Succinct representations of labeled graphs

Journées Géométrie et Informatique

14 juin 2007, Inria Sophia-Antipolis

INRIA SOPHIA ANTIPOLIS

Luca Castelli Aleardi

ÉCOLE POLYTECHNIQUE

Jeremy Barbay Meng He J. Ian Munro

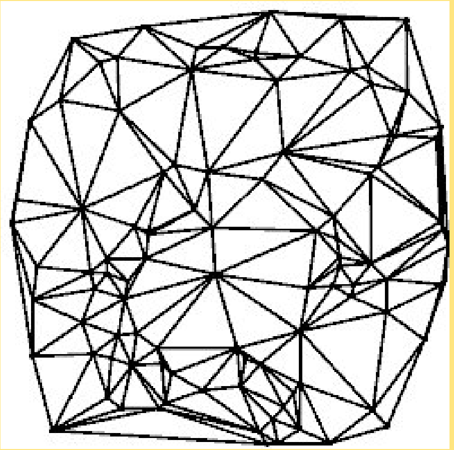
University of Waterloo

David R. Cheriton School of Computer Science

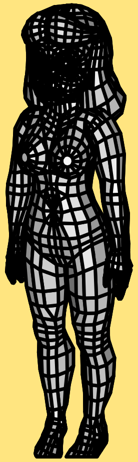
Domain and motivations

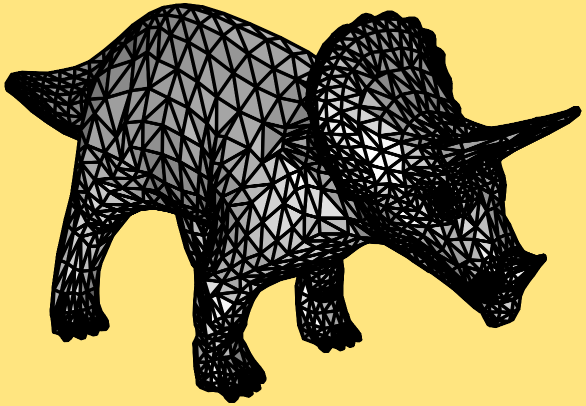
Geometric data

Triangulations and graphs

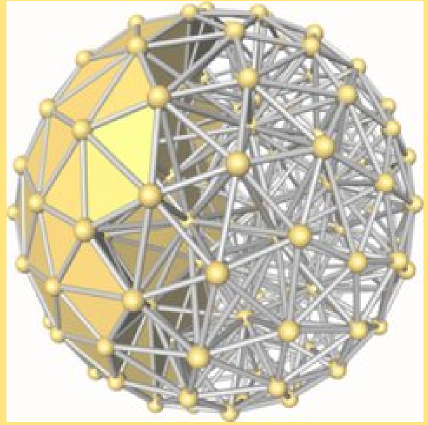


3D meshes



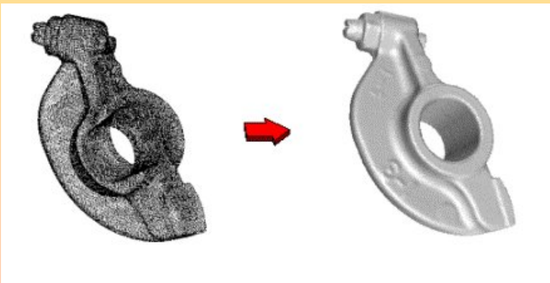


Tetrahedral Volume Meshes

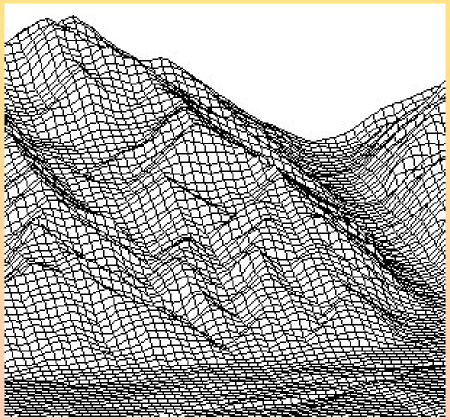


Applications

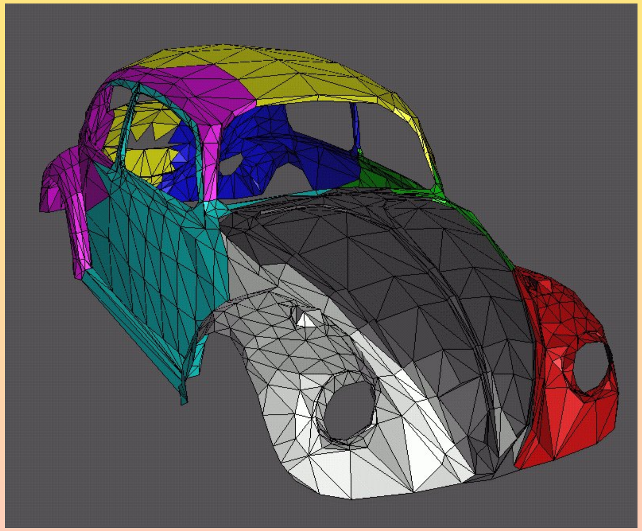
Surface recontruction



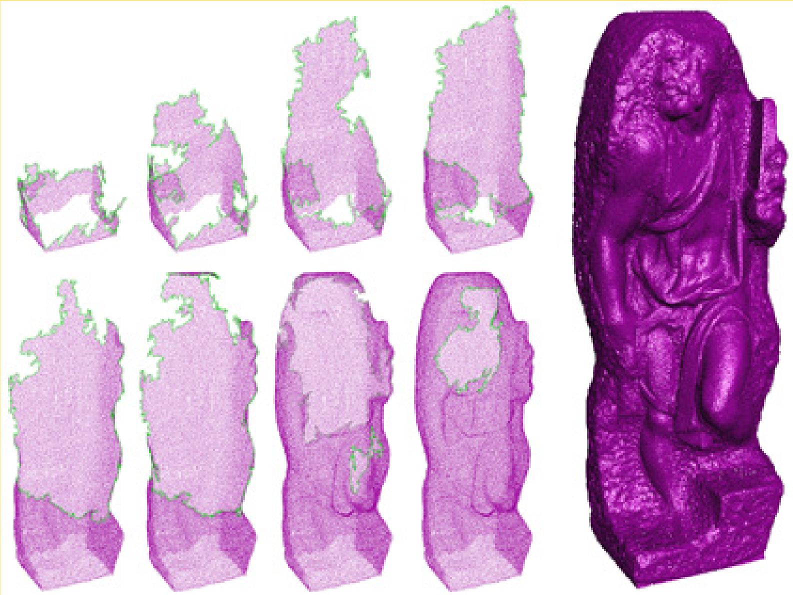
GIS Technology



Geometric modeling



Very large geometric data



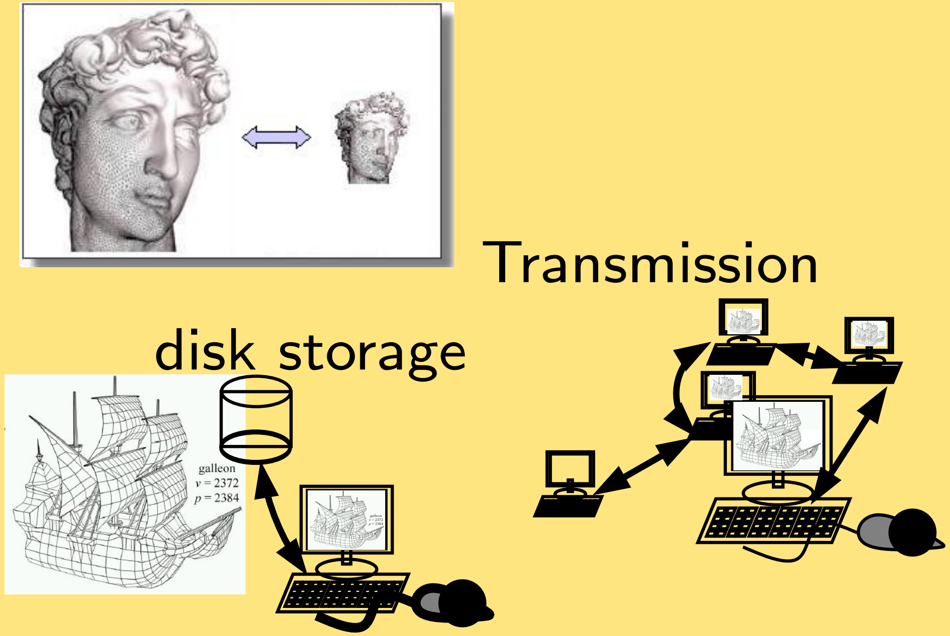
St. Matthew (Stanford's Digital Michelangelo Project, 2000)
186 millions vertices
6 Giga bytes (for storing on disk)
tens of minutes (for loading the model from disk)

David statue (Stanford's Digital Michelangelo Project, 2000)
2 billions polygons
32 Giga bytes (without compression)

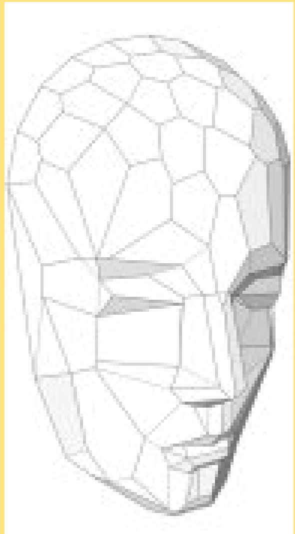
No existing algorithm nor data structure for dealing with the entire model

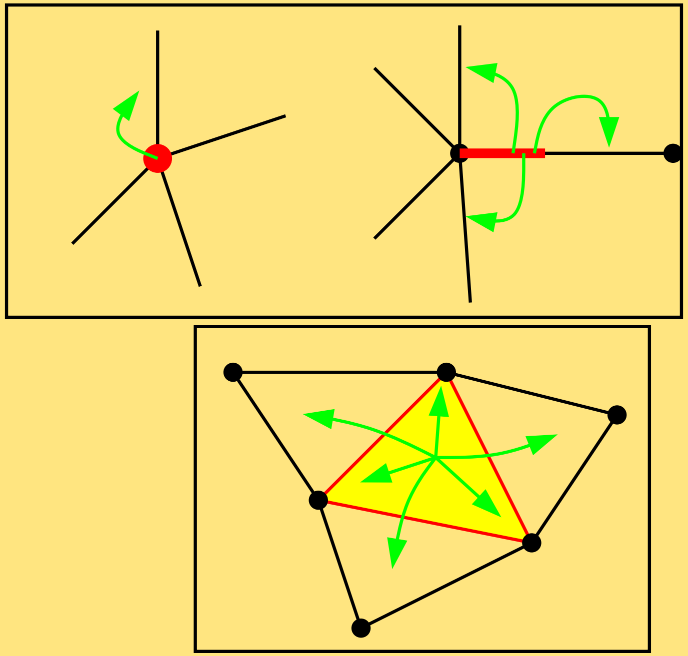
Research topics

Mesh compression

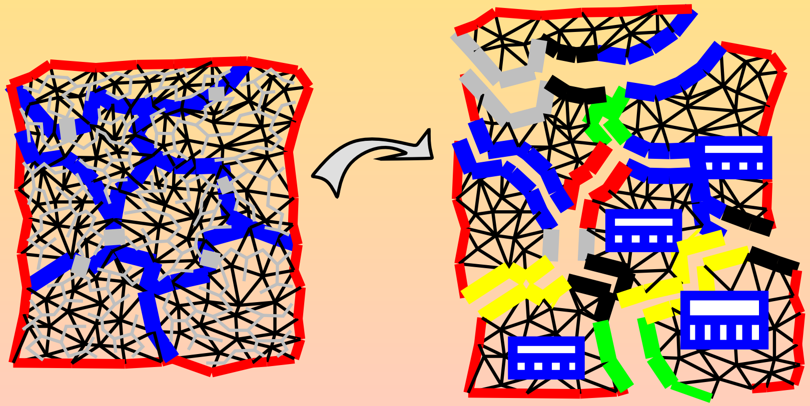


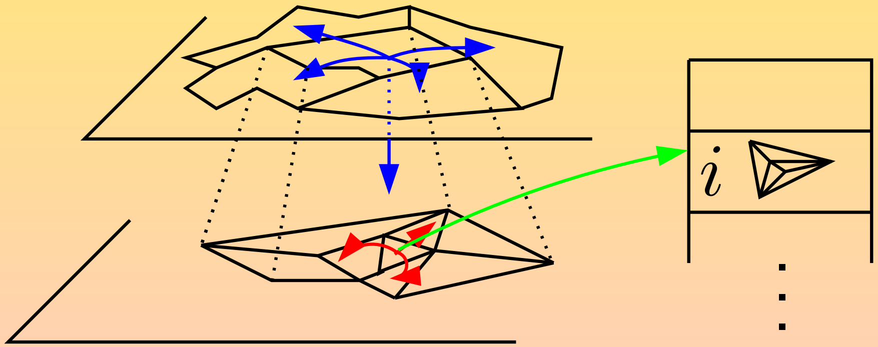
Geometric data structures



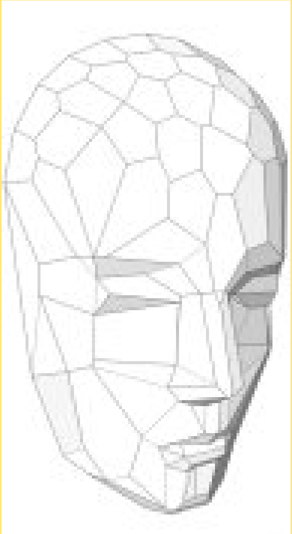
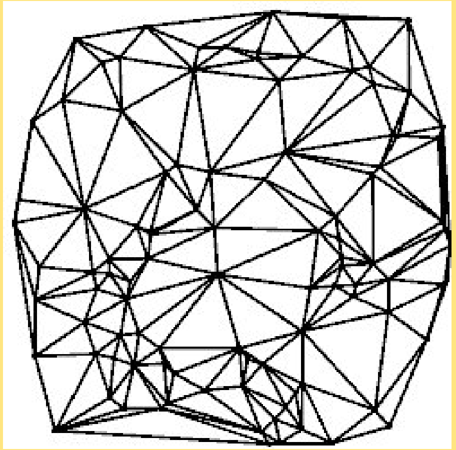
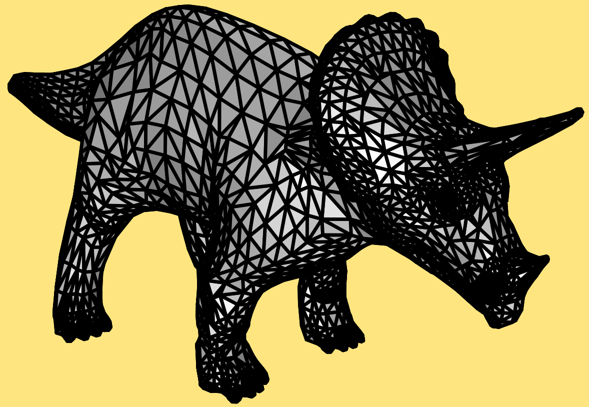


Compact representations of geometric data structures

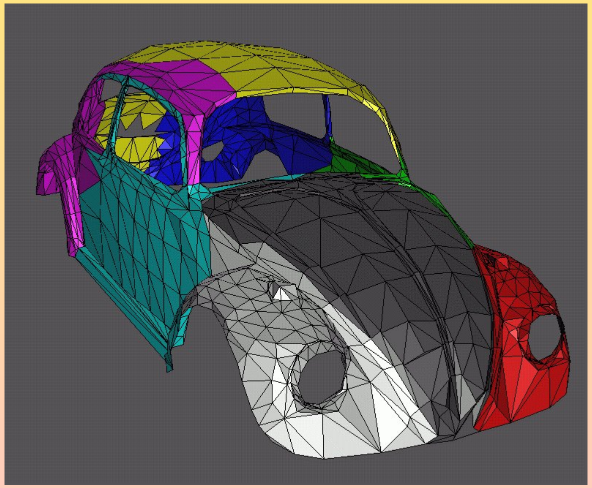
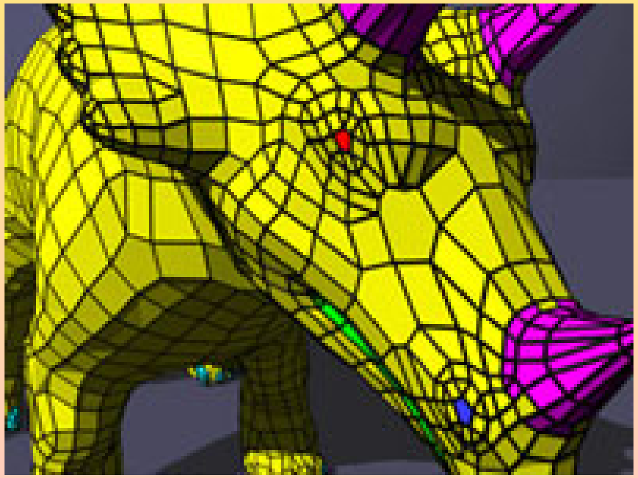




Unlabeled vs. labeled structures

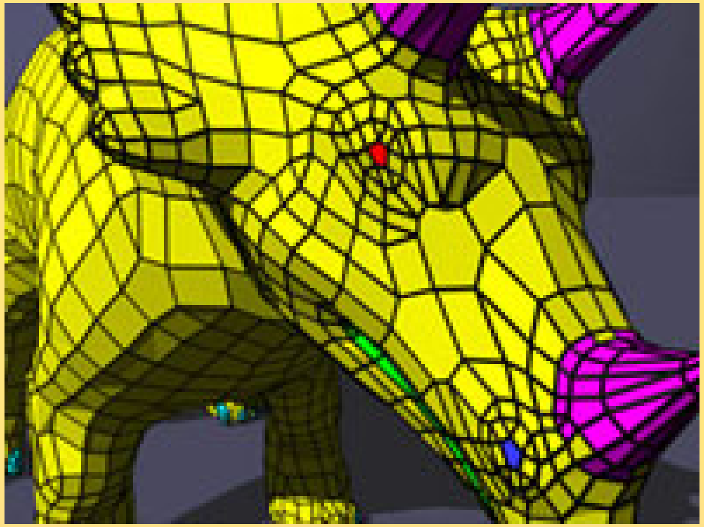
Unlabeled graphs, meshes with no properties

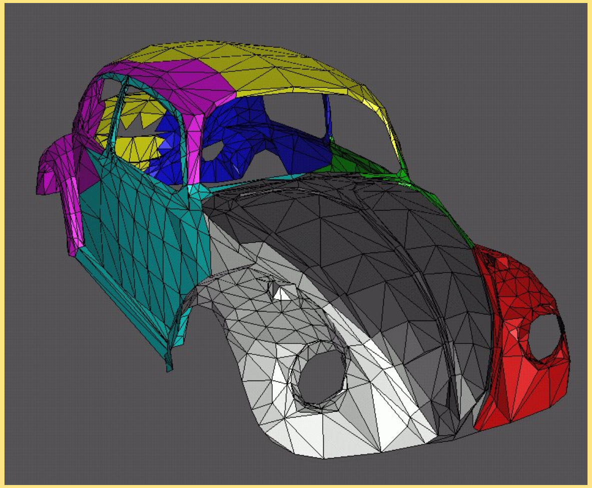



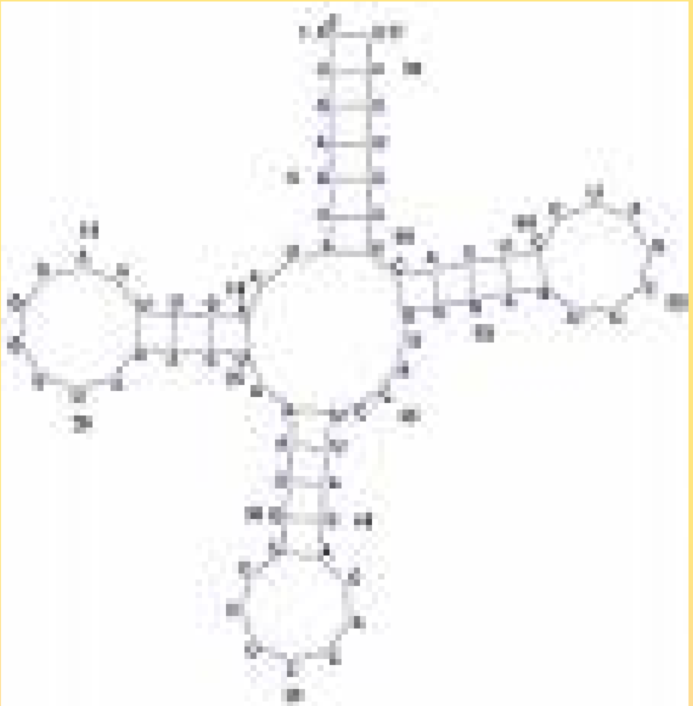
Labeled graphs
meshes with properties

Efficiently representing labeled structures

Goal: efficient dealing with additional attributes associated to elementary cells of the mesh



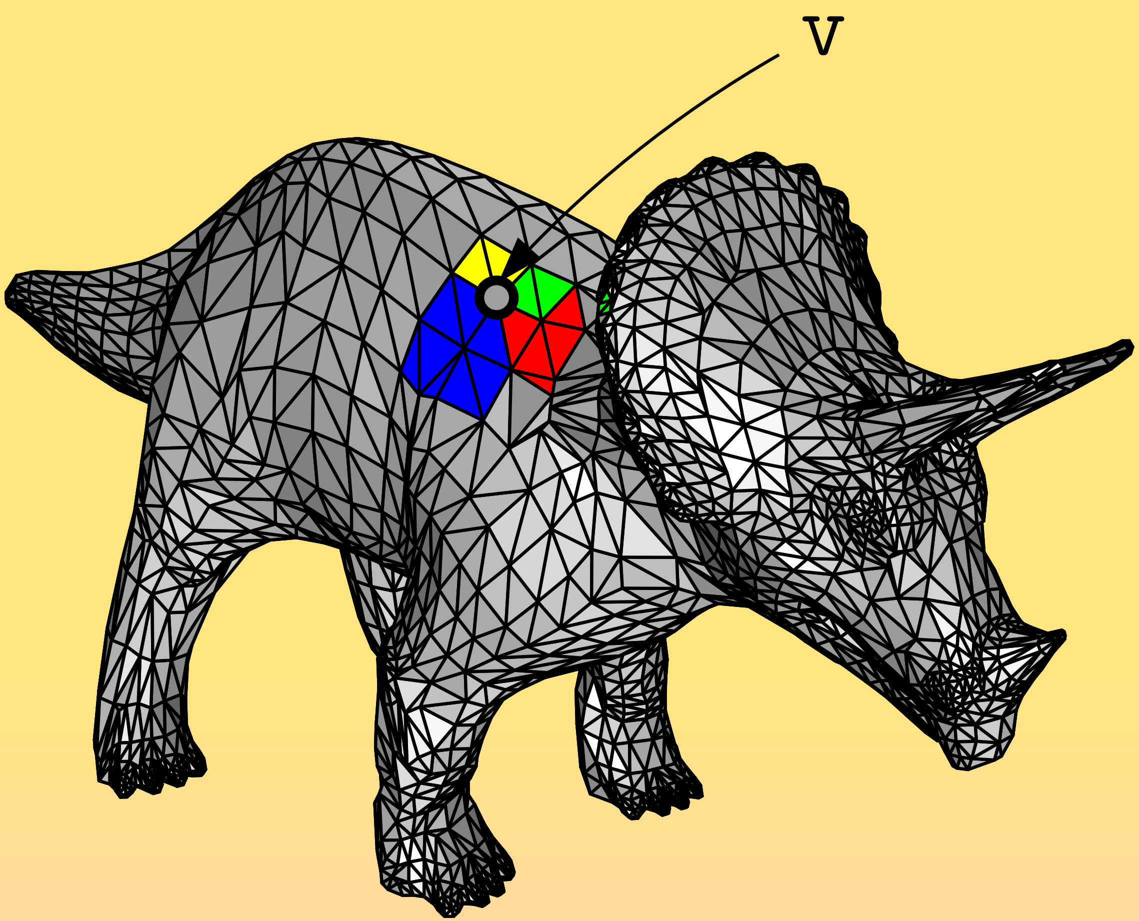




Example: vertex coordinates, face colors and normals, additional tags, ...

Efficiently representing labeled structures

Labeled based navigation operators



$\texttt{lab\_degree}(\alpha, \texttt{v})$

$\texttt{lab\_select}(\alpha, \texttt{v}, \texttt{i})$

$\texttt{lab\_rank}(\alpha, \texttt{v}, \texttt{w})$

$\texttt{lab\_degree}(\texttt{c}_1, \texttt{v}) = 2$

$\texttt{lab\_degree}(\texttt{c}_2, \texttt{v}) = 1$

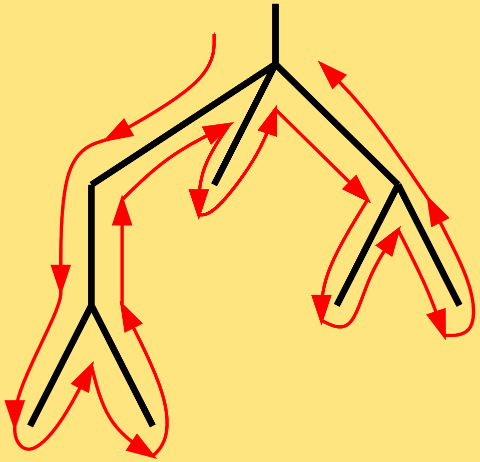
Problem: find adjacent cells with a given label

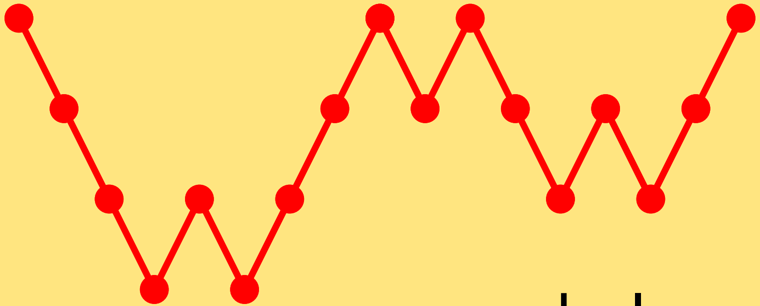
using $o(\lg \sigma)$ bits per label in almost $O(1)$ time

Compact representations

An example: (unlabeled) plane trees

ordered tree with n edges





balanced parenthesis word

$1\,1\,1\,0\,1\,0\,0\,0\,1\,0\,1\,1\,0\,1\,0\,0$

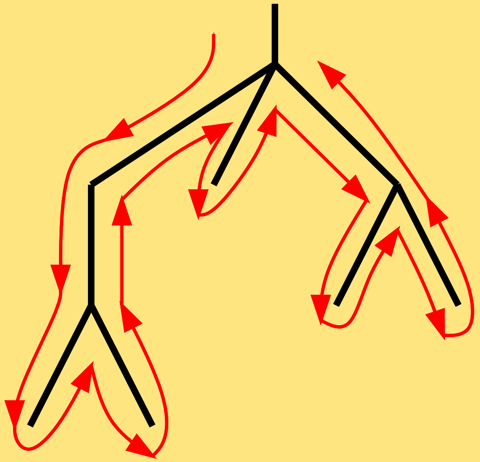
$\Rightarrow 2n$ bits for encoding a tree with n edges

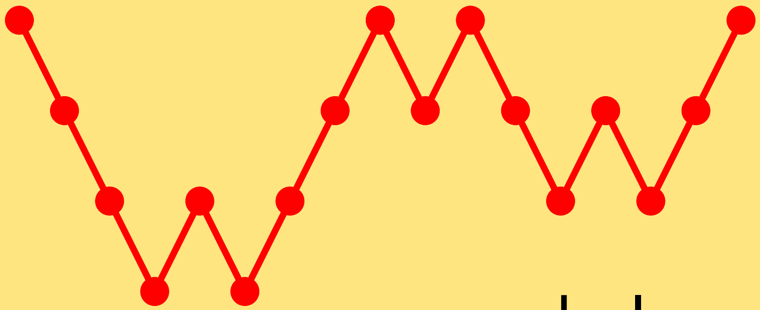
Enumeration of plane trees with n edges

$$\|\mathcal{B}_n\| = \frac{1}{n+1}\binom{2n}{n} \approx 2^{2n} n^{-\frac{3}{2}}$$

An example: (unlabeled) plane trees

ordered tree with n edges





balanced parenthesis word

$1\,1\,1\,0\,1\,0\,0\,0\,1\,0\,1\,1\,0\,1\,0\,0$

$\Rightarrow 2n$ bits for encoding a tree with n edges

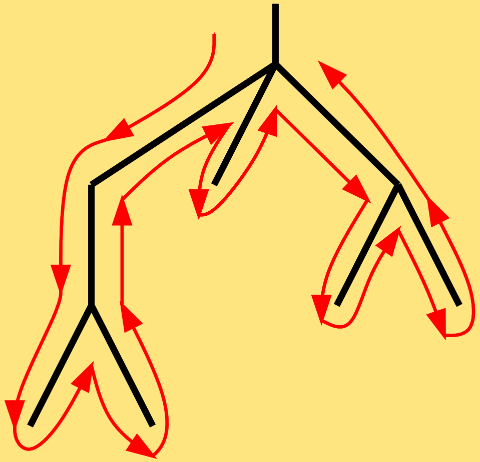
Asymptotic optimal encoding

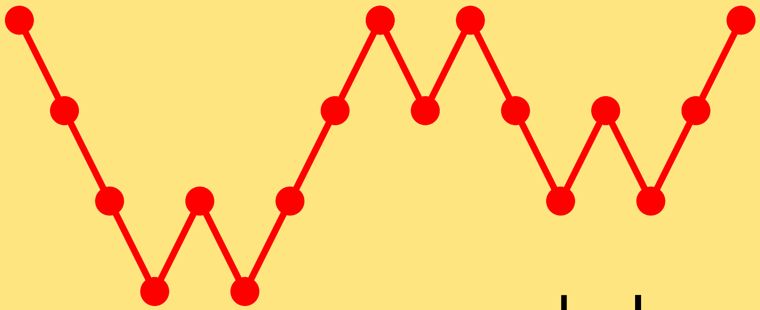
- the cost of an object matches asymptotically the entropy

$$log_2 \|\mathcal{B}_n\| = 2n + O(\lg n)$$

An example: (unlabeled) plane trees

ordered tree with n edges





balanced parenthesis word

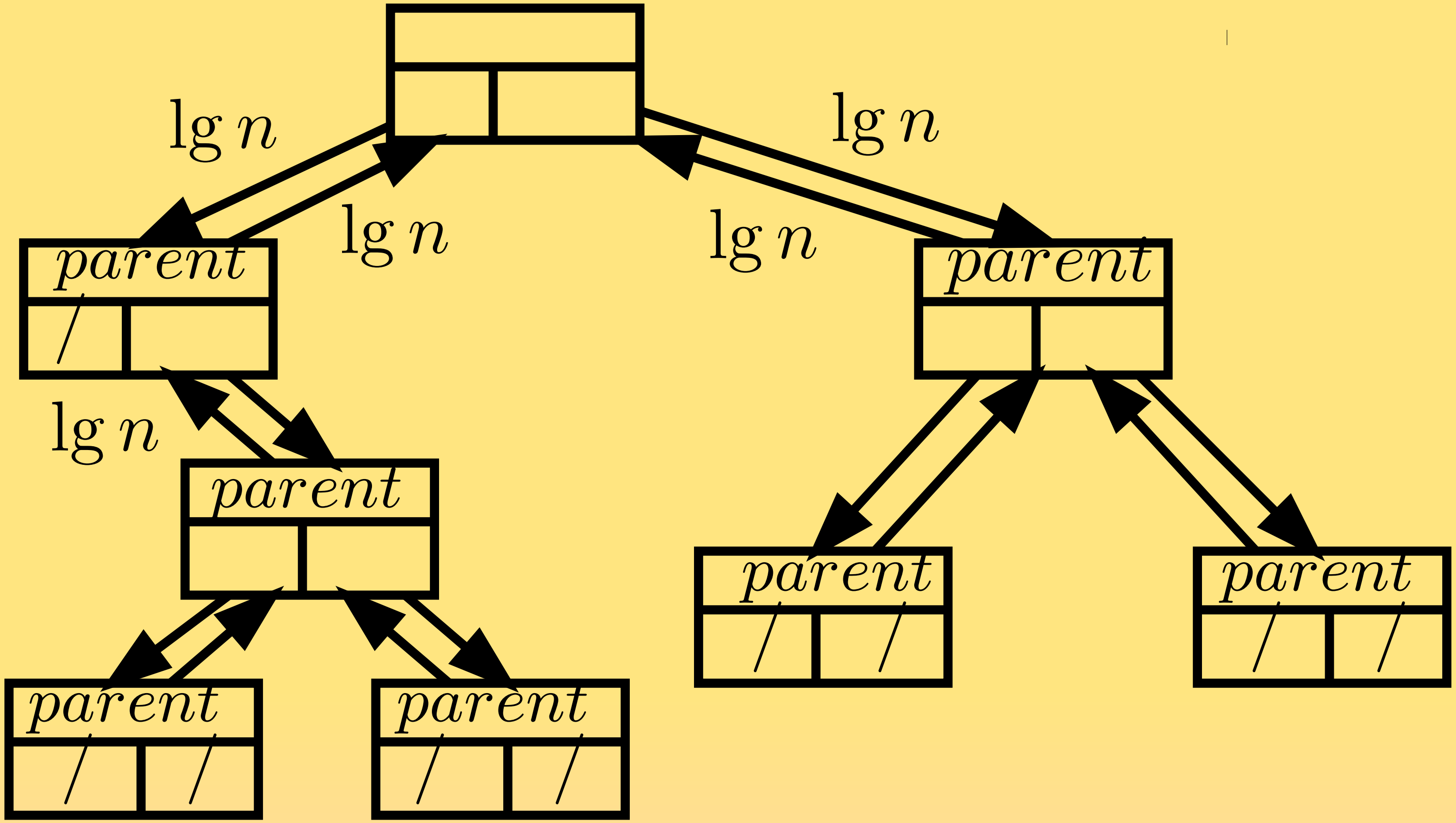
$1\,1\,1\,0\,1\,0\,0\,0\,1\,0\,1\,1\,0\,1\,0\,0$

$\Rightarrow 2n$ bits for encoding a tree with n edges

No efficient implementation of local adjacency queries

An example: (unlabeled) plane trees

Explicit pointers based representation



adjacency queries between vertices in $O(1)$ time

not optimal encoding: we need $\Theta(n \lg n)$ bits

Could we do better?

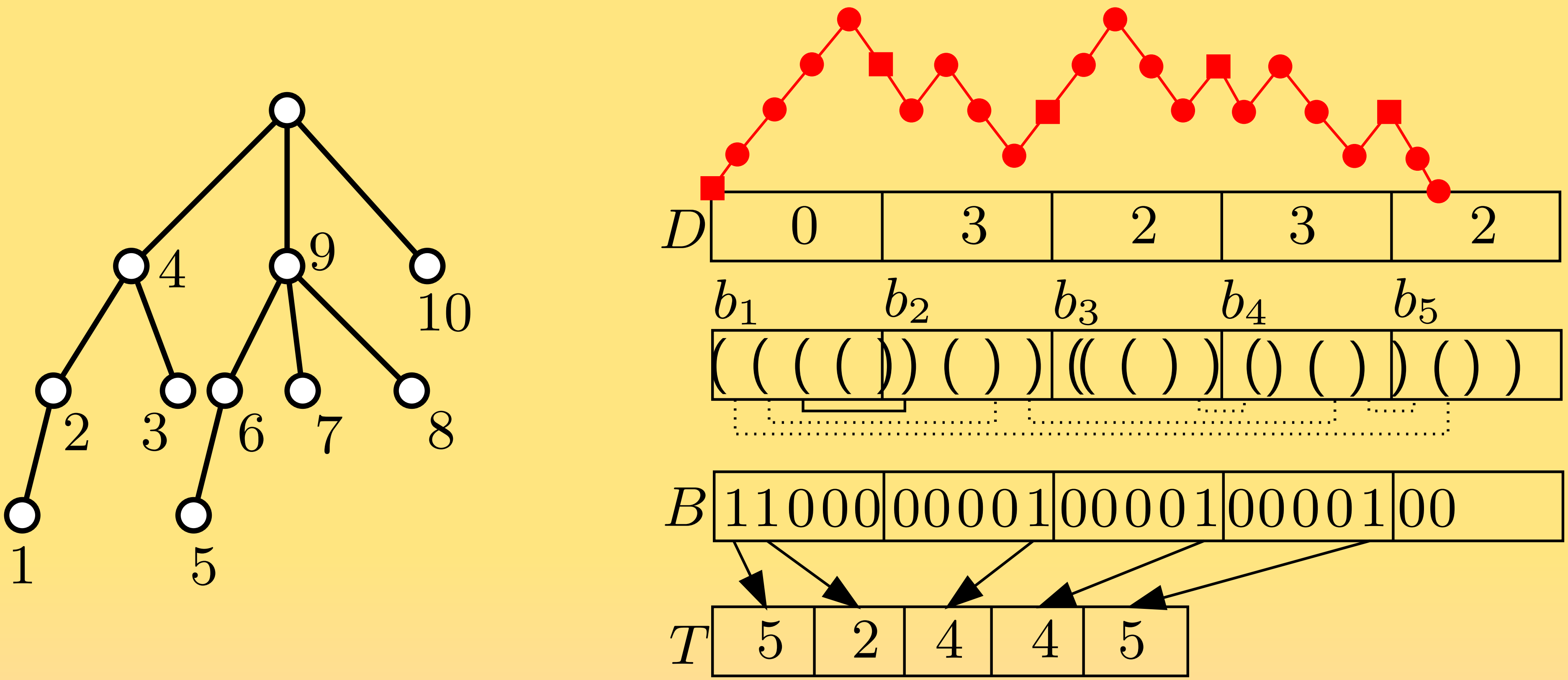
a compact encoding (asymptotically optimal)

testing adjacency queries efficiently (in $O(1)$ time)

An example: binary and ordered trees

(Jacobson, Focs89, Munro et Raman Focs97)

For trees and parenthesis words the anser is... YES



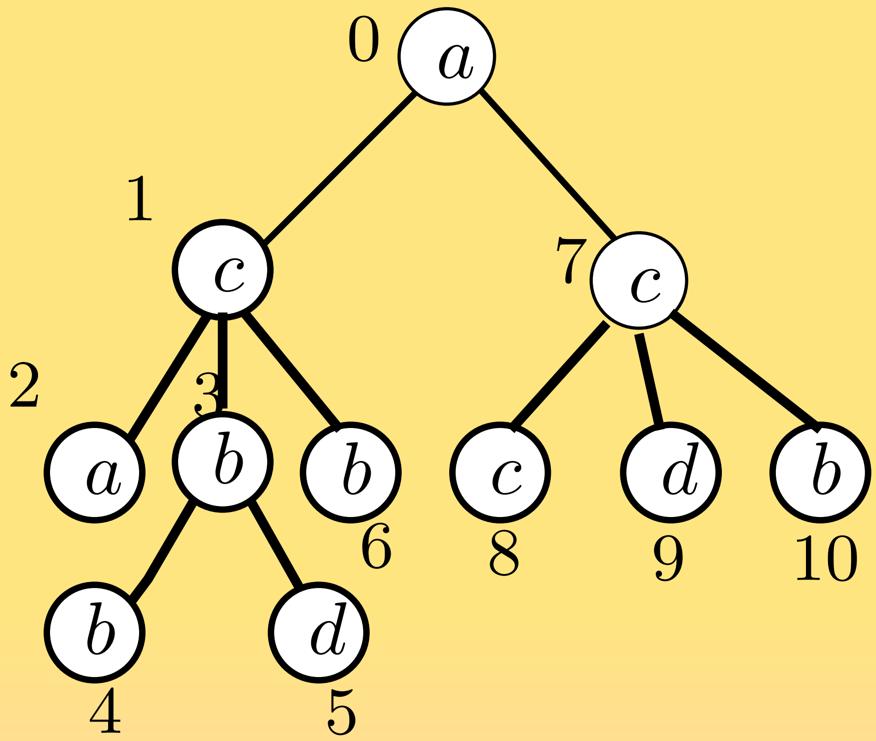
it is possible to test adjacency between vertices in $O(1)$ time
with the guarantee the the encoding is still asymptotically optimal

$2n + o(n)$ bits are sufficient

And for labeled trees?

Labeled ordinal trees

(Barbay et al. '06)



$\texttt{lab\_degree}(\alpha, \texttt{v})$

$\texttt{lab\_child}(\alpha, \texttt{v}, \texttt{i})$

$\texttt{lab\_rank}(\alpha, \texttt{v}, \texttt{w})$

$\texttt{lab\_child}(\texttt{d}, 7, 1) = 9$

$\texttt{lab\_degree}(\texttt{b}, 1) = 2$

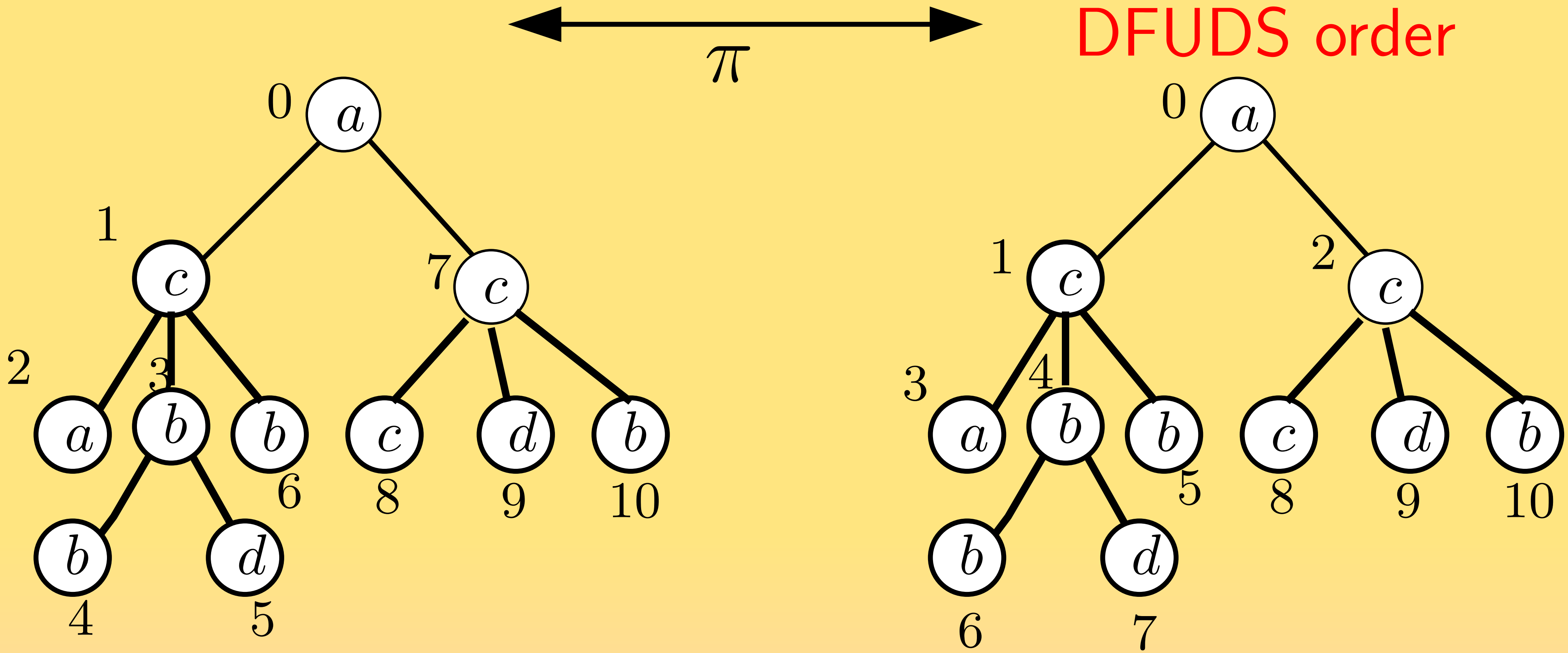
Labeled ordinal trees (Barbay et al. '06)

Prefix order

Descendants are listed consecutively

Children are listed consecutively

DFUDS order



S_1										
a	c	a	b	b	b	d	c	c	d	b

S_2										
a	c	c	a	b	b	b	b	c	d	b

Labeled ordinal trees (Barbay et al. '06)

Succinct encodings of strings (Golynski et al. '06)

S_1	a	c	a	b	b	b	c	c	d	b

$\mathtt{string\_access}(4) = \mathtt{b}$

$\mathtt{string\_rank}(\mathtt{b}, 9) = 3$

$\mathtt{string\_select}(\alpha, \mathtt{c}, 2) = 7$

Space cost: $n(\lg \sigma + o(\lg \sigma))$

Time complexity: $O(\lg \lg \sigma)$

Prefix order $\xleftrightarrow[\pi]{\text{in } O(1) \text{ time}}$ DFUDS order

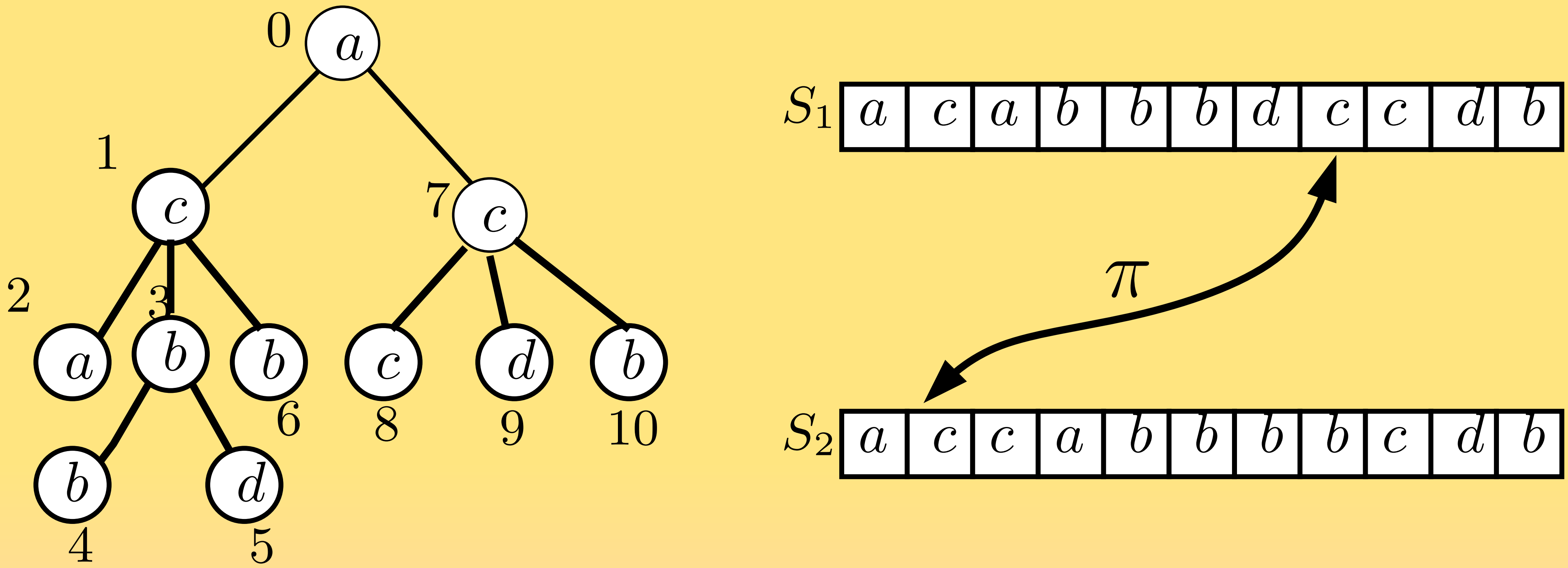
S_1	a	c	a	b	b	b	d	c	c	d	b

S_2	a	c	c	a	b	b	b	b	c	d	b

Labeled ordinal trees (Barbay et al. '06)

Navigation

Prefix order



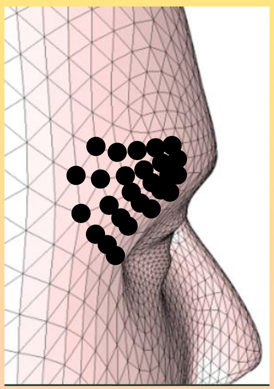
$\texttt{lab\_child}(\texttt{d}, 7, 1) = 9 \longleftrightarrow$ $\texttt{string\_rank}(\ldots)$ $\texttt{string\_select}(\ldots)$

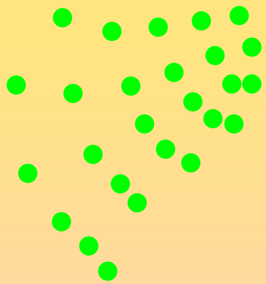
Geometric data

Which information?

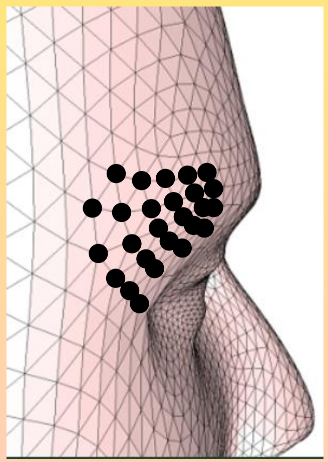
Geometry and connectivity

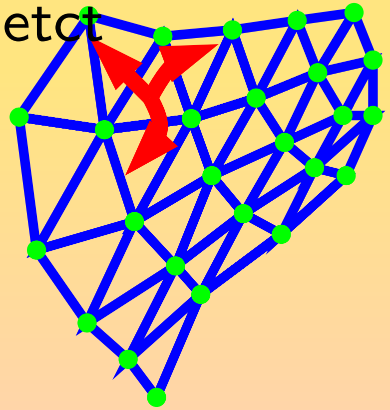
Geometric object





Combinatorial objetct

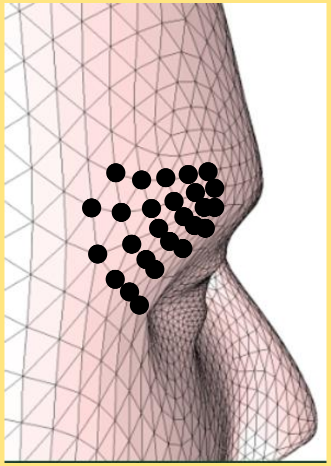


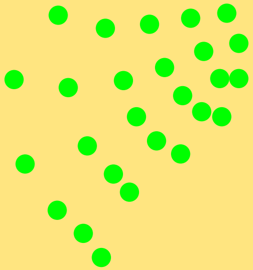


Geometric information

Geometric information

Geometric object





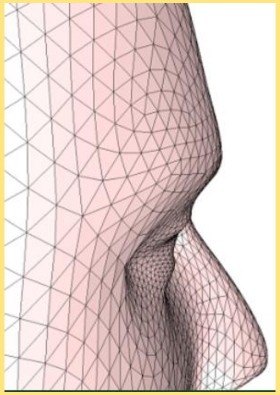
$$\begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

between 30 et 96 bits/vertex

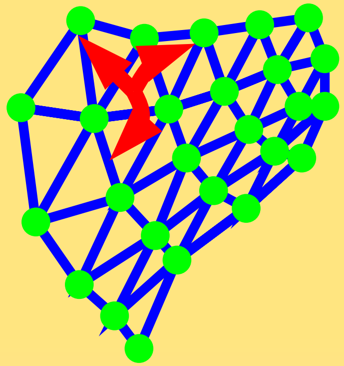
Connectivity information

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

Combinatorial object



$\log n$ ou 32 bits



$$2 \times n \times 6 \times \log n$$
$$n \times 1 \times \log n$$
$$13n \log n$$

$416n$ bits connectivity

3D triangle meshes

simple compression

VRML, $\boxed{\text{288 or 114 bits/vertex}}$

[Edgebreaker] $\boxed{\text{3.67bits/vertex}}$ (guaranteed upper bound)

[Poulalhon Schaeffer] $\boxed{\text{3.24bits/vertex}}$ (optimal)

[Touma Gotsman] $\approx\boxed{\text{2bits/vertex}}$ (heuristic)

PS $110100011000001001000001100100000000$

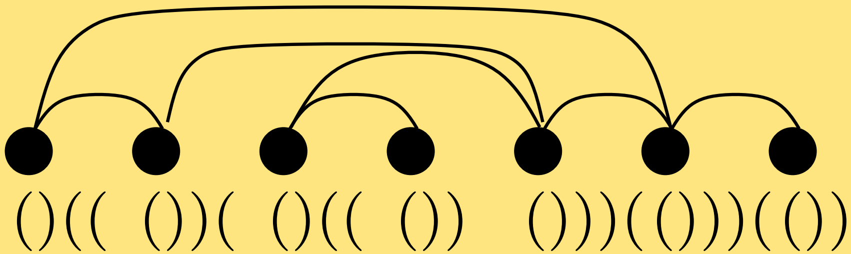
TG $V_5V_5V_6V_5V_4V_5V_8V_5V_5V_4S_4V_3V_4$

EB $CCCRCCRCCRECRRELCRE$

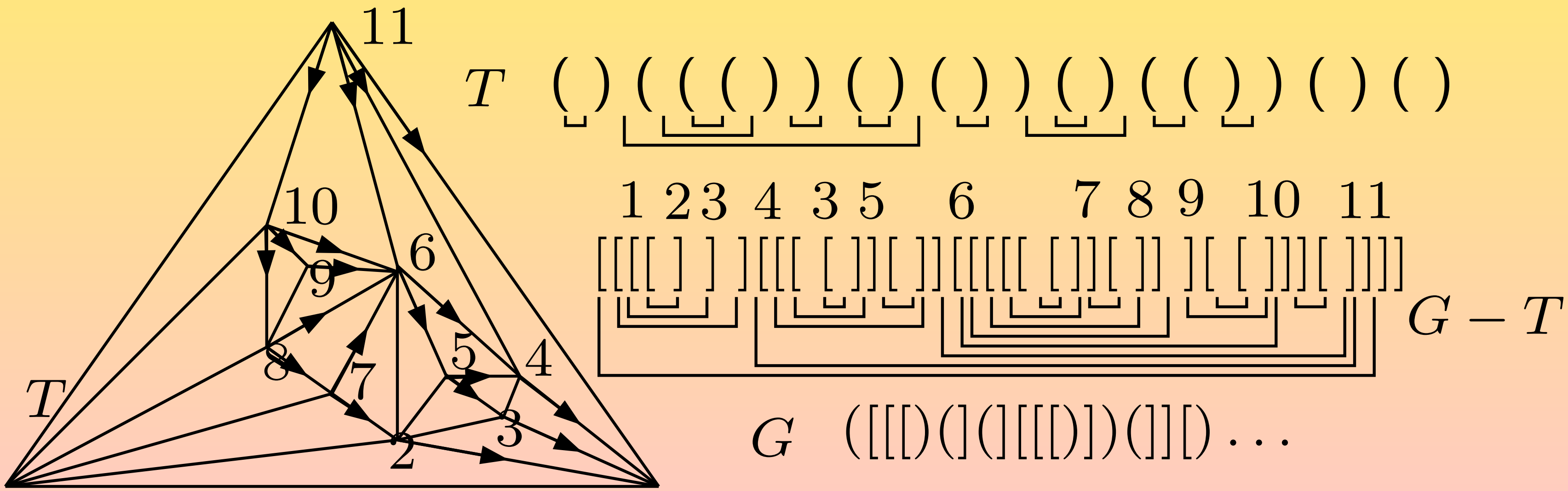
implicit encodings

Succinct encodings of unlabeled graphs: existing works

planar graphs: book embeddings and canonical orderings



Codage	3-conn.	triang.
Jacobson	$64n$	$64n$
Munro Raman	$8n + 2e$	$7m$
Chuang et al.	$2e + 2n$	$3.5m$
Chiang et al.	$2e + 2n$	$4m$
Blandford et al.	$O(n)$	$O(m)$
Castelli et al.	$2e$	$1.62m$



$\boxed{1.62m}$

Optimal bound

Existing works

Unlabeled case — Mesh compression algorithms — Labeled case

[Edgebreaker] 1.84bits/face

[Poulalhon Schaeffer] 1.62bits/face

[Touma Gotsman] ≈1bits/face

[Facefixer] Isenburg and Snoeyink

Unlabeled case — Compact and succinct representations

Codage	3-conn.	triang.
Jacobson	$64n$	$64n$
Munro Raman	$8n+2e$	$7m$
Chuang et al.	$2e+2n$	$3.5m$
Chiang et al.	$2e+2n$	$4m$
Blandford et al.	$O(n)$	$O(m)$
Castelli et al.	$2e$	$1.62m$

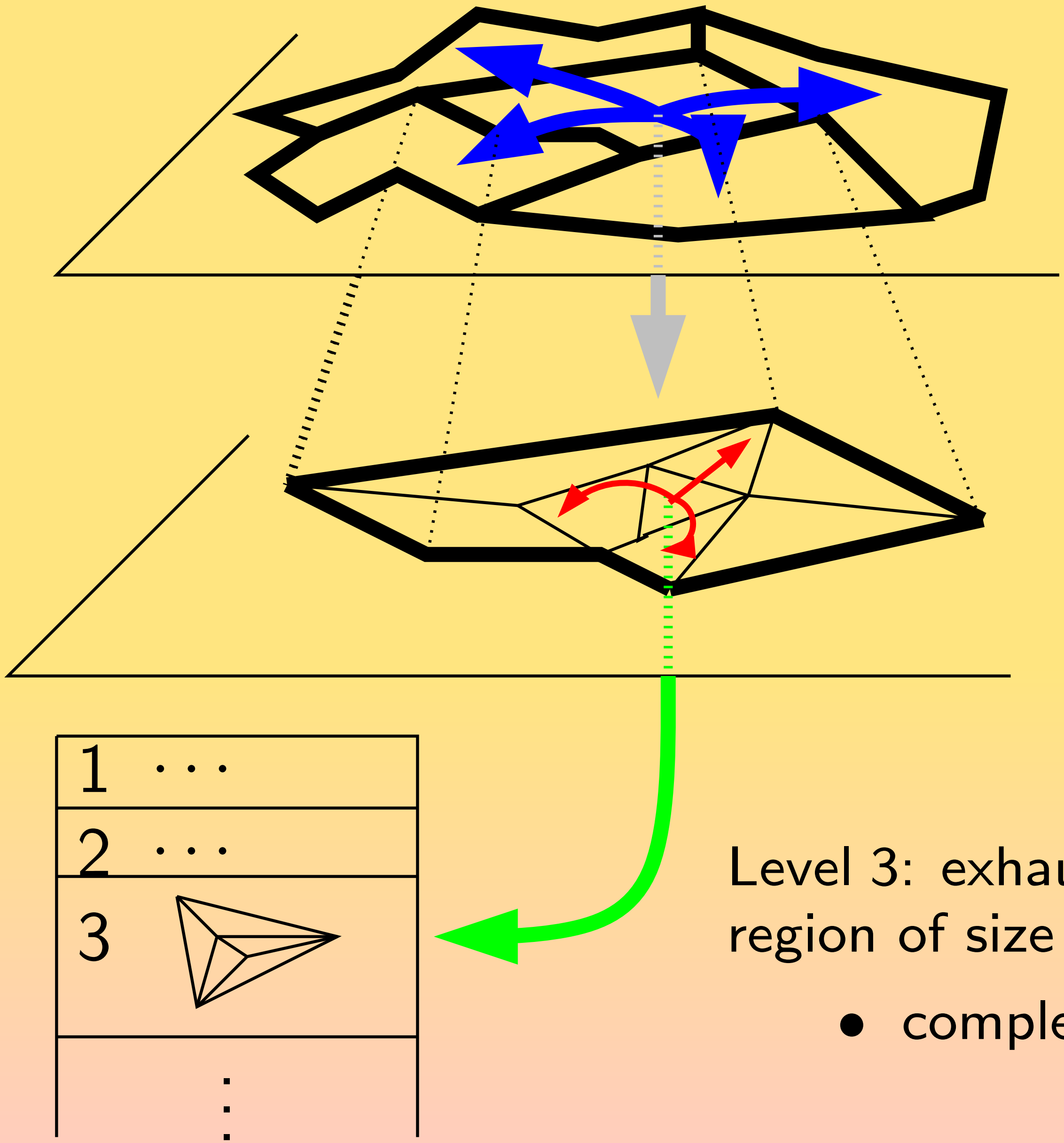
Labeled case

—

Our solution for succinctly representing graphs

Our scheme

Overview of the hierarchical structure



Level 1:

- $\Theta(\frac{n}{\log^2 n})$ regions of size $\Theta(\log^2 n)$, represented by pointers to level 2

Level 2:
in each of the $\frac{n}{\log^2 n}$ regions

- $\Theta(\log n)$ regions of size $C \log n$, represented by pointers to level 3

Level 3: exhaustive catalog of all different region of size $i < C \log n$:

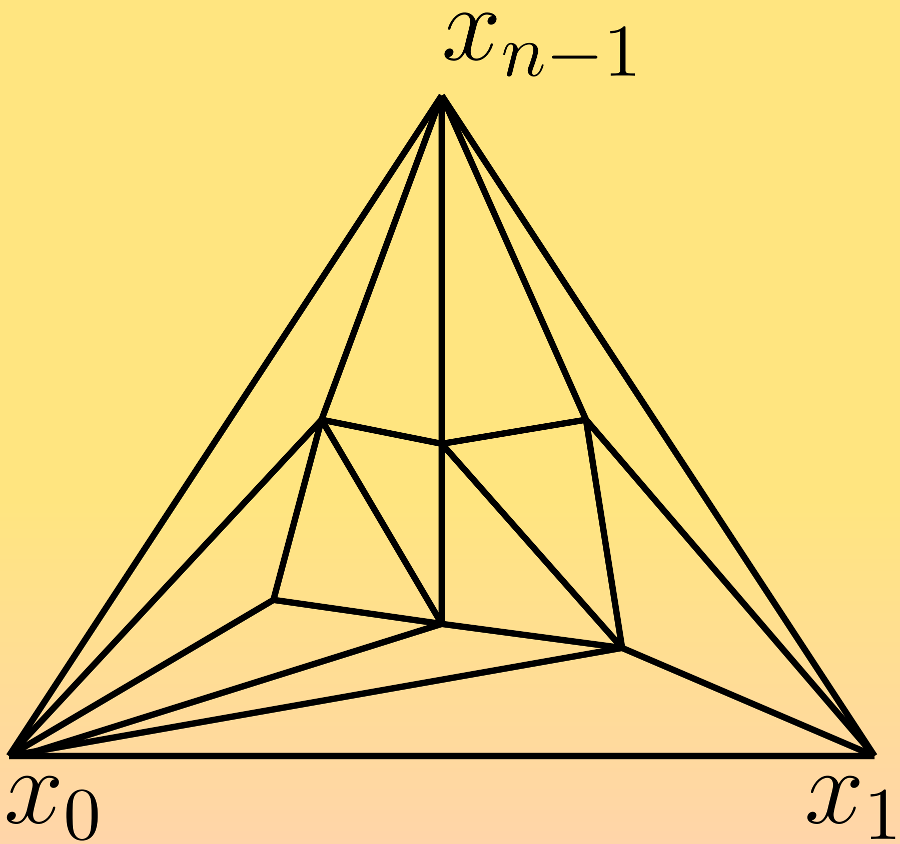
- complete explicit representation.

Our solution for representing labeled graphs

A nice characterization of planar graphs

Realizer of a planar triangulation (Schnyder '90)

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

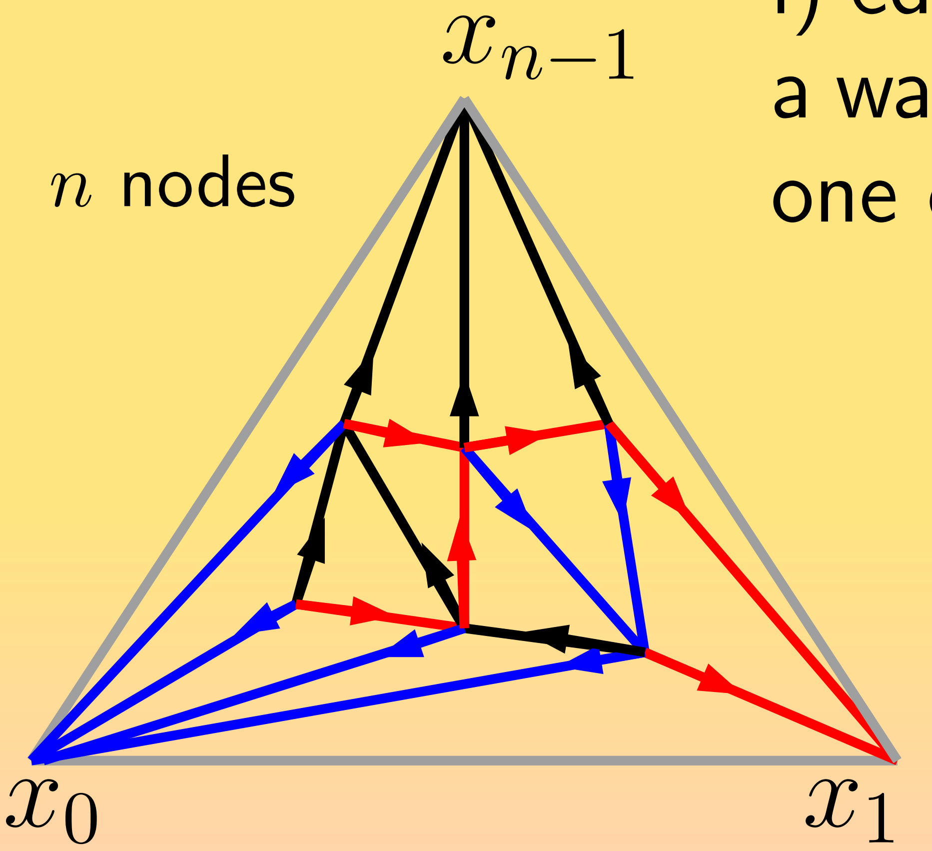


A nice characterization of planar graphs

Realizer of a planar triangulation (Schnyder '90)

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

n nodes

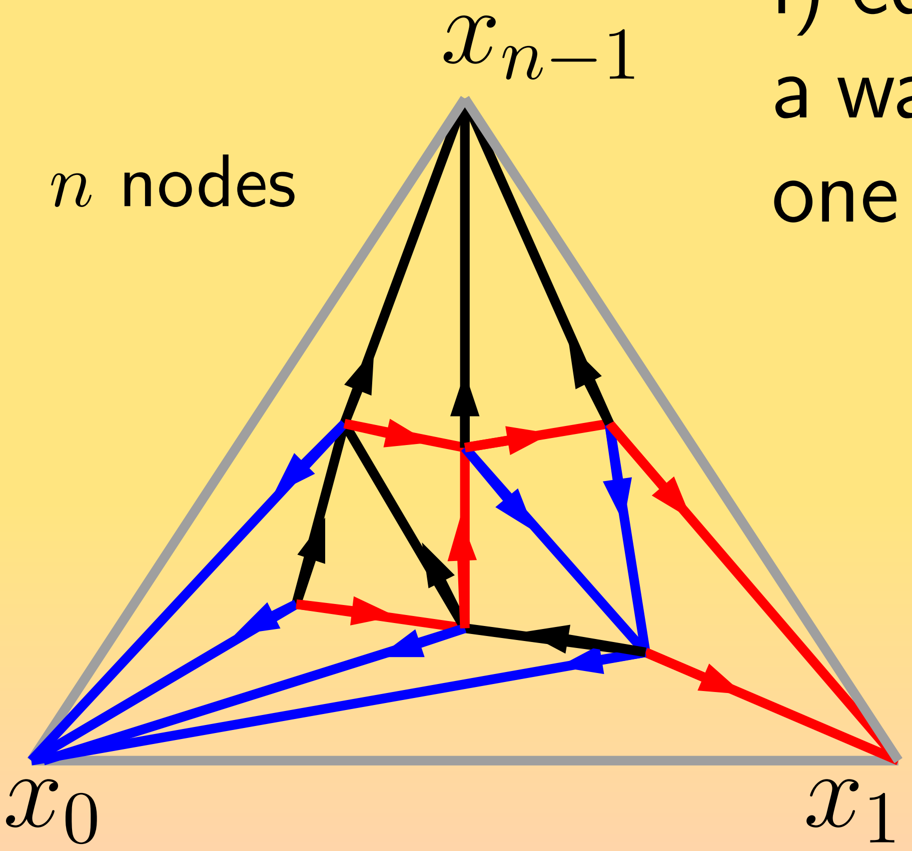


i) edge are colored and oriented in such a way that each inner nodes has exaclty one outgoing edge of each color

A nice characterization of planar graphs

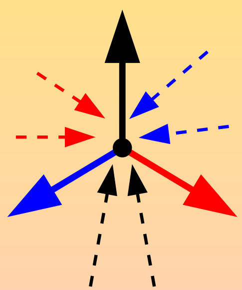
Realizer of a planar triangulation (Schnyder '90)

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 nodes has exaclty one outgoing edge of each color

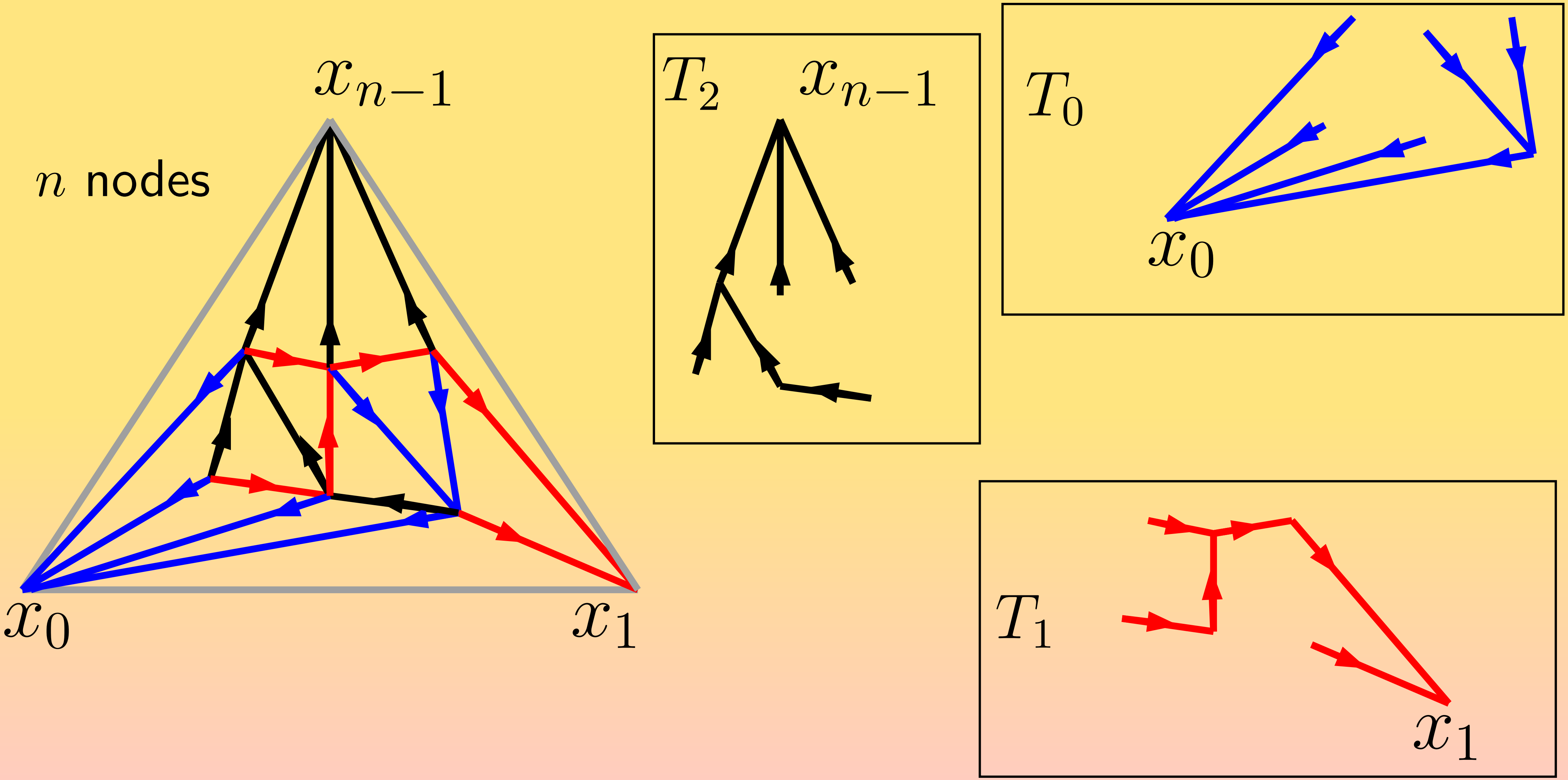
ii) colors and orientations must respect the local Schnyder condition



A nice characterization of planar graphs

Realizer of a planar triangulation (Schnyder '90)

T_0, T_1, T_2 are vertex spanning trees of T:



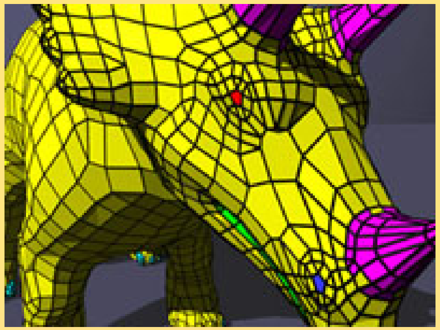
Succinct labeled triangulations

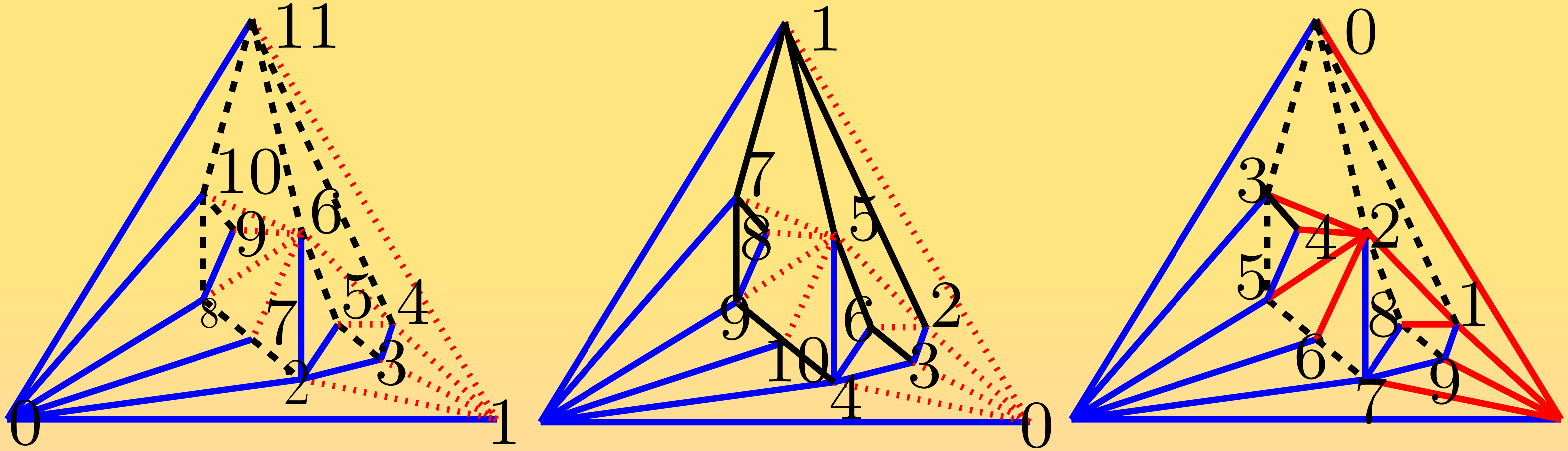
Main idea: find good orders on the vertices of a graph

Our solution: general overview

Main idea: find good orders on the vertices of a graph

Solution | Combinatorial properties of planar graphs | + | rank/select on strings |





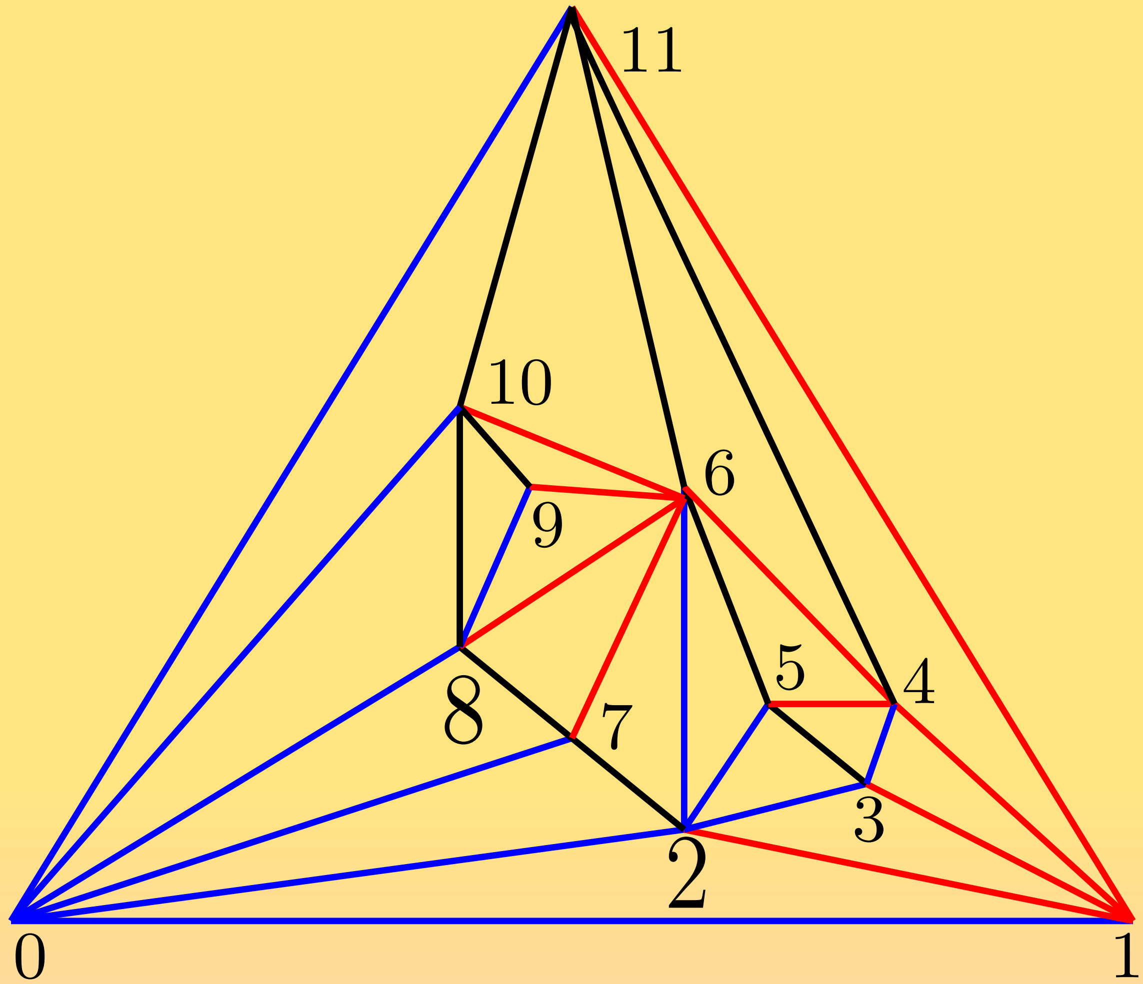
S (([[[[ ) ( ] ( ] ( ] { [[ ) {) (}] {) (}] { [[[[ ) {) (}] {) (}] ( ] {) {) (} }] {) (} } }]))

0 1 1 2 3 4 4 3 5 6 7 8 9 10 11

$$\pi_1 \begin{bmatrix} 11 & 4 & 3 & 2 & 6 & 5 & 10 & 9 & 8 & 7 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \end{bmatrix} \qquad \pi_2 \begin{bmatrix} 4 & 6 & 10 & 9 & 8 & 7 & 2 & 5 & 3 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \end{bmatrix}$$

Encoding the triangulation

Multiple parenthesis string



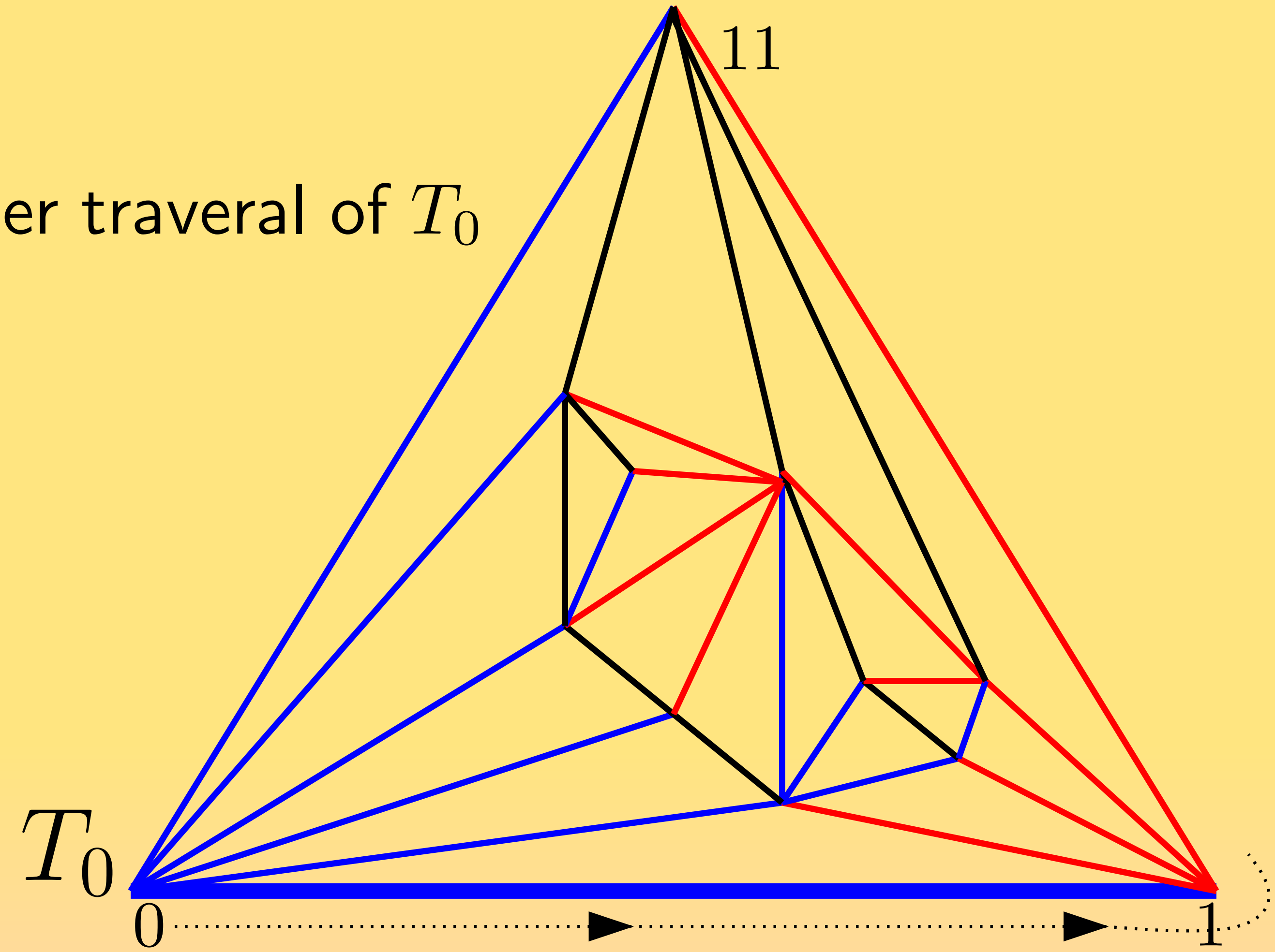
S $(([[[[)(](](]\{[[)\{)(\}]\{)(\}]\{[[[[)\{)(\}]\{)(\}](]\{)\{)(\}\}]\{)(\}\}\}]))$

01 12 3 4 4 35 6 7 8 9 10 11

Encoding the triangulation

Constructing the parenthesis representation

ccw preorder traveral of T_0



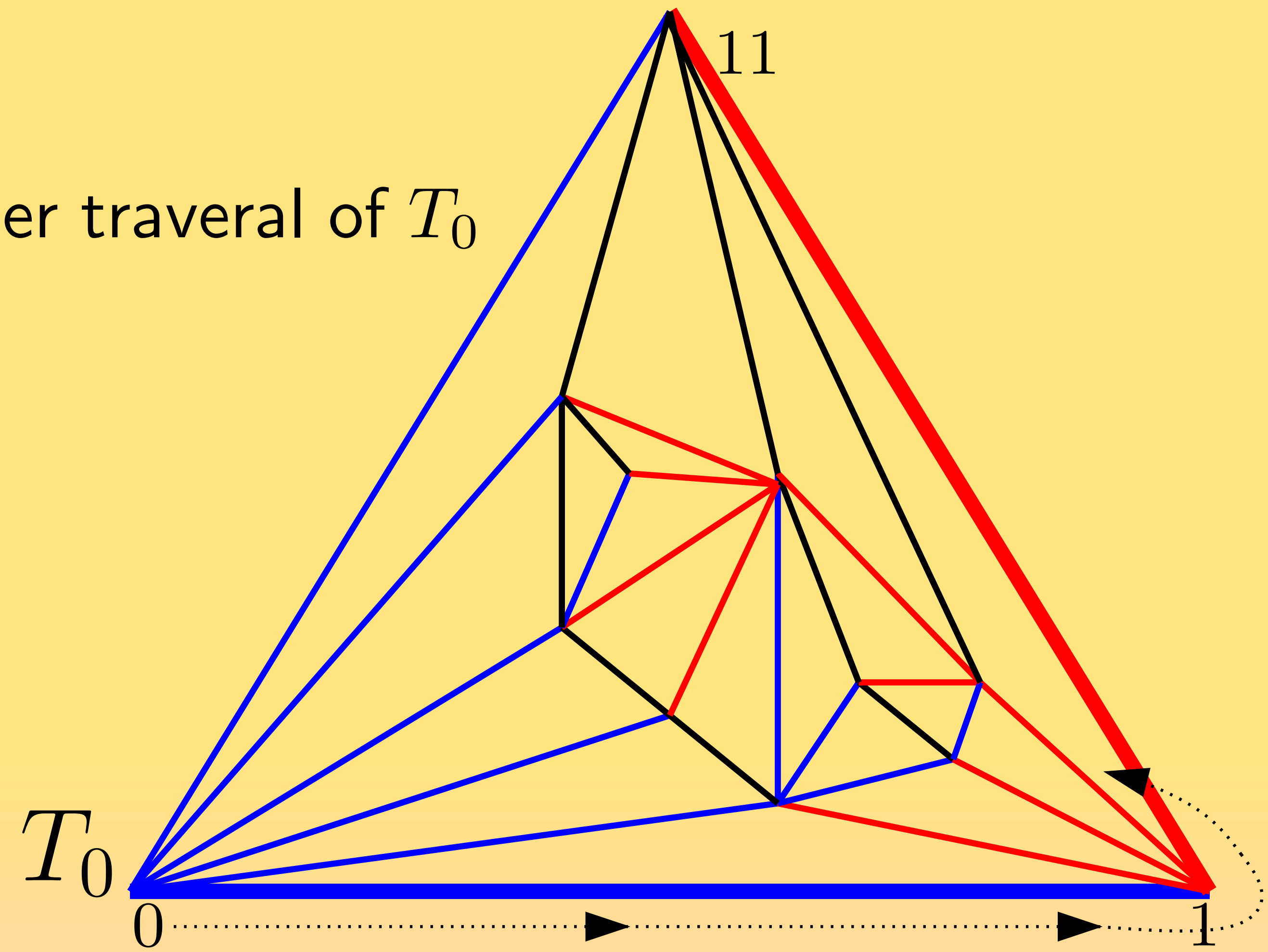
S $(($

01

Encoding the triangulation

Constructing the parenthesis representation

ccw preorder traveral of T_0



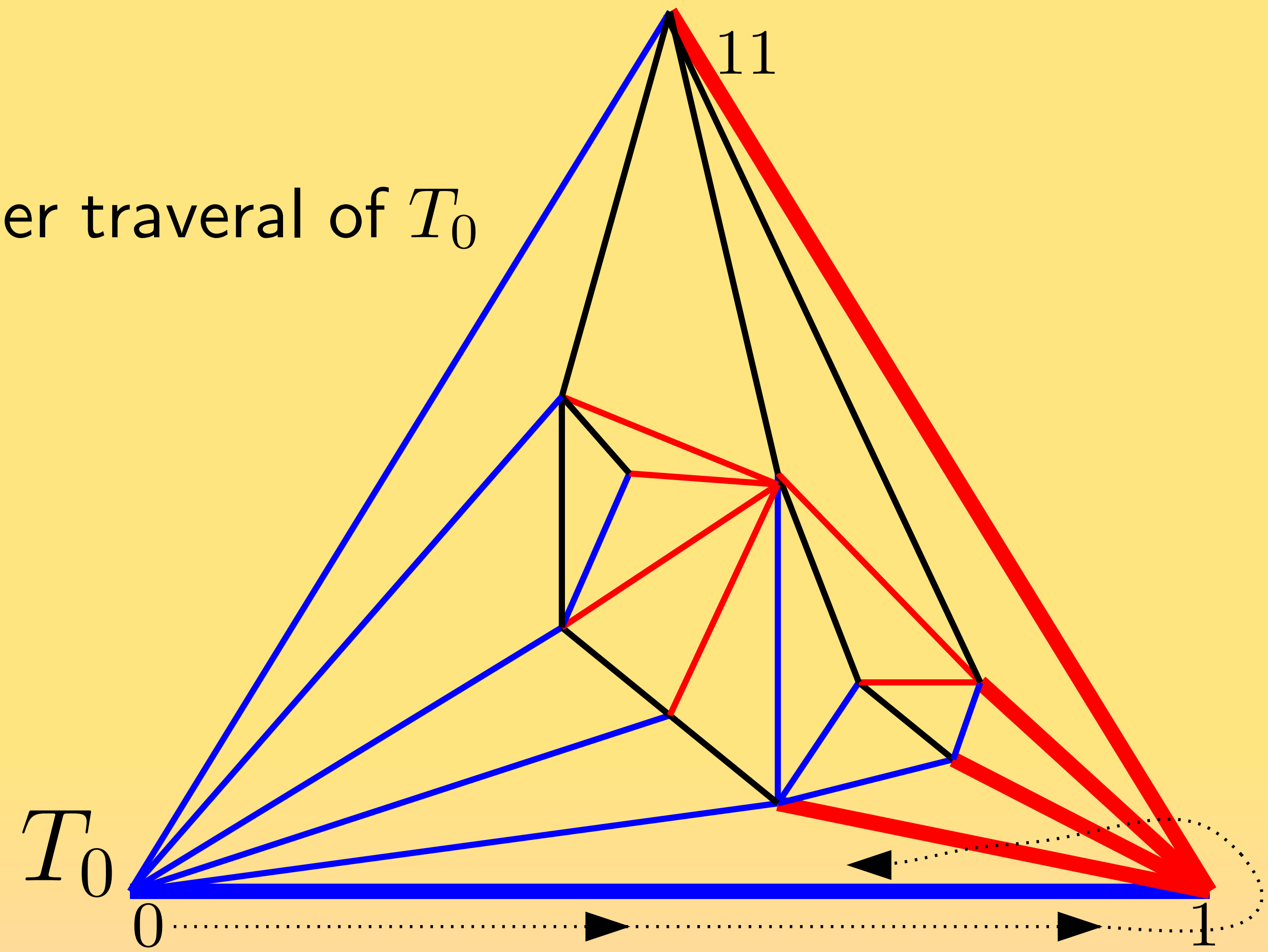
S $(([$

01

Encoding the triangulation

Constructing the parenthesis representation

ccw preorder traveral of T_0



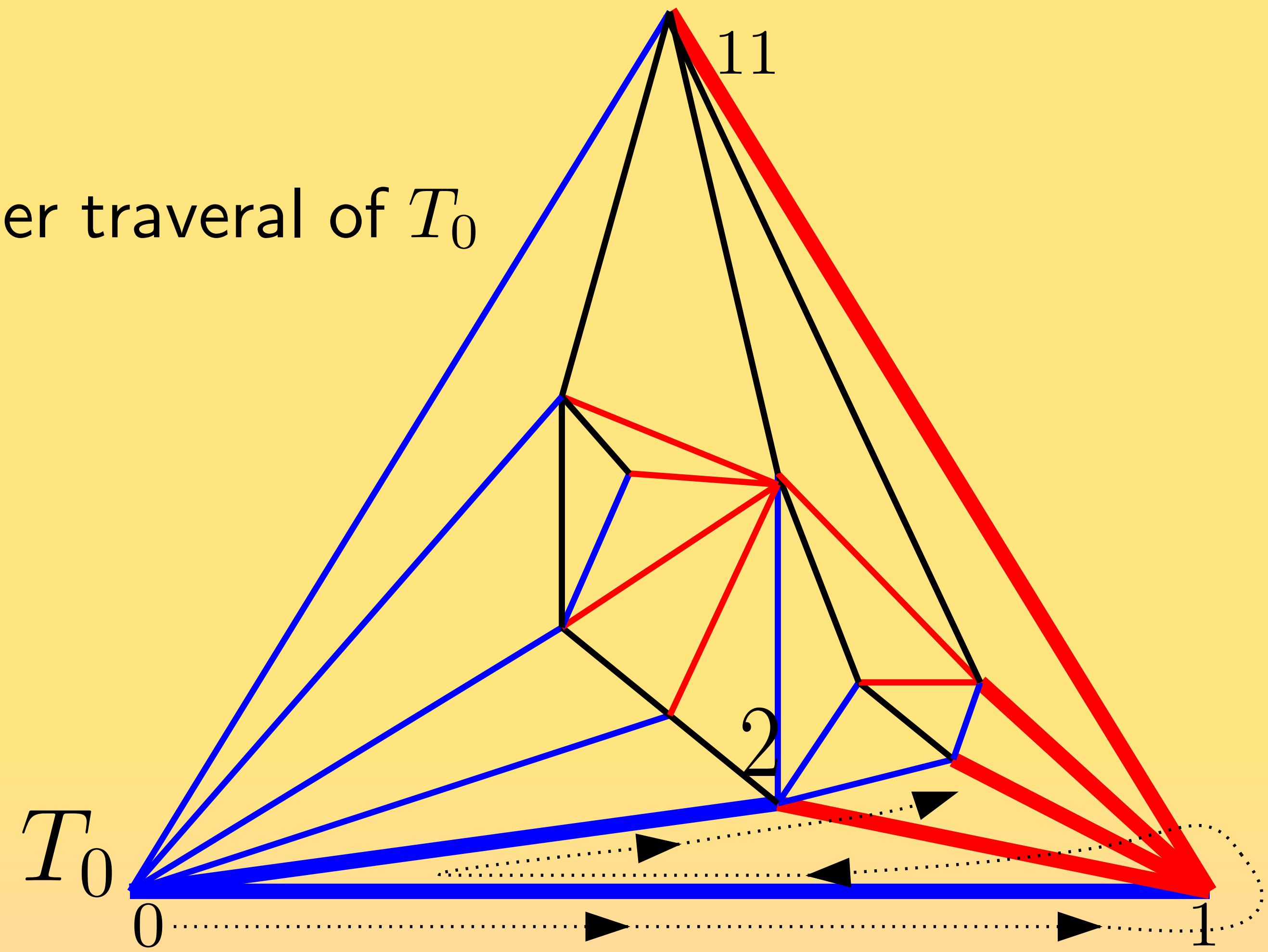
S (([[[[

01

Encoding the triangulation

Constructing the parenthesis representation

ccw preorder traveral of T_0



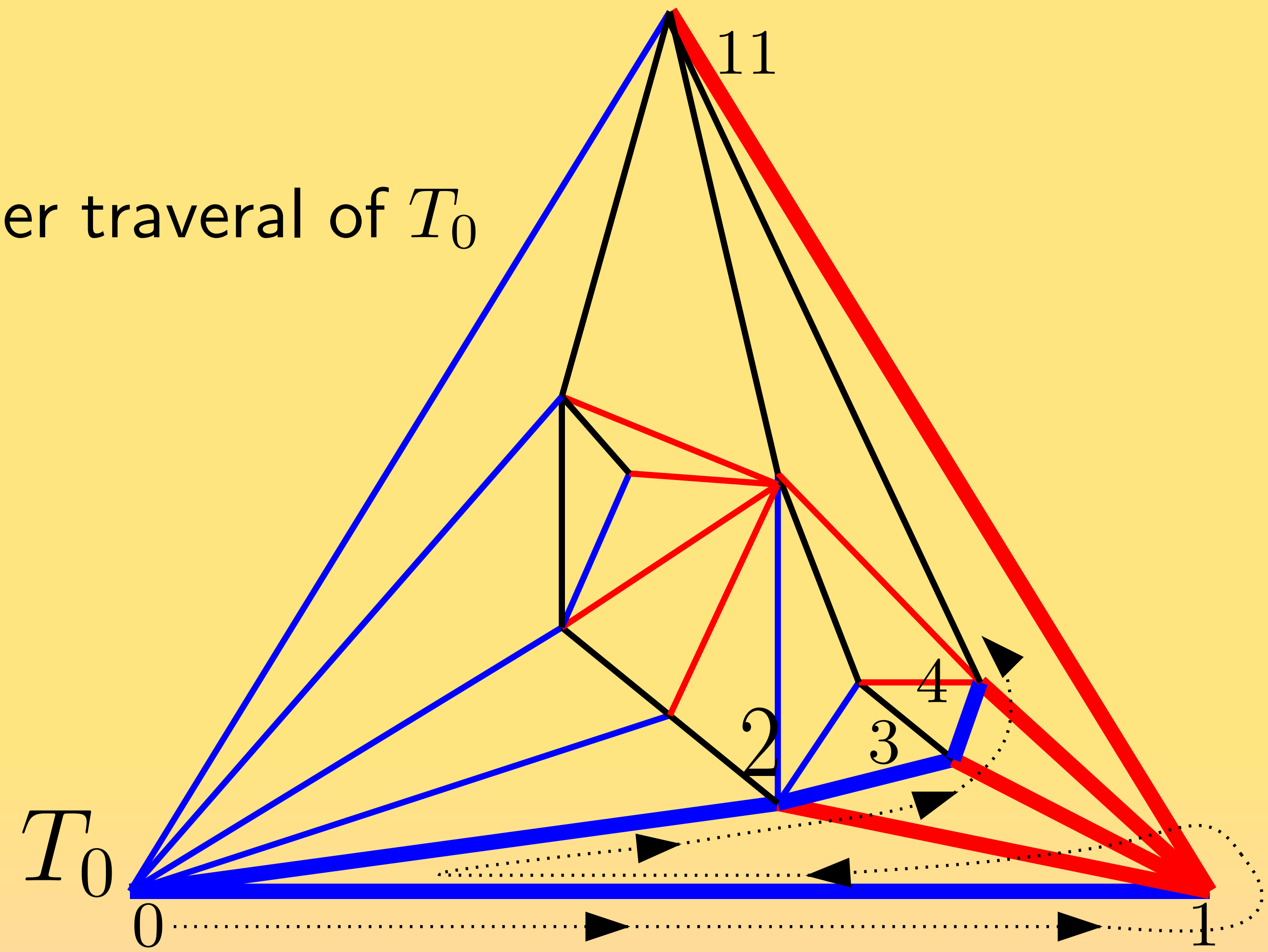
S (([[[[)([

01 12

Encoding the triangulation

Constructing the parenthesis representation

ccw preorder traveral of T_0



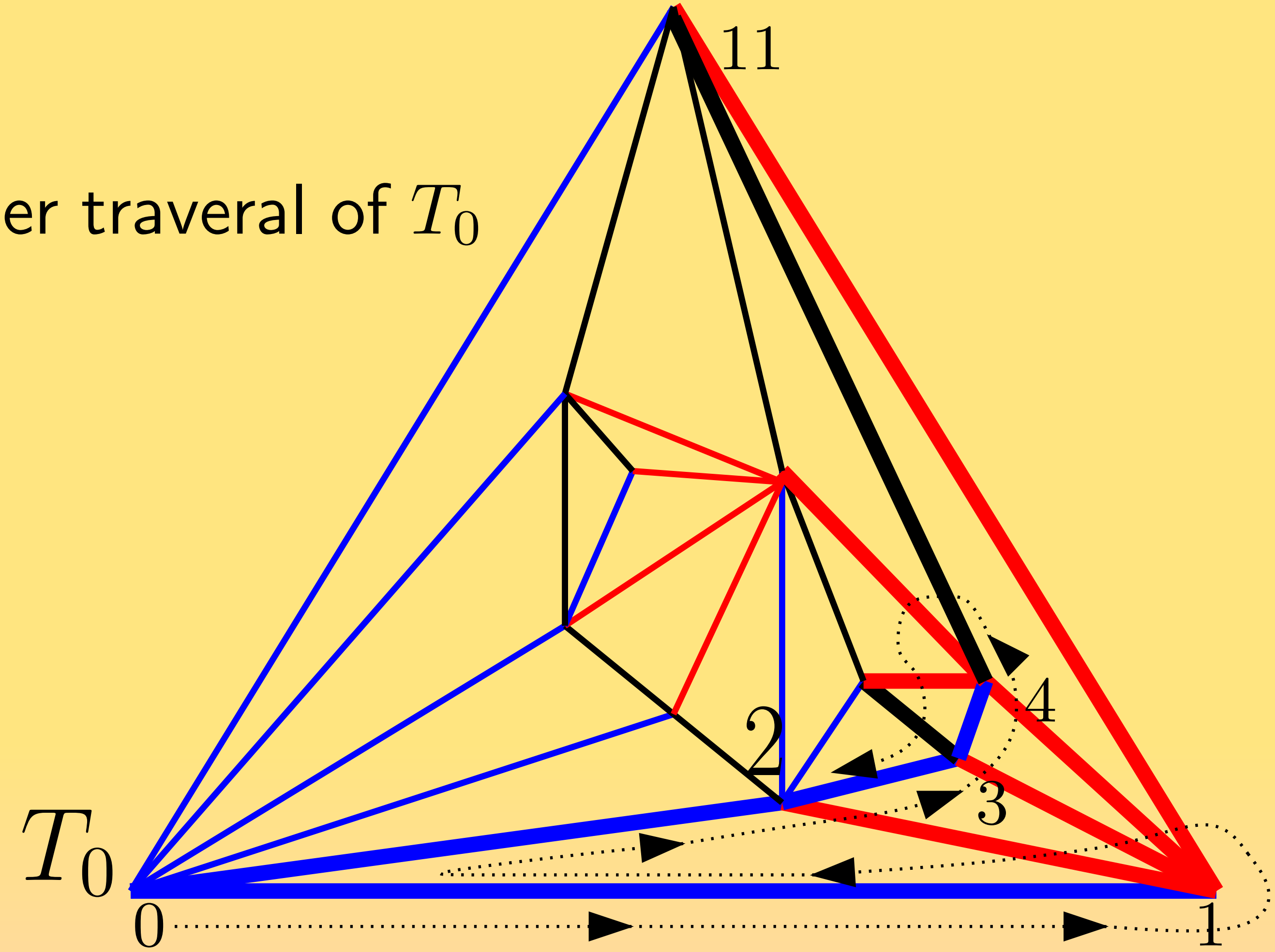
S (([[[[)(] (] (]

01 12 3 4

Encoding the triangulation

Constructing the parenthesis representation

ccw preorder traveral of T_0



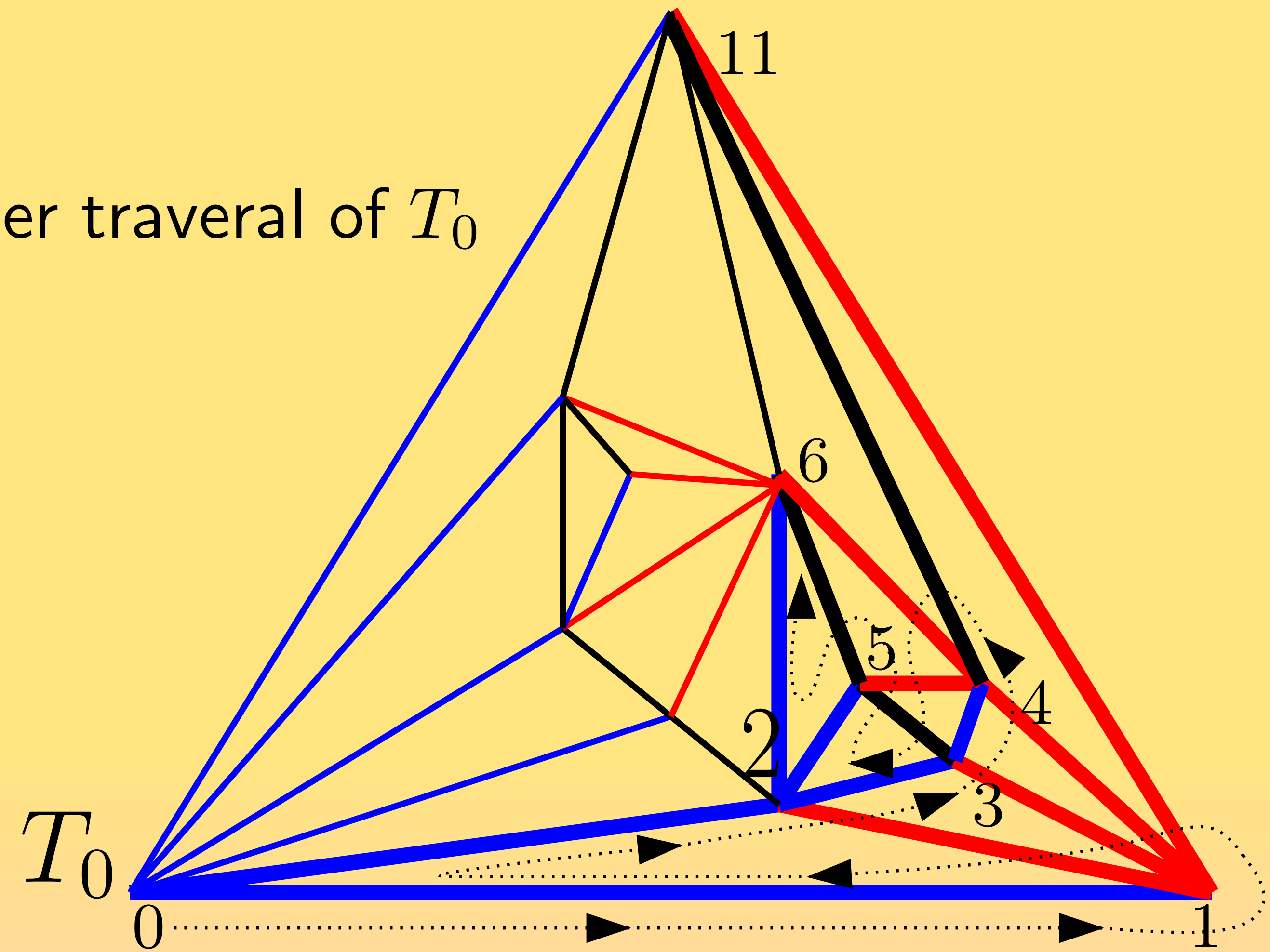
S (([[[[)(](](]{[[){)

01 12 3 4 4 3

Encoding the triangulation

Constructing the parenthesis representation

ccw preorder traveral of T_0



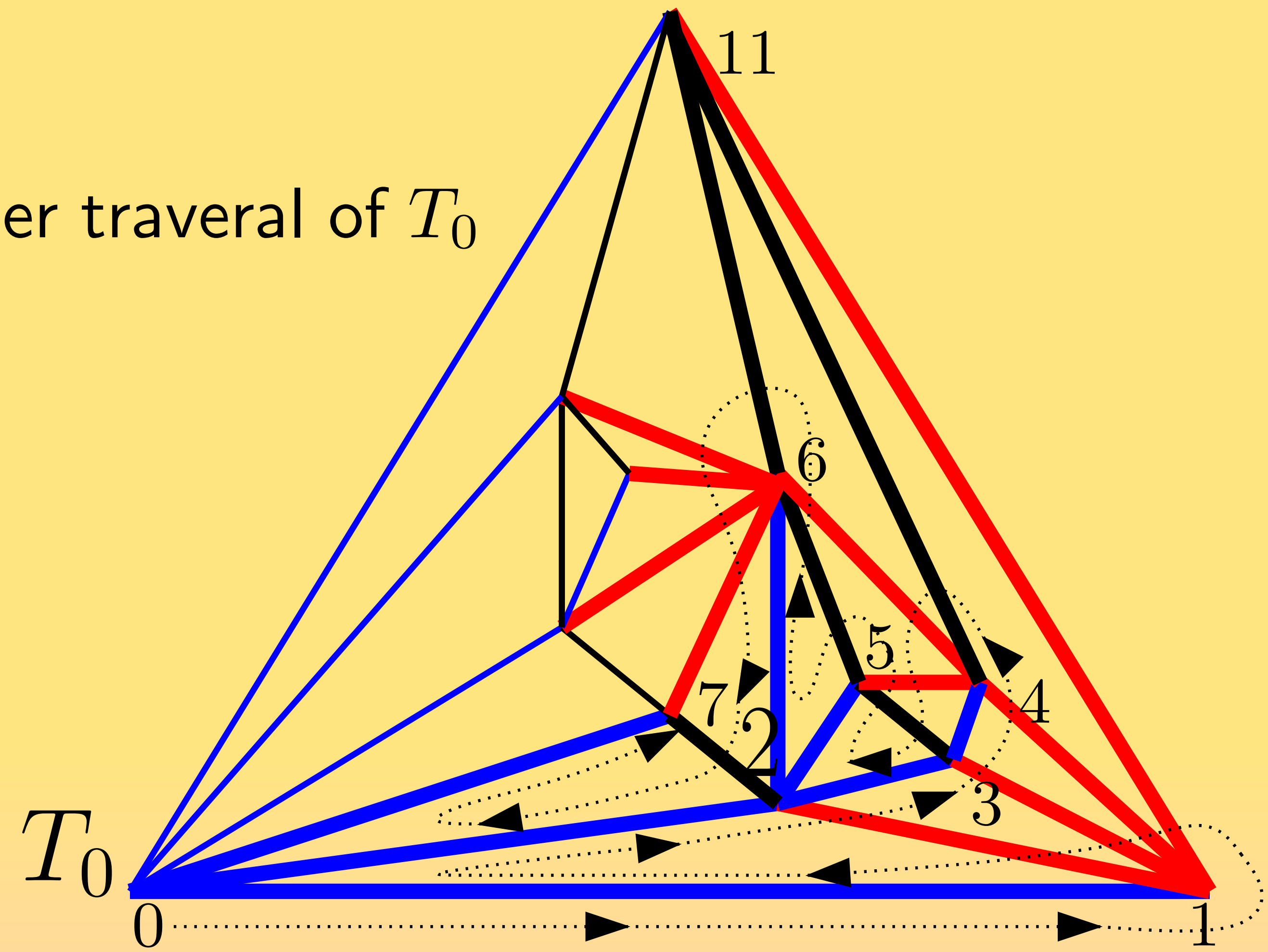
S $(([[[[)(](](]\{[[)\{)(\}]\{)($

$01 \quad 12\,3\,4 \quad 4\,35 \quad 6$

Encoding the triangulation

Constructing the parenthesis representation

ccw preorder traveral of T_0



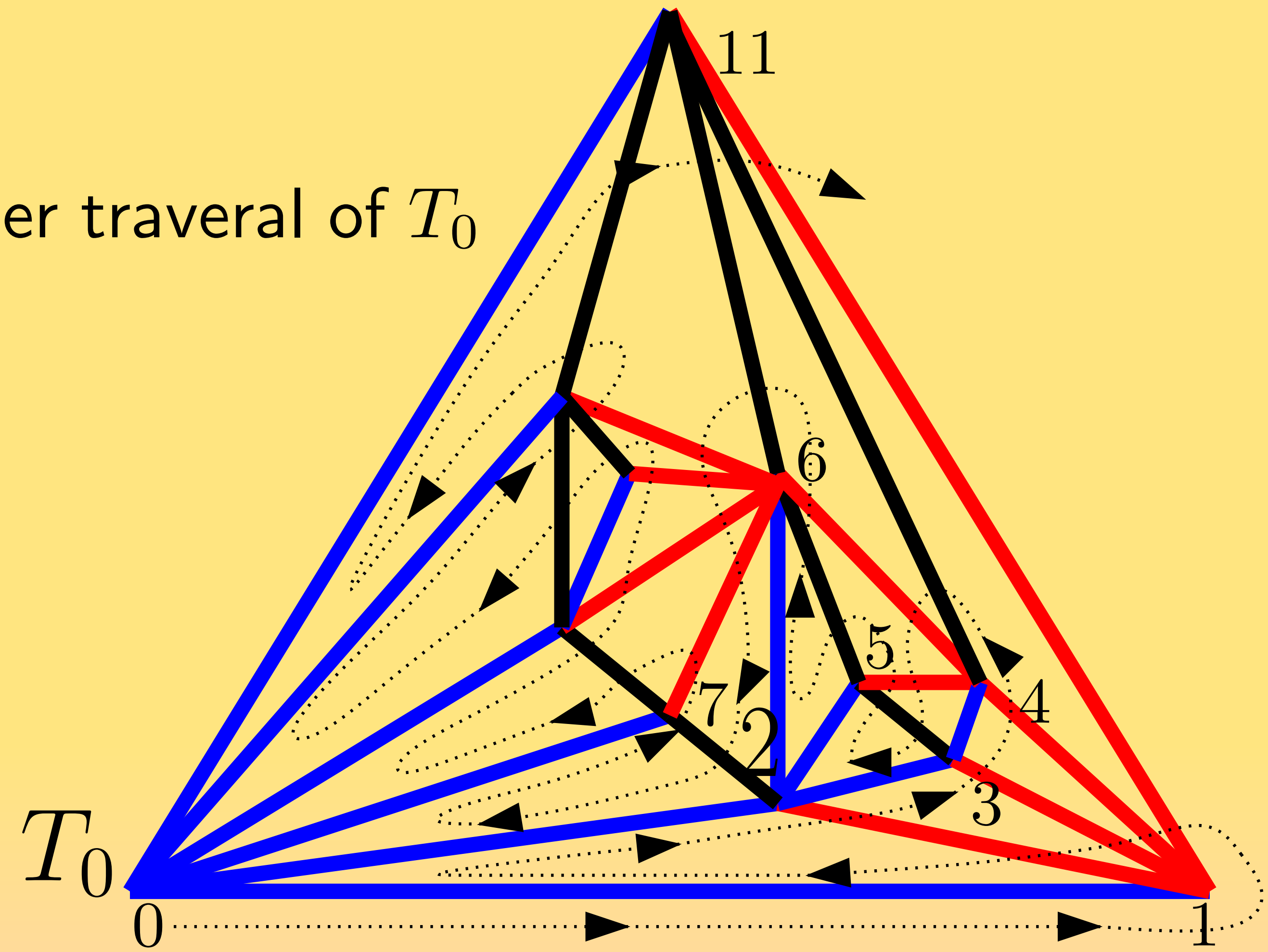
S (([[[[)(](](]{[[){)(}]{)(}]{[[[[){)(

01 12 3 4 4 35 6 7

Encoding the triangulation

Constructing the parenthesis representation

ccw preorder traveral of T_0

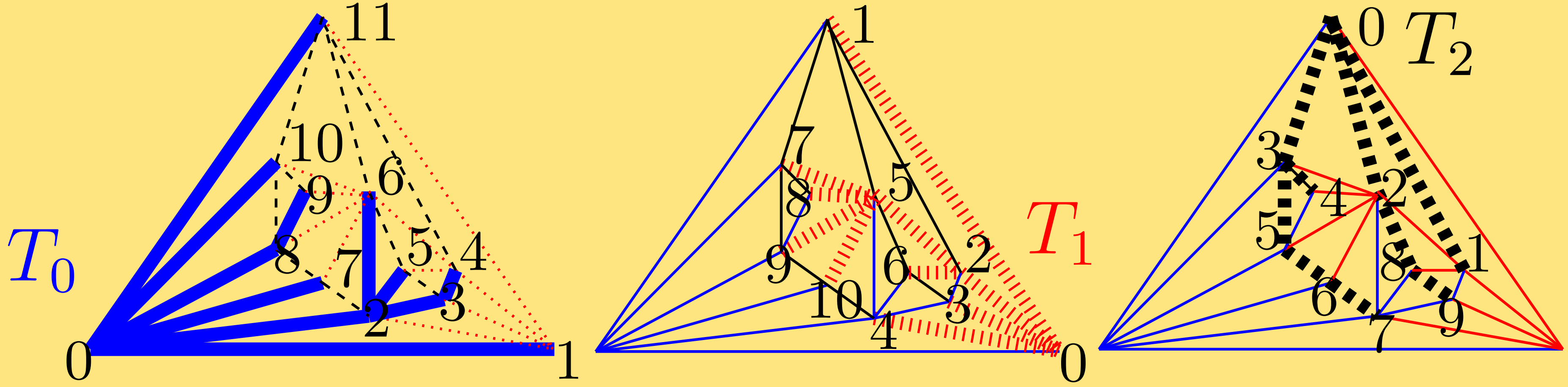


S (([[[[)(](](]{[[){)(}]{)(}]{[[[[){)(}]{)(}](]{){)(}}]{)(}}}]))

01 12 3 4 4 35 6 7 8 9 10 11

Last step

Defining 3 orders on the vertices of the graph



S (([[[[)(](](]{[[){)(}]{)(}]{[[[[){)(}]{)(}](]{){)(}}]{)(}}]))

S_0 (( ) ((( )) ( ) ( )) ( ) (( )) ( ) ( ))
0 1 2 3 4 5 6 7 8 9 10 11

S_1 [[[[ ]]] [[ ]] [[[[ ]]]]]
11 4 3 2 6 5 10 9 8 7

S_2 { { } { } { { } { } { { } } { } } }
3 5 2 7 8 9 10 6 4

$$\pi_1 \begin{bmatrix} 11 & 4 & 3 & 2 & 6 & 5 & 10 & 9 & 8 & 7 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \end{bmatrix}$$

$$\pi_2 \begin{bmatrix} 4 & 6 & 10 & 9 & 8 & 7 & 2 & 5 & 3 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \end{bmatrix}$$

A succinct representation of (vertex) labeled graphs

Theorem (Barbay, Castelli Aleardi, He and Munro)
For labeled planar triangulations with e edges and n vertices , with σ labels associated with vertices, there exists a succinct representation which requires

$$(2\log_2 6 + \varepsilon)e + o(e) + \boxed{n \cdot \underline{o}(\lg \sigma)} \text{ bits}$$

while supporting labeled based navigation in

$$O((\lg\lg\lg \sigma)^2 \lg\lg \sigma) \text{ time}$$

For large σ

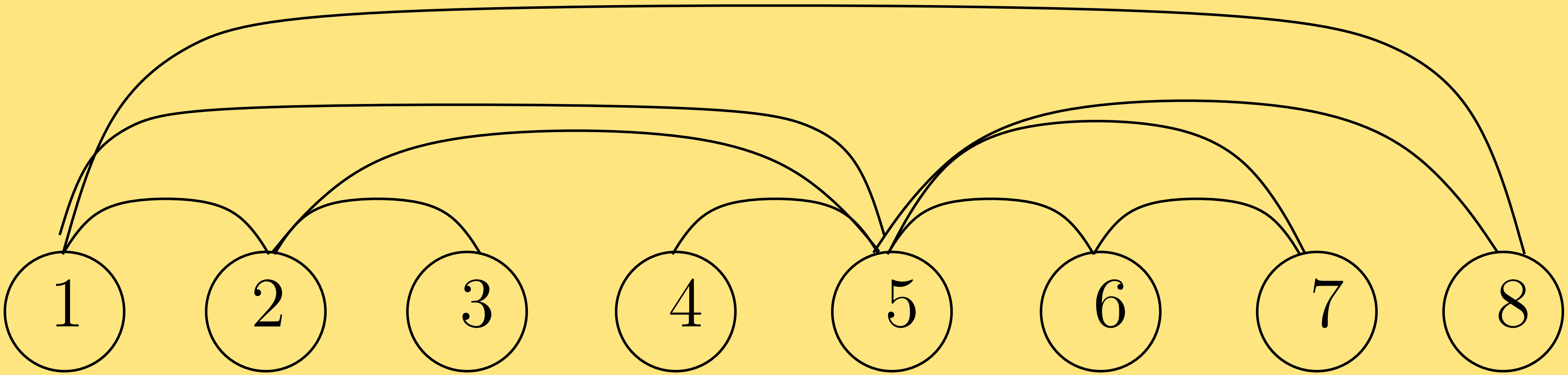
$$(2\log_2 6)e \approx 3(2\log_2 6)n << n \cdot \lg \sigma$$

Another result

(for edge labeled graphs)

Book embedding and planar graphs

One page graph (outerplanar graph)



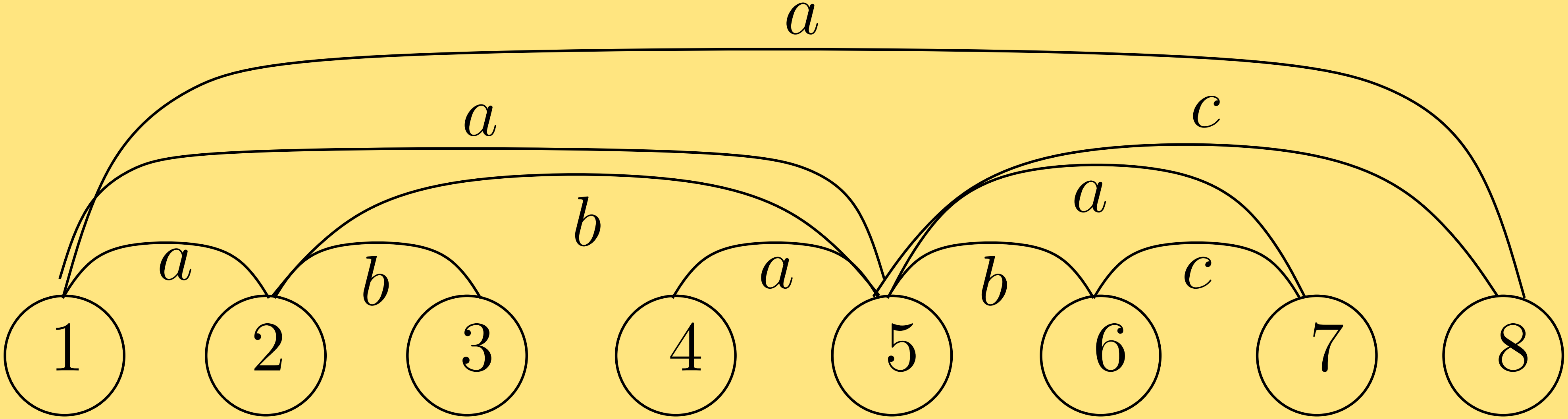
((( )(( ) ( )))((( )( ))))

Planar graphs admit a 4-pages book embedding

Book embedding and planar graphs

One page (edge) labeled graph

$$\Sigma = \{a, b, c\}$$
$$n = 8$$

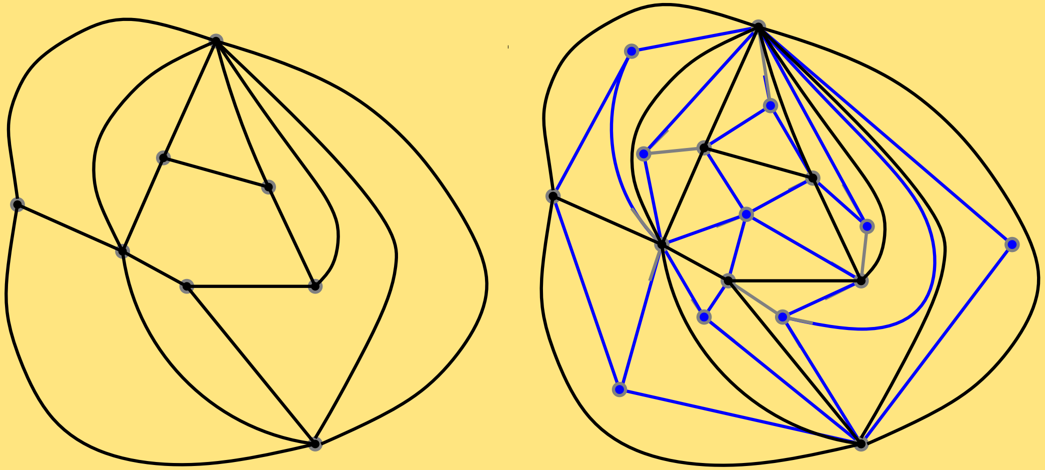


(((	)((	)	(	)))(((	)(	))	))
1000	1000	10	10	1000000	100	100	100
aaa	bb		a	cab	c		
	a	b		aba	b	ca	ca

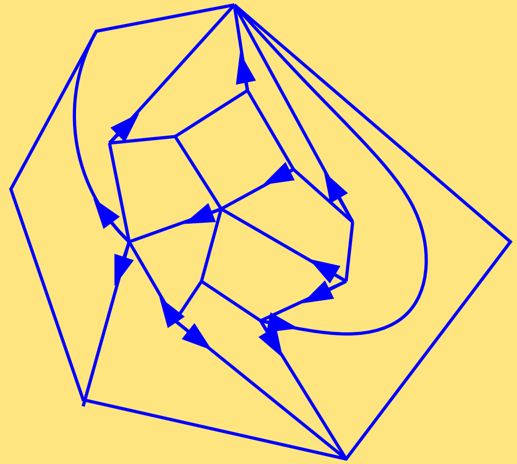
future work

Future work

- Extension to poygonal meshes



3-connected planar graph



quadrangulation

- Improving the dominant term

$$(2\log_2 6)e + o(e) + n\,o(\lg\sigma) \xrightarrow{?} \boxed{1.08e} + o(e) + n\,o(\lg\sigma)$$

(optimal) entropy bound

- new (more interesting) label navigation operations