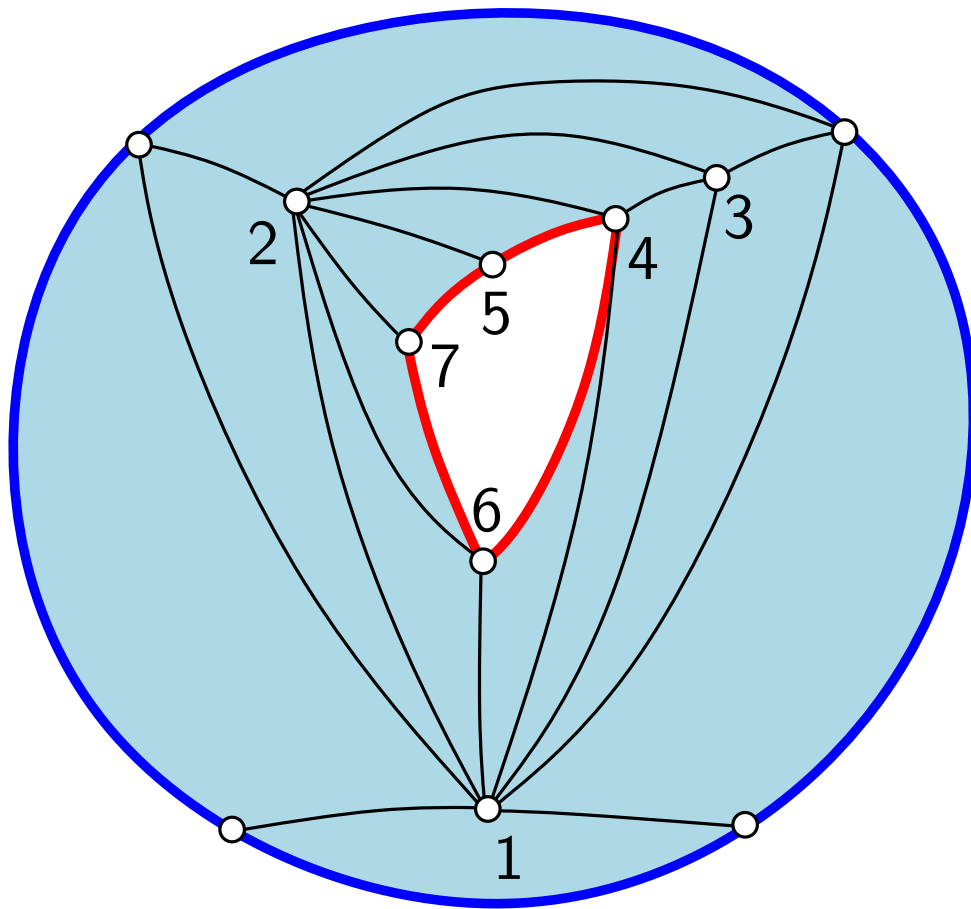
Extension to the cylinder: drawing algorithm



$$\text{Width} = 2n \quad \text{Height} \leq n(n - 3)/2$$

Can also deal with chordal edges incident to outermost cycle

Extension to the cylinder: drawing algorithm



Each edge has vertical extension at most w

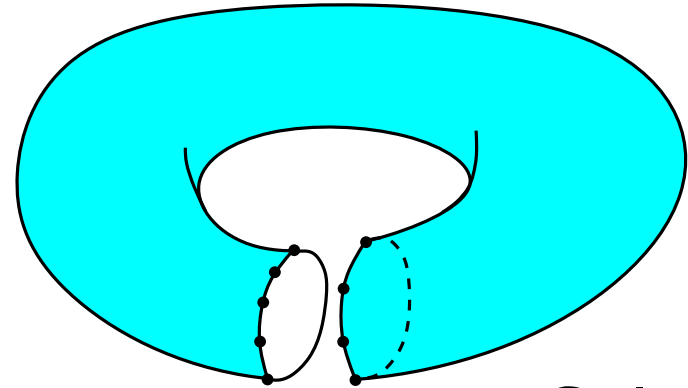
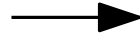
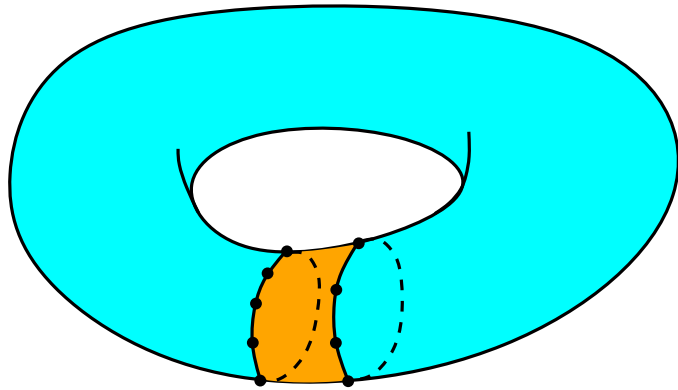
$$\Rightarrow h \leq n(2d + 1)$$

with d the graph-distance between the two boundaries

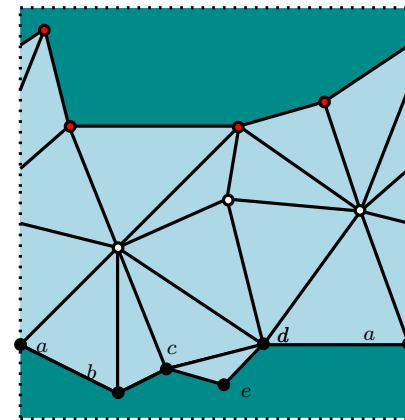
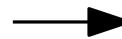
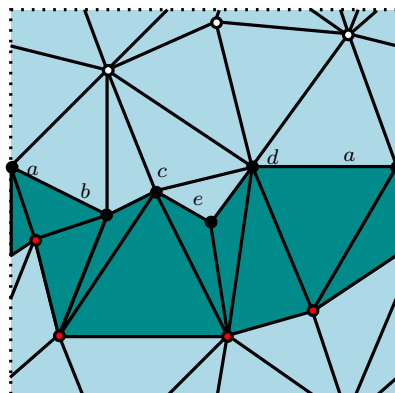
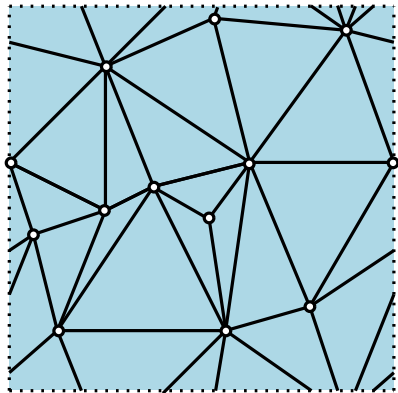
Getting toroidal drawings

Every toroidal triangulation admits a “tambourine”
[Bonichon, Gavoille, Labourel'06]

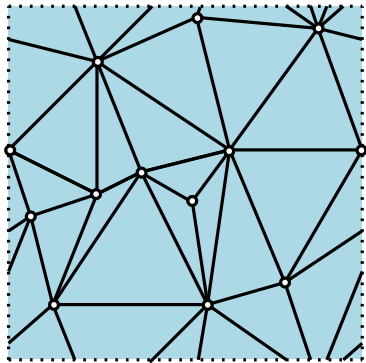
Torus



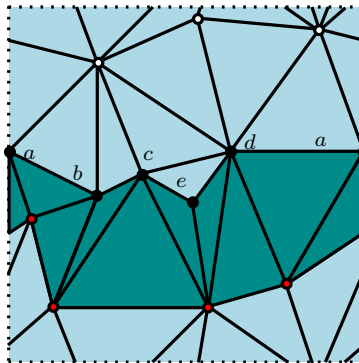
Cylinder



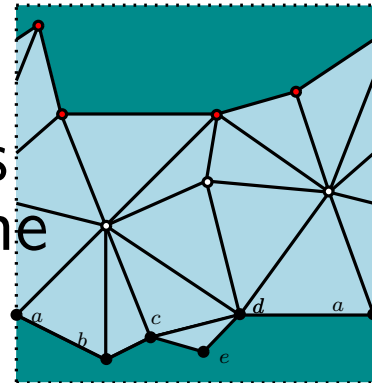
Getting toroidal drawings



compute
tambourine



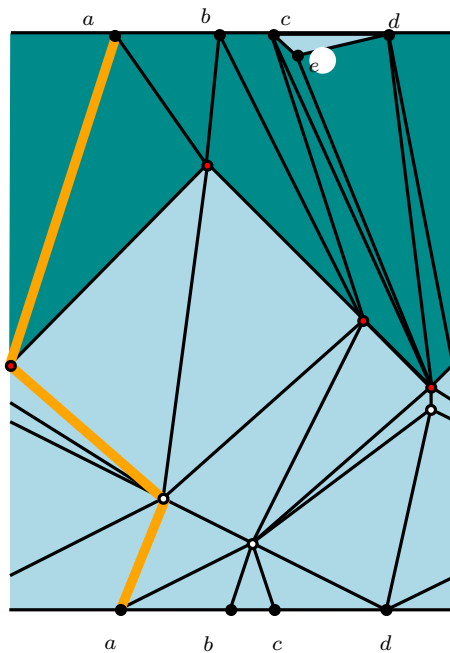
delete edges
in tambourine



Torus

Cylinder

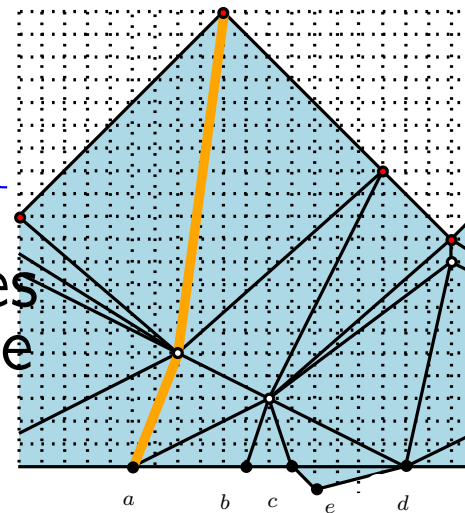
drawing algo.
on cylinder



$c=3$

$$\Delta h \leq 2n + 1$$

resinsert edges
in tambourine



$$w \leq 2n$$

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

$d=2$

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

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

Schnyder woods for toroidal graphs

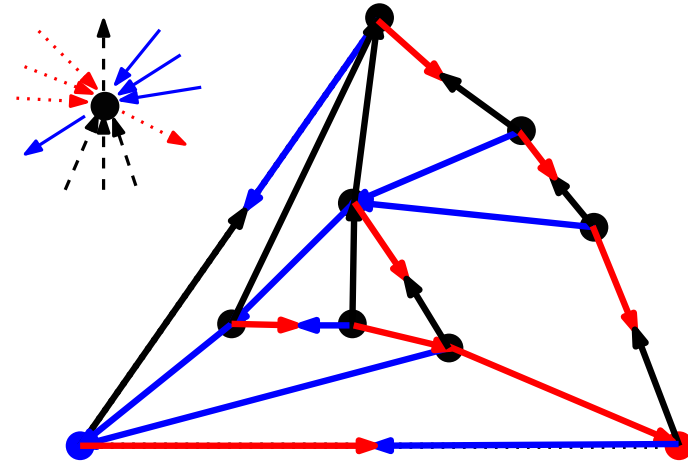
Toroidal Schnyder woods: definition

[Goncalves Lévêque, DCG'14]

$$g = 1 \quad e = 3n$$

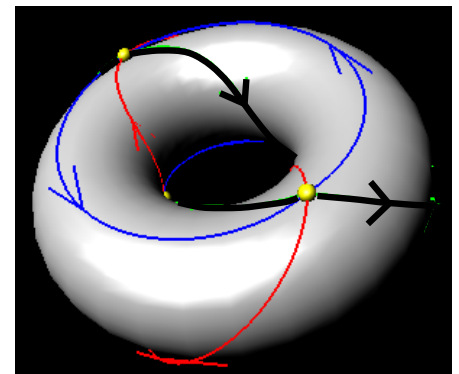
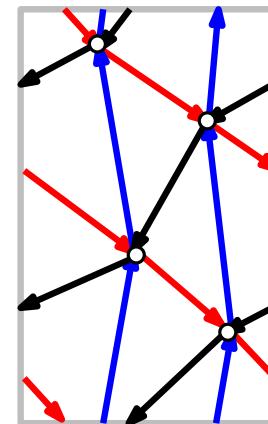
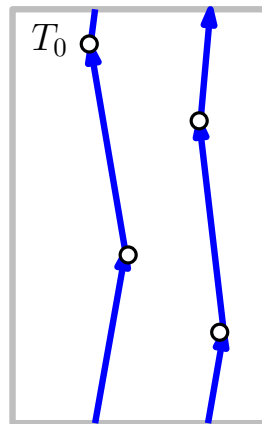
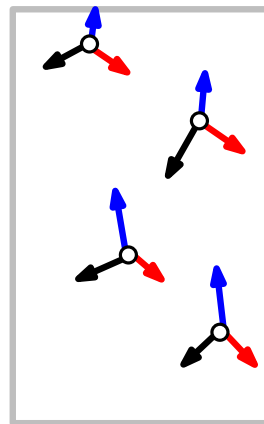
Planar Schnyder woods [Felsner 2001]

- Schnyder local rule (for half-edges)
- no monochromatic cycles



Toroidal Schnyder woods [Goncalves Lévêque, DCG'14]

- Schnyder local rule (for half-edges)
- every monochromatic cycle intersects at least one monochromatic cycle of each color

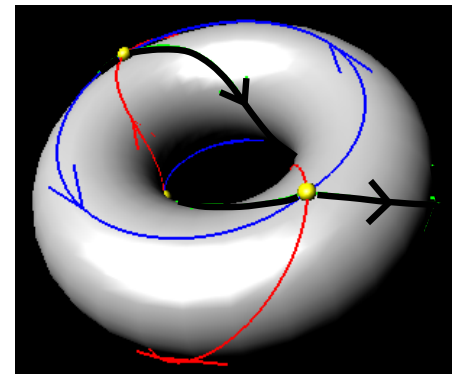
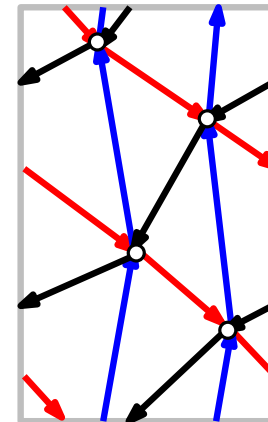
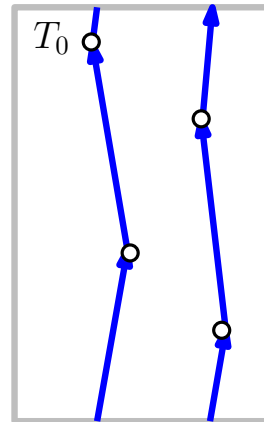
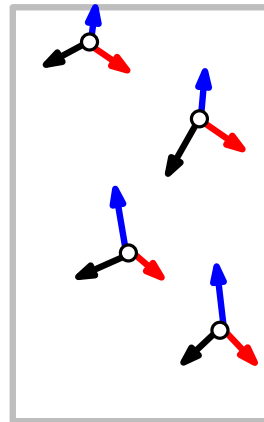
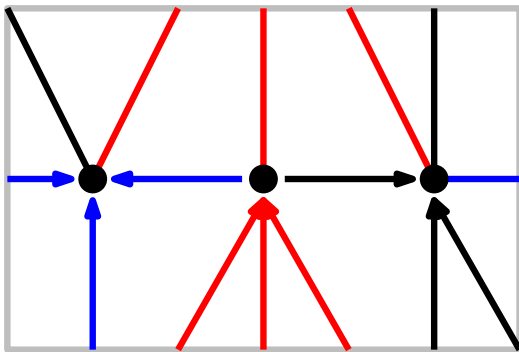


Toroidal Schnyder woods: existence

[Goncalves Lévêque, DCG'14]

$$g = 1 \quad e = 3n$$

no pair of intersecting monochromatic cycles

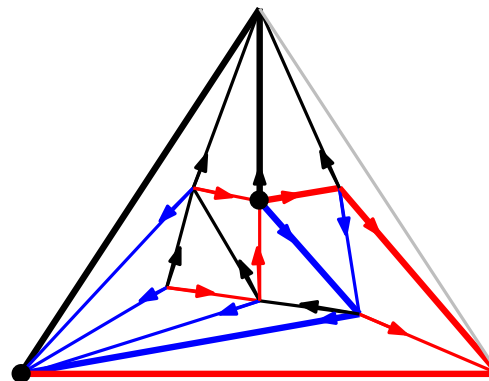
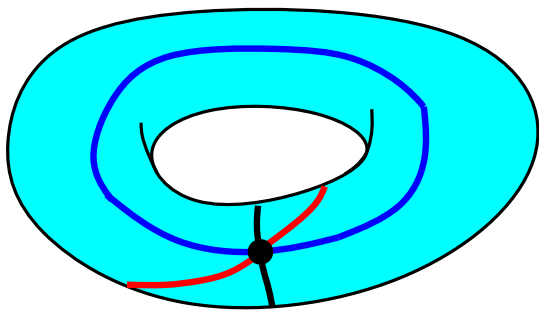
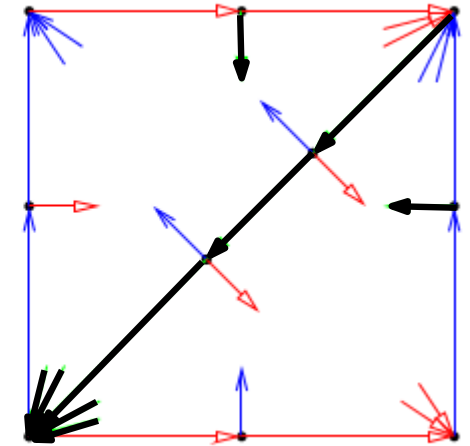
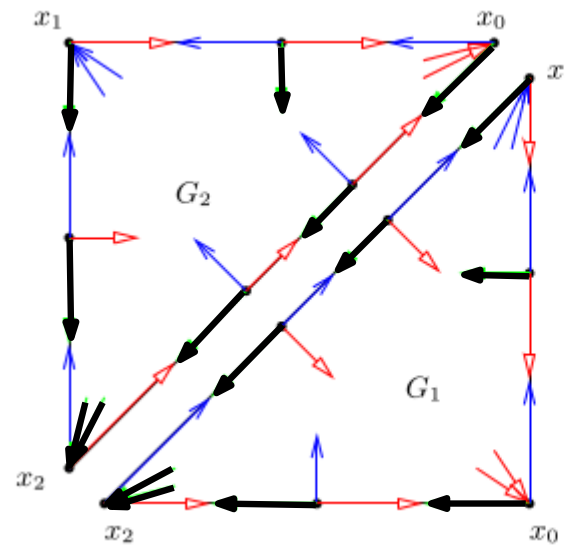
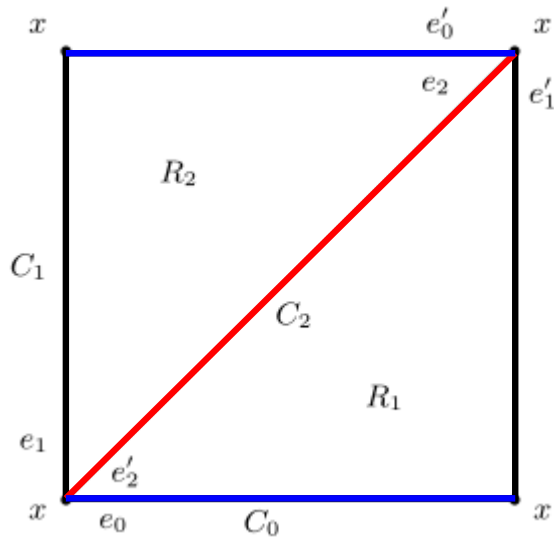


Toroidal Schnyder woods: existence

Thm[Fijavz]

(planar simple triangulations)

A simple toroidal triangulation contains three non-contractible and non-homotopic cycles that all intersect on one vertex and that are pairwise disjoint otherwise.



Toroidal Schnyder woods: drawing

Thm[Goncalves Lévêque]

(planar simple triangulations)

A simple toroidal triangulation admits a straight-line periodic drawing on a grid of size $O(n^2 \times n^2)$

