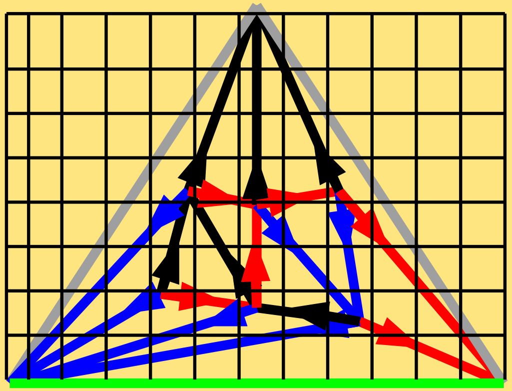
Schnyder woods for higher genus triangulated surfaces

20 mai 2008, TGGT'08, ENS, Paris



Luca Castelli Aleardi

CS Departement, ULB, Bruxelles

ULB UNIVERSITÉ LIBRE DE BRUXELLES

Eric Fusy

Simon Fraizer University, Vancouver

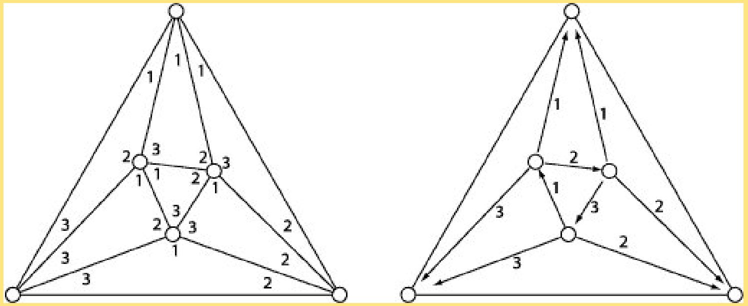
Thomas Lewiner

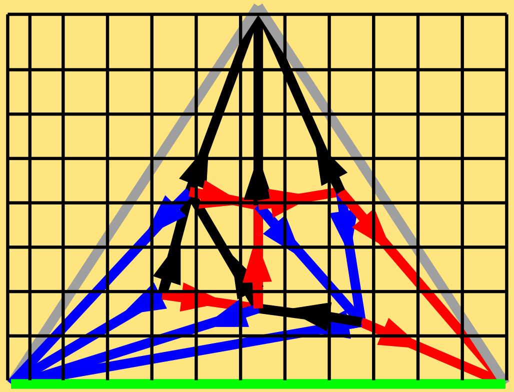
PUC University, Rio de Janeiro

Motivation and applications
Schnyder woods (or Schnyder trees or realizers)

edge orientation and coloration for triangulations

Combinatorics of maps

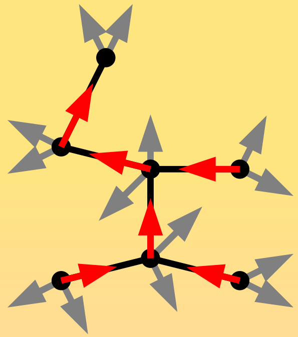




- enumeration problems

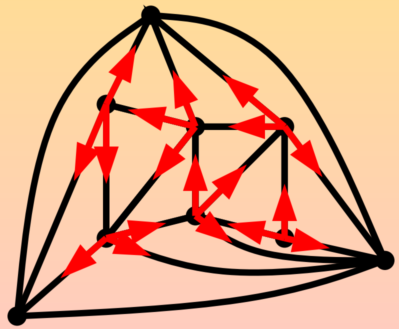
Graph drawing

- draw a planar graph, with vertices having integer coordinates (using as few as possible coordinates)



Compression and succinct encoding

- Reduce the amount of (memory) space used by the connectivity of a graph.
- Supporting efficient navigation, using small space
  Example: adjacency queries between vertices

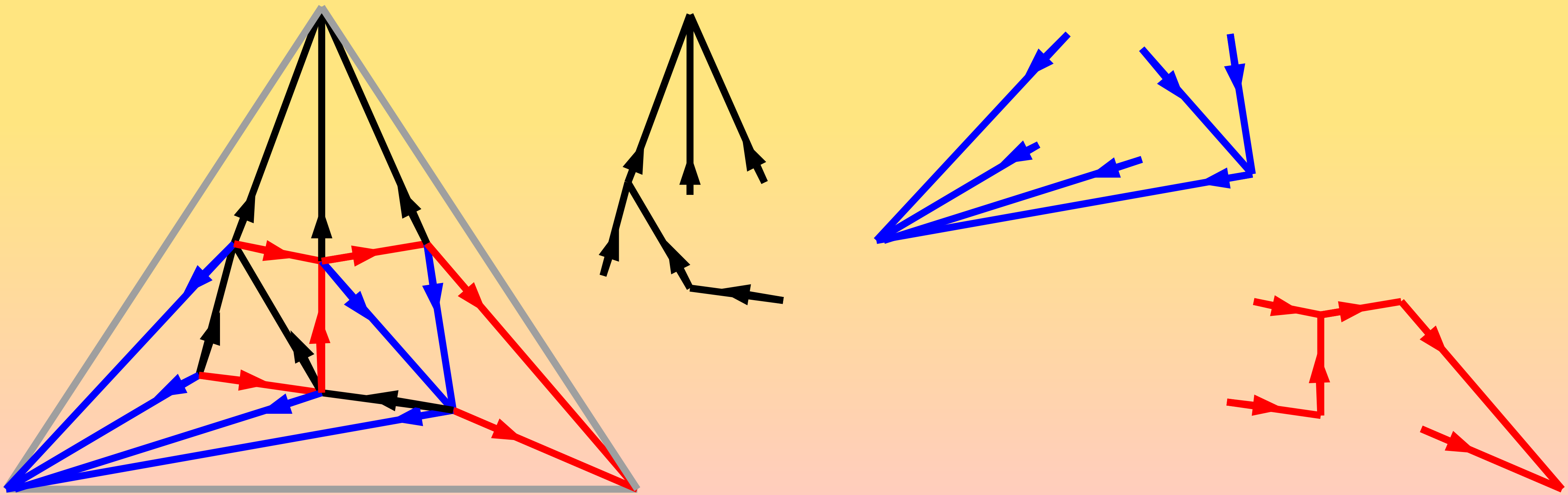


A nice characterization of planar graphs: Schnyder woods

Edge orientations, tree decompositions and dimension of a graph

Theorem (Schnyder, Felsner, Trotter)
A graph G is planar if and only if its dimension is at most 3

Theorem (Schnyder '90)
A graph G is planar if and only if the dimension of its incidence poset is at most 3

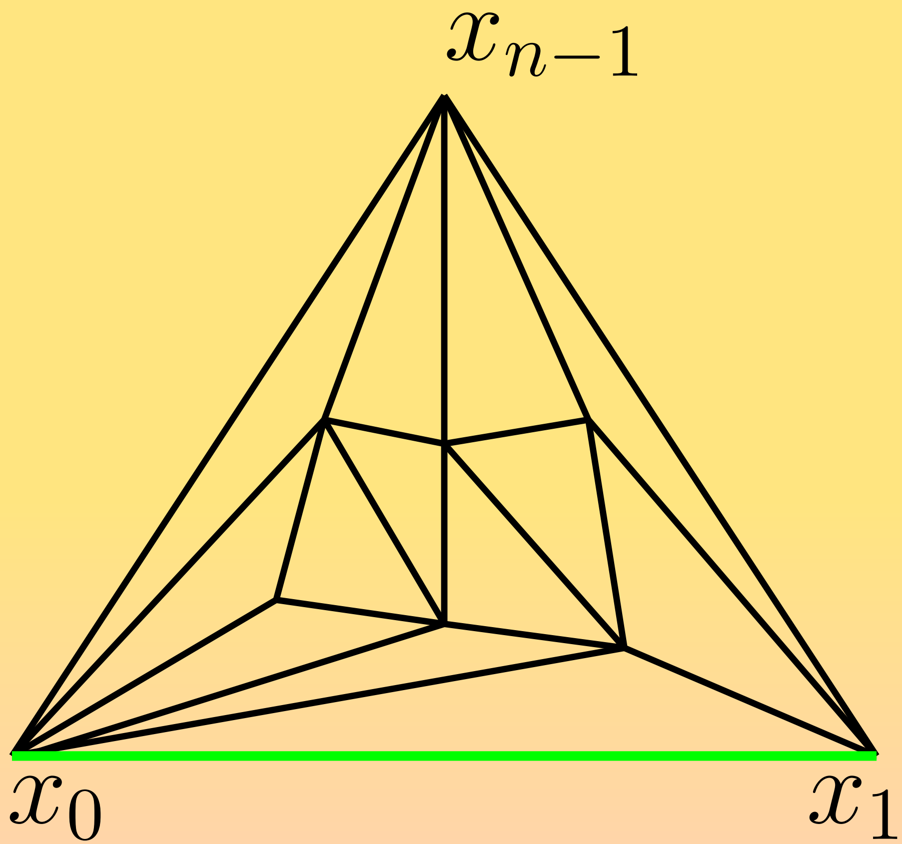


The definition in the planar (triangulated) case

A nice characterization of planar graphs

(Schnyder '90)

Let T be a triangulation having outer face $\{x_0, x_1, x_{n-1}\}$.
with n nodes

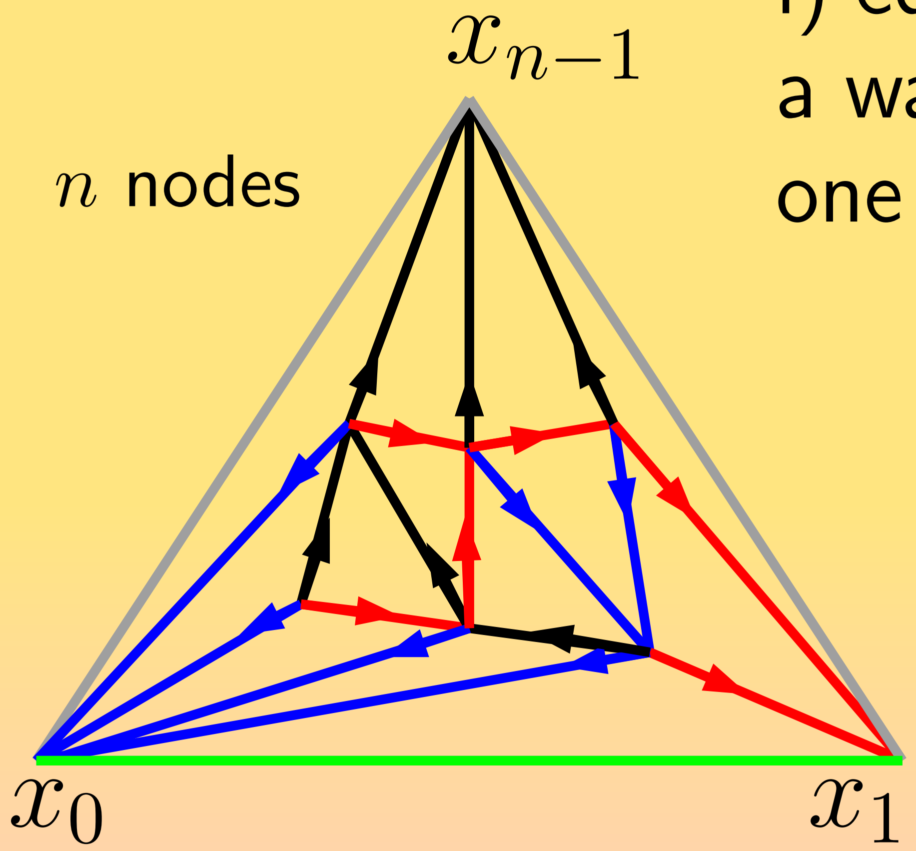


A nice characterization of planar graphs

(Schnyder '90)

A Schnyder wood of a triangulation is

a partition T_0, T_1, T_2 of the internal edges of T s.t. :



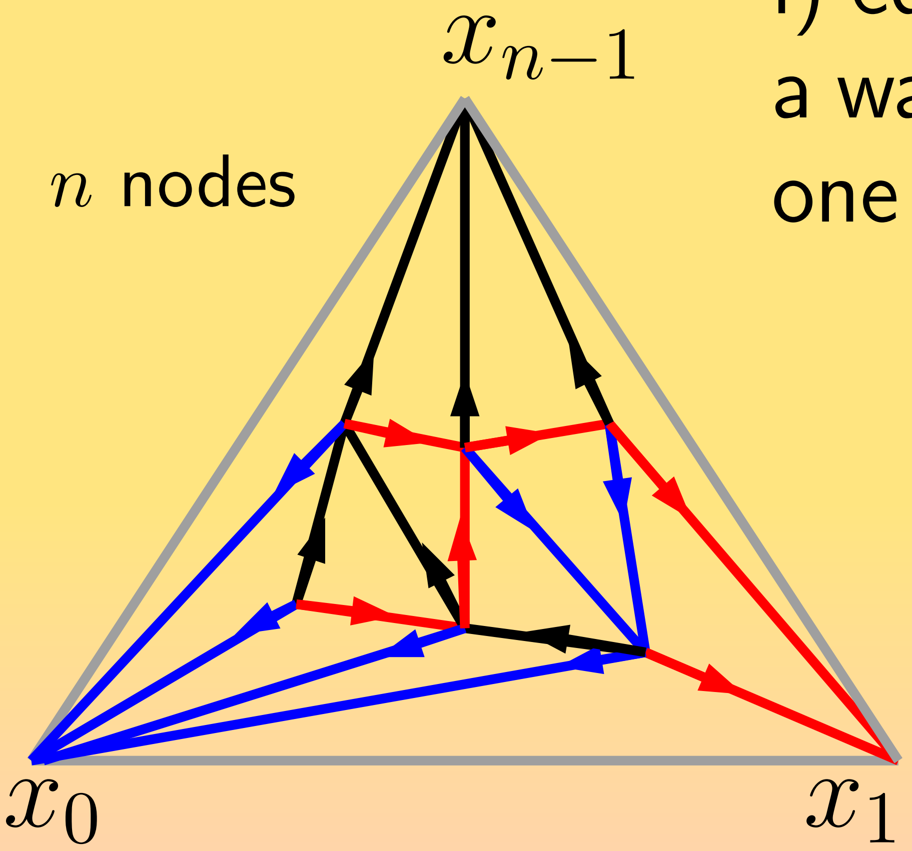
i) edge are colored and oriented in such a way that each inner node has exactly one outgoing edge of each color

A nice characterization of planar graphs

(Schnyder '90)

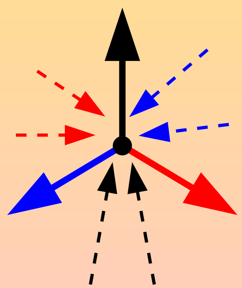
A partition T_0, T_1, T_2 of the internal edges of T s.t. :

n nodes



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

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

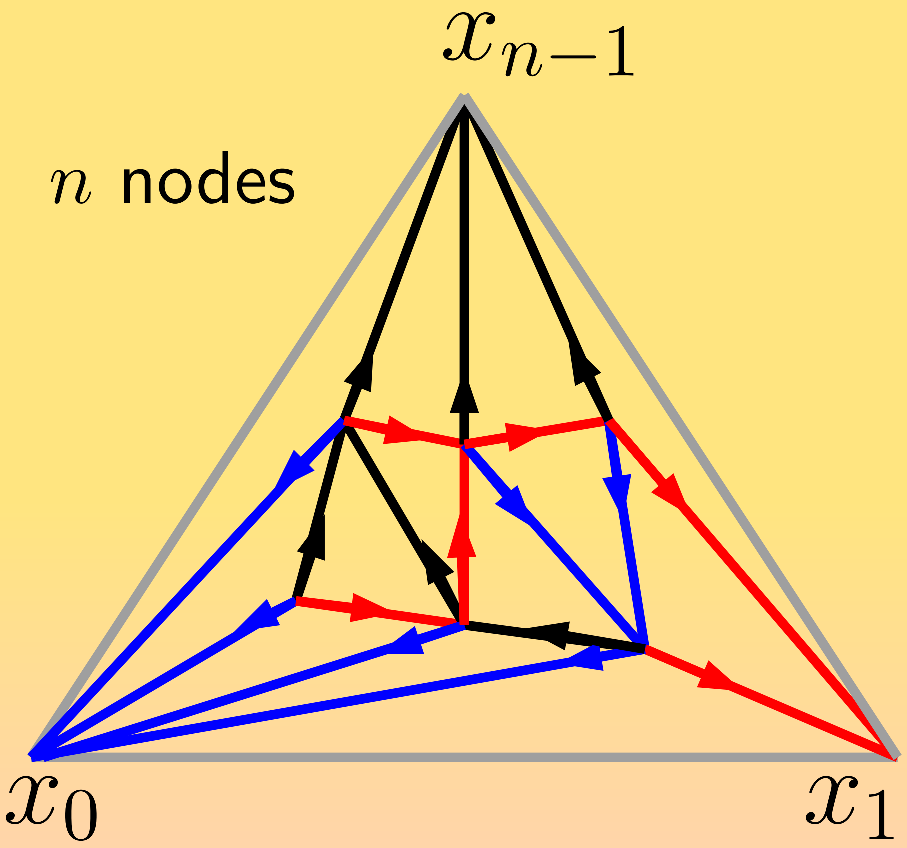


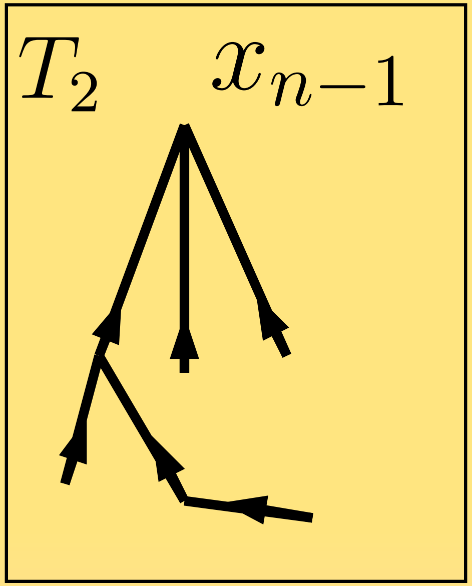
Important facts about Schnyder woods

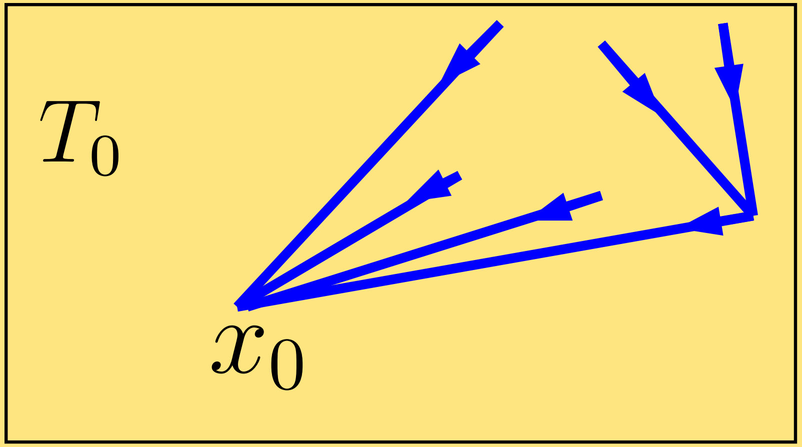
A first fundamental fact: 3 tree decomposition

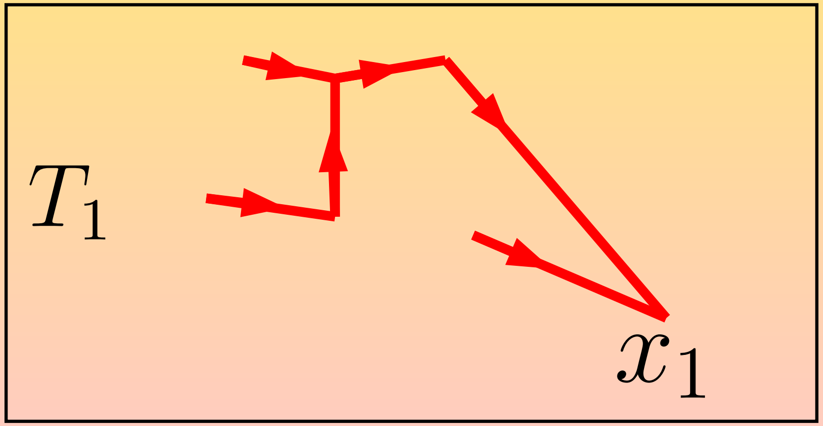
(Schnyder '90)

T_0, T_1, T_2 are spanning trees of (the inner nodes of) T:



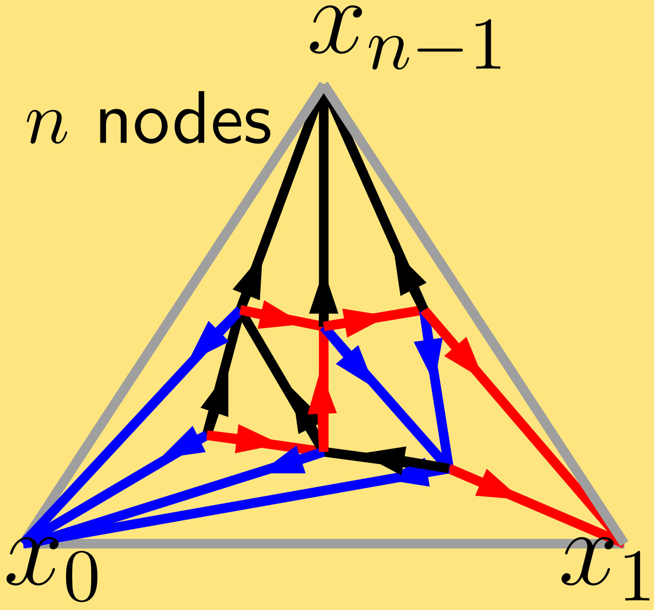


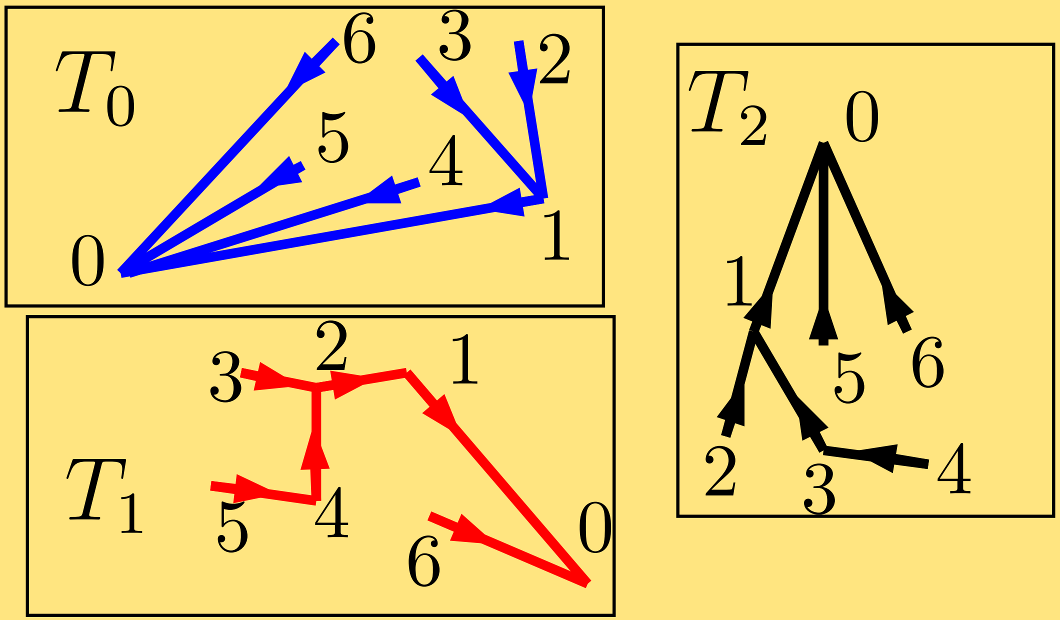




Second fact: dimension of a graph

L_0, L_1, L_2 are three orders on the vertices of T:





$$\boxed{\begin{array}{l} L_0 : v_1 < v_2 < v_3 < v_4 < v_5 < v_6 \\ L_1 : v_2 < v_3 < v_6 < v_4 < v_5 < v_1 \\ L_2 : v_2 < v_3 < v_6 < v_1 < v_3 < v_2 \end{array}}$$

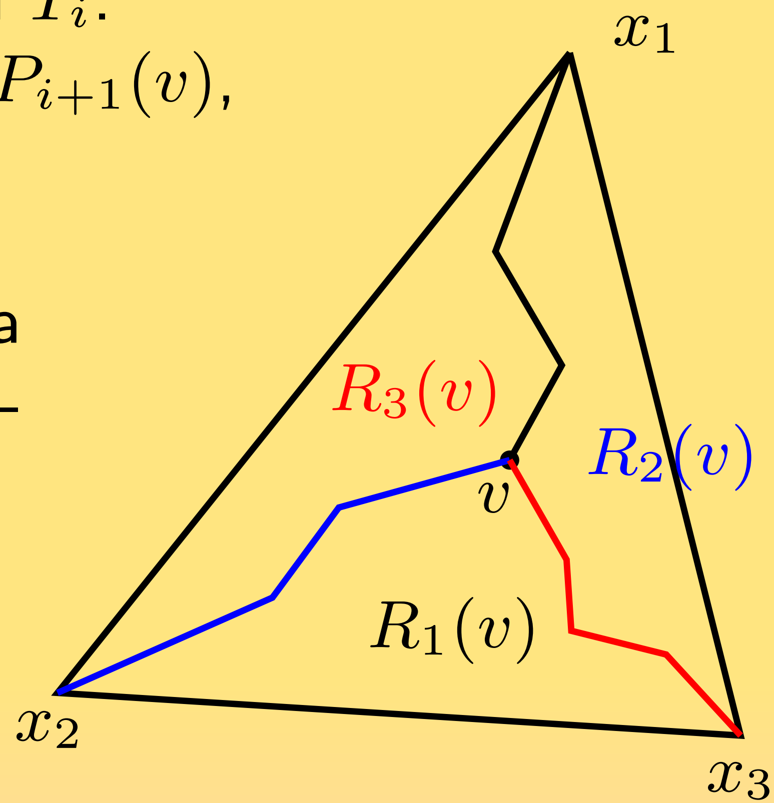
The first motivation: barycentric drawing
Combinatorial interpretation

How to use Schnyder woods:

- Let $P_i(v)$ be the path from v to x_i in T_i.
- Let$R_i(v)$ be the region defined by $P_{i+1}(v)$, $P_{i+2}(v)$ and (x_{i+1}, x_{i+2}).

The combinatorial equivalent of the area is given by the number of triangles enclosed in each region R_i:

$$v_i = \frac{|R_i(v)|}{|T|}$$

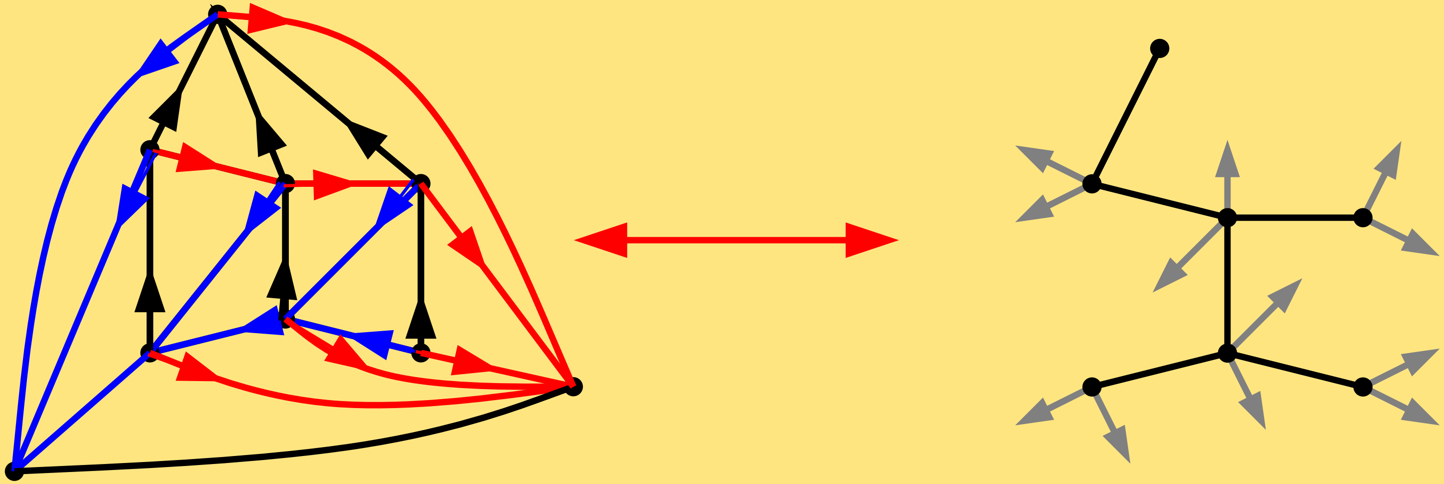


Theorem

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 $x_3 = (0, 2n-5)$.

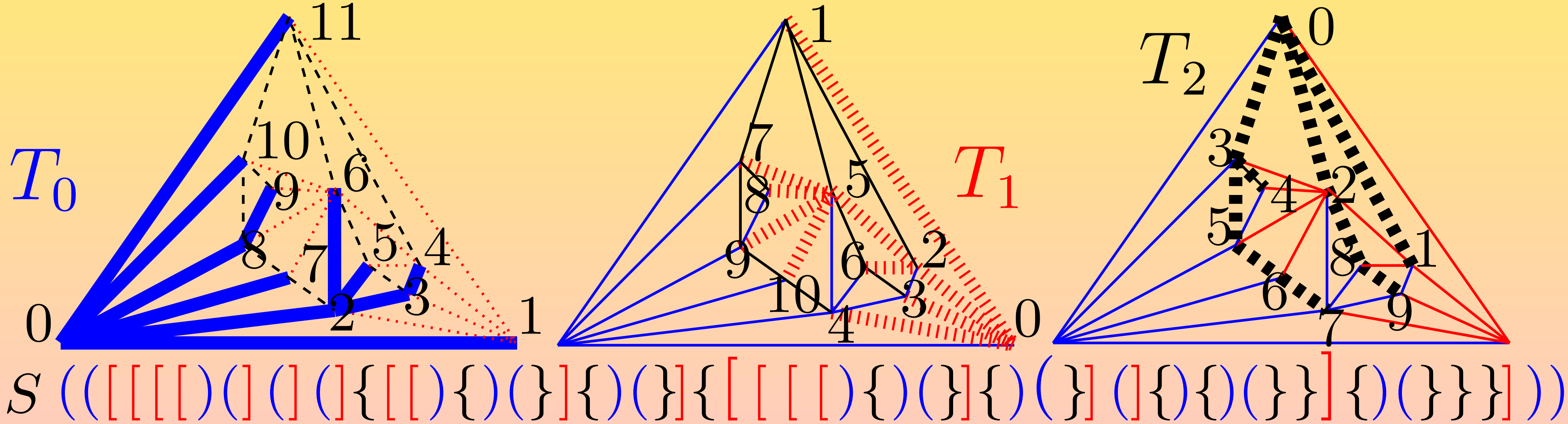
Application: graph compression and succinct encoding

Optimal compression (Poulalhon-Schaeffer, Icalp '03)



Succinct encoding (Chuang-Garg-He-Kao-Lu Icalp '98)

(Barbay-Castelli Aleardi-He-Munro Isaac'07)



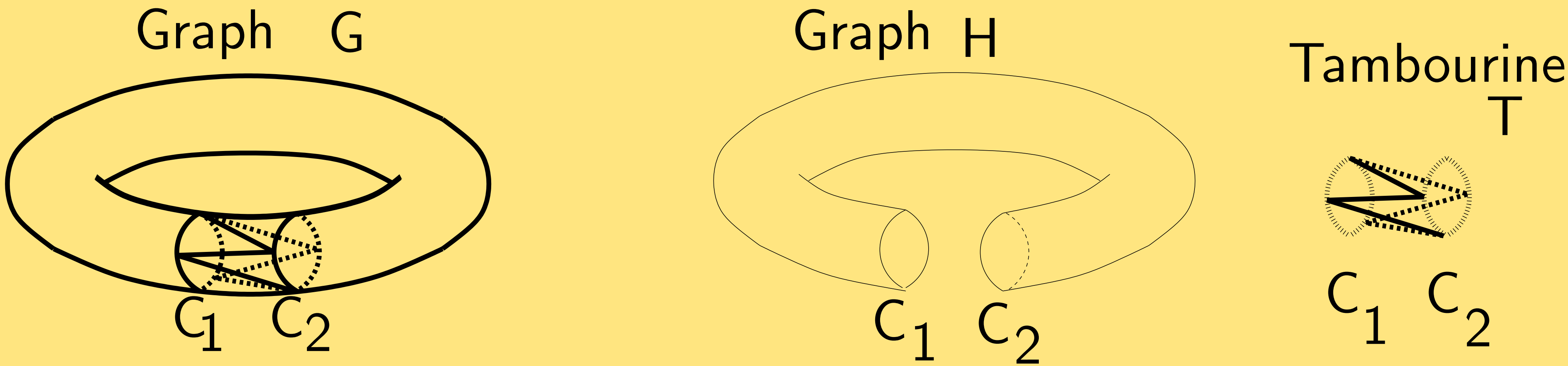
Tree decompositions in higher genus

Related works: tree decompositions of toroidal graphs the "tambourine" solution

(Bonichon, Gavoille, Labourel, ICGT '05)

Main idea of this approach:

Compute a pair of adjacent non contractible cycles



Result: T_0, T_1, T_2 vertex spanning trees

Inconvenients:

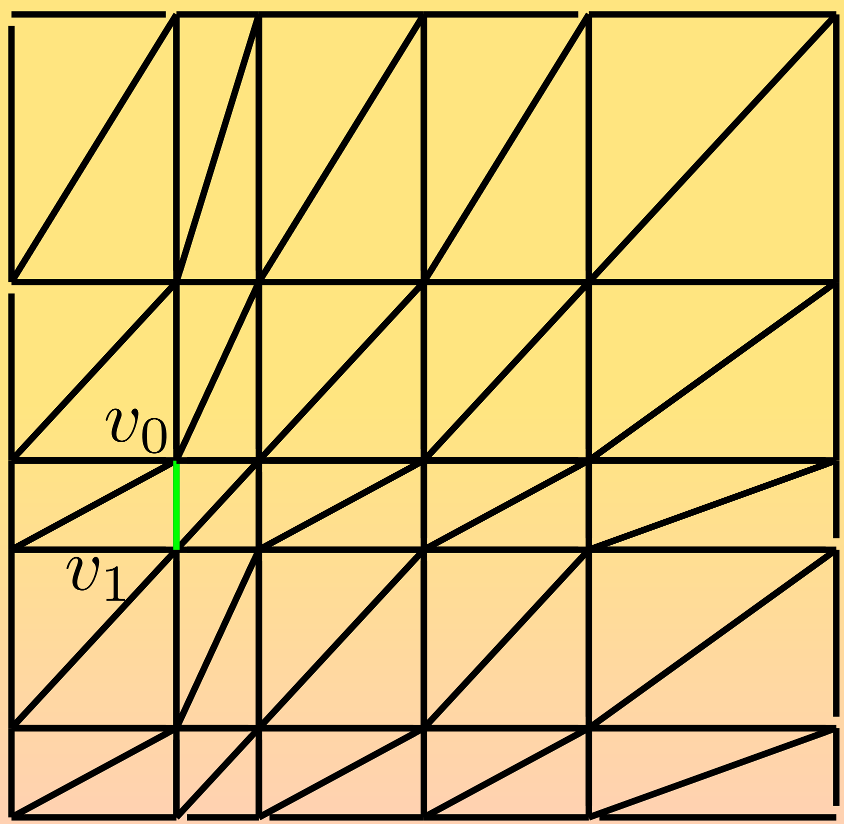
- valid only for toroidal triangulation (genus 1)
- potentially large number of vertices not satisfying the local condition

Our main contribution

a generalized higher genus definition

Schnyder Woods: the higher genus case

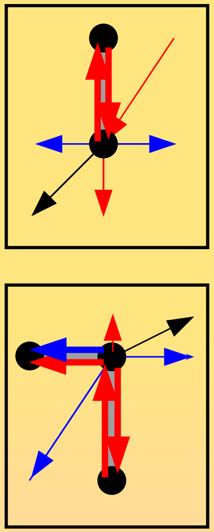
Given a rooted triangulated surface of genus g



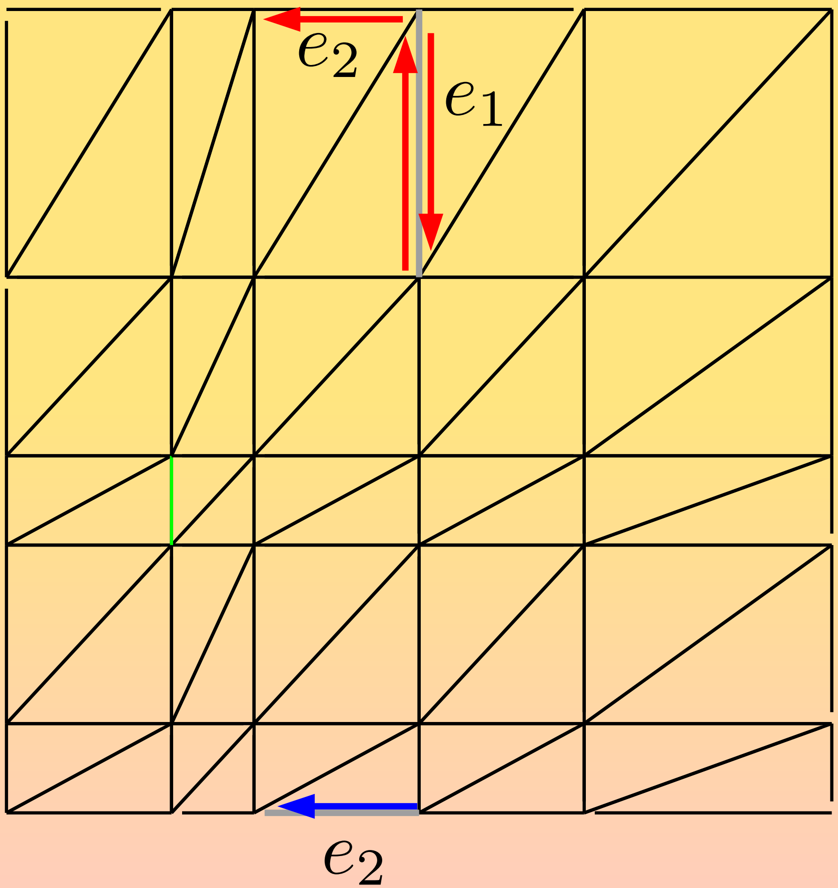
Schnyder Woods: the higher genus case

i) a small set $\mathcal{E}^s$ of *special* edges, doubly oriented and colored

$$|\mathcal{E}^s| = 2g$$



$\mathcal{E}^s = \{e_1, e_2\}$



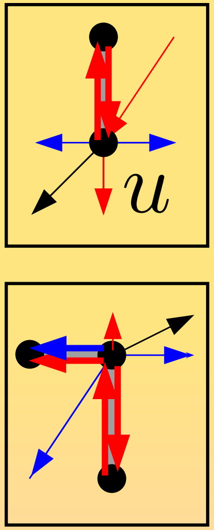
Schnyder Woods: the higher genus case

i) a small set $\mathcal{E}^s$ of *special* edges, doubly oriented and colored

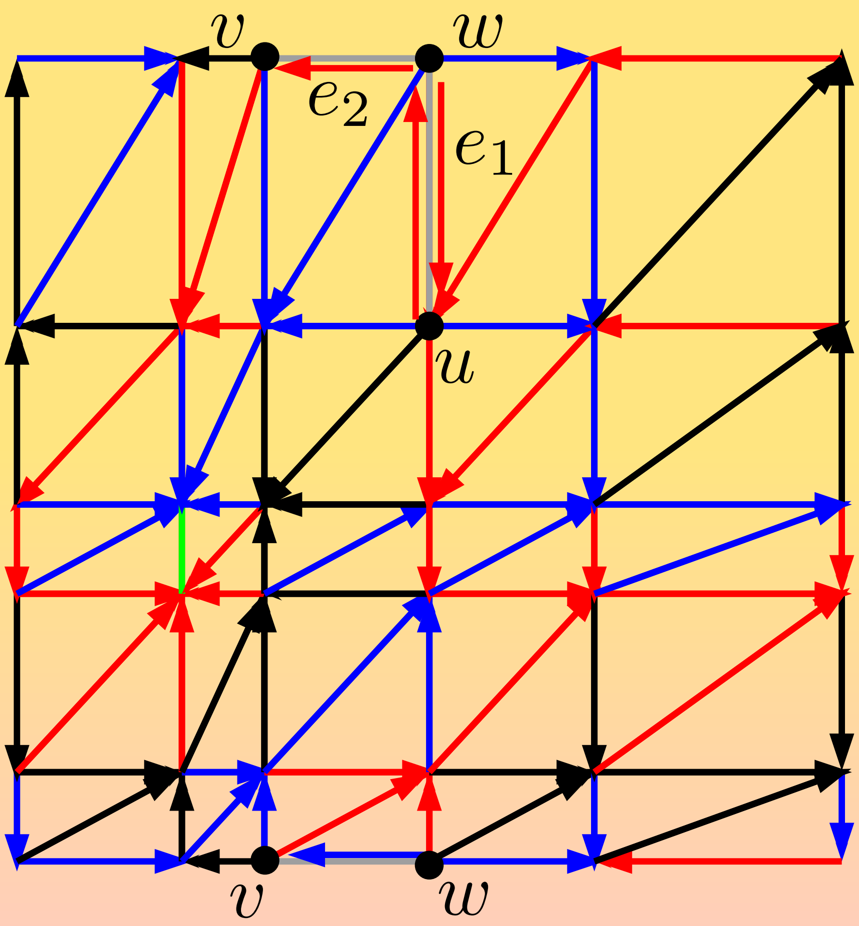
(u, v, w)

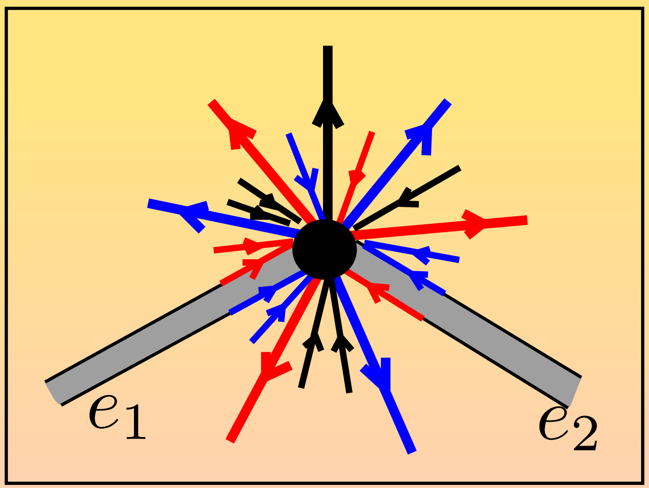
at most $2 \cdot 2g$ multiple vertices (incident to special edges)

ii) a new local condition for edges in a sector incident to a multiple vertex



$\mathcal{E}^s = \{e_1, e_2\}$

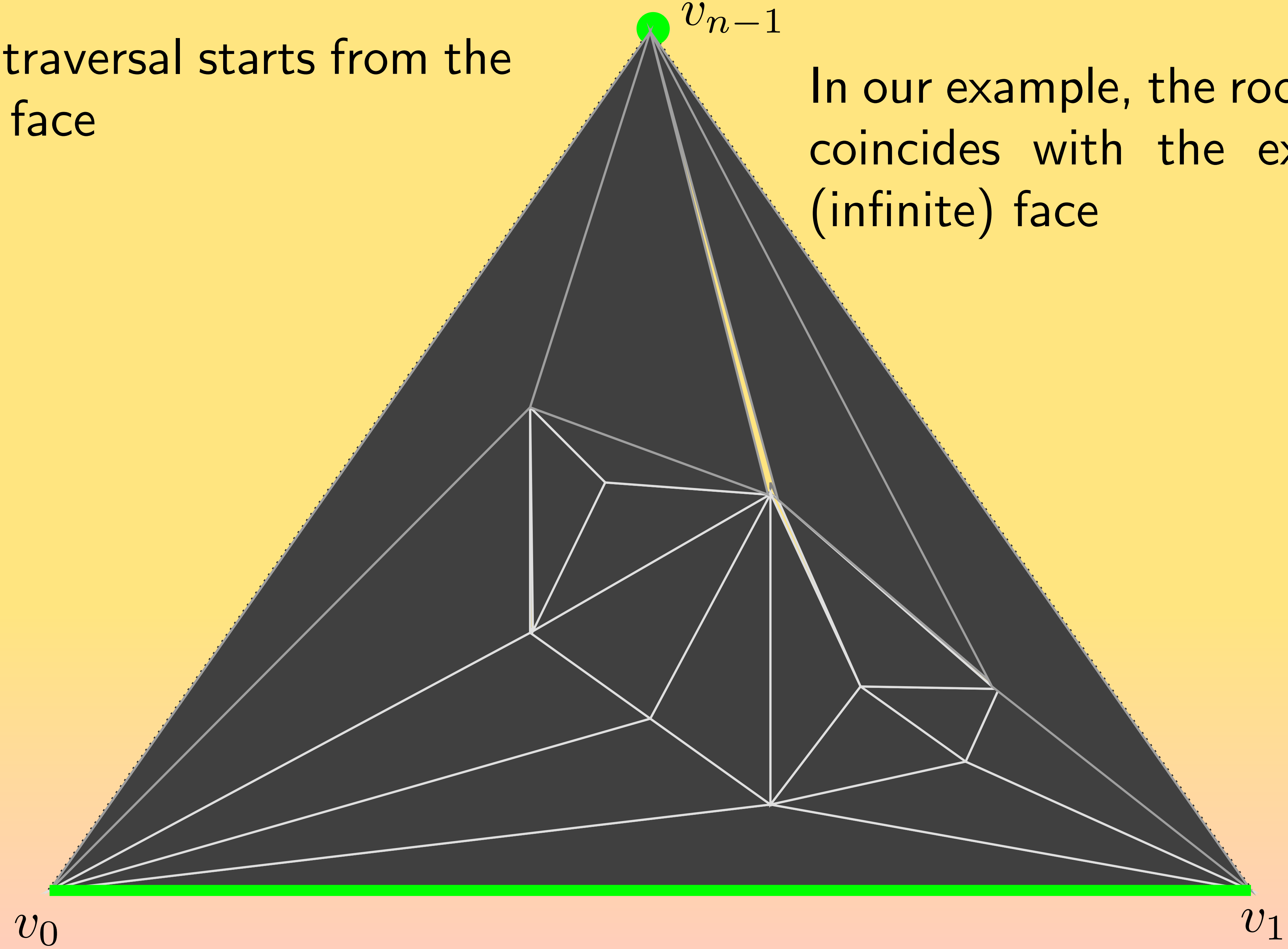




Computing Schnyder Woods (in the plane)

Incremental vertex conquest (Brehm's approach)

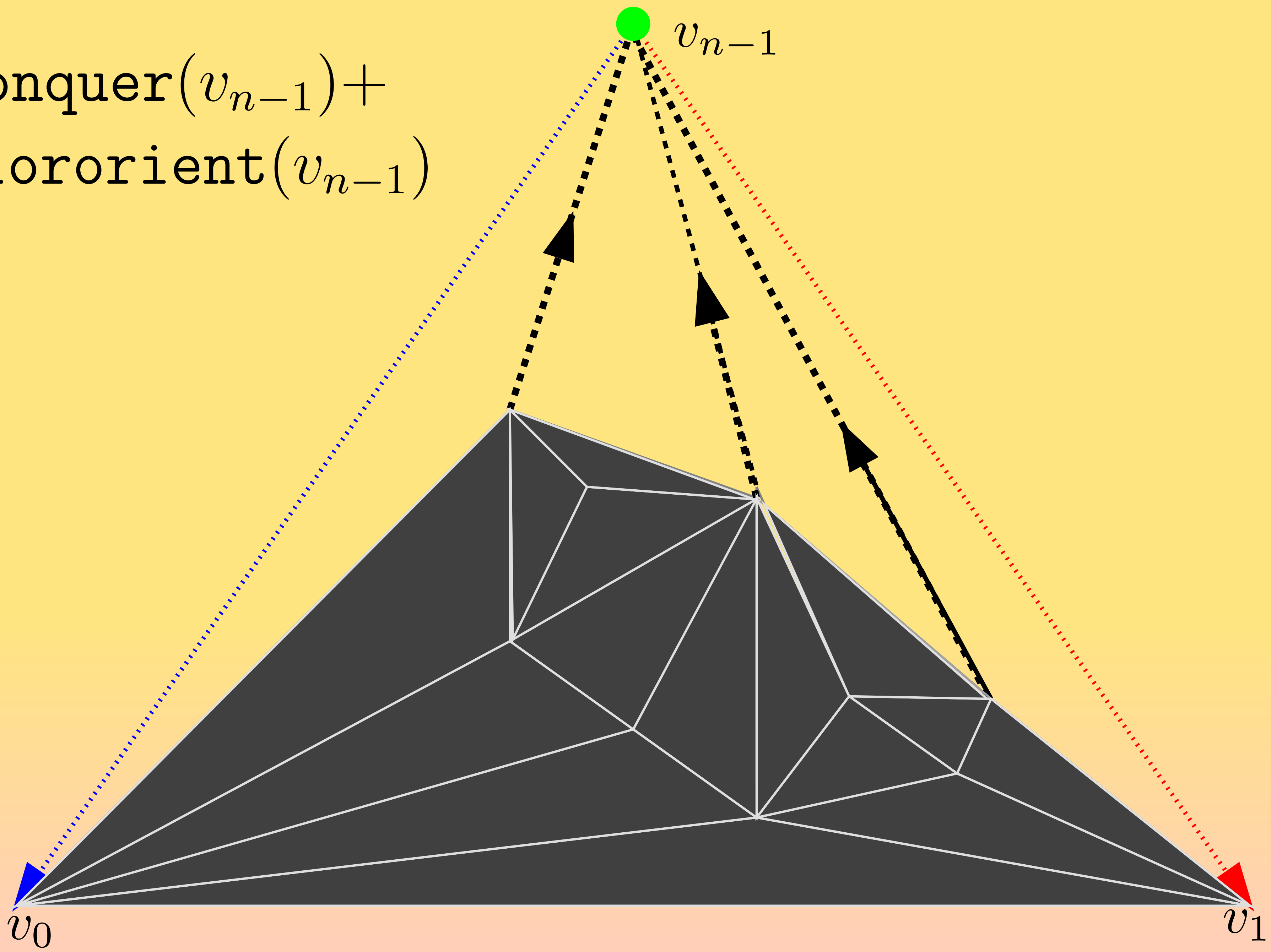
The traversal starts from the root face



In our example, the root face coincides with the exterior (infinite) face

Incremental vertex conquest

$\texttt{conquer}(v_{n-1})+$
$\texttt{colororient}(v_{n-1})$

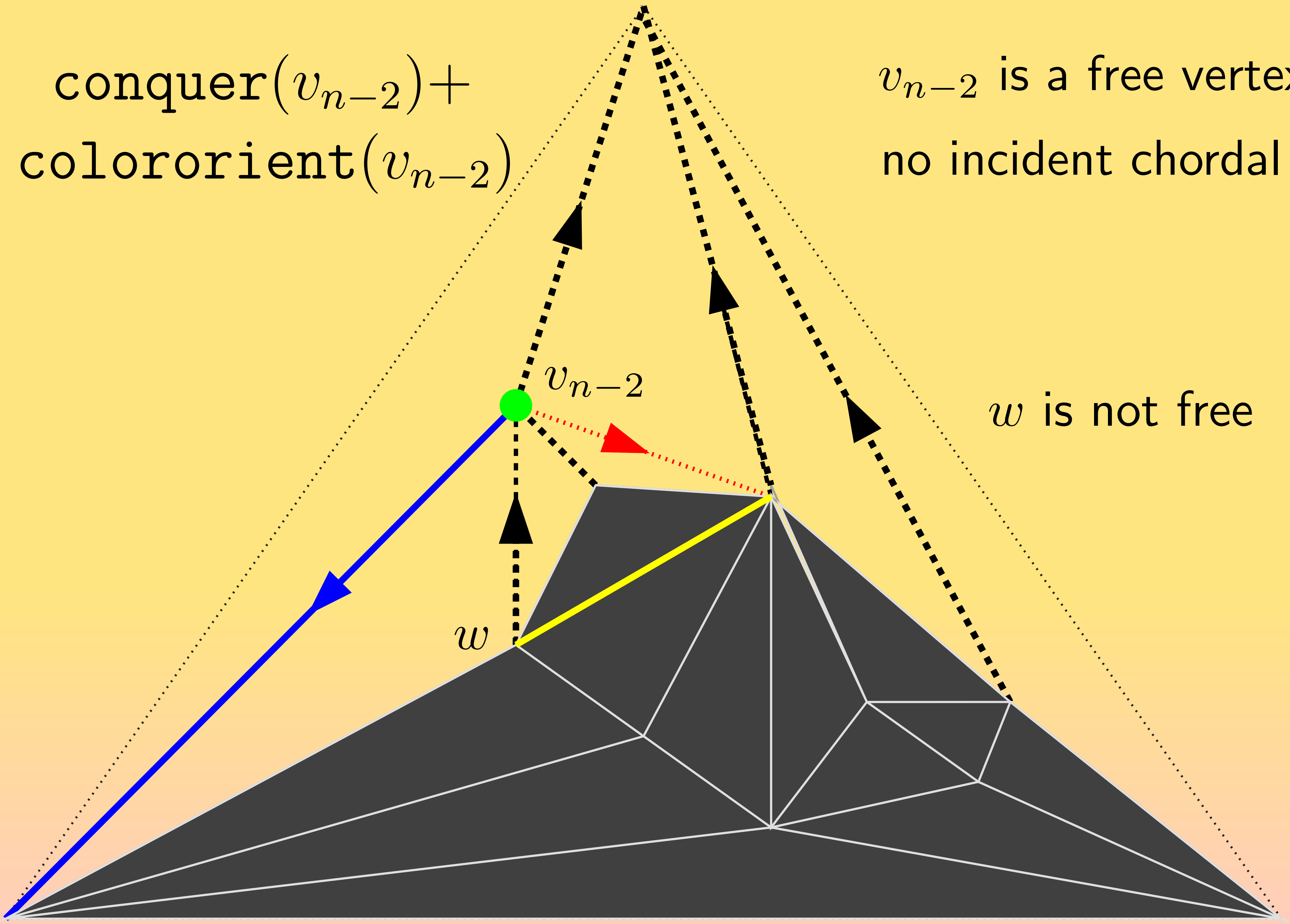


Incremental vertex conquest

$\texttt{conquer}(v_{n-2})+$
$\texttt{colororient}(v_{n-2})$

v_{n-2} is a free vertex

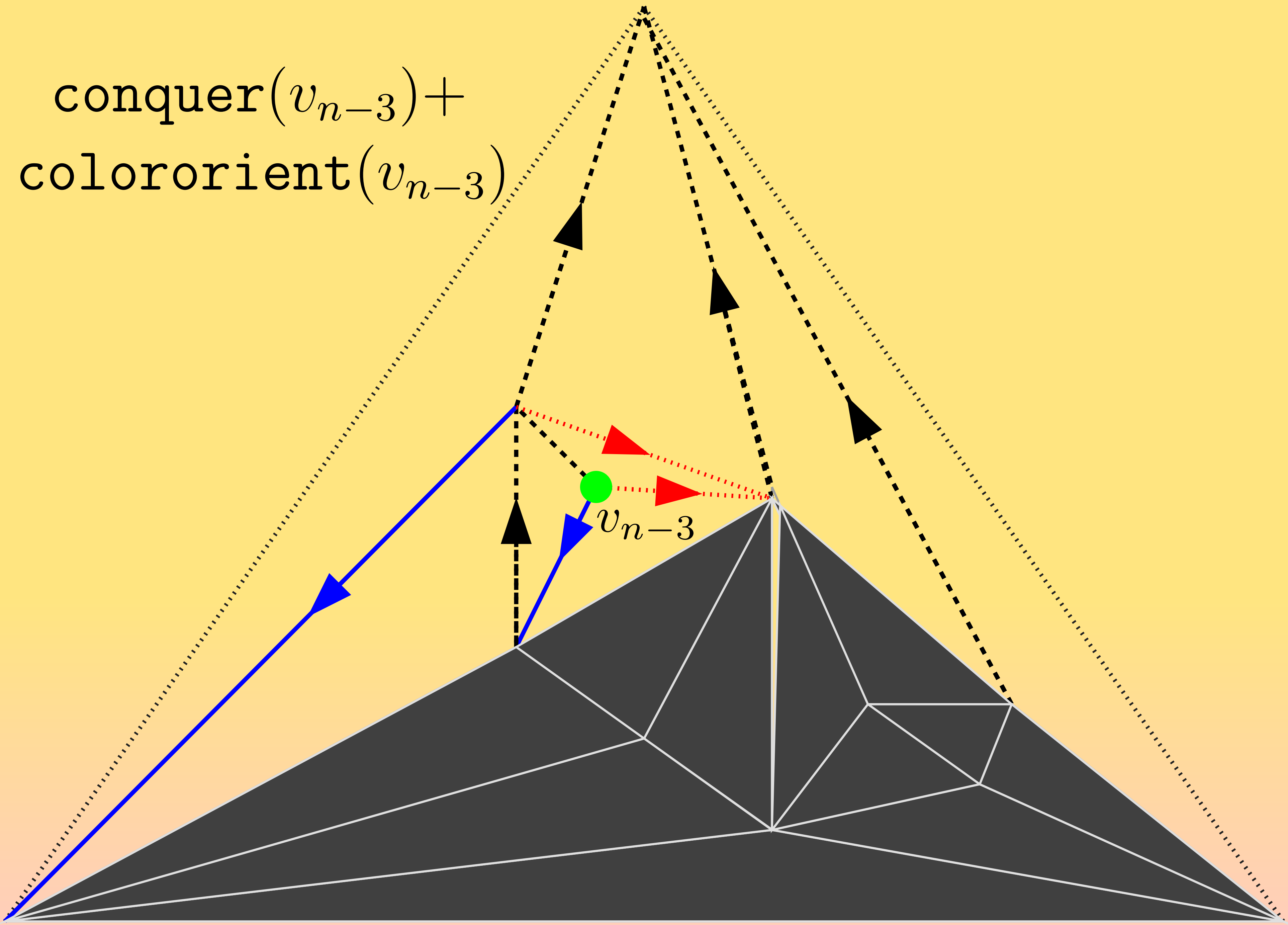
no incident chordal edges



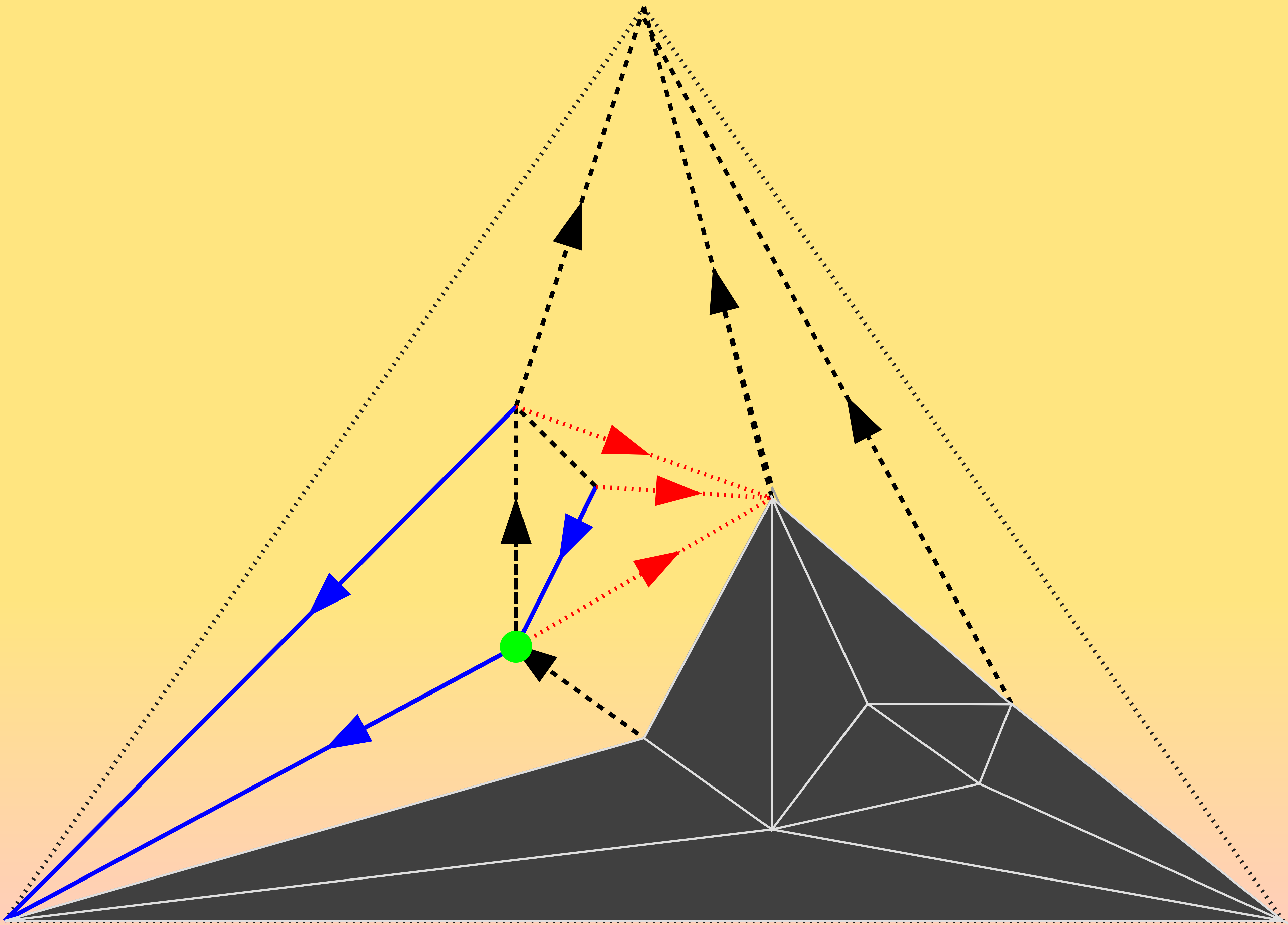
w is not free

Incremental vertex conquest

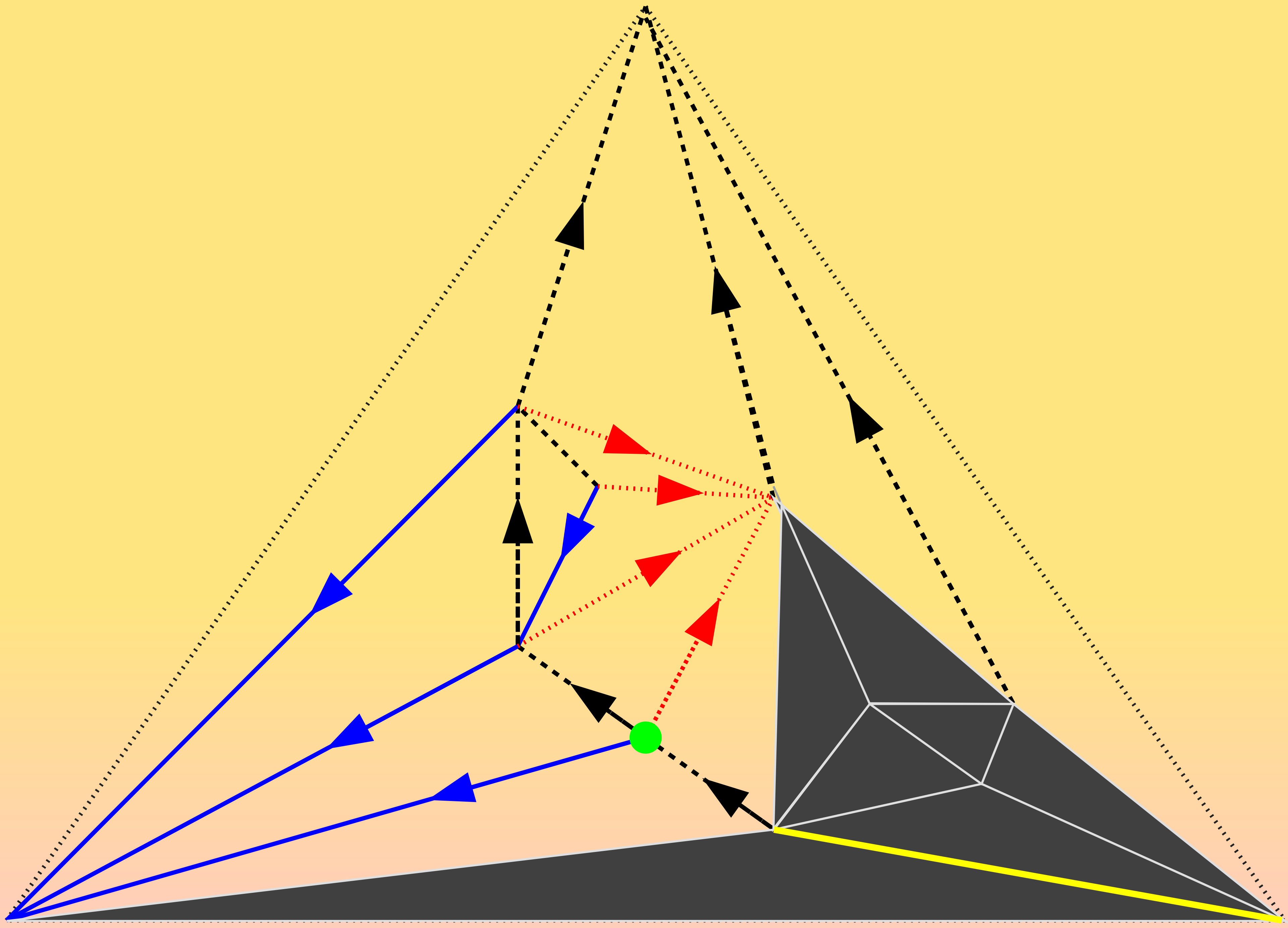
$\texttt{conquer}(v_{n-3})+$
$\texttt{cololorient}(v_{n-3})$



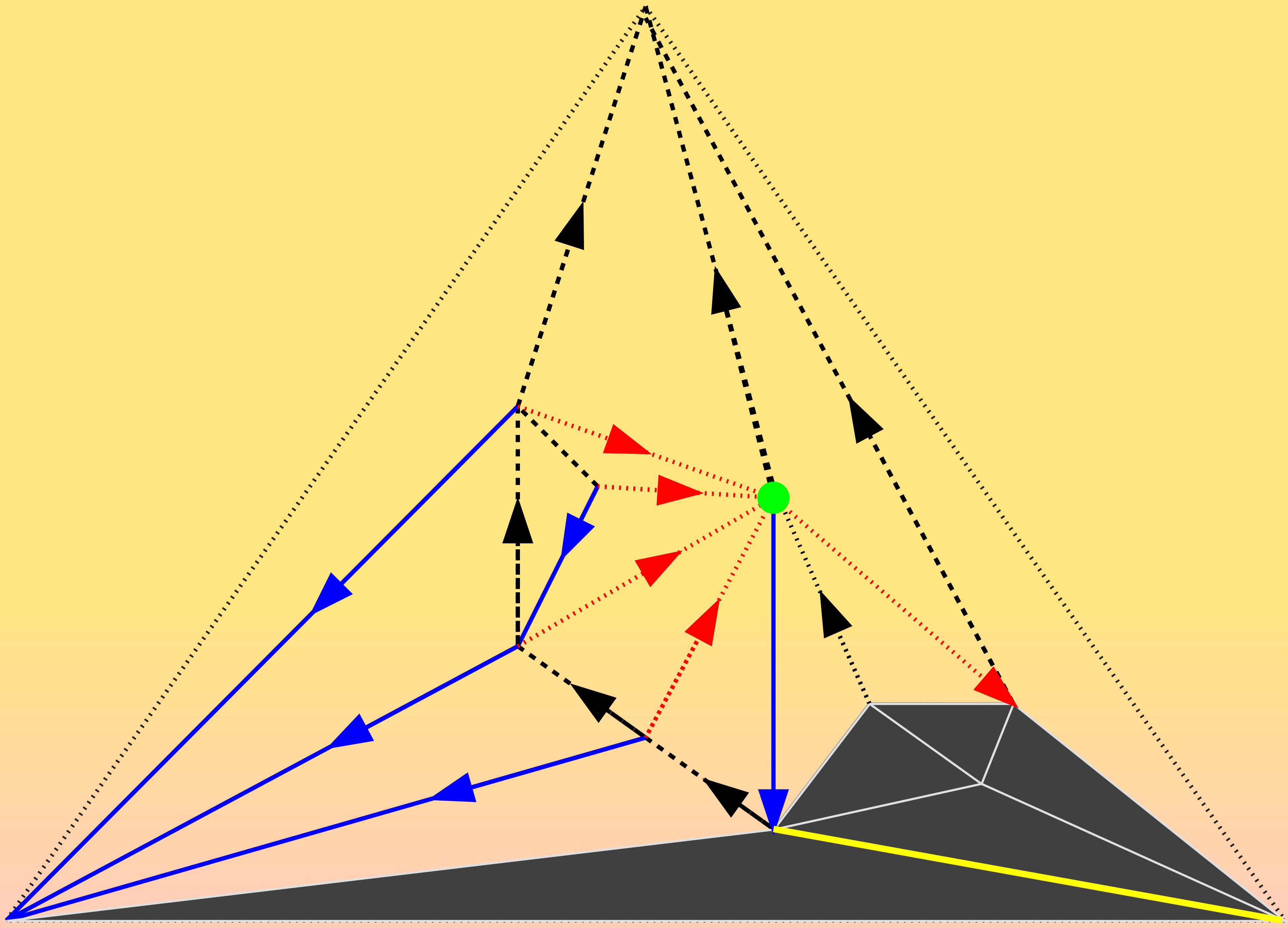
Incremental vertex conquest



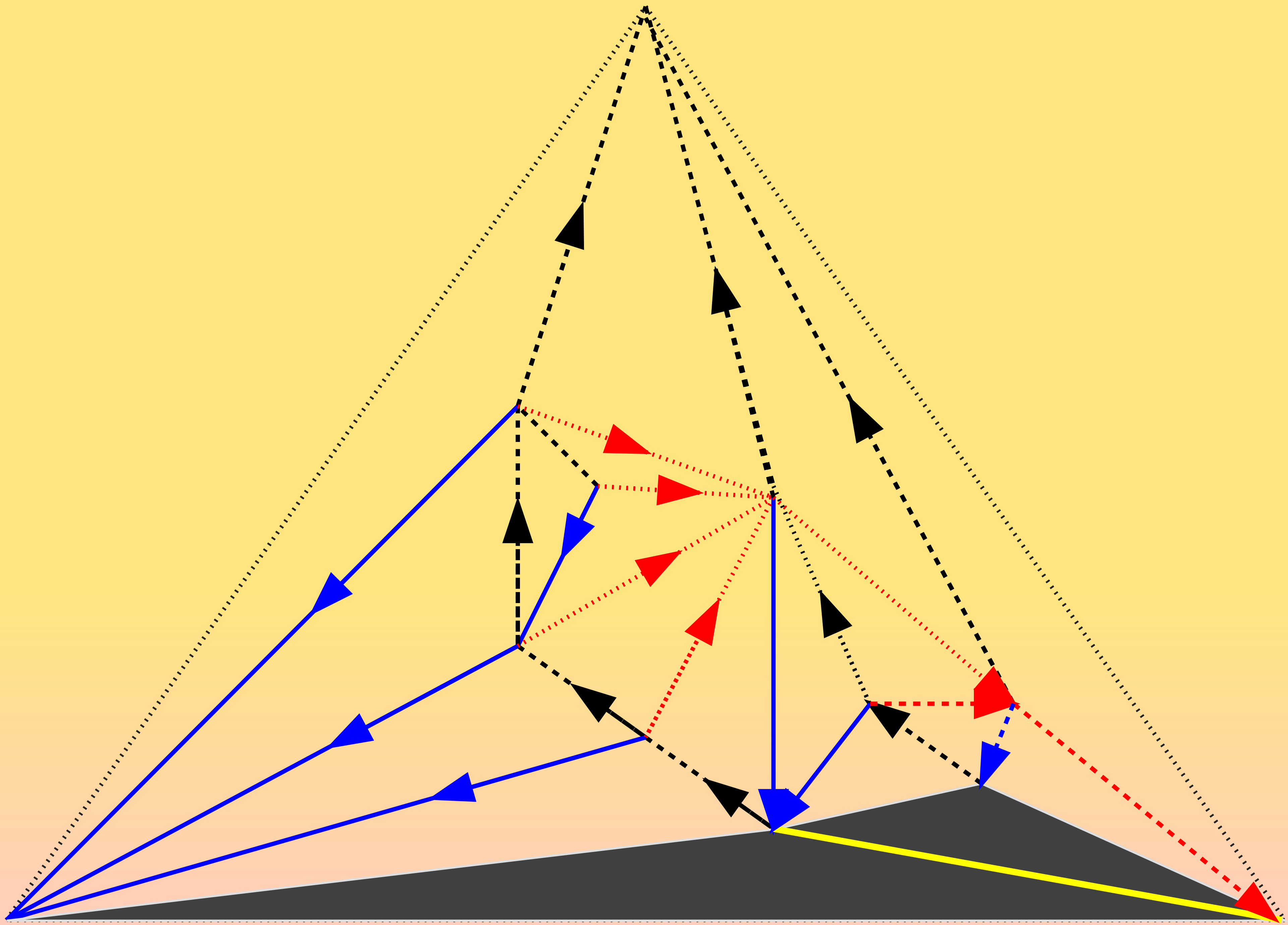
Incremental vertex conquest



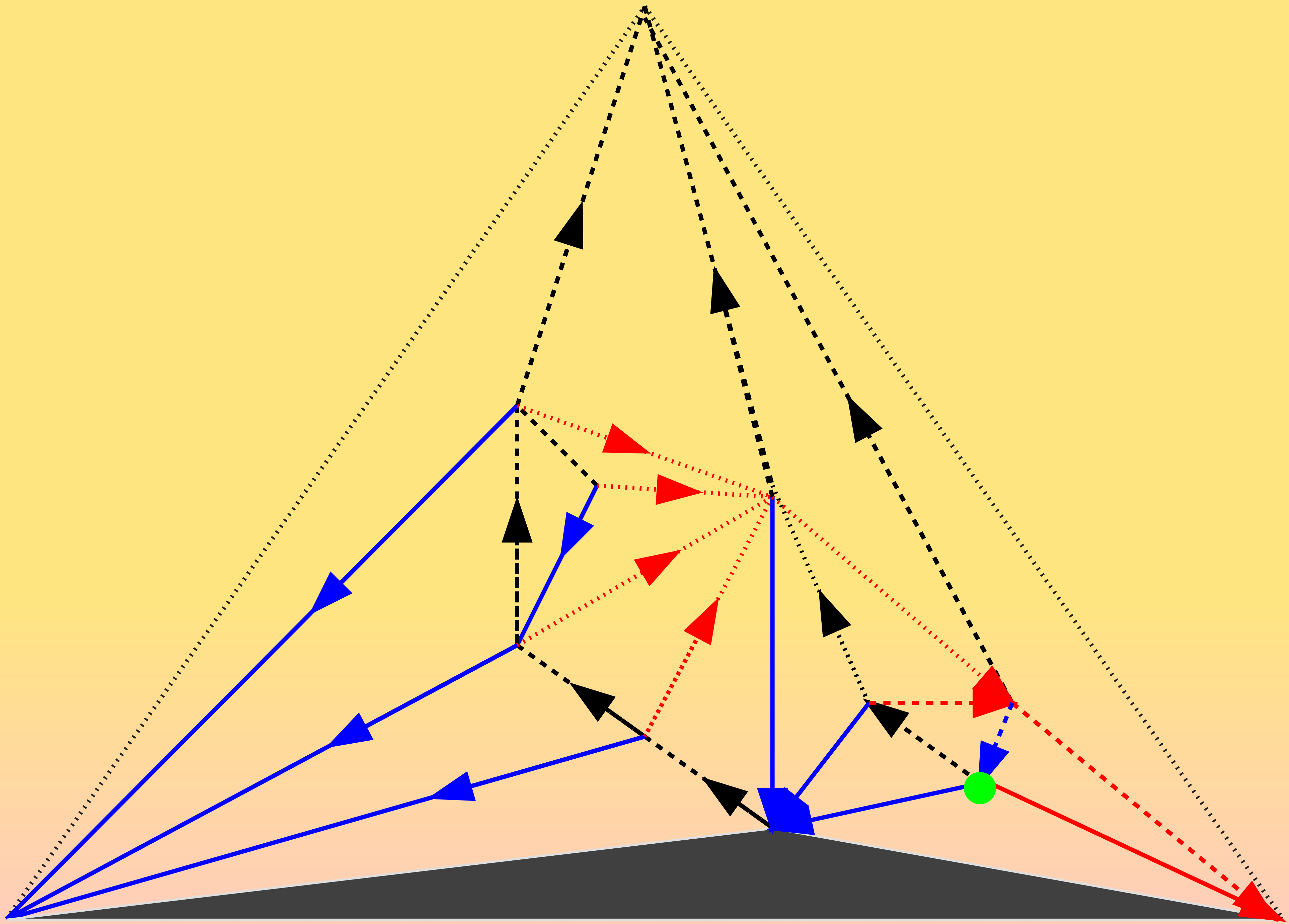
Incremental vertex conquest



Incremental vertex conquest



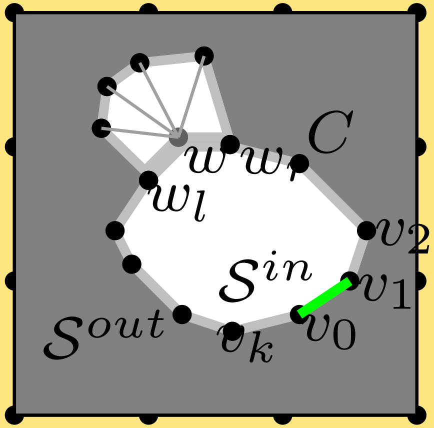
Incremental vertex conquest



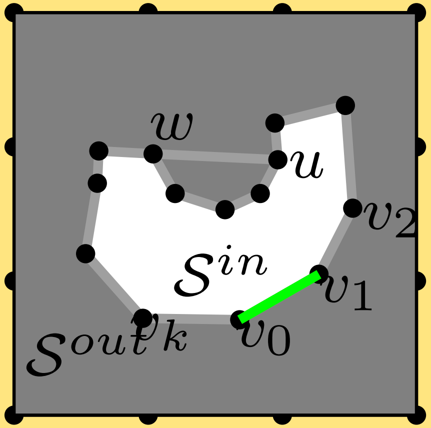
Computing g-Schnyder Woods

New handle operators: split and merge

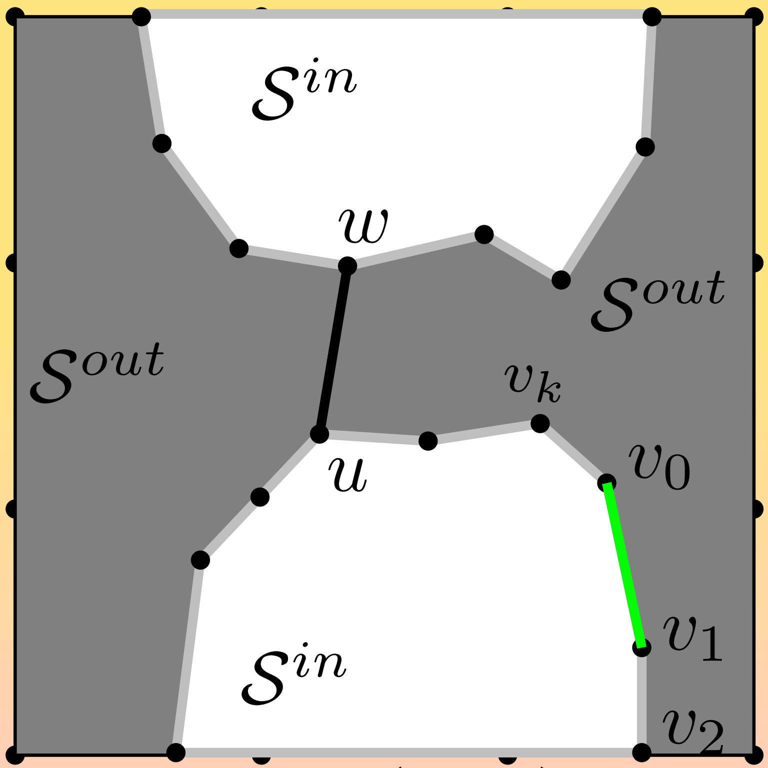
conquer(w)



chordal edge (u, w)

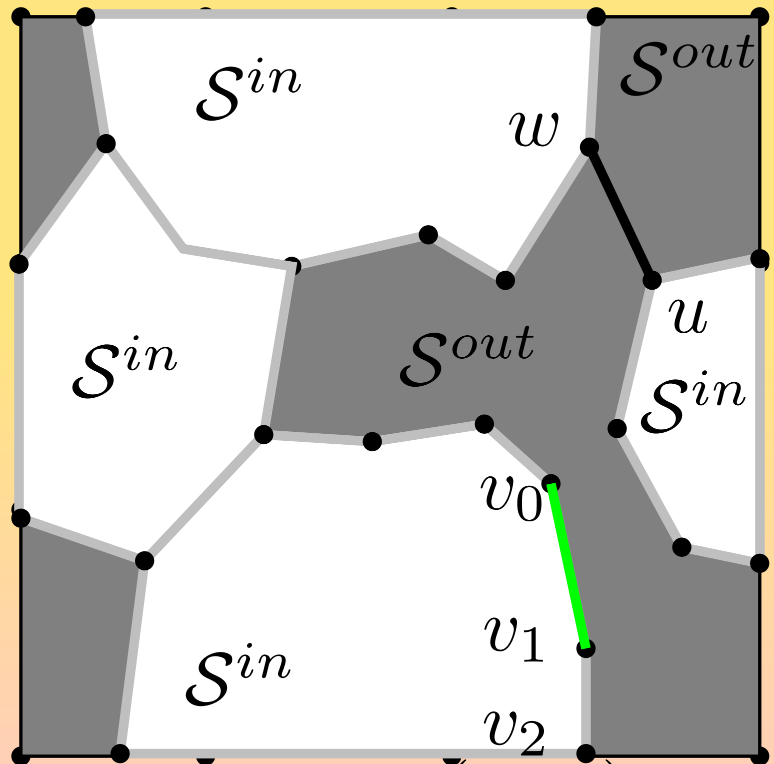


(u, w) chordal edge



split(u, w)

(u, w) defines a non-trivial cycle



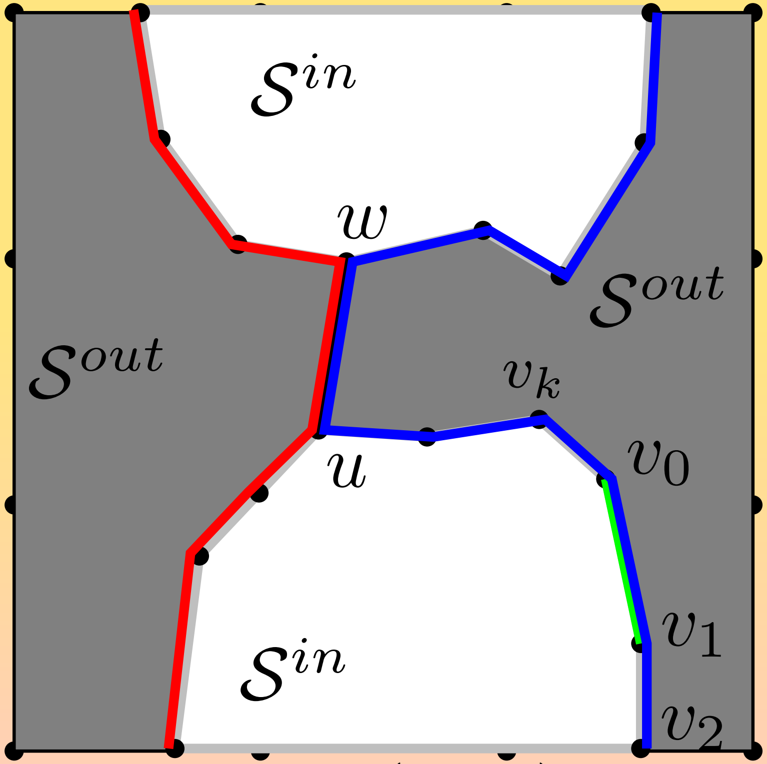
merge(u, w)

New handle operators: split and merge

(u, w) chordal edge defining a non-trivial cycle

(u, w) split edge

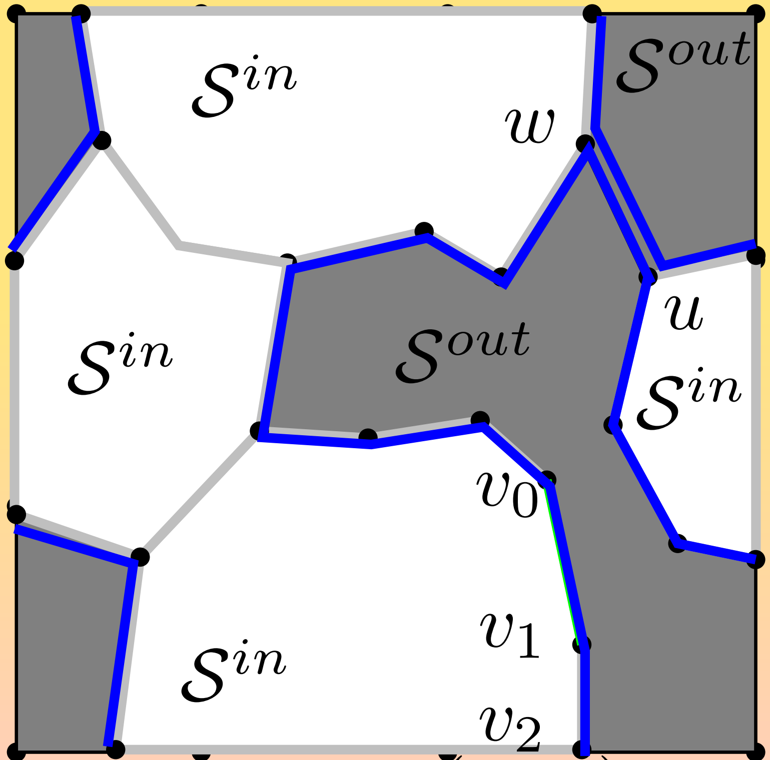
split a boundary into 2 boundary components



`split`(u, w)

(u, w) merge edge

merge two differents boundary components

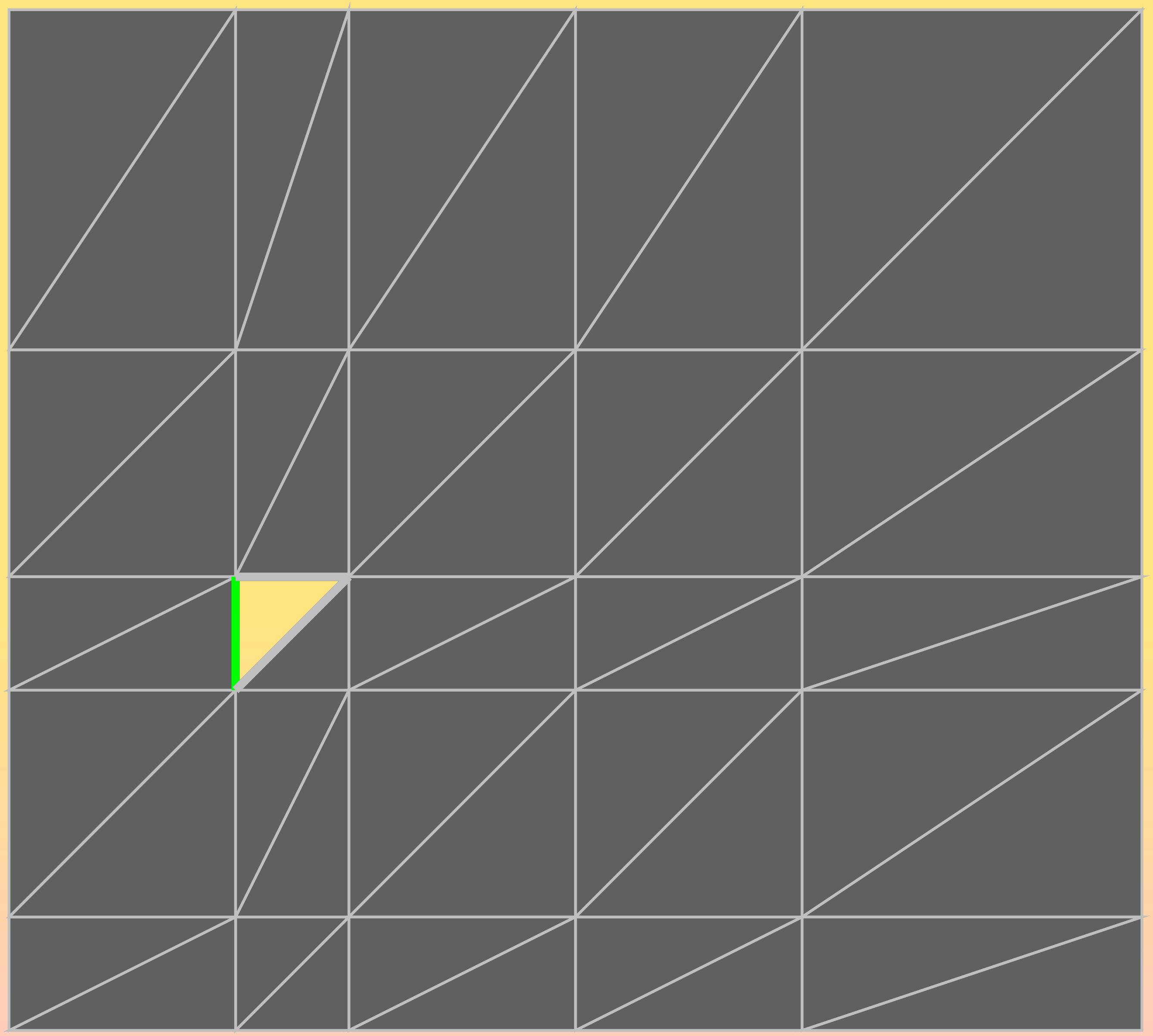


`merge`(u, w)

Example of execution of our algorithm (toroidal case)

Starts the traversal from the root face (green edge)

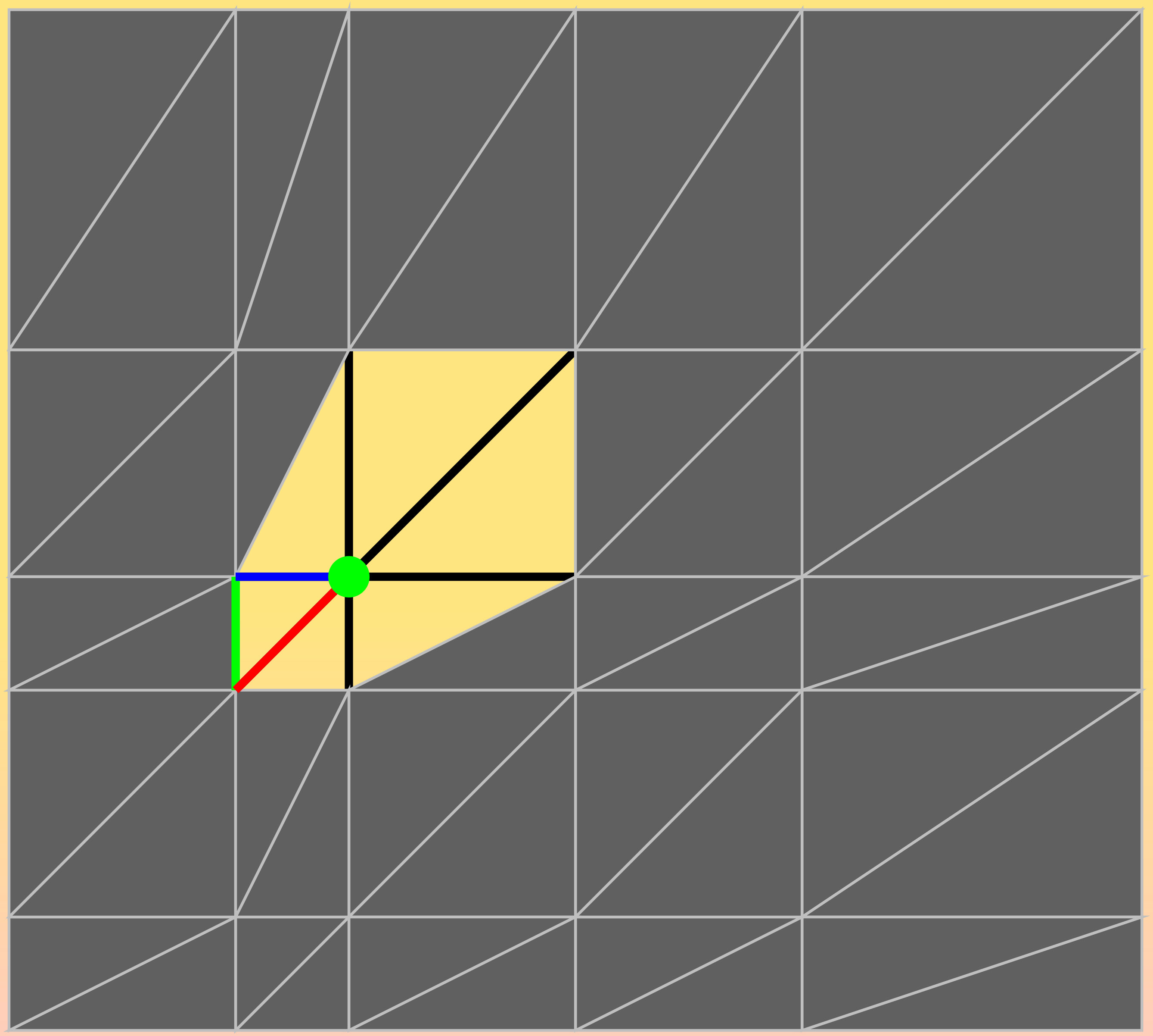
$\mathcal{S}$



Example of execution of our algorithm (toroidal case)

conquer$(w)+$
colororient(w)

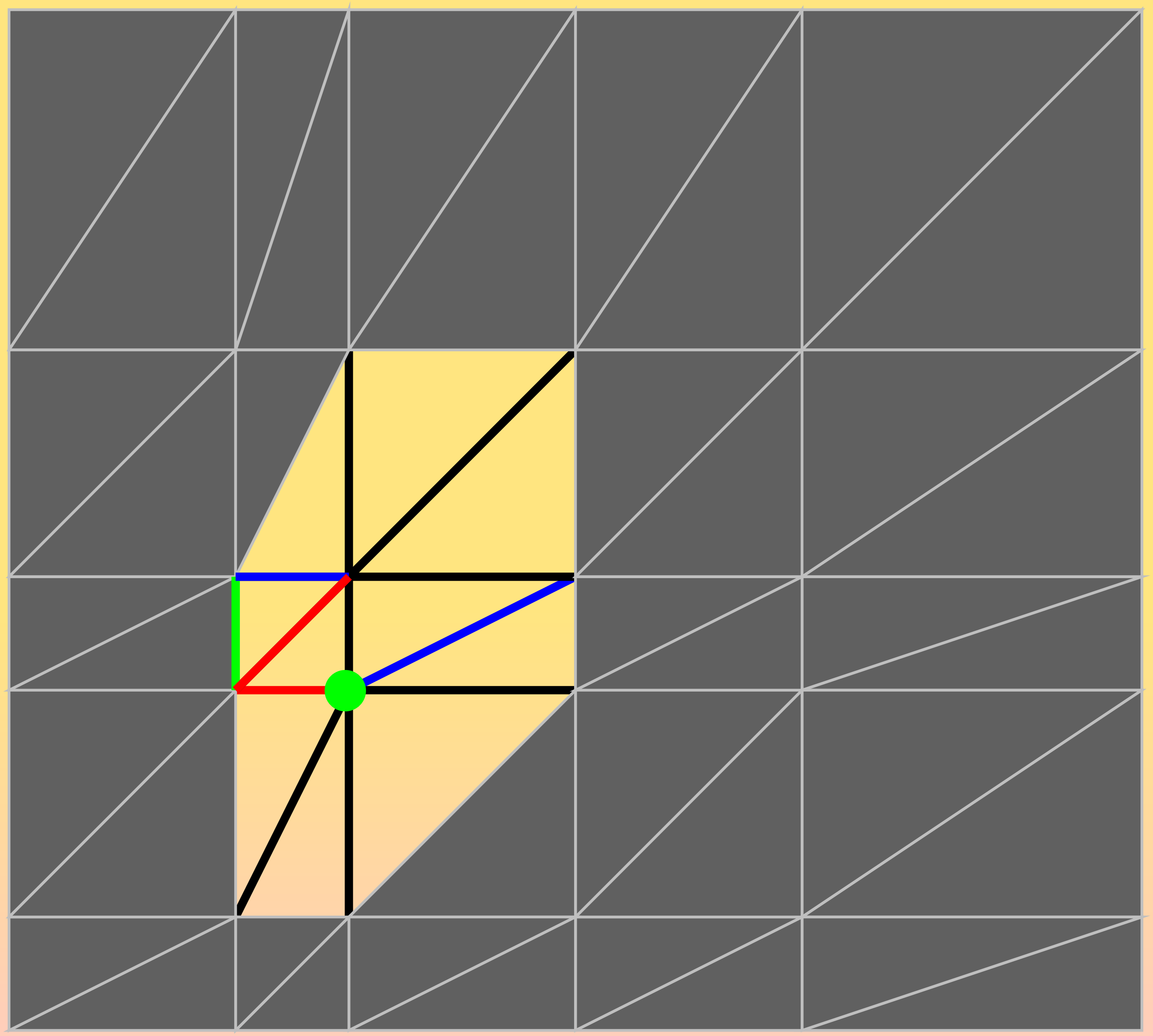
$\mathcal{S}$



Example of execution of our algorithm (toroidal case)

$\texttt{conquer}(w)+$
$\texttt{colororient}(w)$

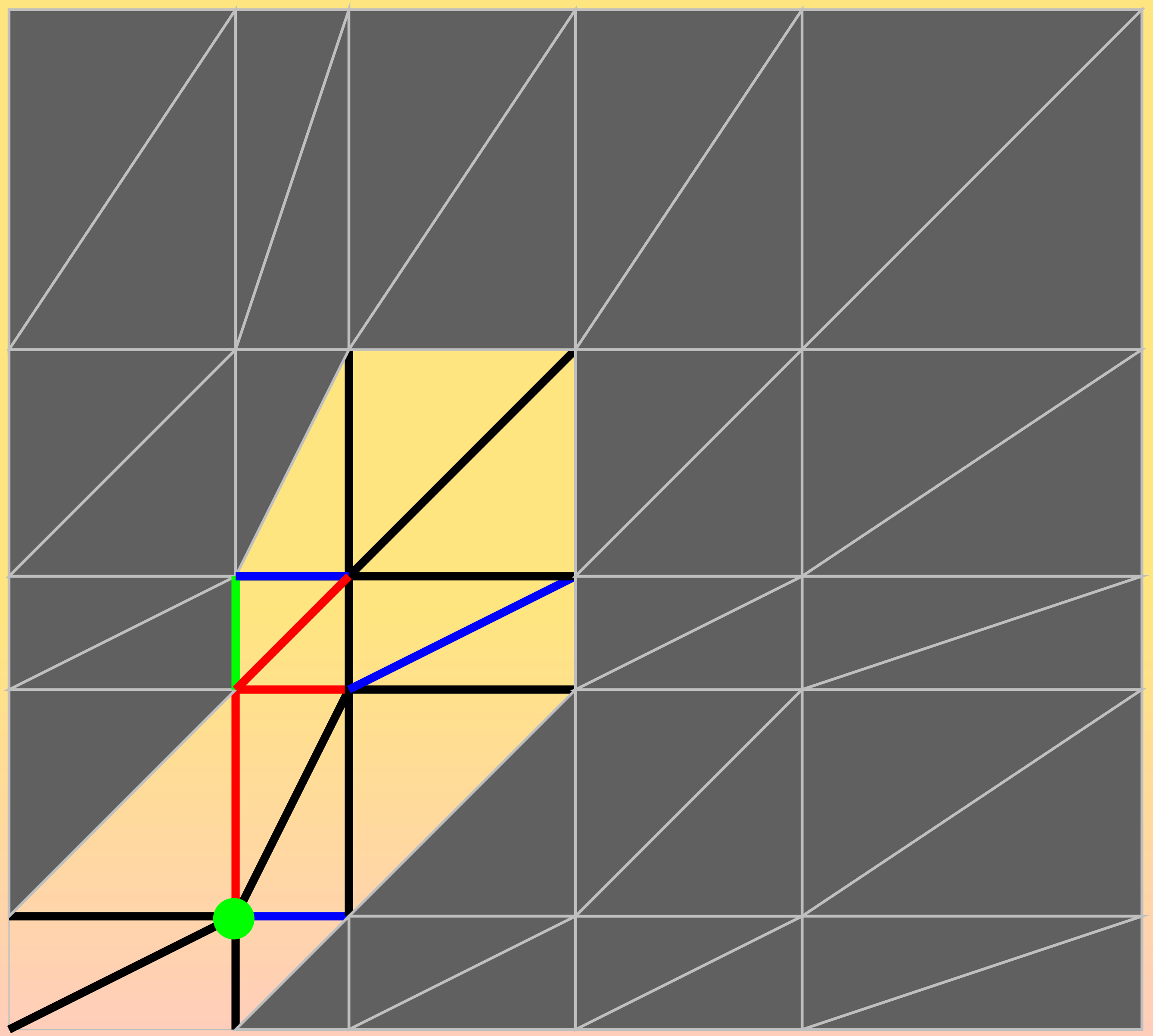
$\mathcal{S}$



Example of execution of our algorithm (toroidal case)

$\texttt{conquer}(w)+$
$\texttt{colororient}(w)$

$\mathcal{S}$



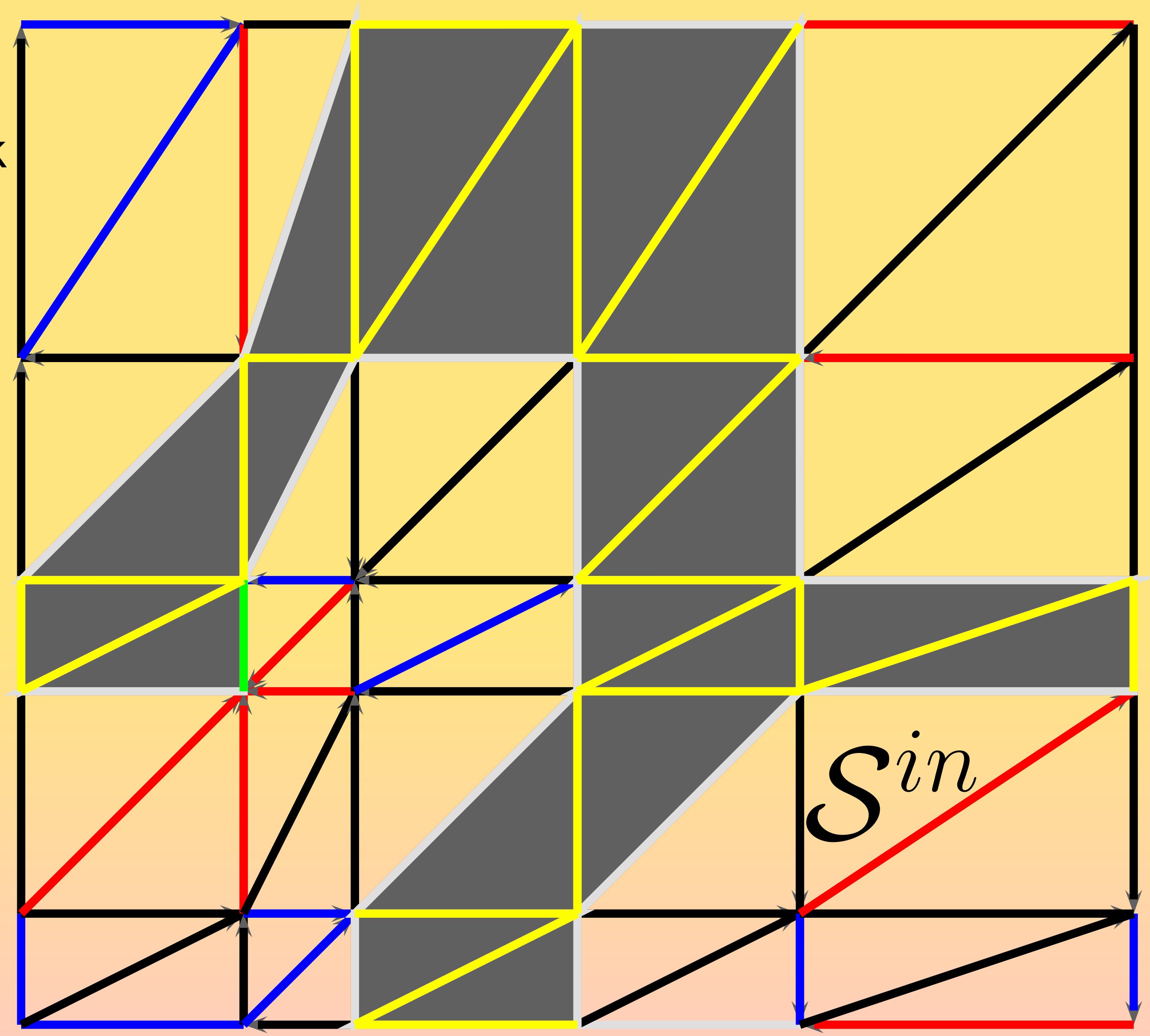
Example of execution of our algorithm (toroidal case)

After a maximal sequence of vertex conquest operations . . .

no more free vertices... the (planar) traversal gets stuck

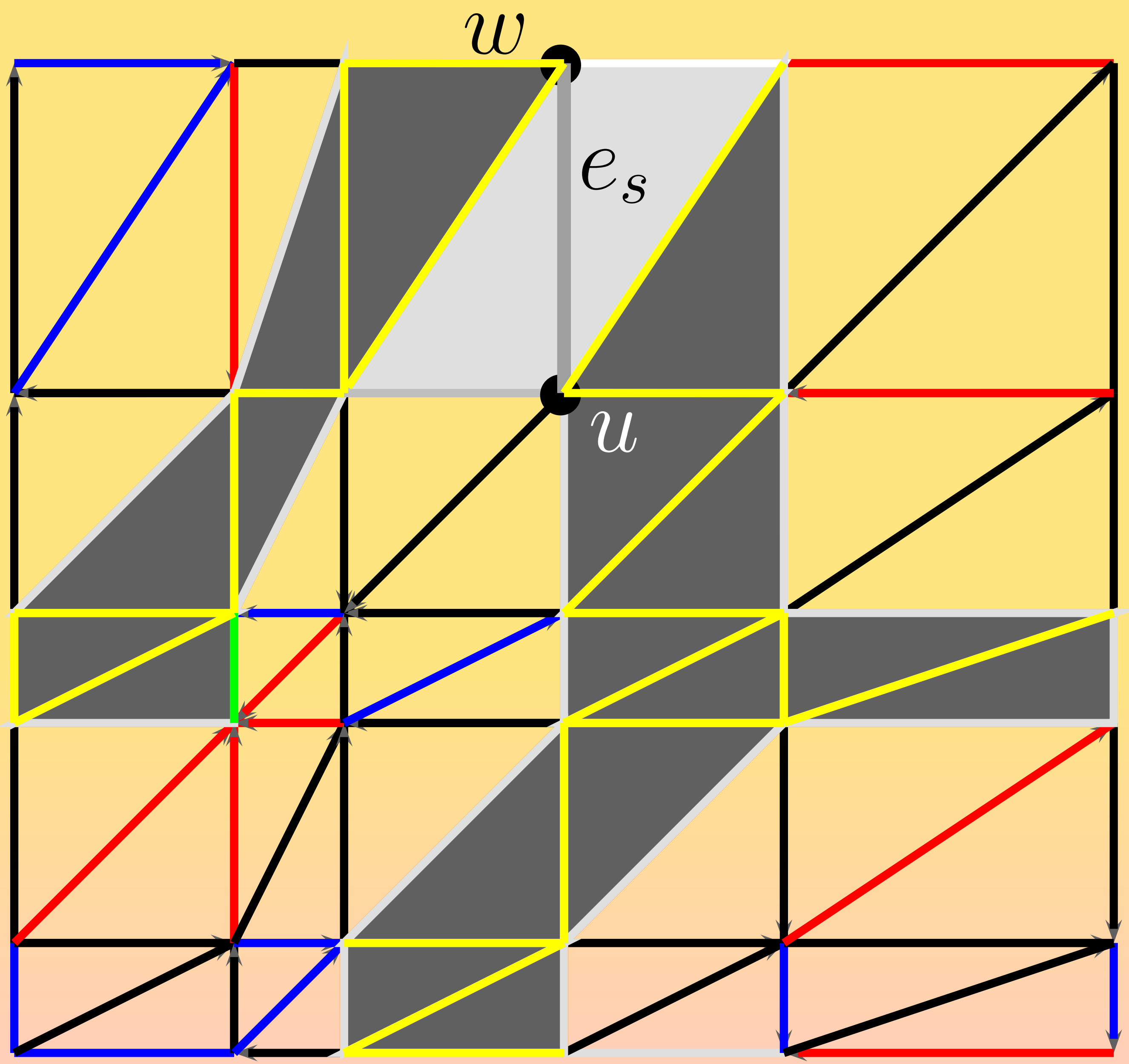
$\mathcal{S}^{in}$ is a topological disk

$\mathcal{S}^{out}$ is a face connected map with a boundary component



Example of execution of our algorithm (toroidal case)

Let us perform a $\texttt{split}(\texttt{u}, \texttt{w})$ operation

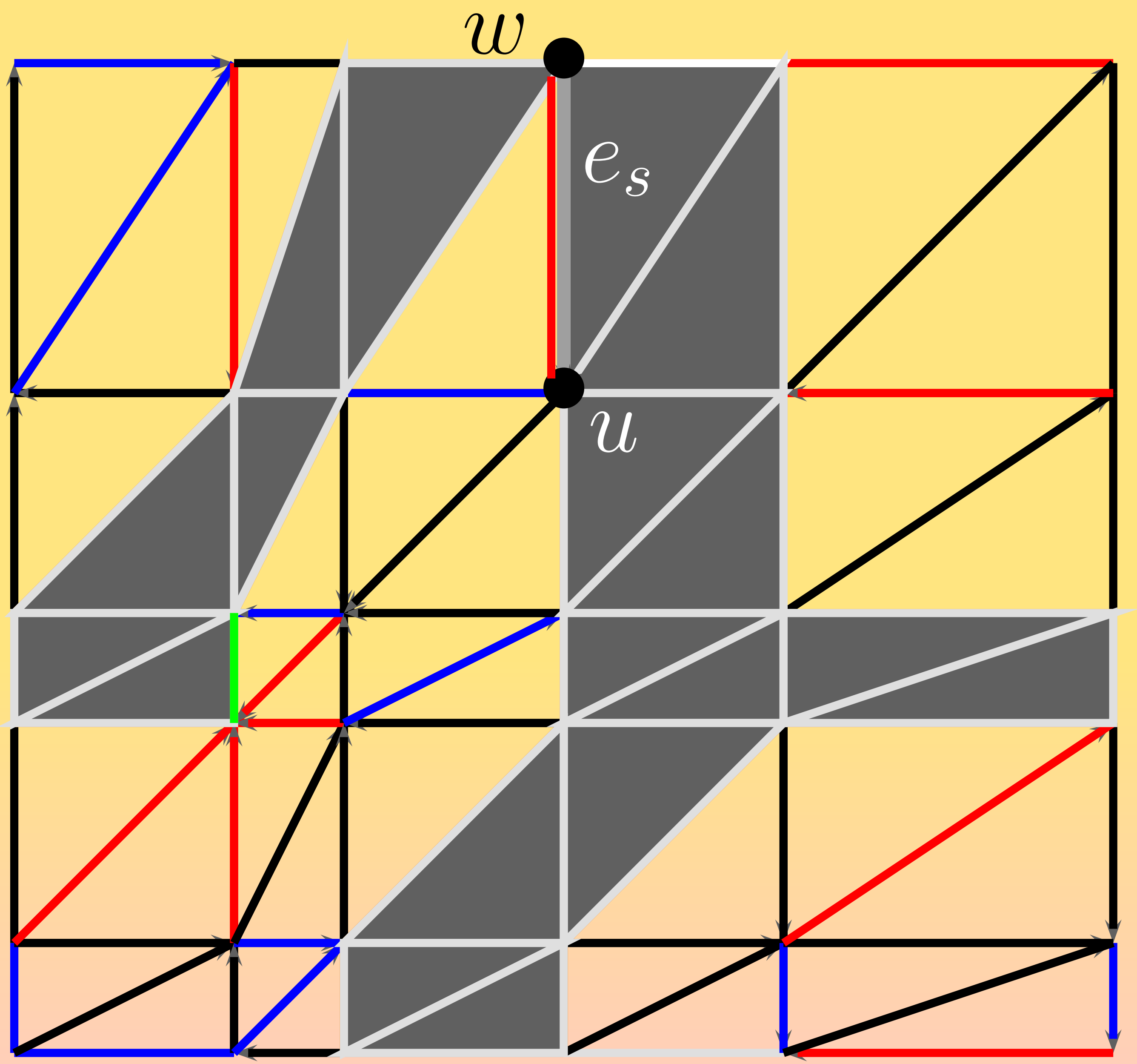


$\mathcal{S}^{out} \setminus (u, w)$ is a face connected map of genus 1 with two boundary components

Example of execution of our algorithm (toroidal case)

We can now perform a $\mathtt{conquer}(u)$ operation

$\mathcal{S}^{out} \setminus (u, w)$ is a face connected map of with two boundary components

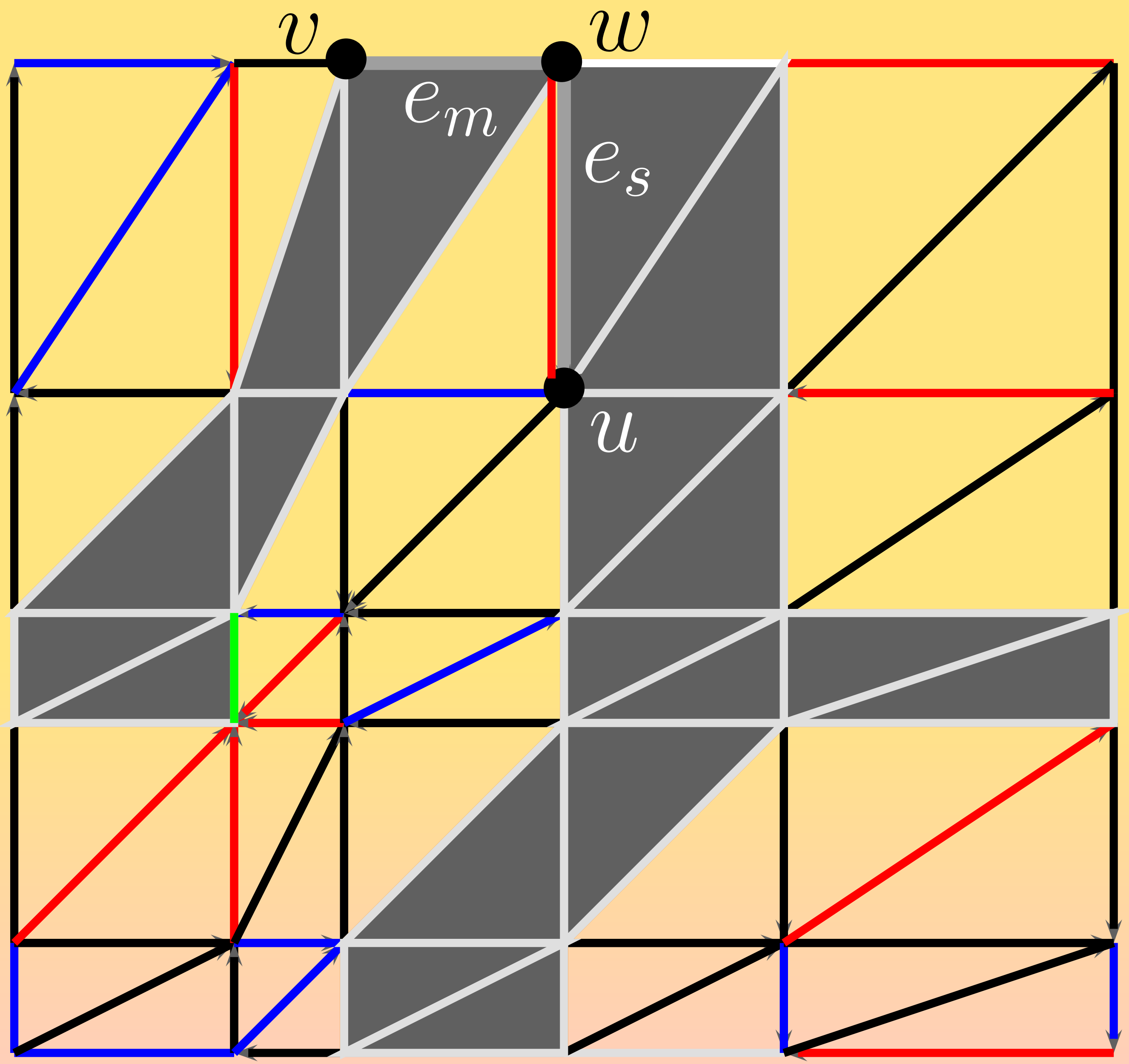


Example of execution of our algorithm (toroidal case)

Let us perform a $\mathtt{merge}(w, v)$ operation

`merge` operations decrease of 1 the number of boundary components

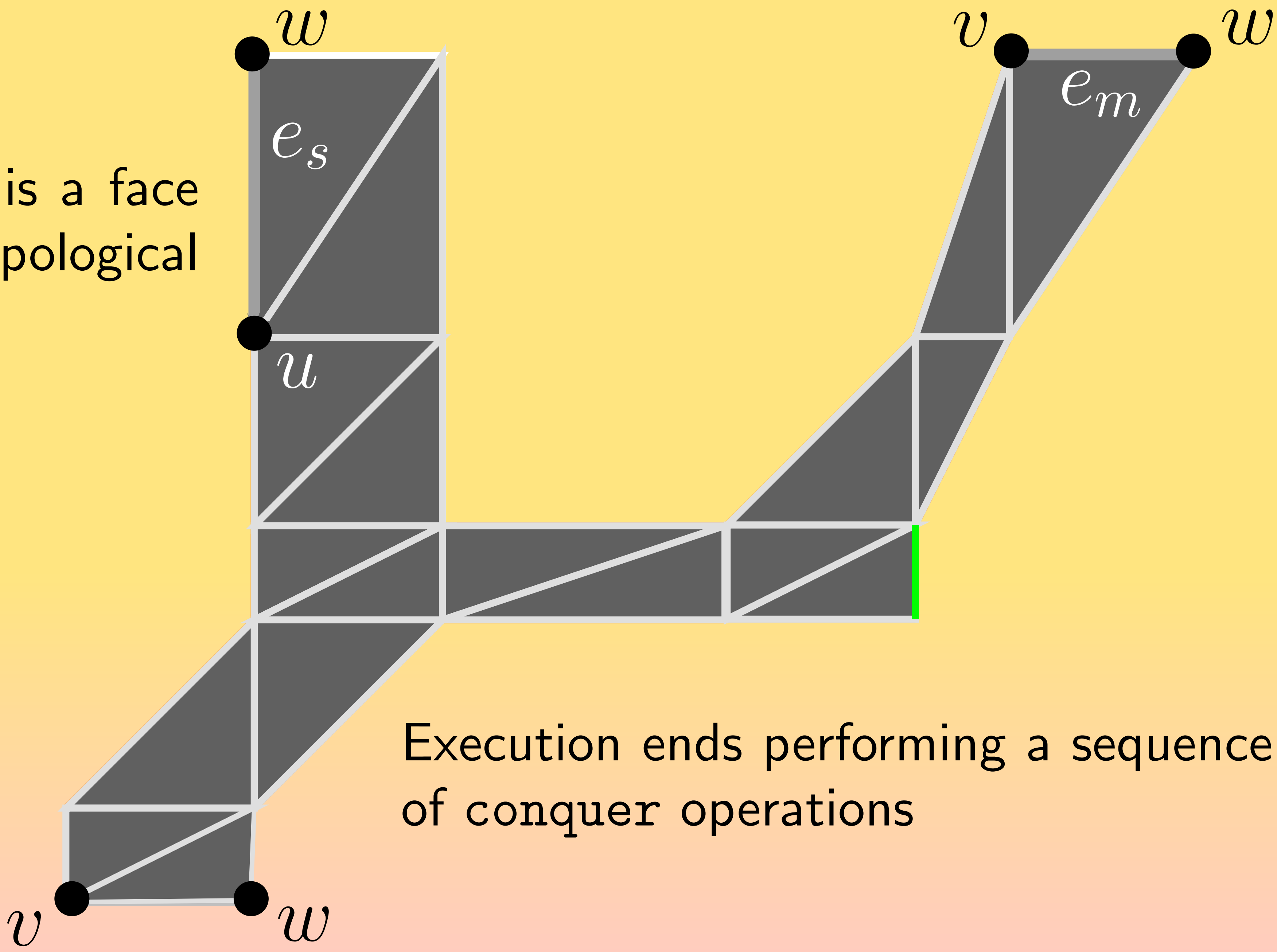
$\mathcal{S}^{out} \setminus (w, v)$ is a face connected topological disk



Example of execution of our algorithm (toroidal case)

Let us see in a better way...

$\mathcal{S}^{out} \setminus (w, v)$ is a face connected topological disk

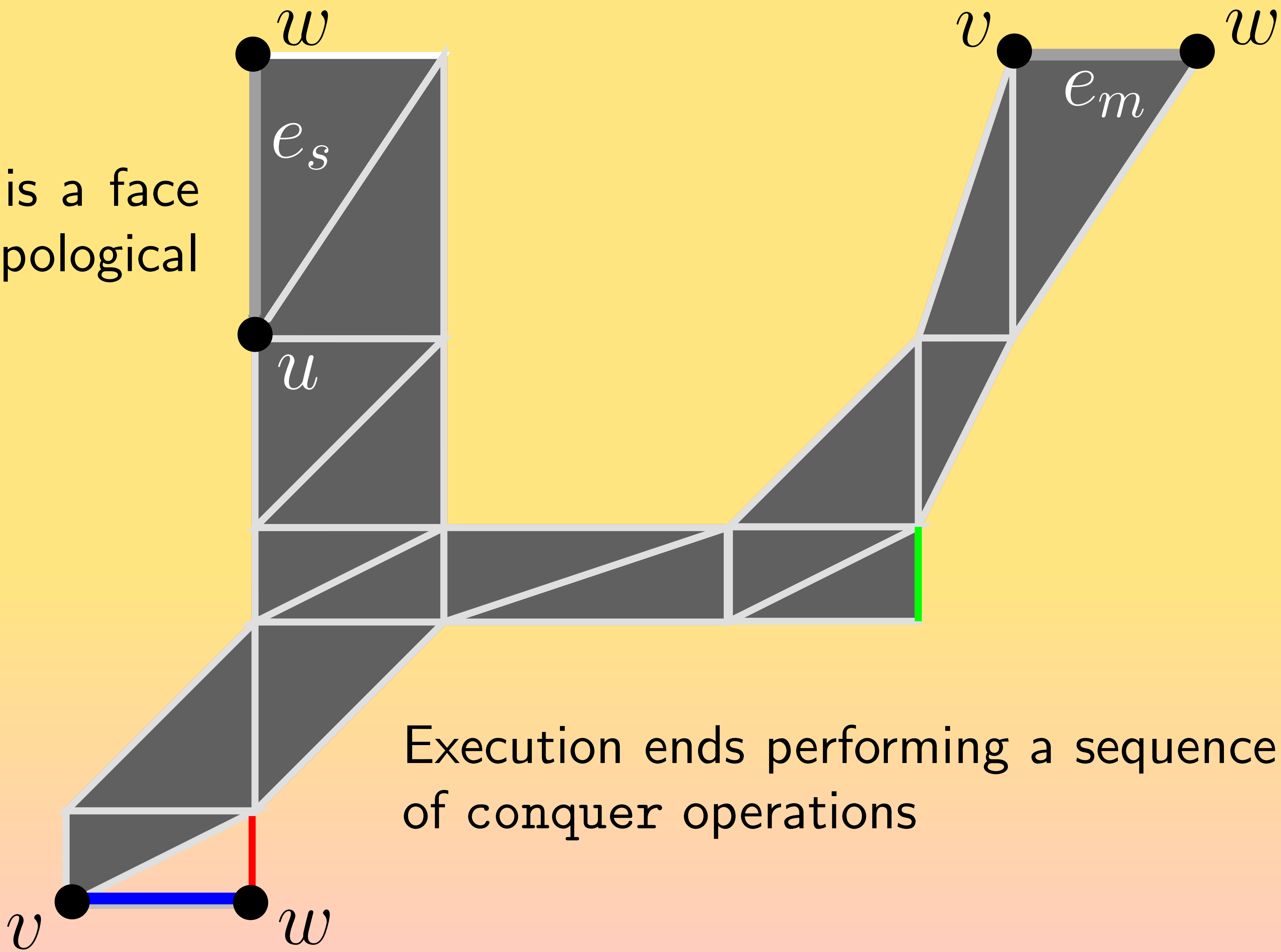


Execution ends performing a sequence of `conquer` operations

Example of execution of our algorithm (toroidal case)

Let us see in a better way...

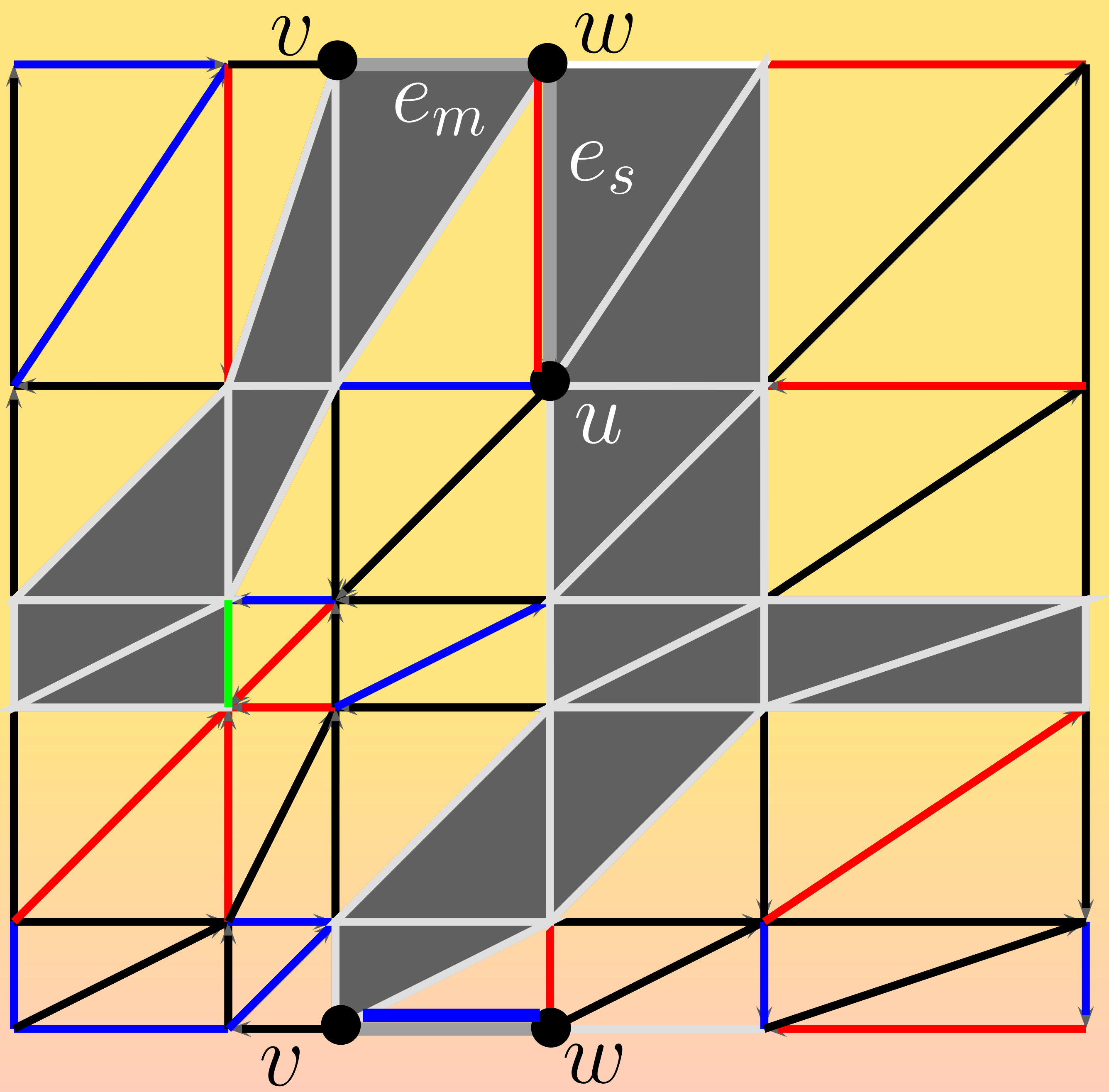
$\mathcal{S}^{out} \setminus (w, v)$ is a face connected topological disk



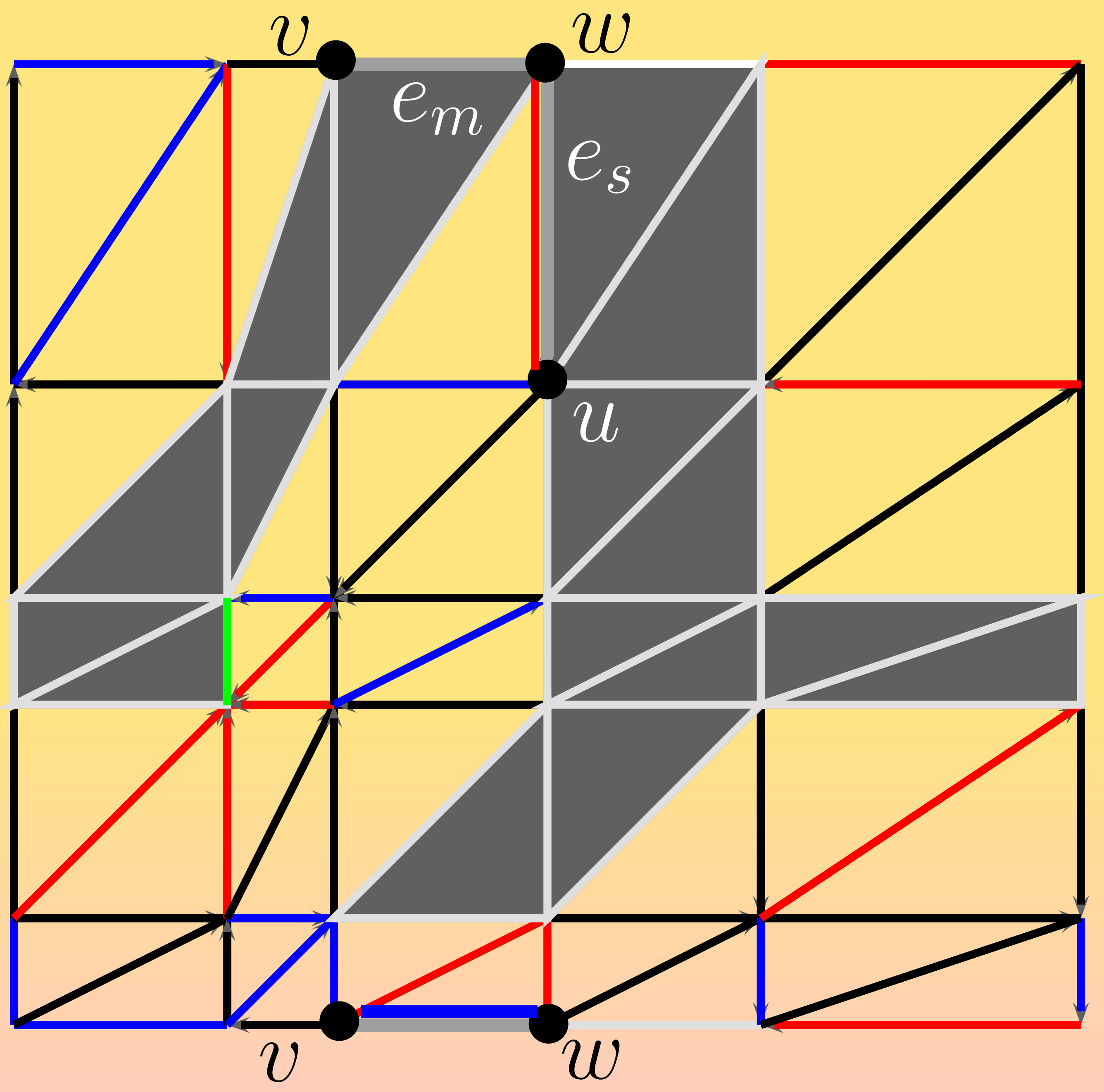
Execution ends performing a sequence of `conquer` operations

Example of execution of our algorithm (toroidal case)

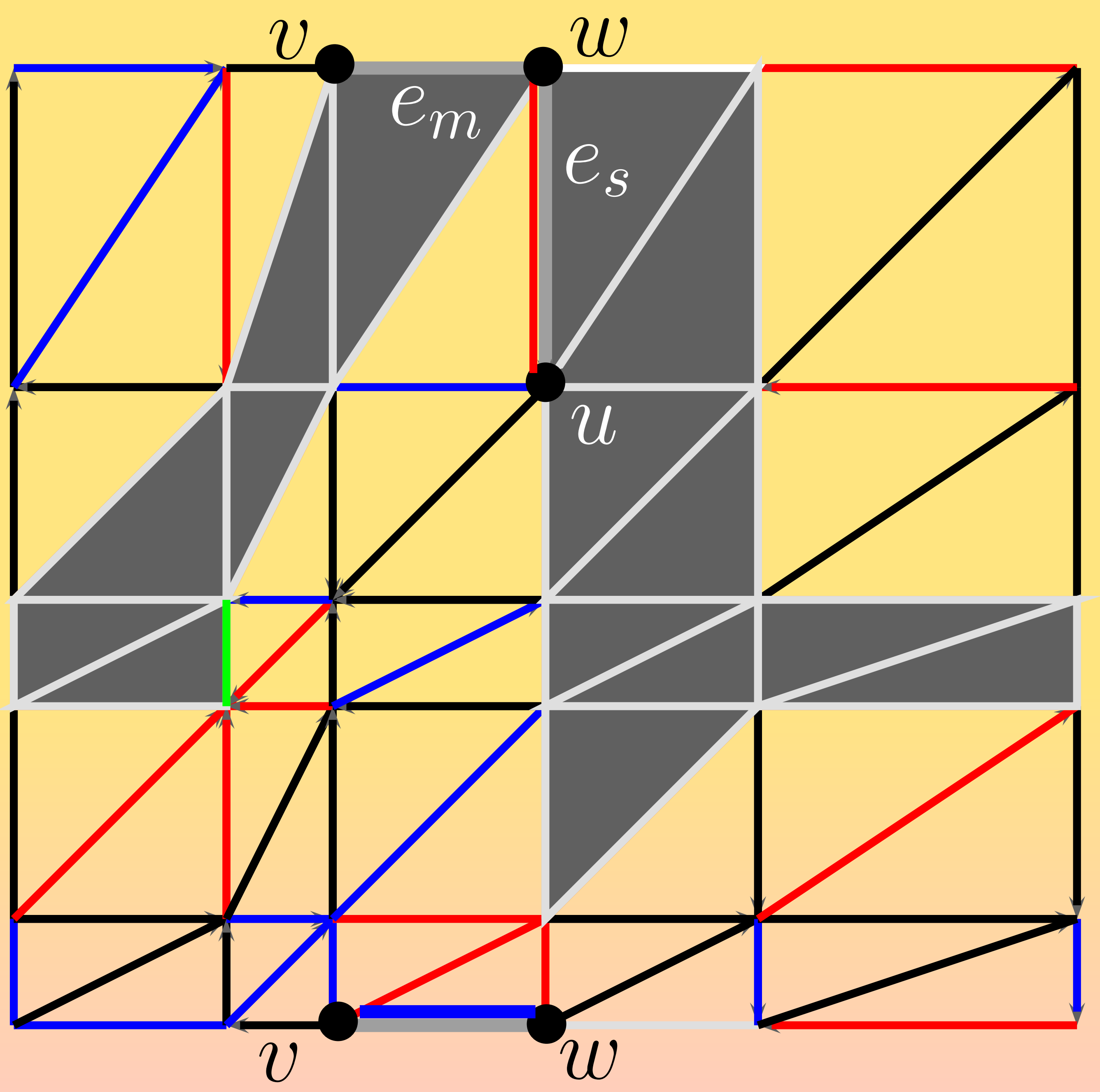
Let us see in a better way...



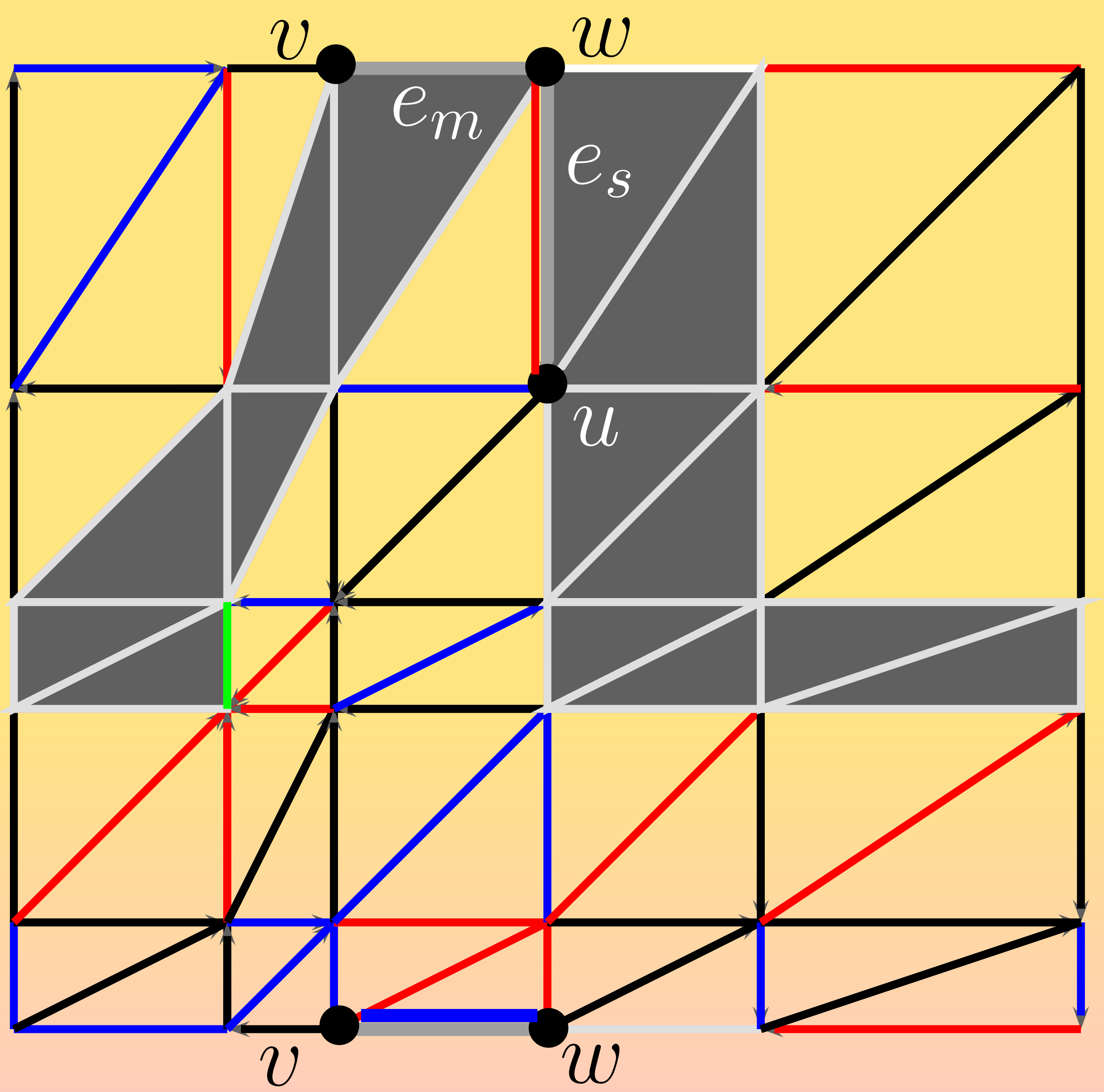
Example of execution of our algorithm (toroidal case)



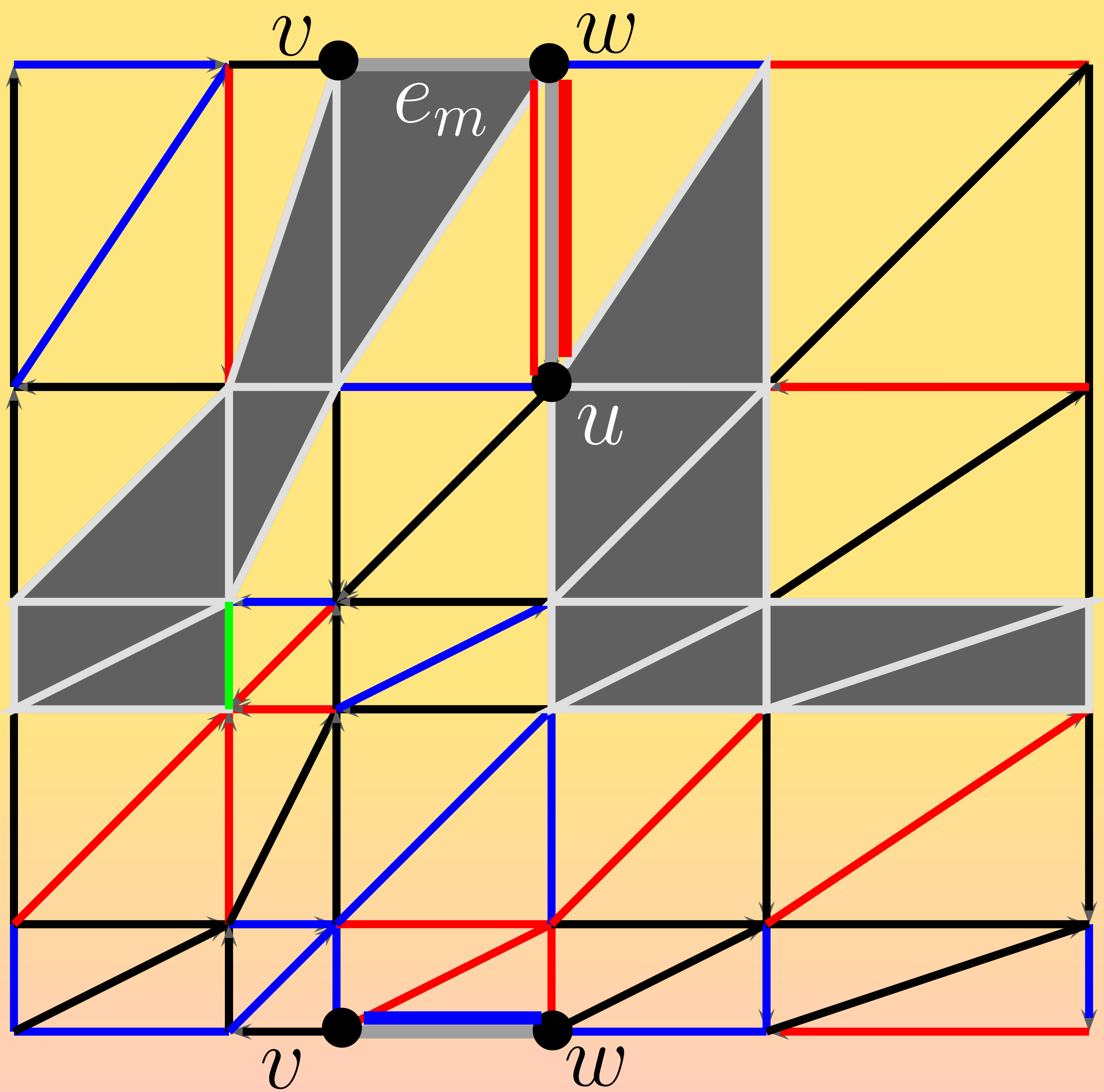
Example of execution of our algorithm (toroidal case)



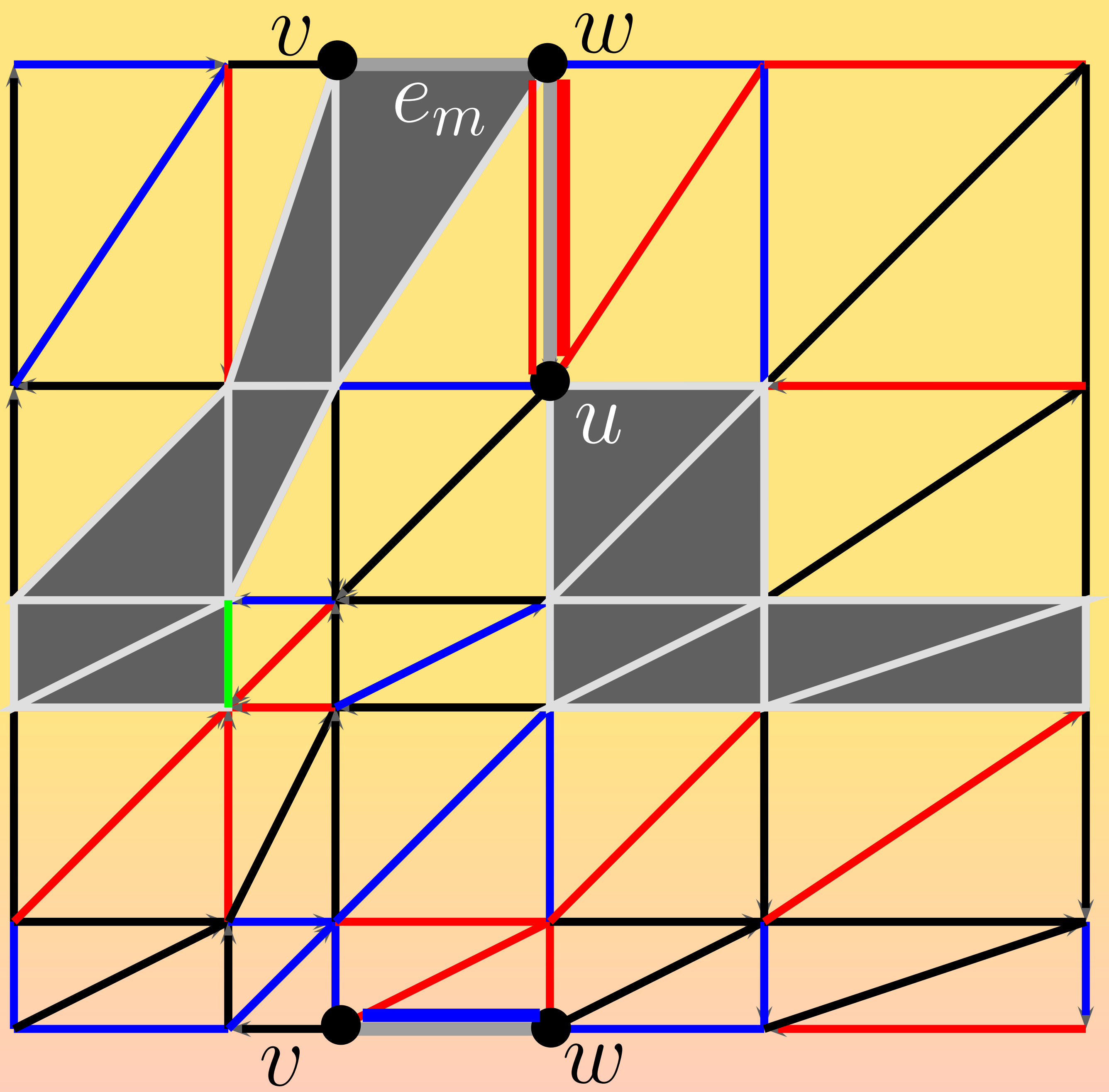
Example of execution of our algorithm (toroidal case)



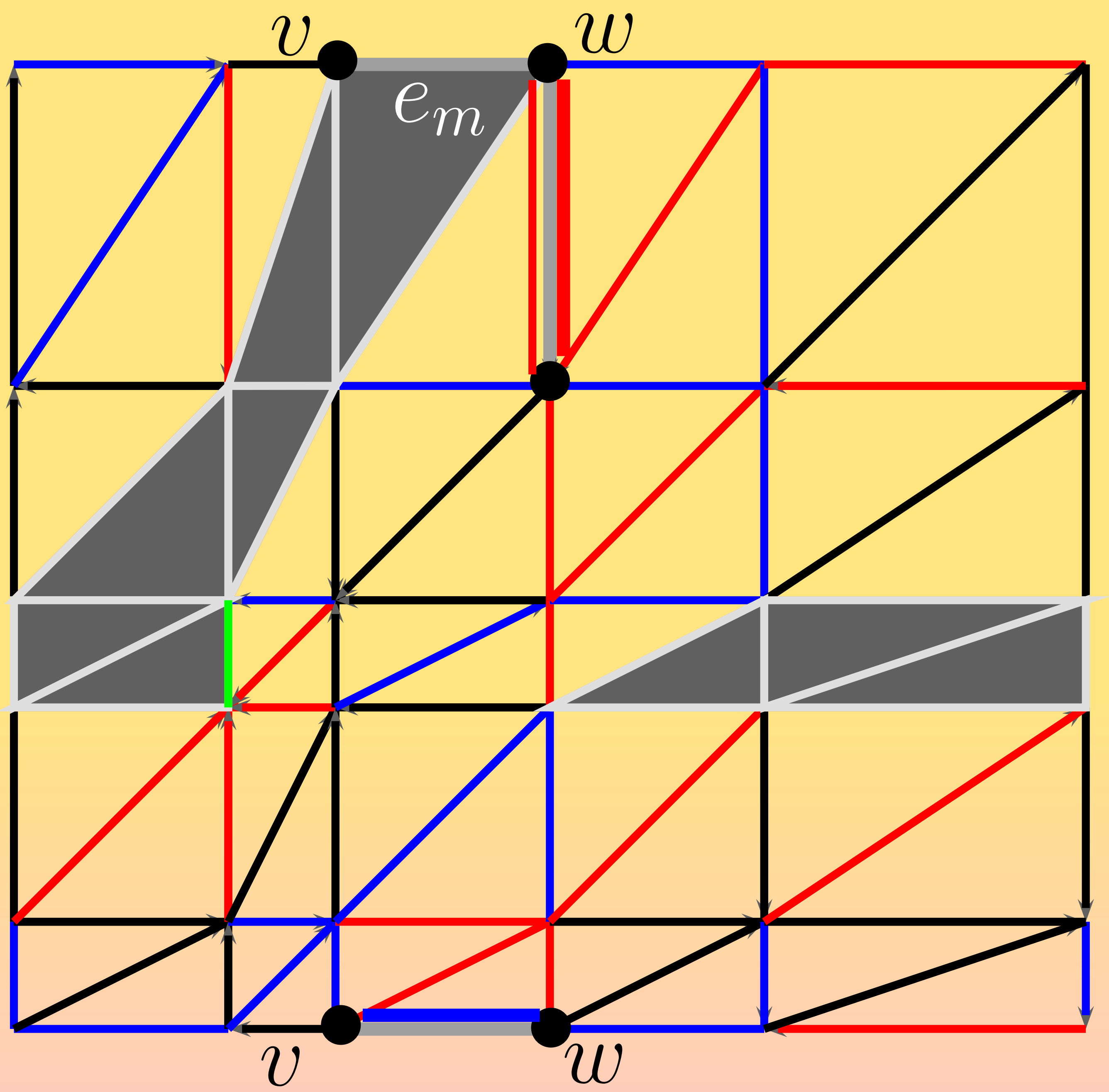
Example of execution of our algorithm (toroidal case)



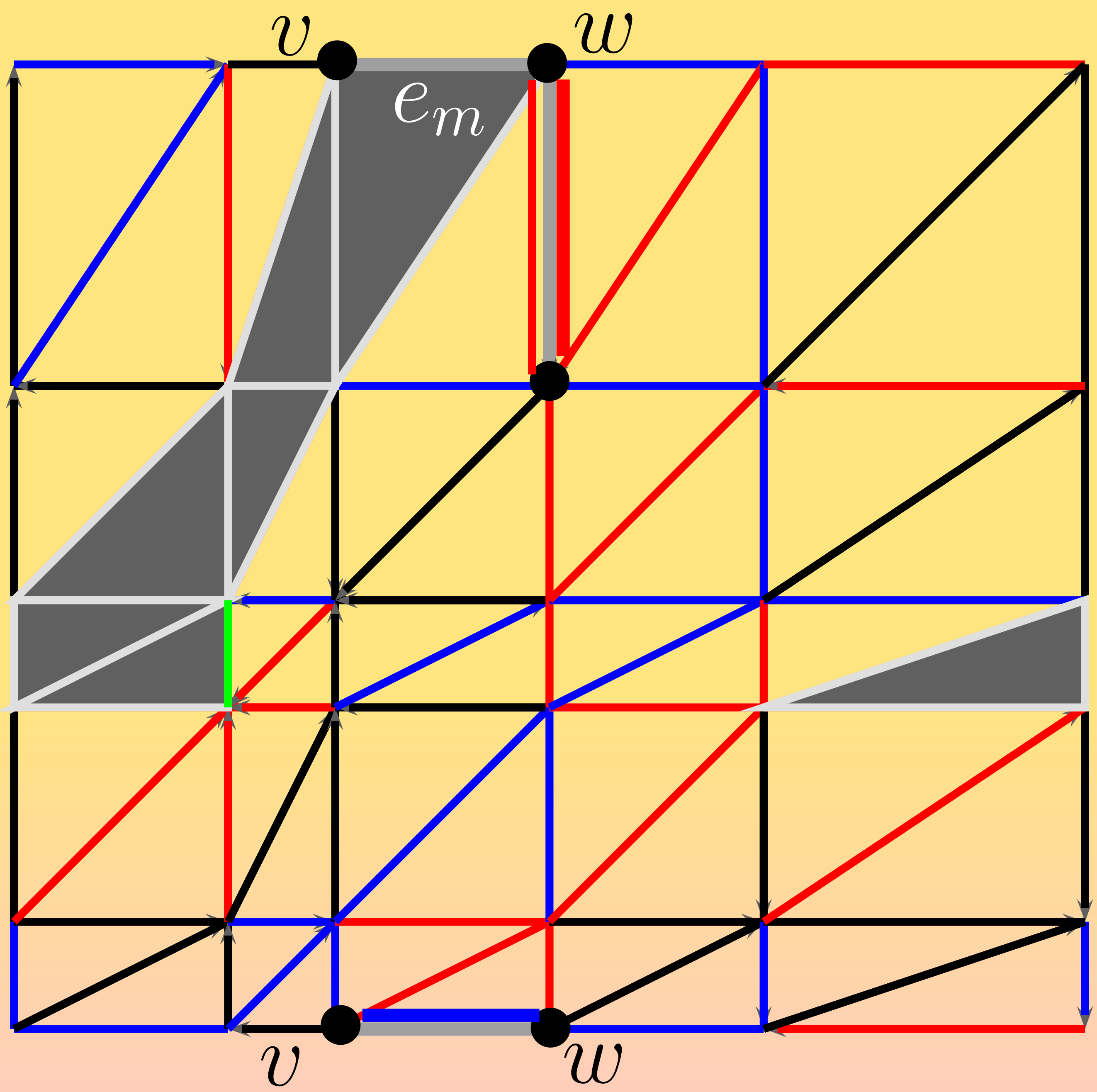
Example of execution of our algorithm (toroidal case)



Example of execution of our algorithm (toroidal case)

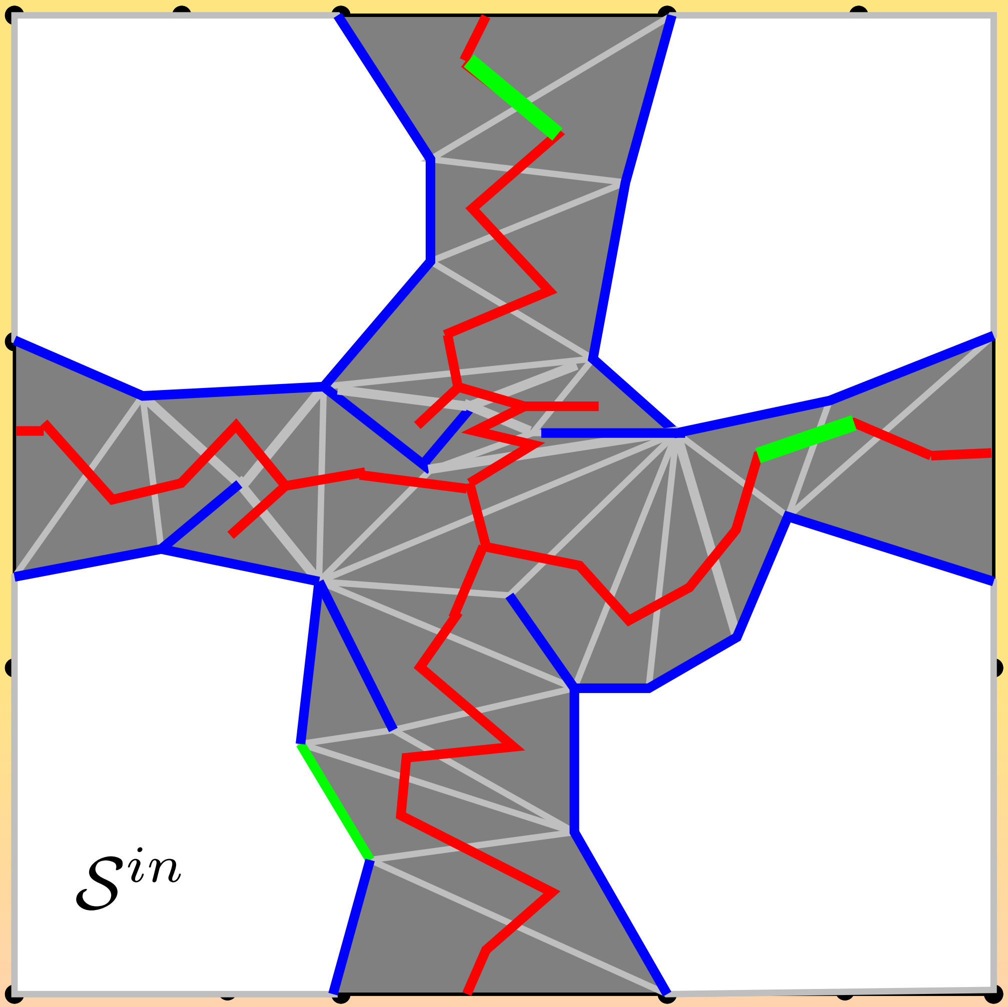


Example of execution of our algorithm (toroidal case)



Correctness and termination

Existence of split and merge edges



T vertex spanning tree of the primal graph, containing the boundary (blue) edges

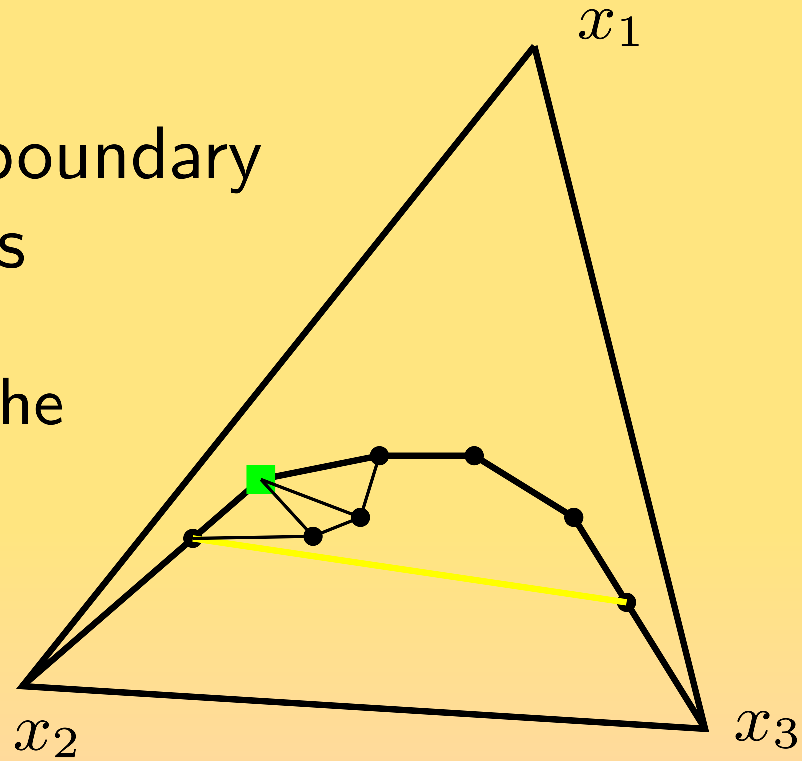
T^* is a 1 face map containing $2g$ non-trivial cycles

T^* contains g split and merge edges

Correctness and termination

A fundamental lemma about chordal edges (the planar case)

Lemma
There always exists at least one boundary vertex, not incident to chordal edges

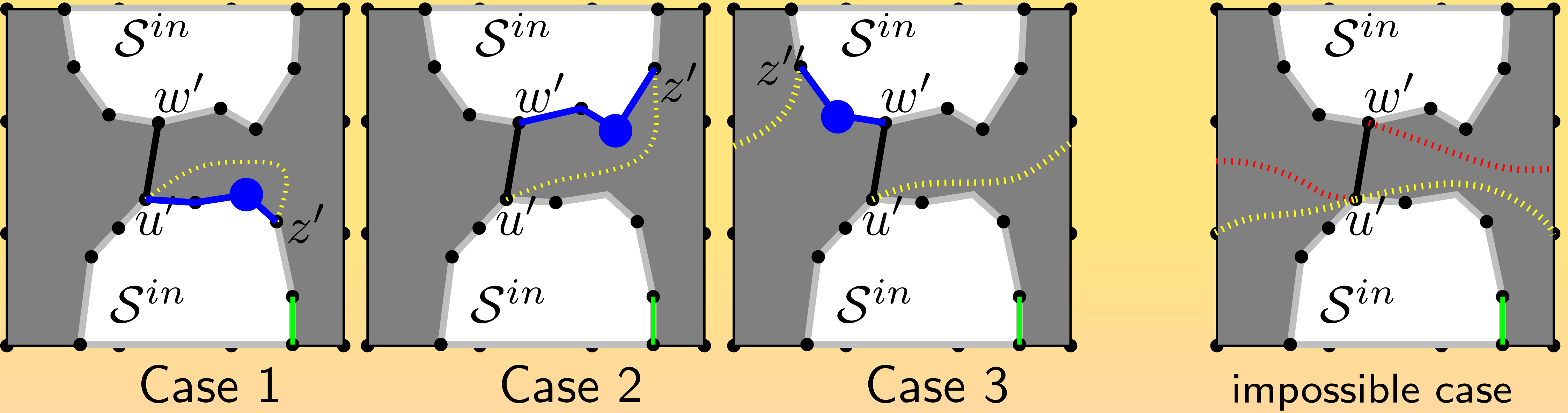


Proof: by induction on the size of the boundary

Theorem Each planar triangulation admits a *canonical ordering* on the vertices

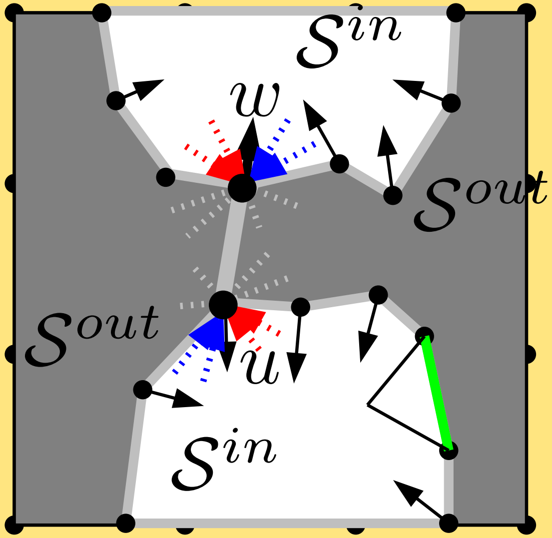
Correctness and termination (higher genus)

Properties of chordal edges in genus g

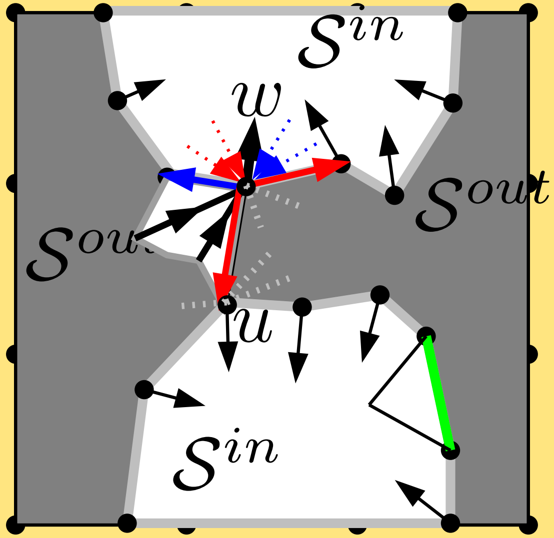


Correctness and termination (higher genus)

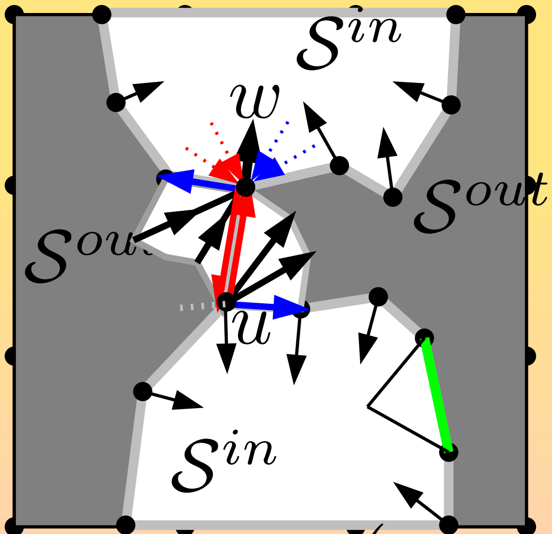
Coloring invariants



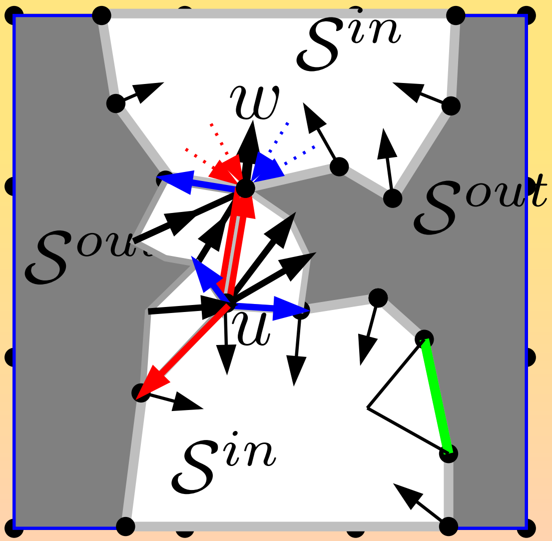
`split`(u, w)



`conquer`$(w)+$
`cololorient`(w)



`conquer`$(u)+$
`cololorient`(u)

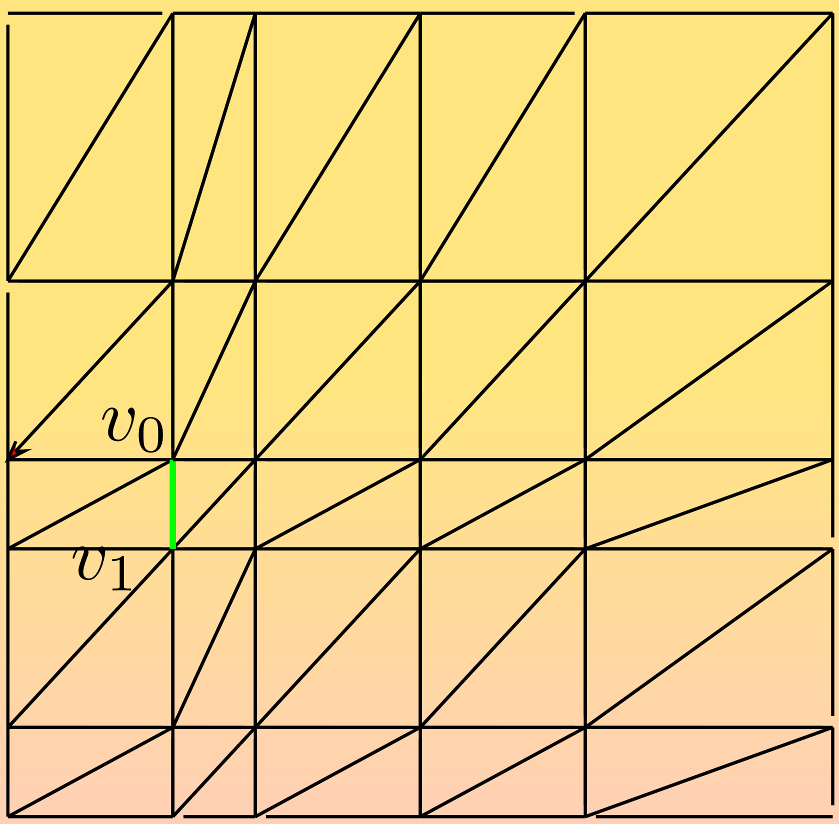


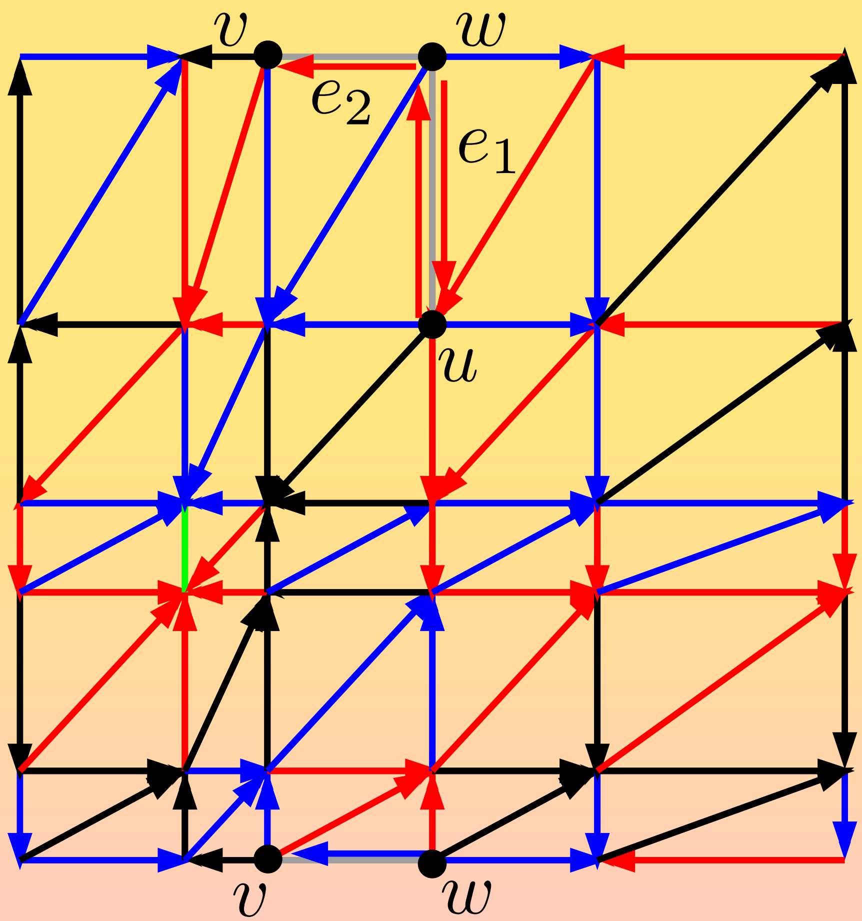
`conquer`$(u)+$
`colorient`(u)

Our main result

Theorem (Castelli Aleardi, Fusy and Lewiner, 2008)
Given a (simple) rooted triangulation $\mathcal{T}$ of genus g and size n, we can compute in $O(n)$ time a g-Schnyder wood of $\mathcal{T}$.

The local Schnyder condition is true almost everywhere in the graph at the exception of multiple vertices



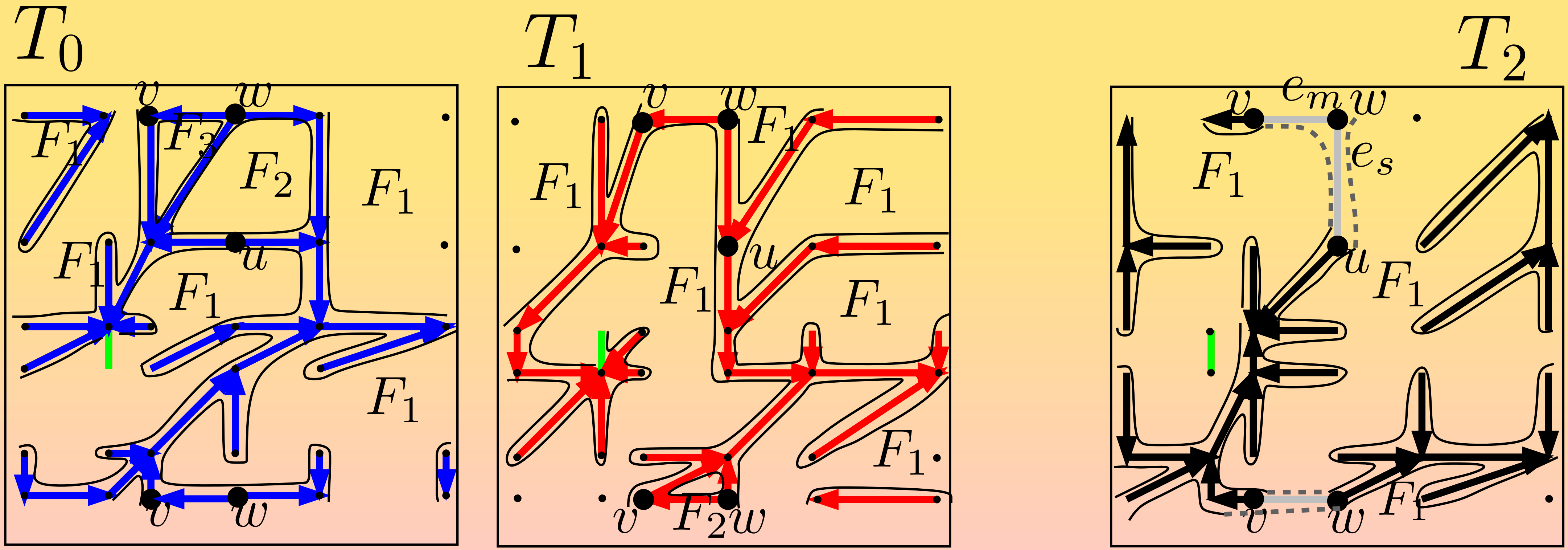


From plane trees to genus g maps

A new characterization in term of g-maps

Theorem (Castelli Aleardi, Fusy and Lewiner, 2008)
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

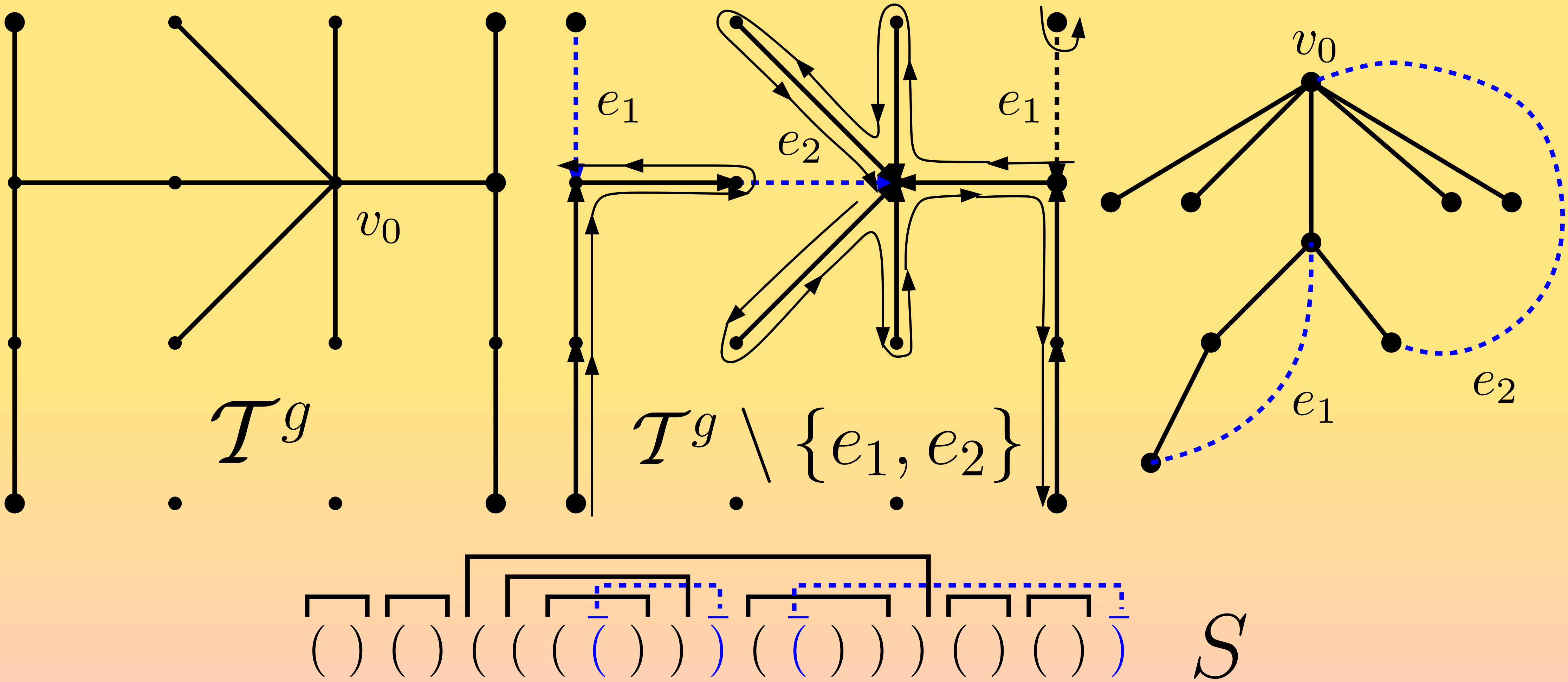


A new encoding application

Encoding g-maps via multiple parentheses words

Corollary

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

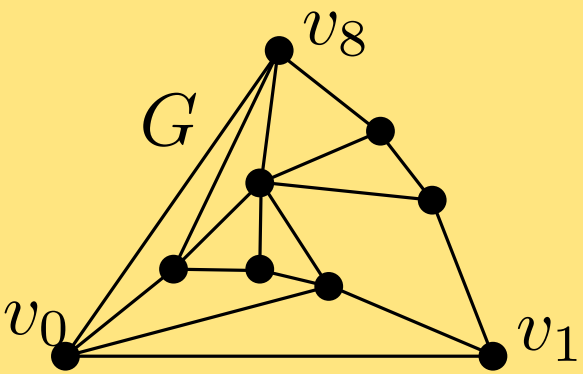


Futur works and open questions

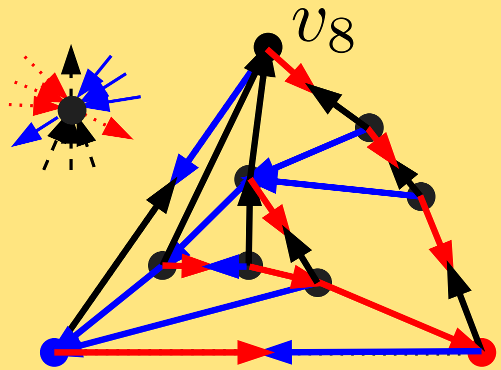
- optimal encoding in higher genus
- lattice structure for the set of Schnyder woods
- extension to the 3-connected case (polygonal meshes)

Futur works

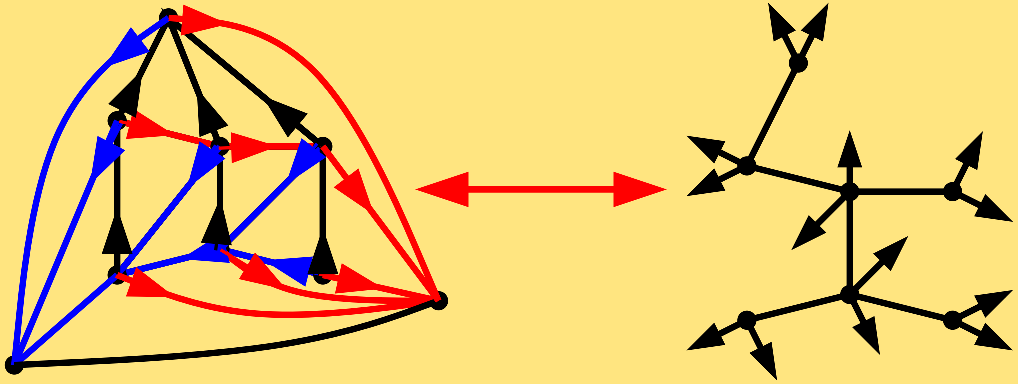
planar graphs



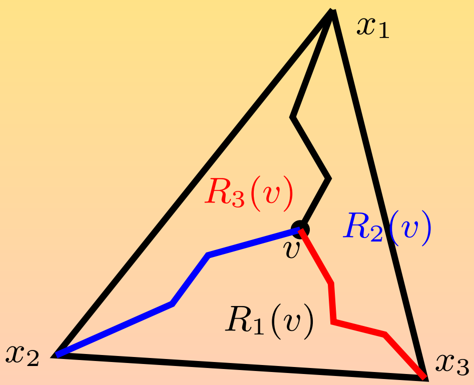
3-connected graph

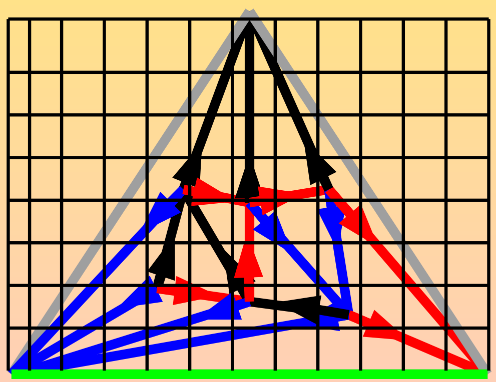


Schnyder wood



bijections between planar maps and trees

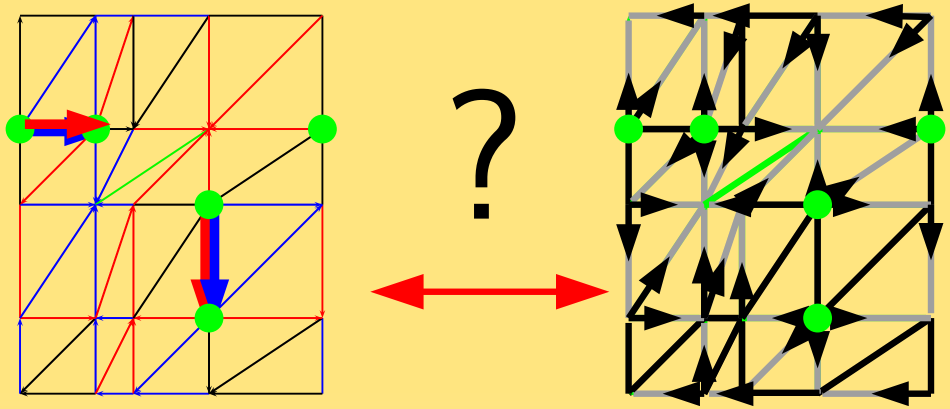




graph drawing

higher genus

?



$\mathcal{S}$

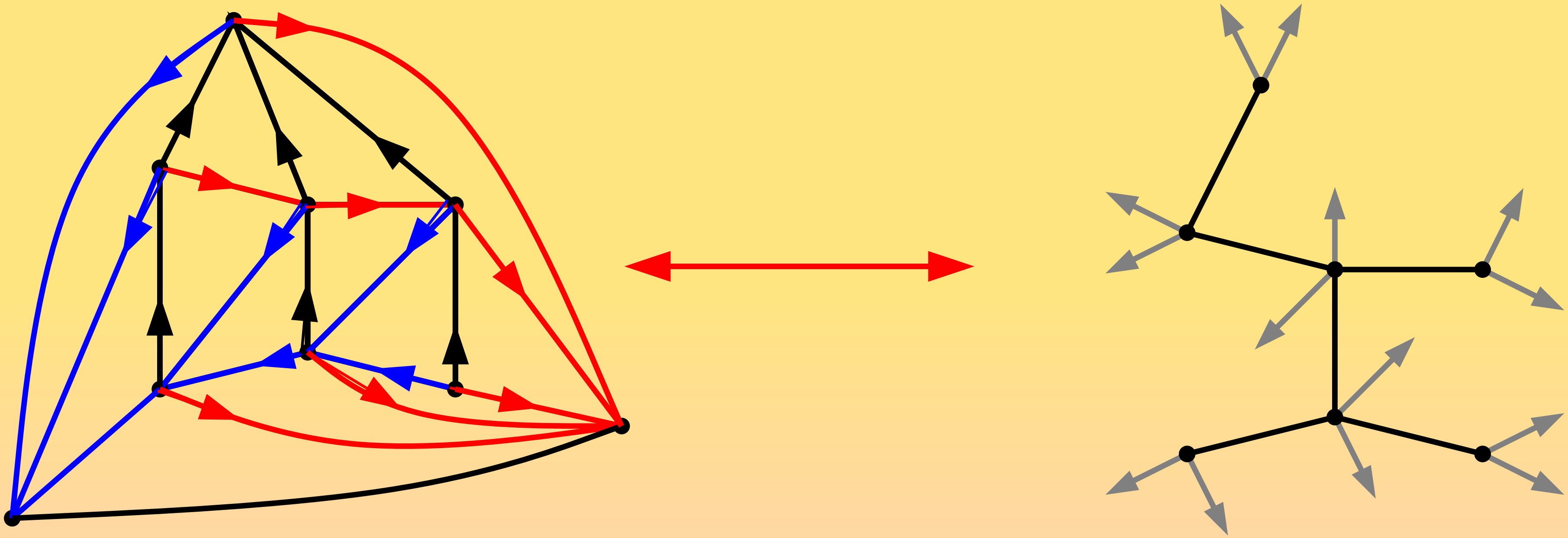
?

Optimal coding and sampling (planar case)

Theorem. (Tutte 62) The number of planar triangulation with $n+2$ vertices is

$$\frac{2(4n-3)!}{(3n-1)!n!} \asymp \left(\frac{256}{27}\right)^n .$$

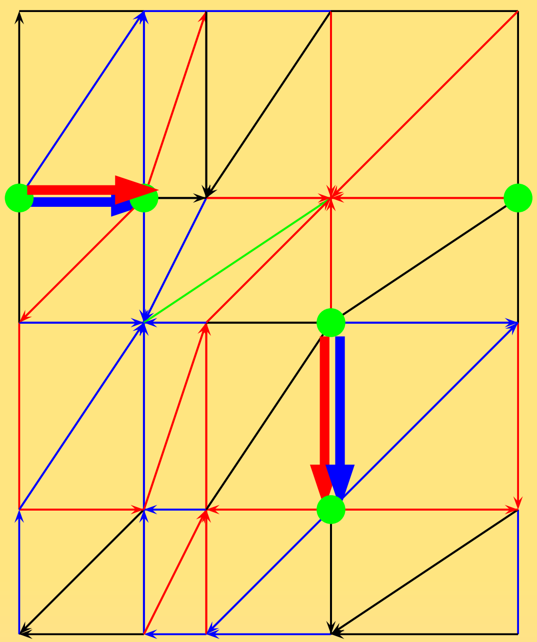
Théorème. (Poulalhon–Schaeffer Icalp 03)
Il existe une bijection entre la classe des arbres de taille n ayant deux bourgeons par noeud, et la classe des triangulations planaires enracinées à $n+2$ sommets.

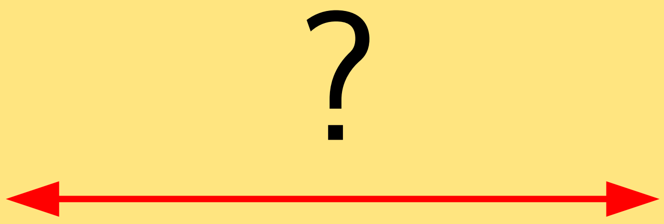


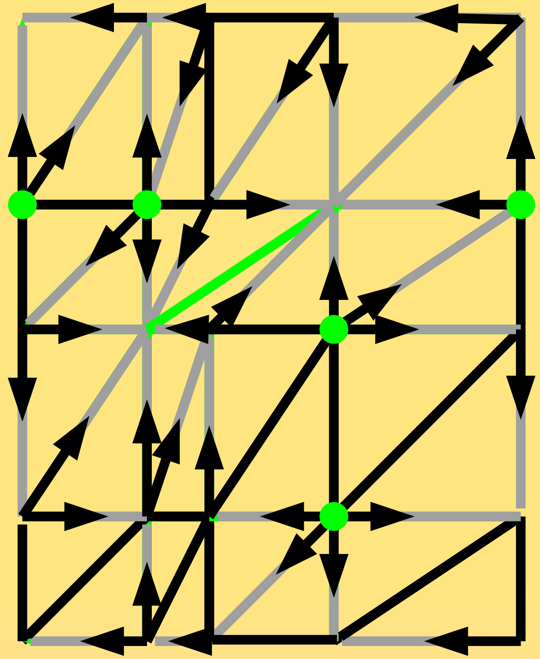
a new nice interpretation of Tutte's formula:

$$|\mathcal{T}_n| = \frac{2}{2n} \cdot |\mathcal{A}_n^{(2)}|.$$

Optimal coding (genus g)







$\mathcal{S}$

$\mathcal{T}^g$

triangulated graph of genus g

one face map of genu g

Acknowledgements

We are extremely grateful for enlightening discussions on Schnyder woods and combinatorics of graphs to many people (most of them being present here today)

O. Bernardi

N. Bonichon

G. Chapuy

E. Colin de Verdière

R. Cori

A. Labourel

C. Gavoille

H. de Fraysseix

P. Ossona de Mendez

D. Poulalhon

G. Schaeffer

P. Rosenstiehl

Thanks to all

Acknowledgements