Succinct representations of labeled graphs

Journées en l'honneur de Donald E. Knuth

30 octobre 2007, LaBRI, Bordeaux

Luca Castelli Aleardi

ULB UNIVERSITÉ LIBRE DE BRUXELLES

ÉCOLE POLYTECHNIQUE

LIX

INRIA SOPHIA ANTIPOLIS

Jeremy Barbay Meng He J. Ian Munro

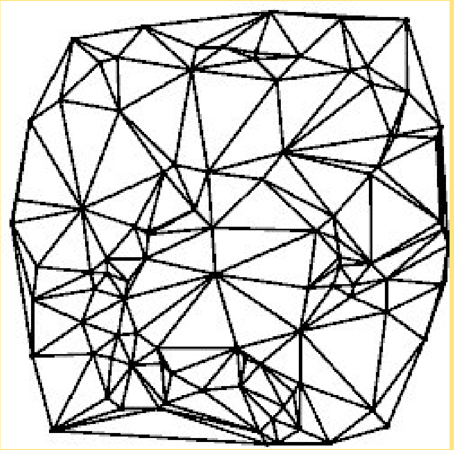
University of Waterloo

David R. Cheriton School of Computer Science

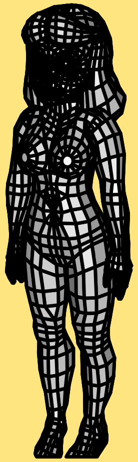
Domain and motivations

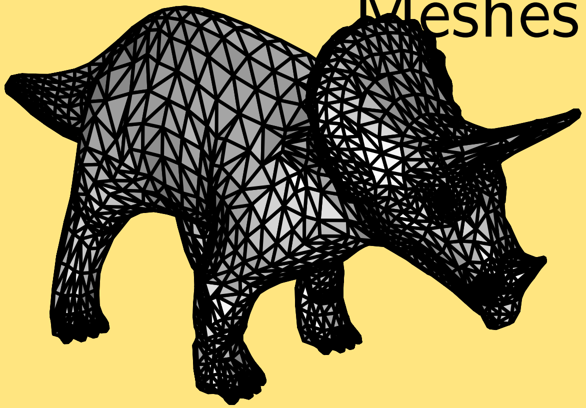
Geometric data

Triangulations and graphs



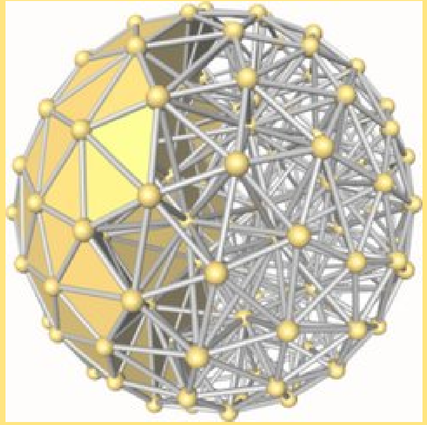
3D meshes





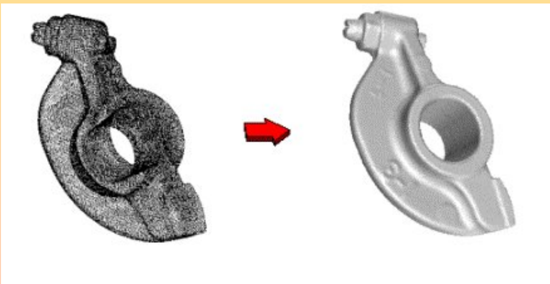
Tetrahedral Meshes

Volume

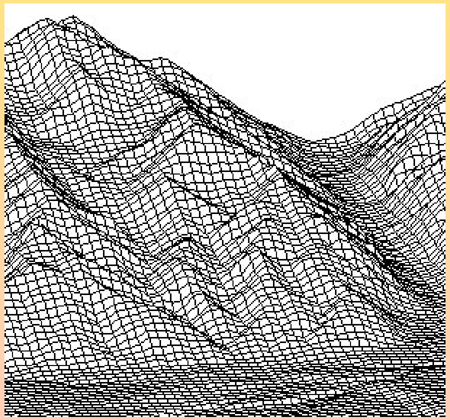


Applications

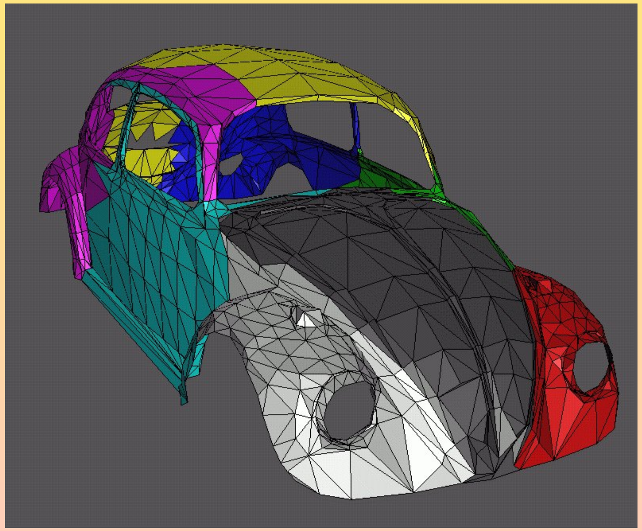
Surface recontruction



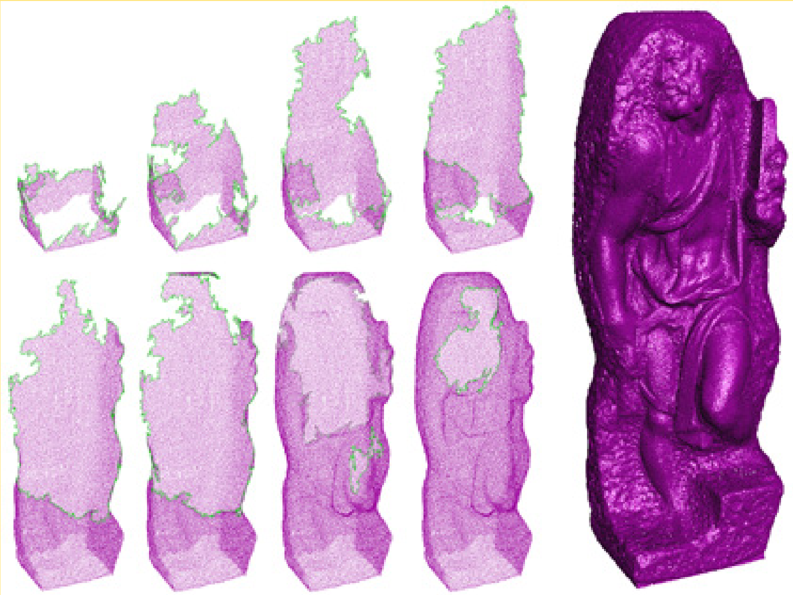
GIS Technology



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)

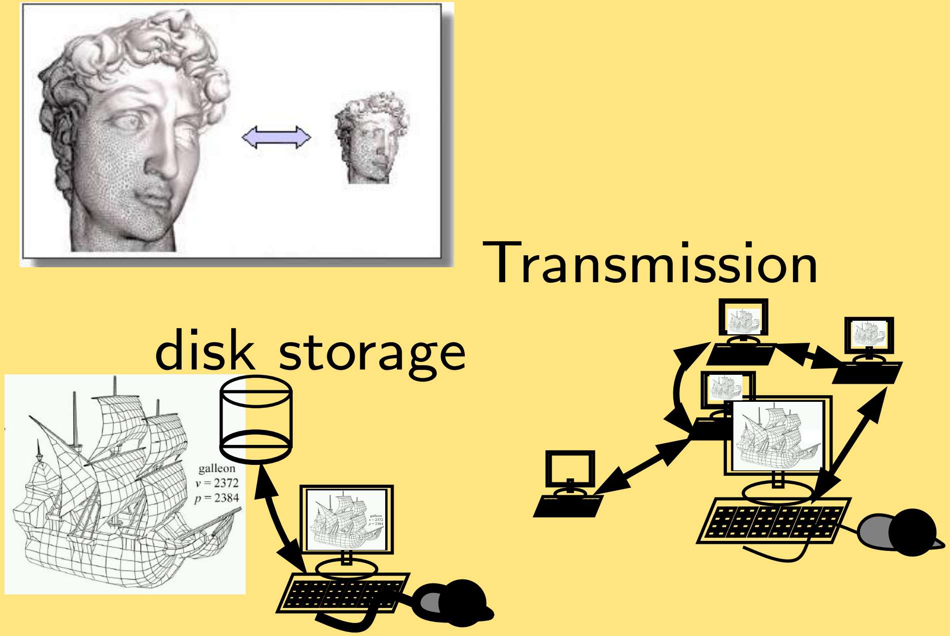
2 billions polygons

32 Giga bytes (without compression)

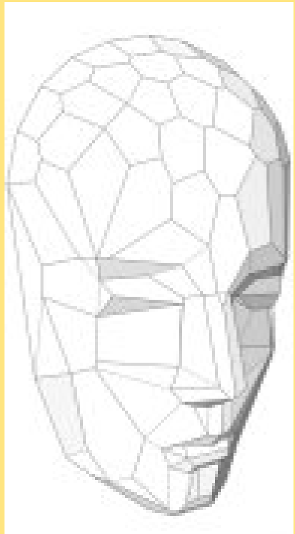
No existing algorithm nor data structure for dealing with the entire model

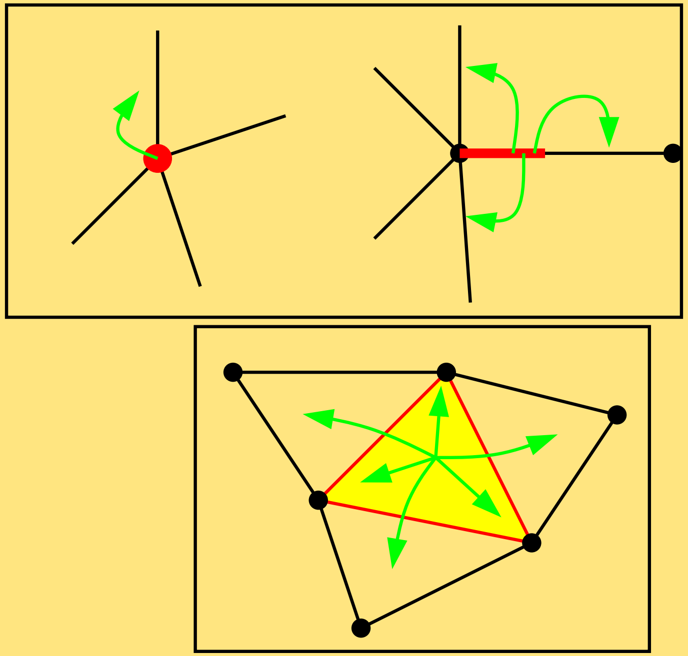
Research topics

Mesh compression

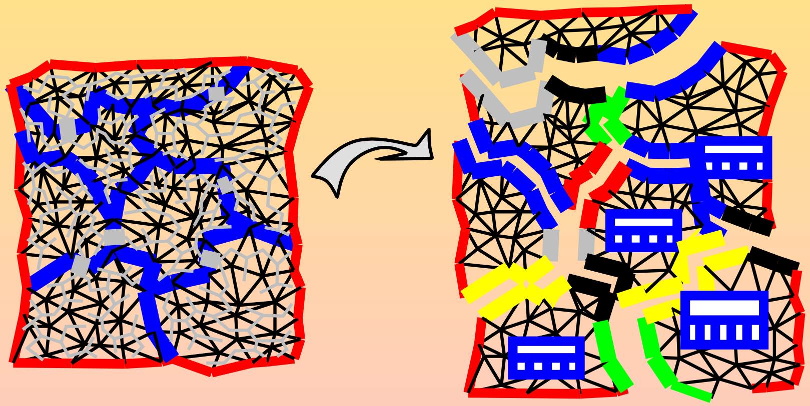


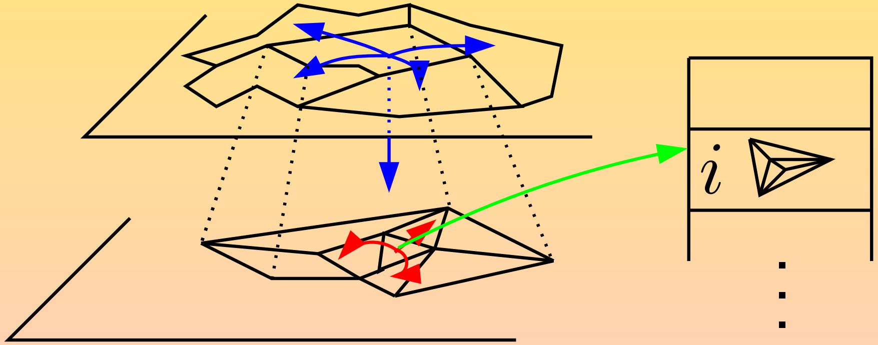
Geometric data structures



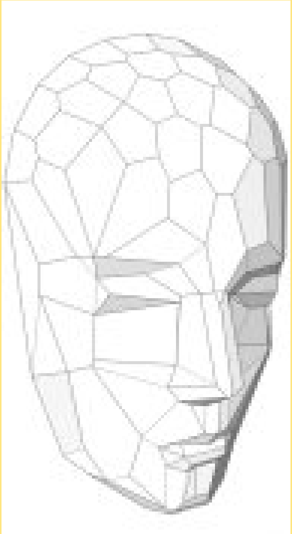
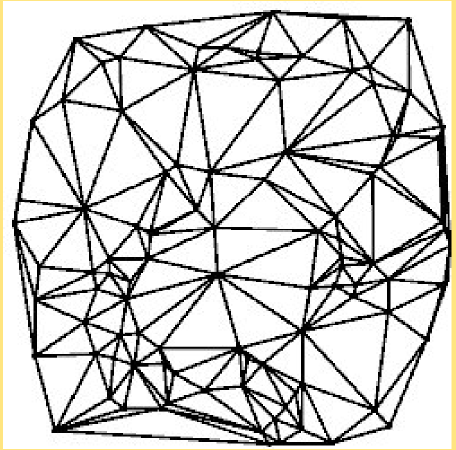
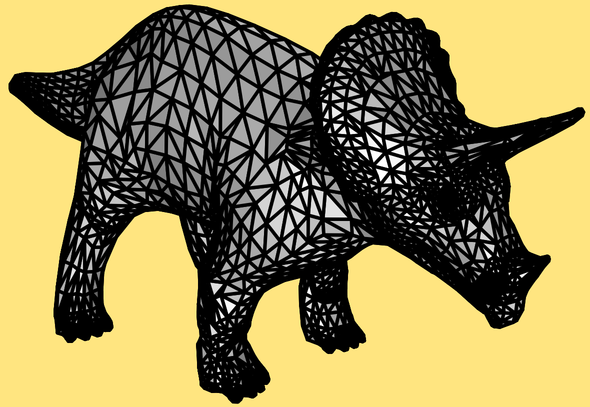


Compact representations of geometric data structures

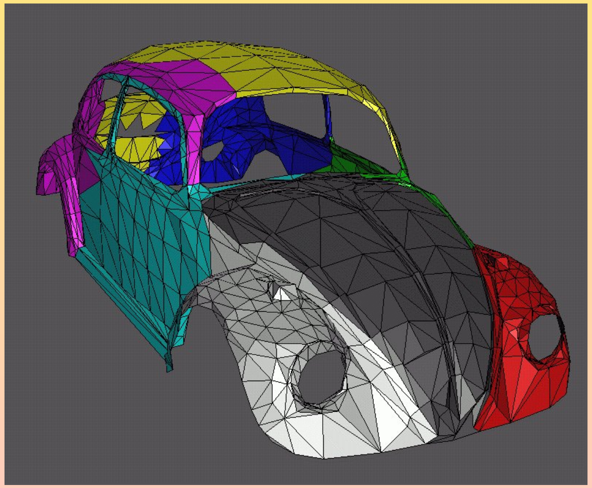
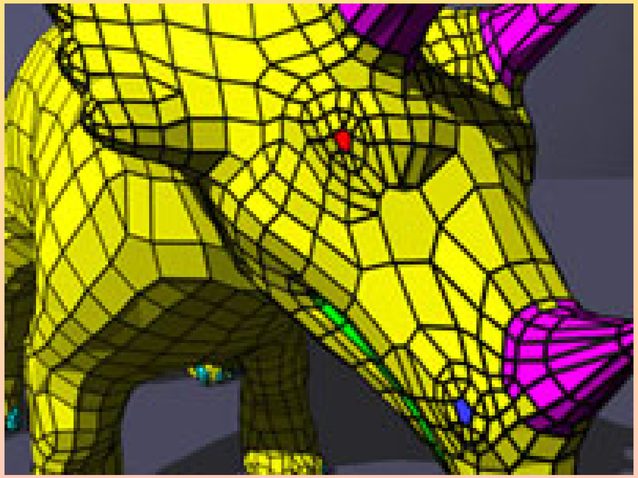




Unlabeled vs. labeled structures

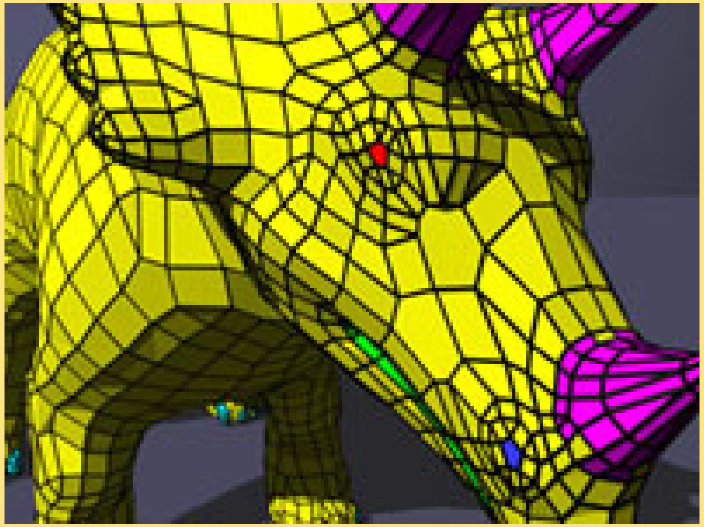
Unlabeled graphs, meshes with no properties

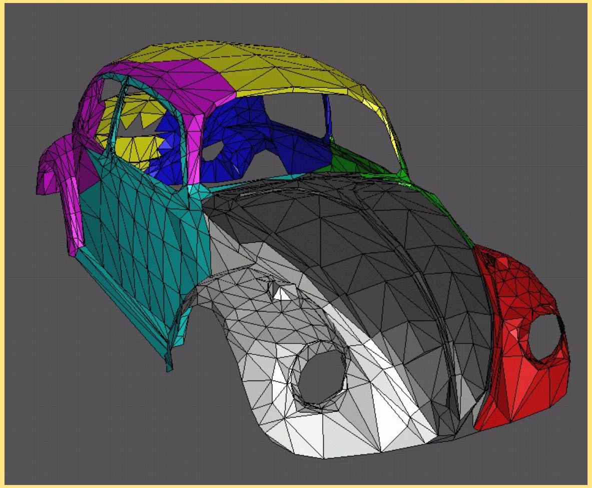
 

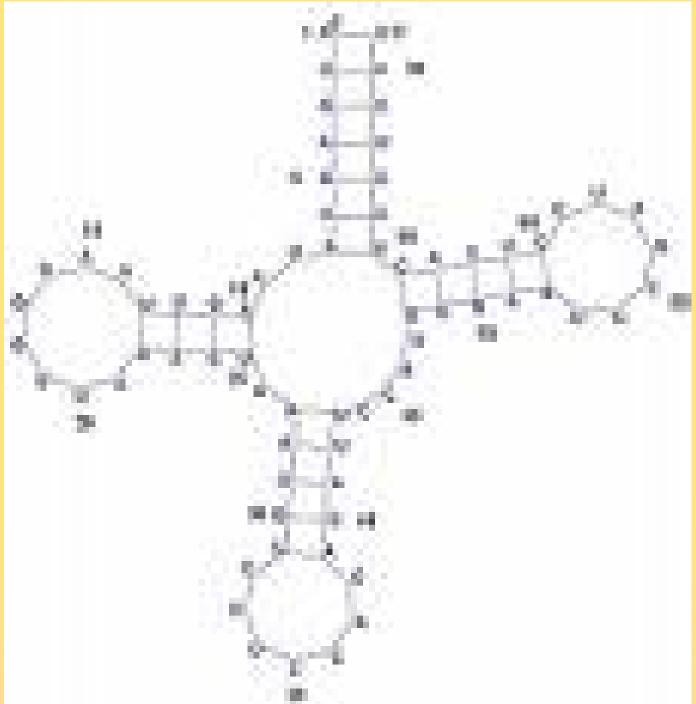
Labeled graphs
meshes with properties

Efficiently representing labeled structures

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



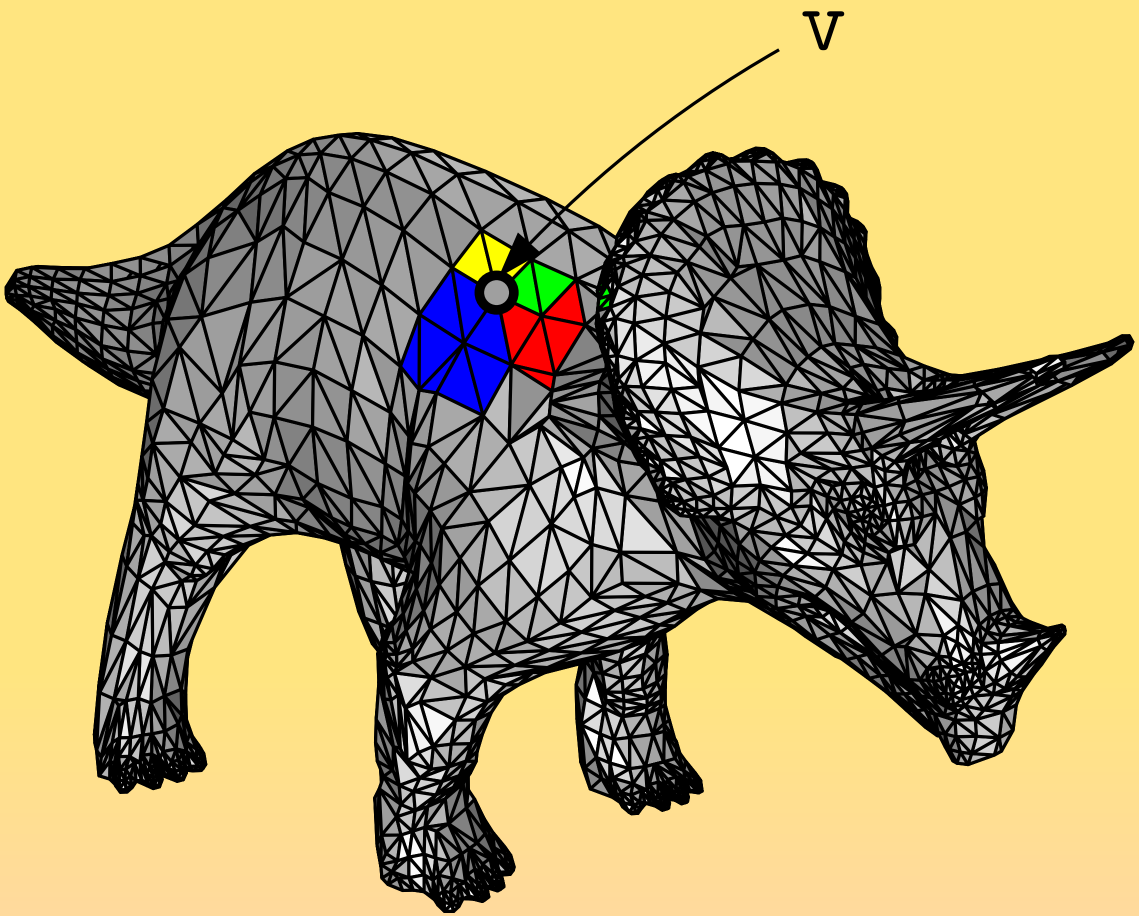




Example: vertex coordinates, face colors and normals, additional tags, ...

Efficiently representing labeled structures

Labeled based navigation operators



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

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

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

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

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

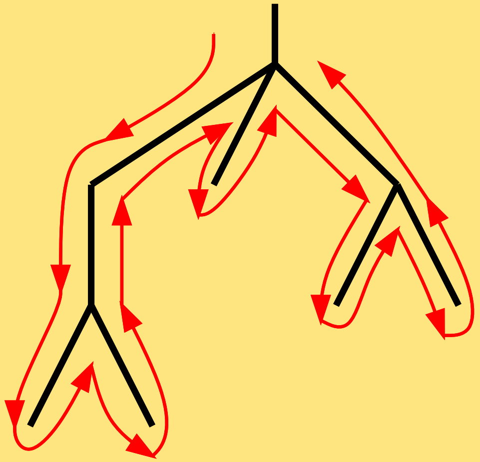
Problem: find adjacent cells with a given label

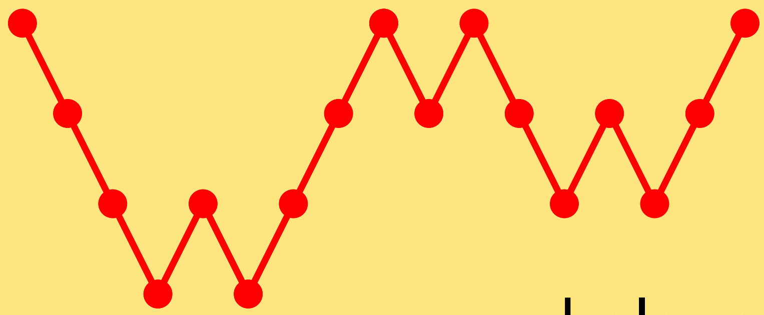
using $o(\lg \sigma)$ bits per label   in almost $O(1)$ time

Compact representations

An example: (unlabeled) plane trees

ordered tree with n edges





balanced parenthesis word

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

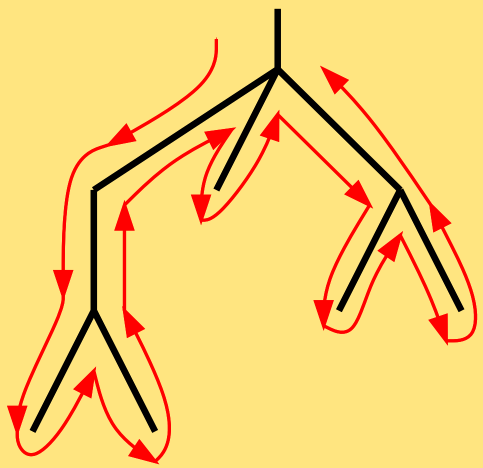
$\Rightarrow 2n$ bits for encoding a tree with n edges

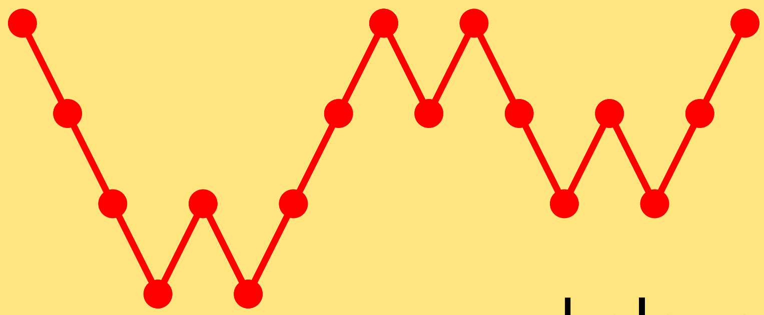
Enumeration of plane trees with n edges

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

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

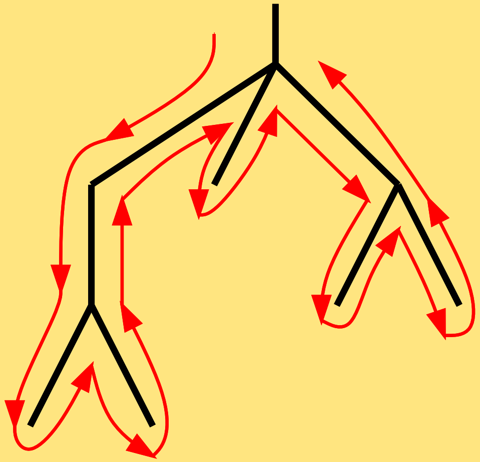
Asymptotic optimal encoding

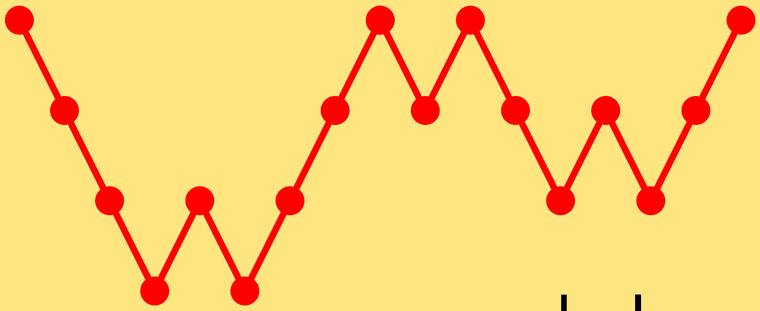
- the cost of an object matches asymptotically the entropy

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

An example: (unlabeled) plane trees

ordered tree with n edges





balanced parenthesis word

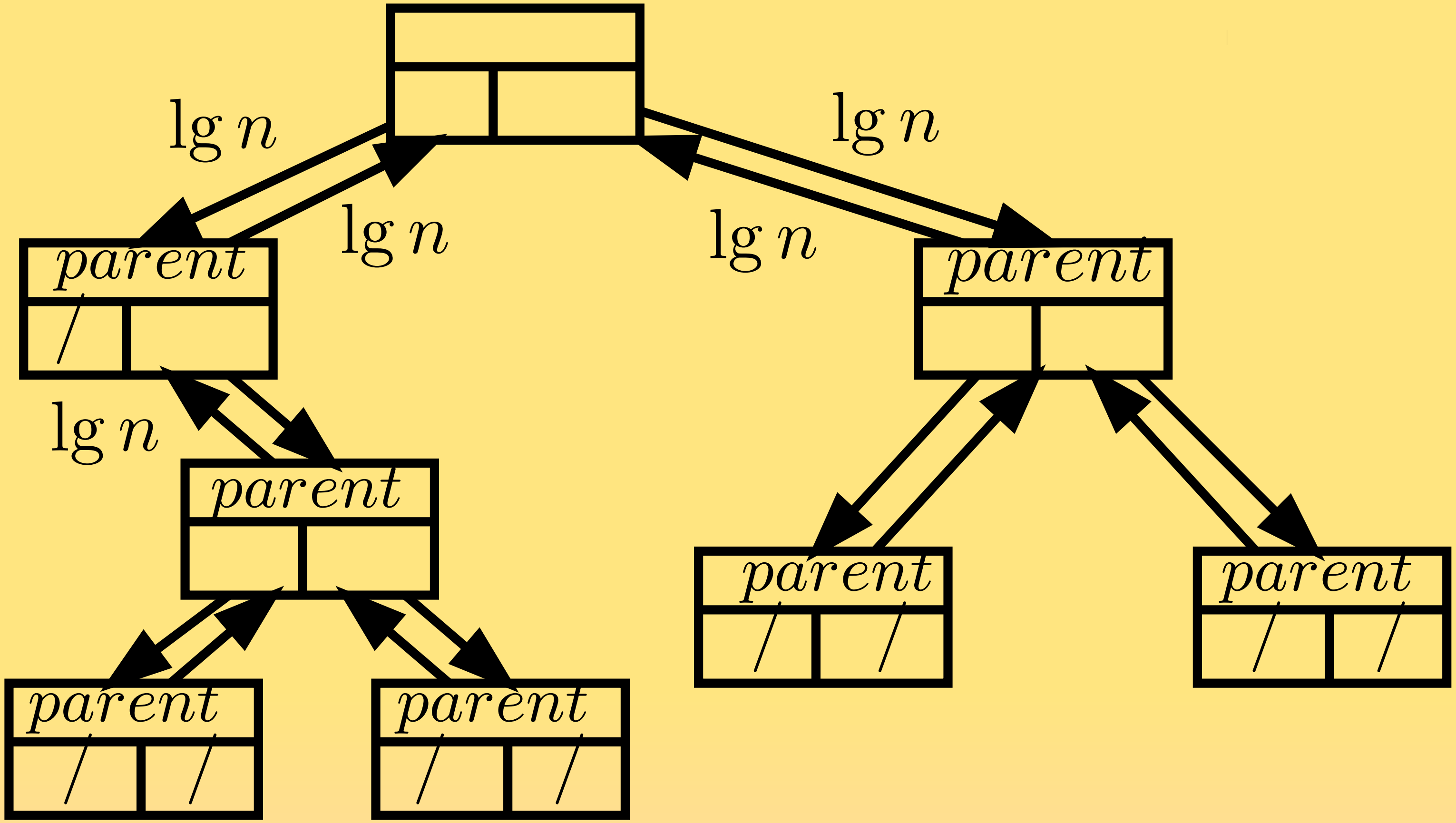
$1\,1\,1\,0\,1\,0\,0\,0\,1\,0\,1\,1\,0\,1\,0\,0$

$\Rightarrow 2n$ 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?

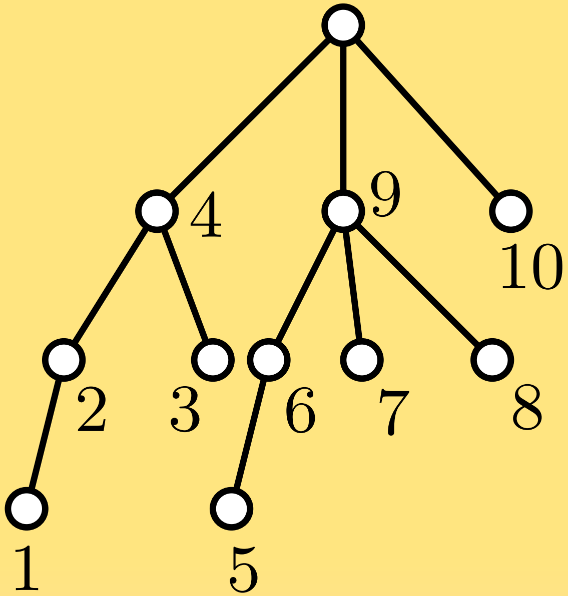
a compact encoding (asymptotically optimal)

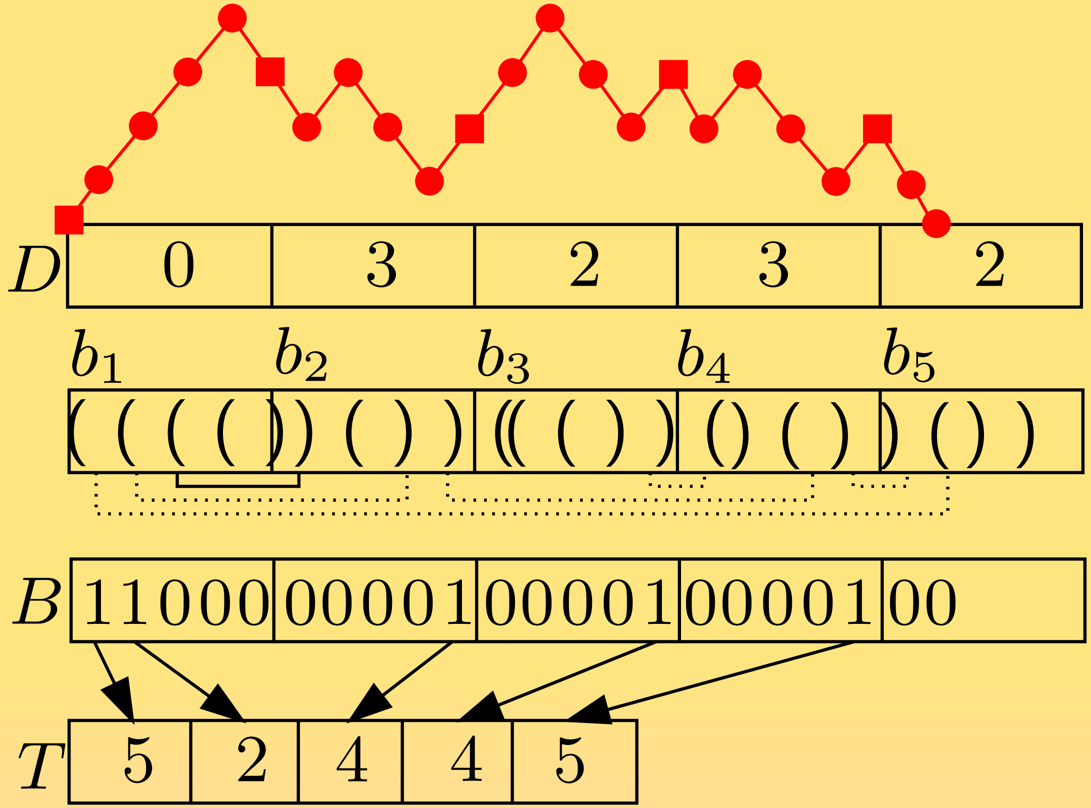
testing adjacency queries efficiently (in $O(1)$ time)

An example: binary and ordered trees (unlabeled)

(Jacobson, Focs89, Munro and Raman Focs97)

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





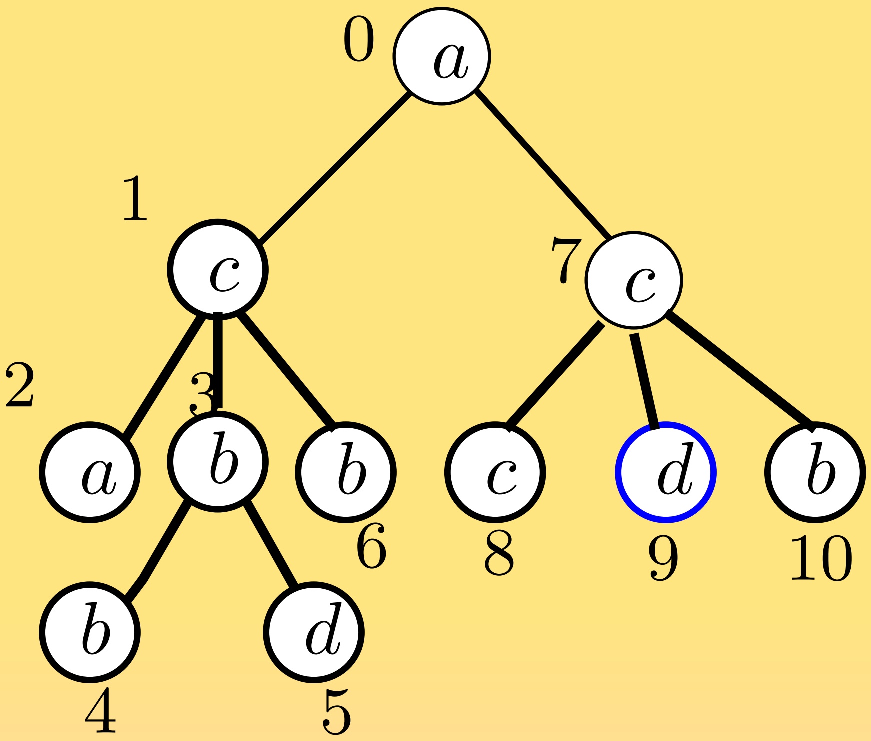
it is possible to test adjacency between vertices in $O(1)$ time
with the guarantee that the encoding is still asymptotically optimal

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

And for labeled trees?

Labeled ordinal trees (Barbay et al. '06)

Labeled based navigation operators



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

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

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

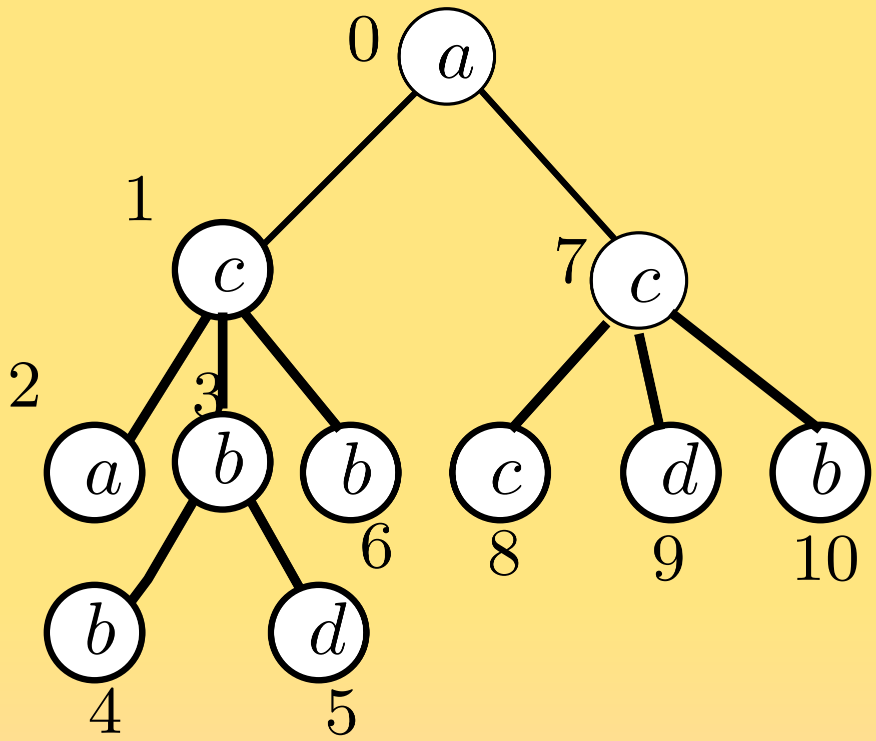
$$\mathtt{lab\_child}(\mathtt{d}, \mathtt{v}_7, 1) = \mathtt{v}_9$$

$$\mathtt{lab\_degree}(\mathtt{b}, \mathtt{v}_1) = 2$$

Labeled ordinal trees (Barbay et al. '06)

Prefix order

Descendants are listed consecutively



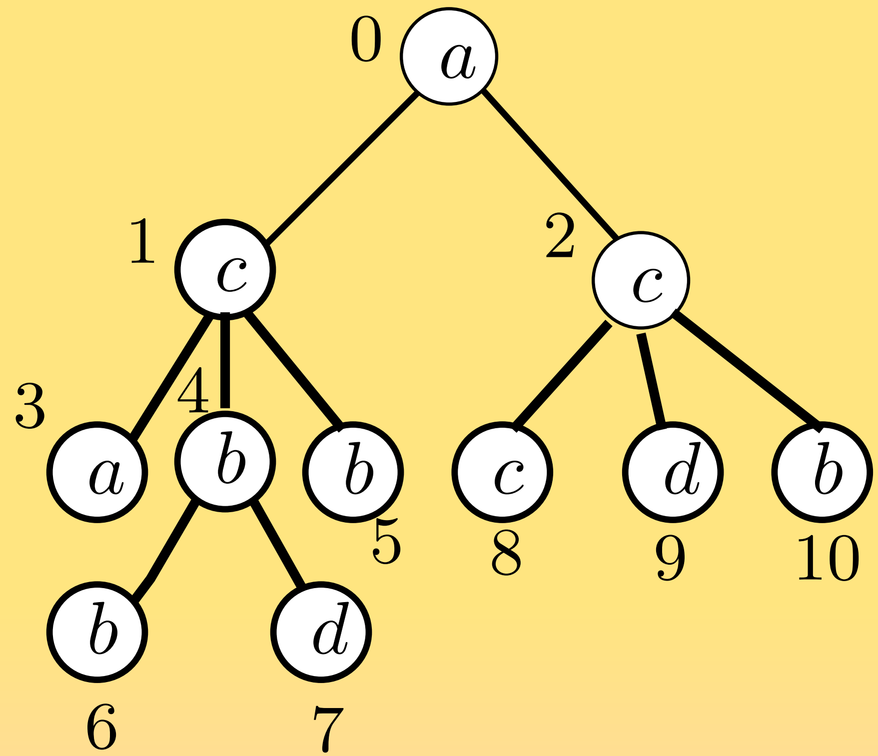
S_1	a	c	a	b	b	b	d	c	c	d	b

Labels listed in prefix order

DFUDS order

(Depth first unary degree sequence)

Children are listed consecutively



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

Labels listed in DFUDS order

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

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

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

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

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

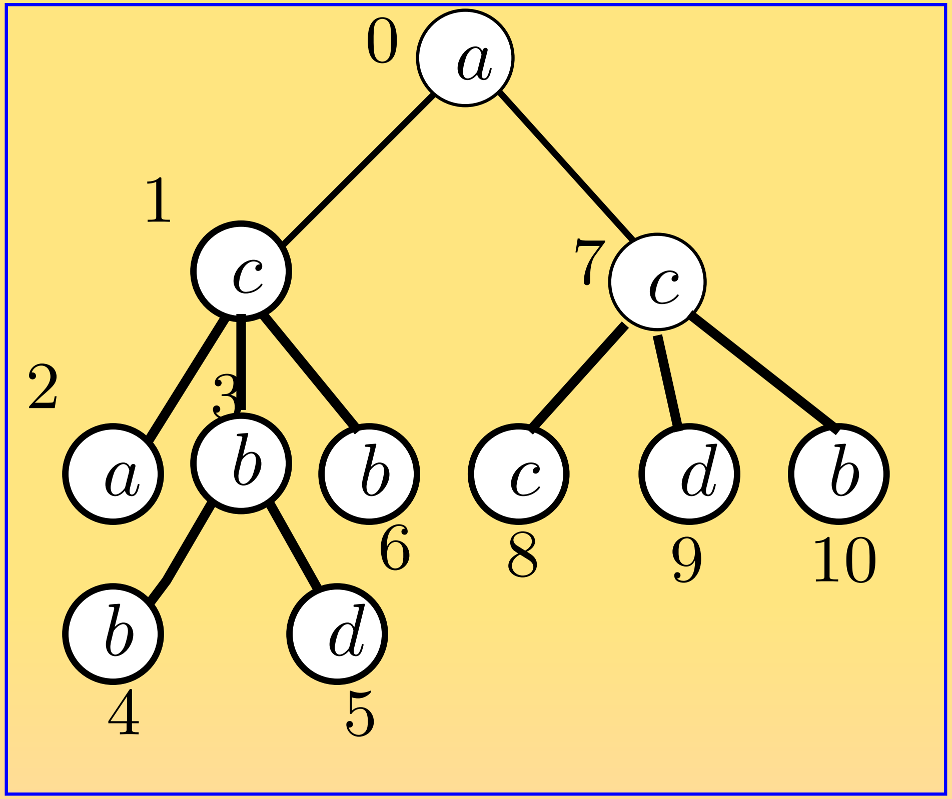
(Barbay et al. '06)

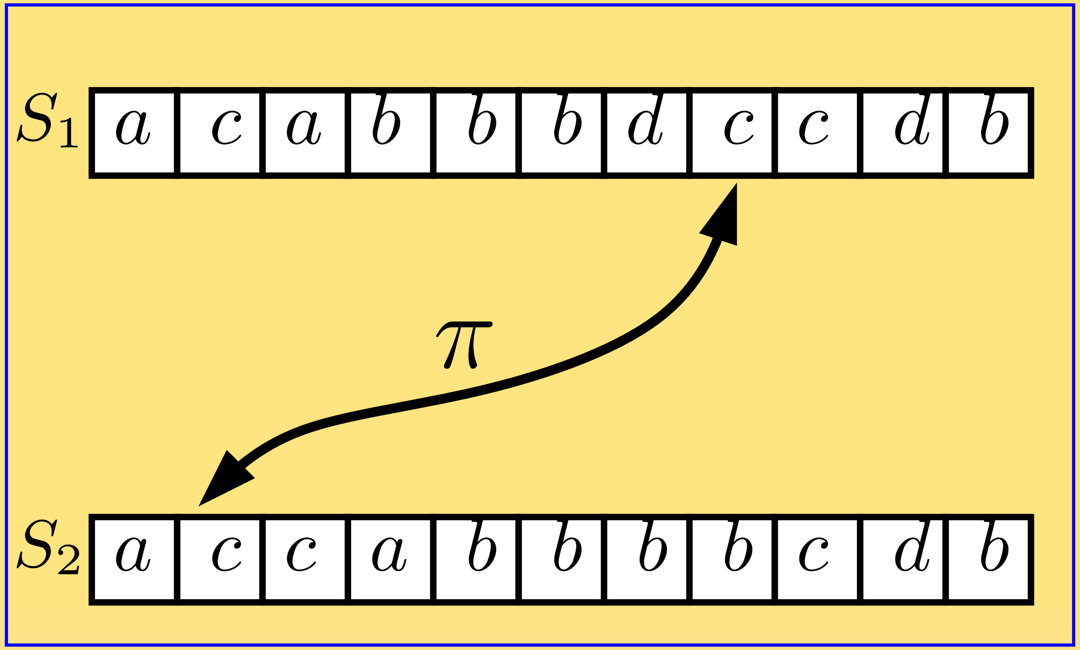
S_1										
a	c	a	b	b	b	d	c	c	d	b

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

Labeled ordinal trees (Barbay et al. '06)

Navigation in the tree





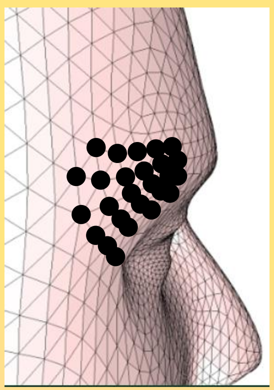
$\texttt{lab\_child}(\texttt{d}, \texttt{v}_7, 1) = \texttt{v}_9 \longleftrightarrow$ `string_rank(...)` `string_select(...)`

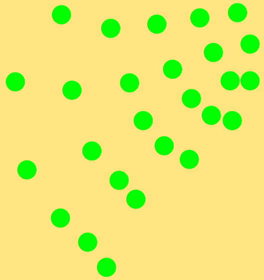
Geometric data

Which information?

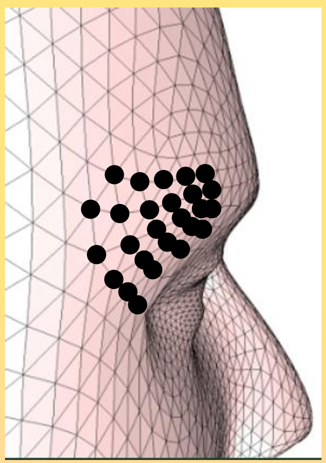
Geometry and connectivity

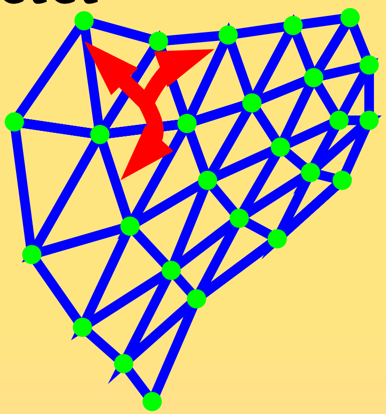
Geometric object





Combinatorial objetct





$\begin{pmatrix} x \\ y \\ z \end{pmatrix}$ between 30 et 96 bits/vertex

$2 \times n \times 6 \times \log n$

$n \times 1 \times \log n$

$13n \log n$

$\boxed{416n \text{ bits connectivity}}$

Existing works (planar triangulations with m faces) $\boxed{1.62m}$

Optimal bound

Mesh compression algorithms

Unlabeled case

[Edgebreaker, '99] 1.84m bits

[Poulalhon Schaeffer, '03] 1.62 m bits

[Touma Gotsman, '98] ≈1m bits

[Facefixer] Isenburg and Snoeyink

Labeled case

Compact and succinct representations

Unlabeled case

Encoding	triangulations
Jacobson '89	$64n$ bits
Munro Raman '97	$7m$ bits
Chuang et al. '98	$3.5m$ bits
Chiang et al. '01	$4m$ bits
Blandford et al. '03	$O(m)$ bits
Castelli et al. '06	$1.62m$ bits

Labeled case

—

A general scheme for succinctly representing graphs

Main ingredient: prepocessing and storage

Arithmétisons donc un peu (La leçon, Eugène Ionesco, 1951)

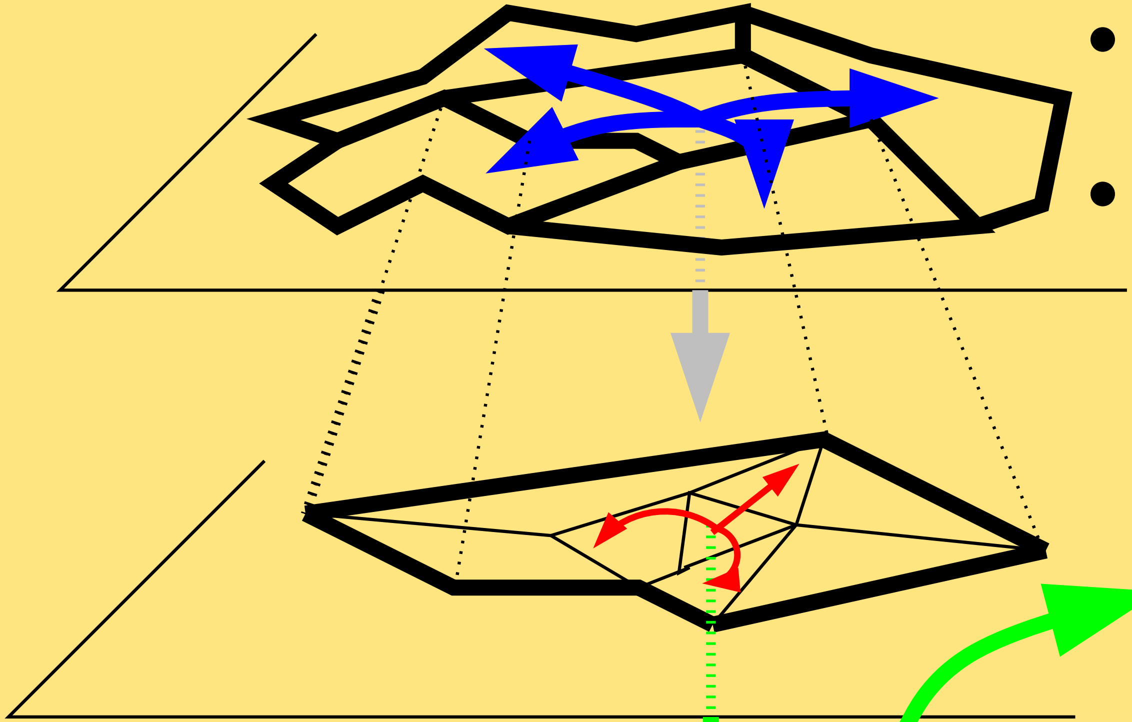
Au cours d'une lecon privée en préparation du doctorat total, une jeune fille discute d'arithmétique avec son professeur.

le professeur Ecoutez-moi, Mademoiselle, si vous n'arrivez pas à comprendre profondément ces principes, ces archétypes arithmétiques, vous n'arriverez jamais à faire correctement un travail de polytechnicien... Encore moins ne pourra-t-on vous charger d'un cours à l'Ecole polytechnique... combien font, par exemple, $3.755.918.261$ multiplié par $5.162.303.508$?
l'élève (très vite) ca fait 193891900145...
le professeur(de plus en plus étonné calcule mentalement) Oui... Vous avez raison... le produit est bien... (il bredouille inintelligiblement)... Mais comment le savez-vous, si vous ne connaissez pas les principes du raisonnement arithmétique?

l'*élève*: C'est simple. Ne pouvant me fier à mon raisonnement, j'ai appris par coeur tous les résultats possibles de toutes les multiplications possibles.

Overview of the hierarchical structure

(Castelli Aleardi, Devillers, Schaeffer, SoCG '06)



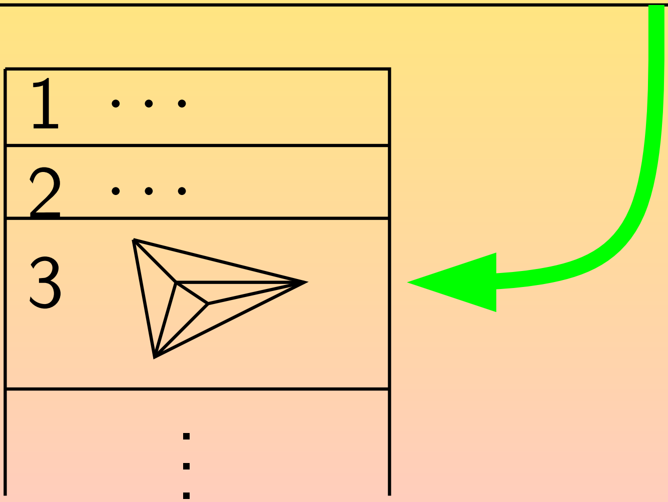
Level1:

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

Leve 2:

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

- $\Theta(\log n)$ micro regions of size $C \log n$, represented by pointers to level 3
- local pointers of size $\boxed{\log \log n.}$



Dominant cost

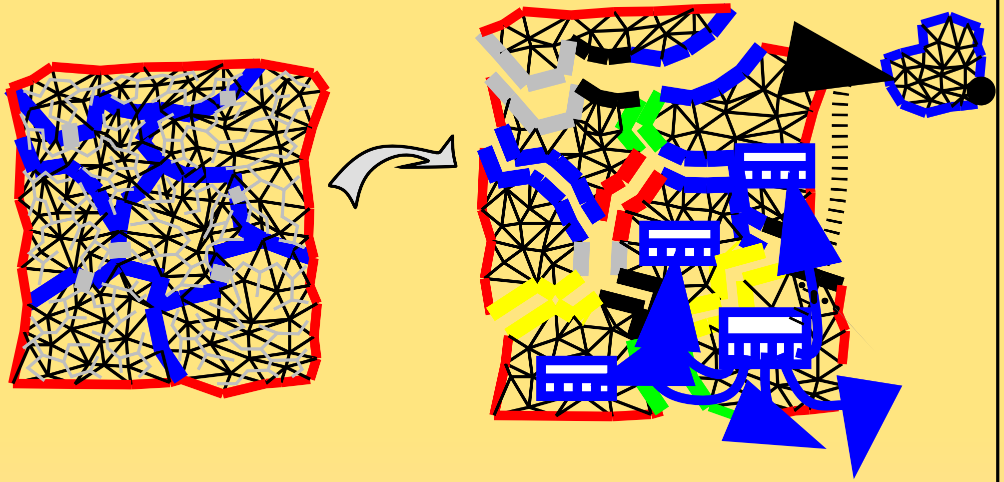
Level 3: exhaustive catalog of micro region of size $i < C \log n$:

- complete explicit description.

Compact representations of labeled graphs

Compact Representations (WADS'05)
(Castelli Aleardi, Devillers, Schaeffer)

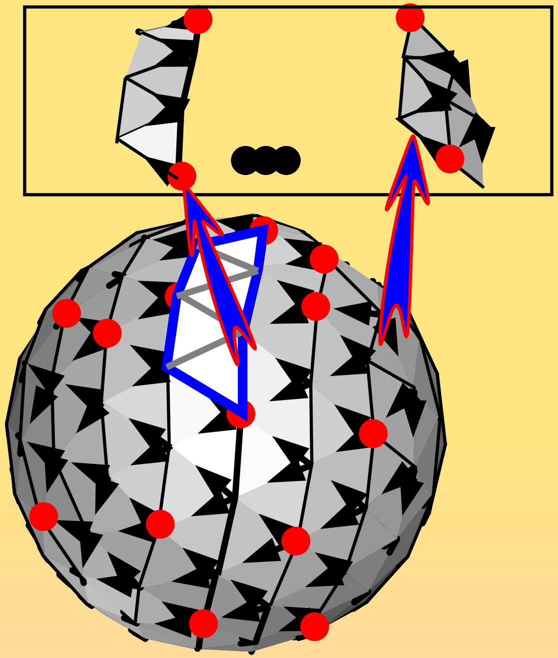
triangulations with a boundary (genus g)



$$2.175m + O(m\frac{\lg\lg m}{\lg m}) \text{ bits}$$

Optimal representations (SoCG'06)
(Castelli Aleardi, Devillers, Schaeffer)

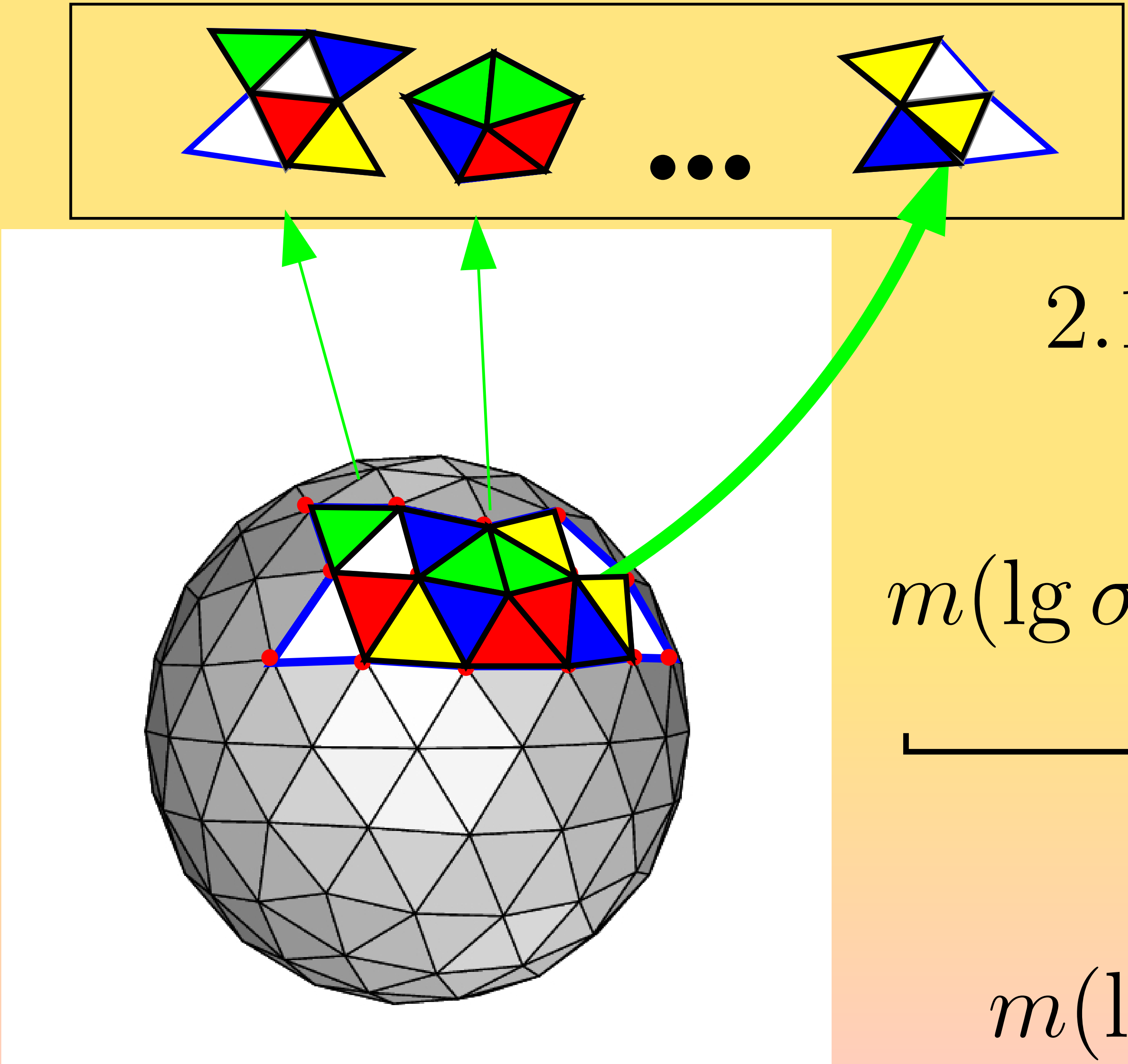
triangulations and 3-connected planar graphs



$$\boxed{1.62m + o(m)} \text{ bits}$$

Compact representations of labeled graphs

A first trade-off



$$\|A\| \approx 2^{\frac{1}{4}2.175r}$$

$2.175m + o(m)$ bits

$+$

$m(\lg\sigma + \boxed{O(\sigma^{\frac{\lg\lg\lg m}{\lg\lg m}})})$ bits

$m(\lg\sigma + o(\lg\sigma)$ bits

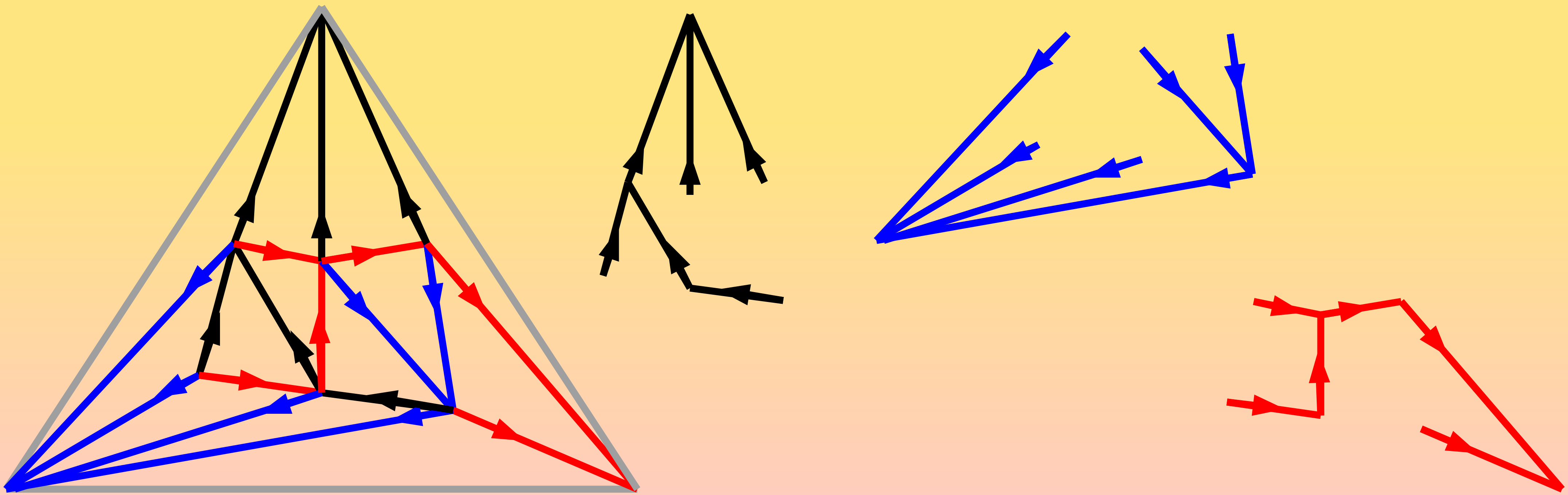
Our solution for representing labeled graphs

A nice characterization of planar graphs

Order dimension, Schnyder's woods and dimension of a graph

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

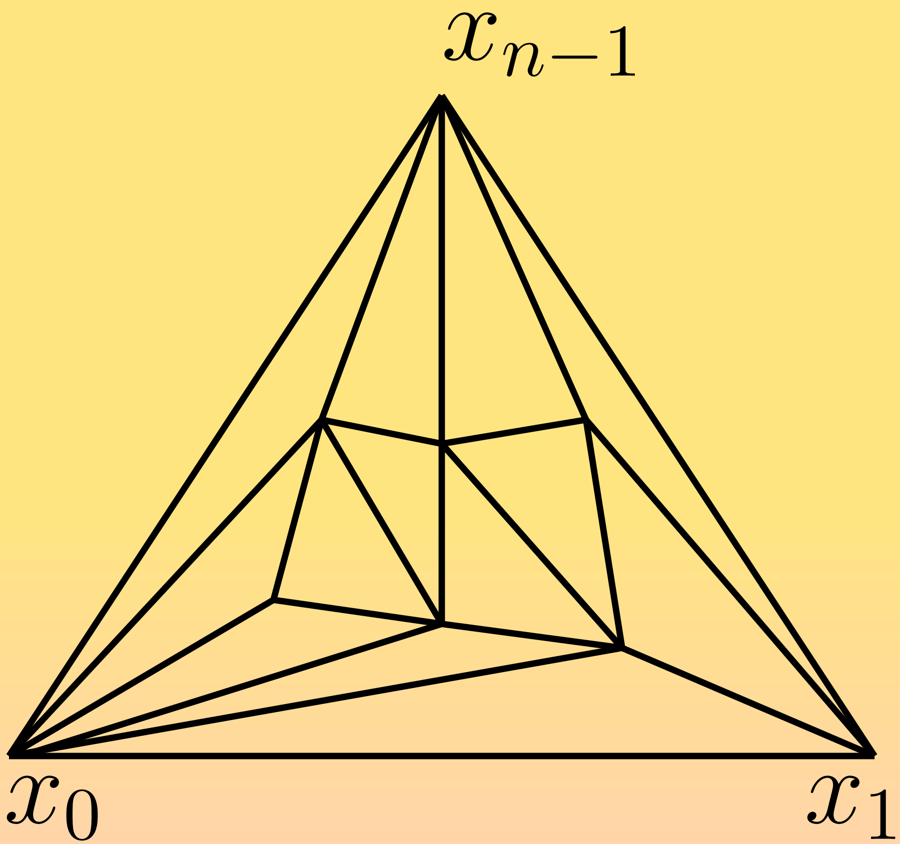
Theorem (Schnyder '90)
A graph G is planar if and only if the dimension of its incidence poset is at most 3



A nice characterization of planar graphs

Realizer of a planar triangulation (Schnyder '90)

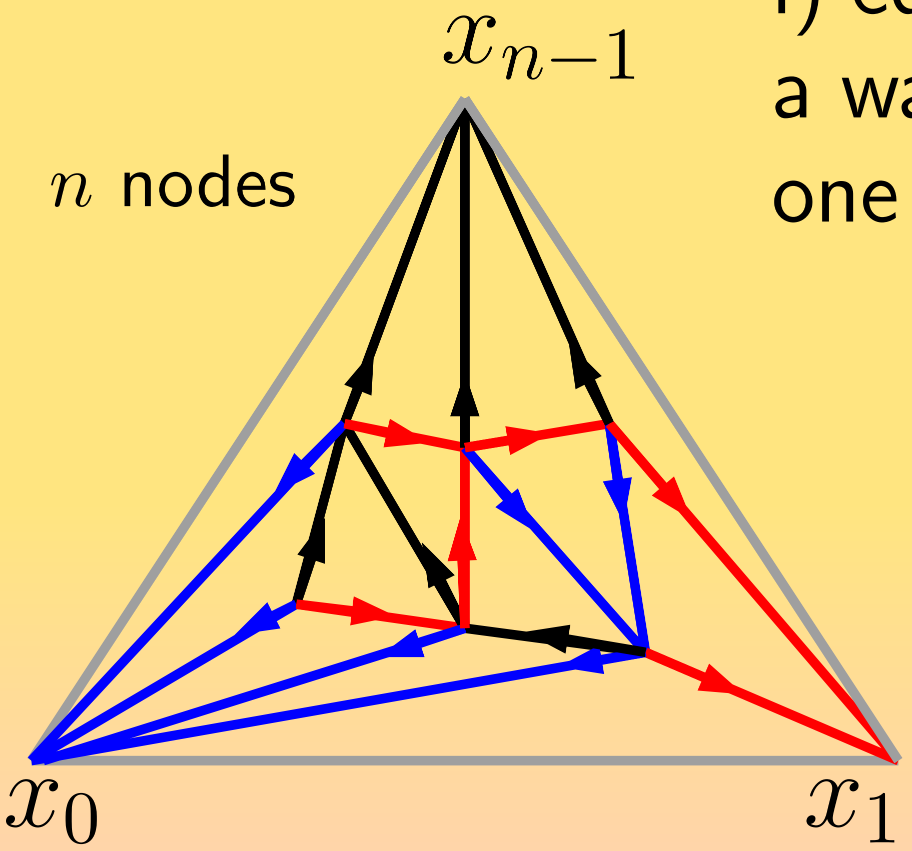
Let T be a triangulation having outer face $\{x_0, x_1, x_{n-1}\}$.
with n nodes



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. :

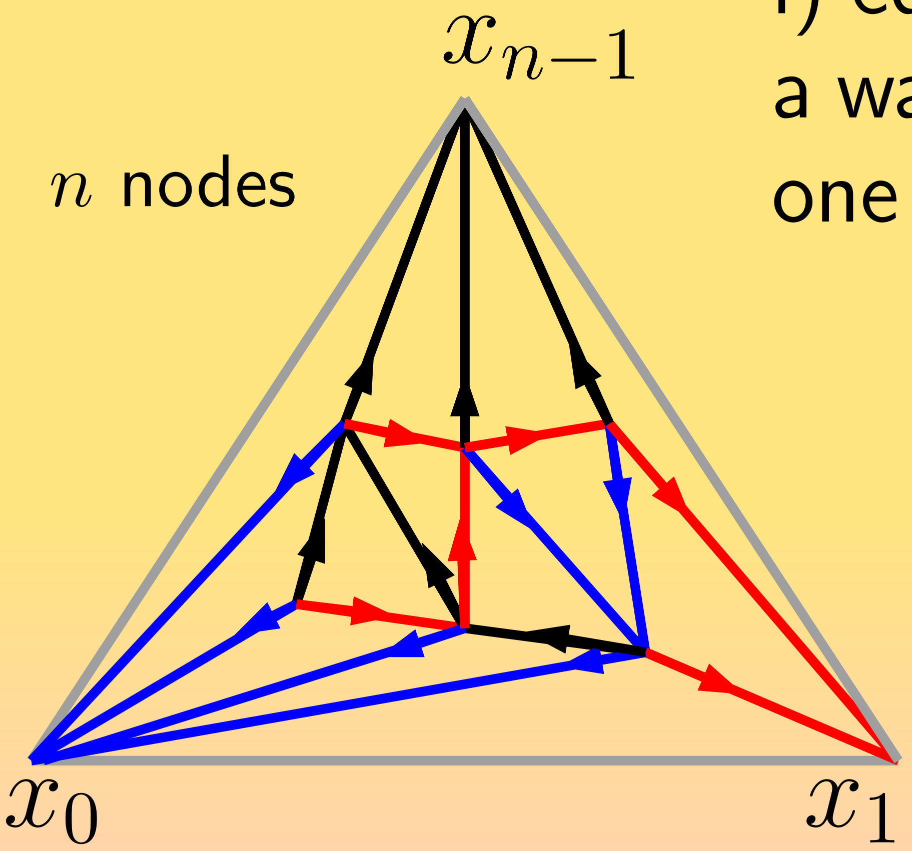


i) edge are colored and oriented in such a way that each inner node has exactly one outgoing edge of each color

A nice characterization of planar graphs

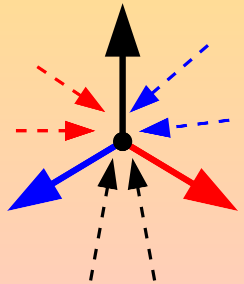
Realizer of a planar triangulation (Schnyder '90)

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

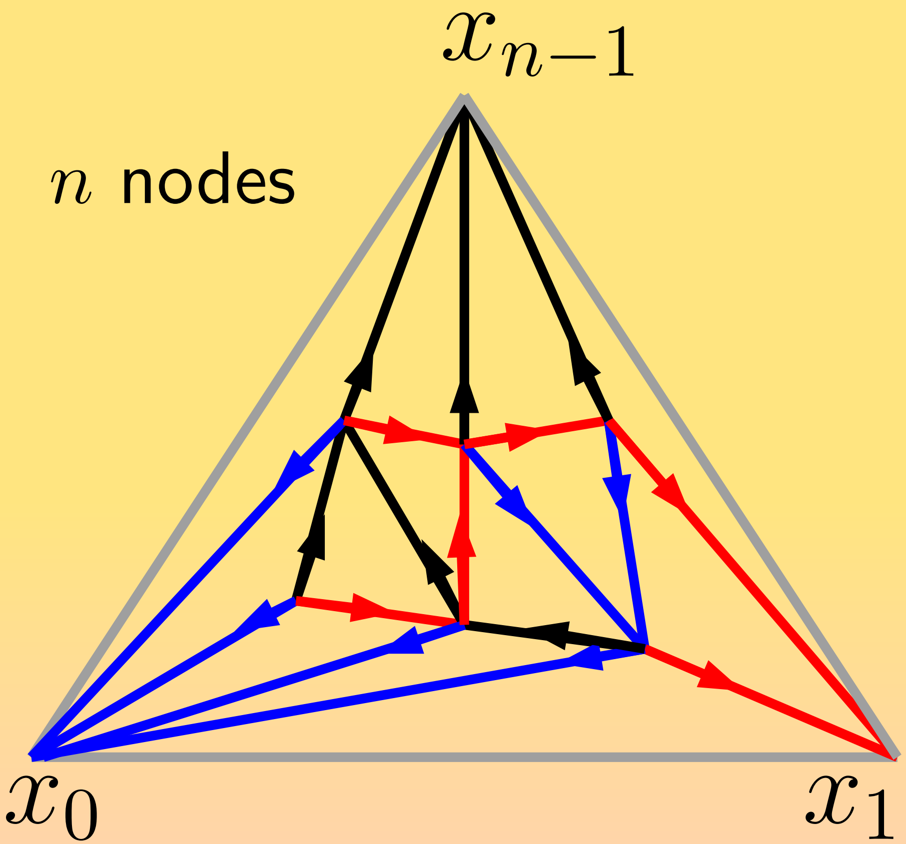
ii) colors and orientations around each inner node must respect the local Schnyder condition

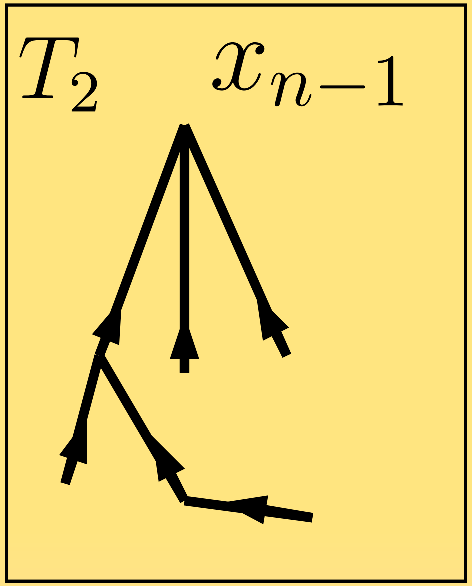


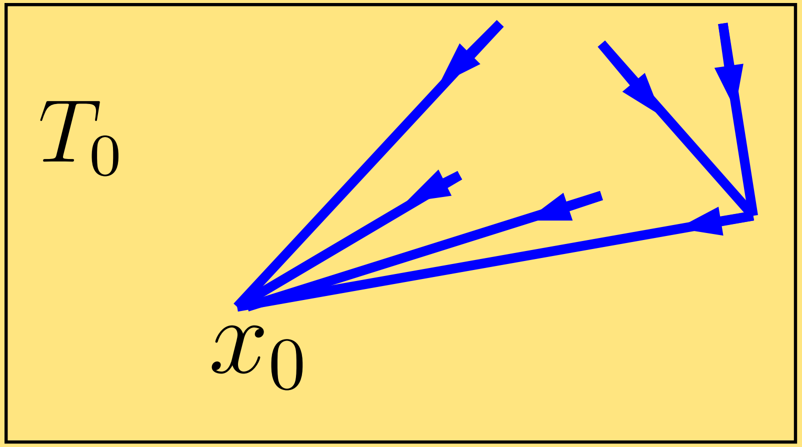
A nice characterization of planar graphs

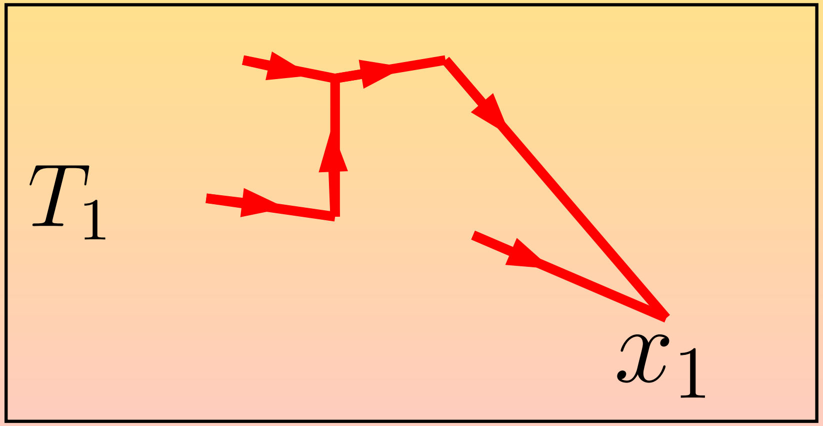
Realizer of a planar triangulation (Schnyder '90)

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



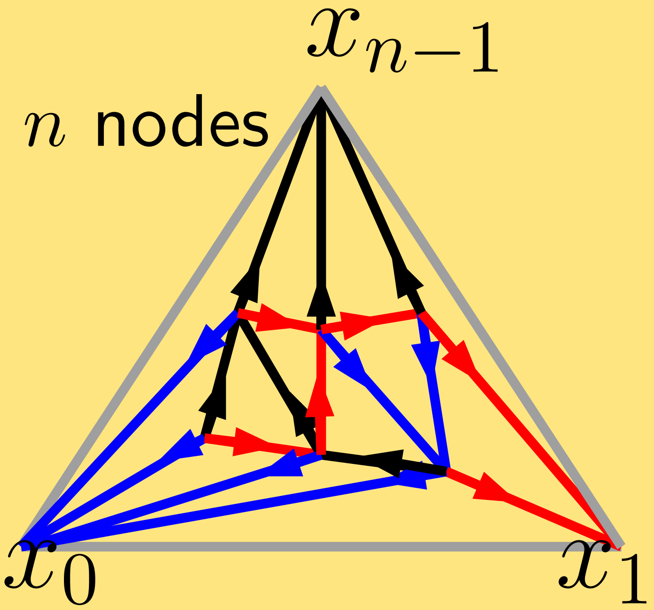


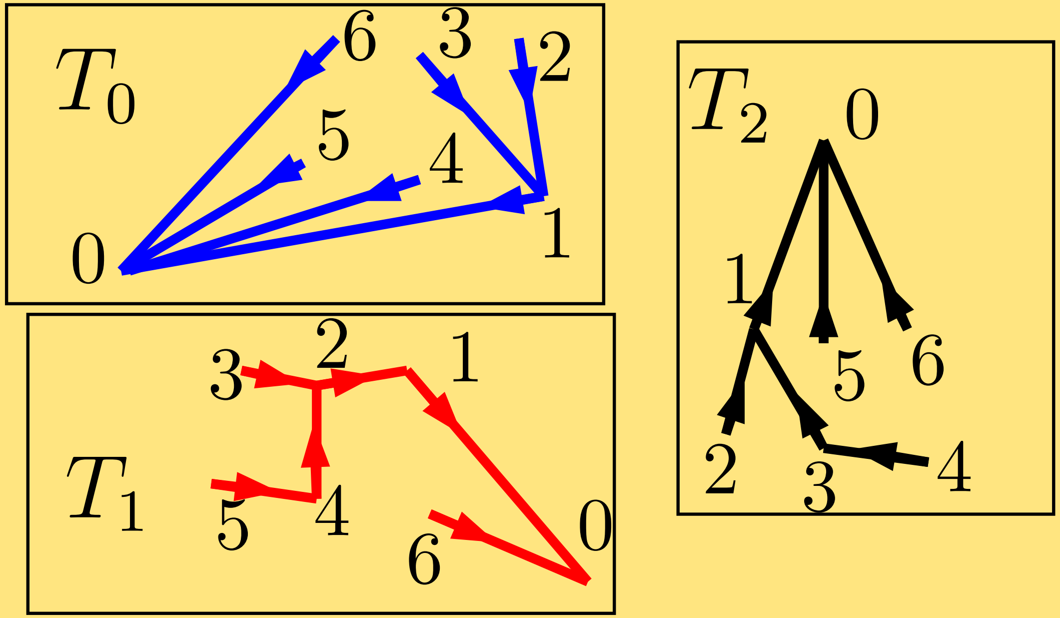




Dimension of a graph

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





$$L_0 : v_1 < v_2 < v_3 < v_4 < v_5 < v_6$$

$$L_1 : v_2 < v_3 < v_6 < v_4 < v_5 < v_1$$

$$L_2 : v_2 < v_3 < v_6 < v_1 < v_3 < v_2$$

these 3 orders do not suffice for succinct encoding

adjacent vertices have not "consecutive" numbers

Succinct labeled triangulations

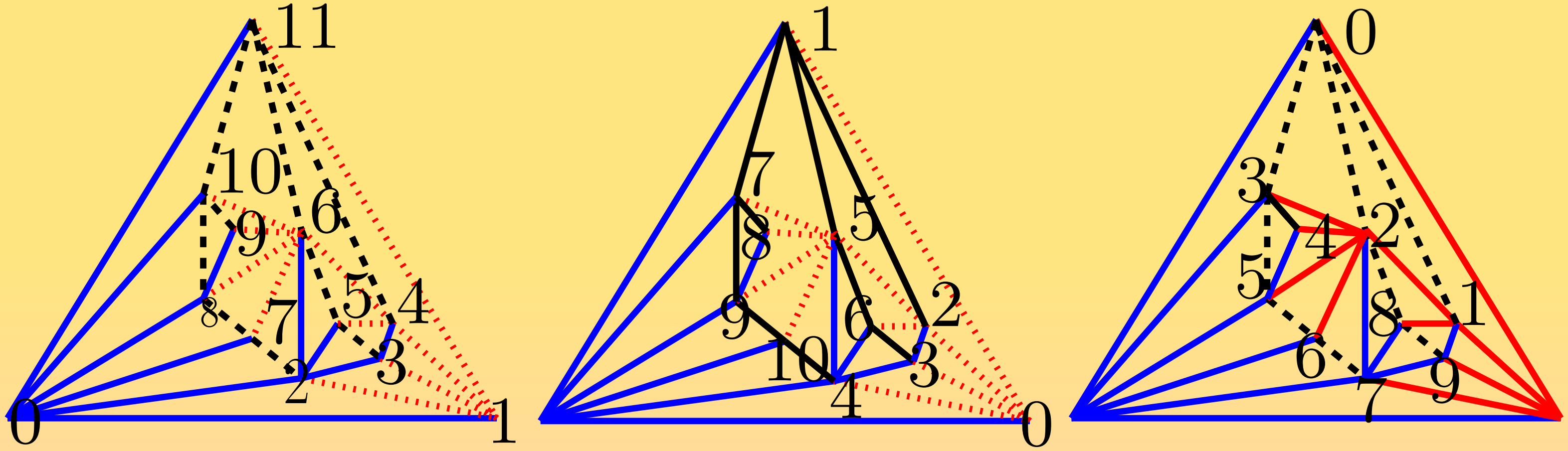
Main idea: find new good orders on the vertices of a graph

Our solution: general overview

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

Solution

Combinatorial properties of planar graphs + 3 new orders + rank/select on strings



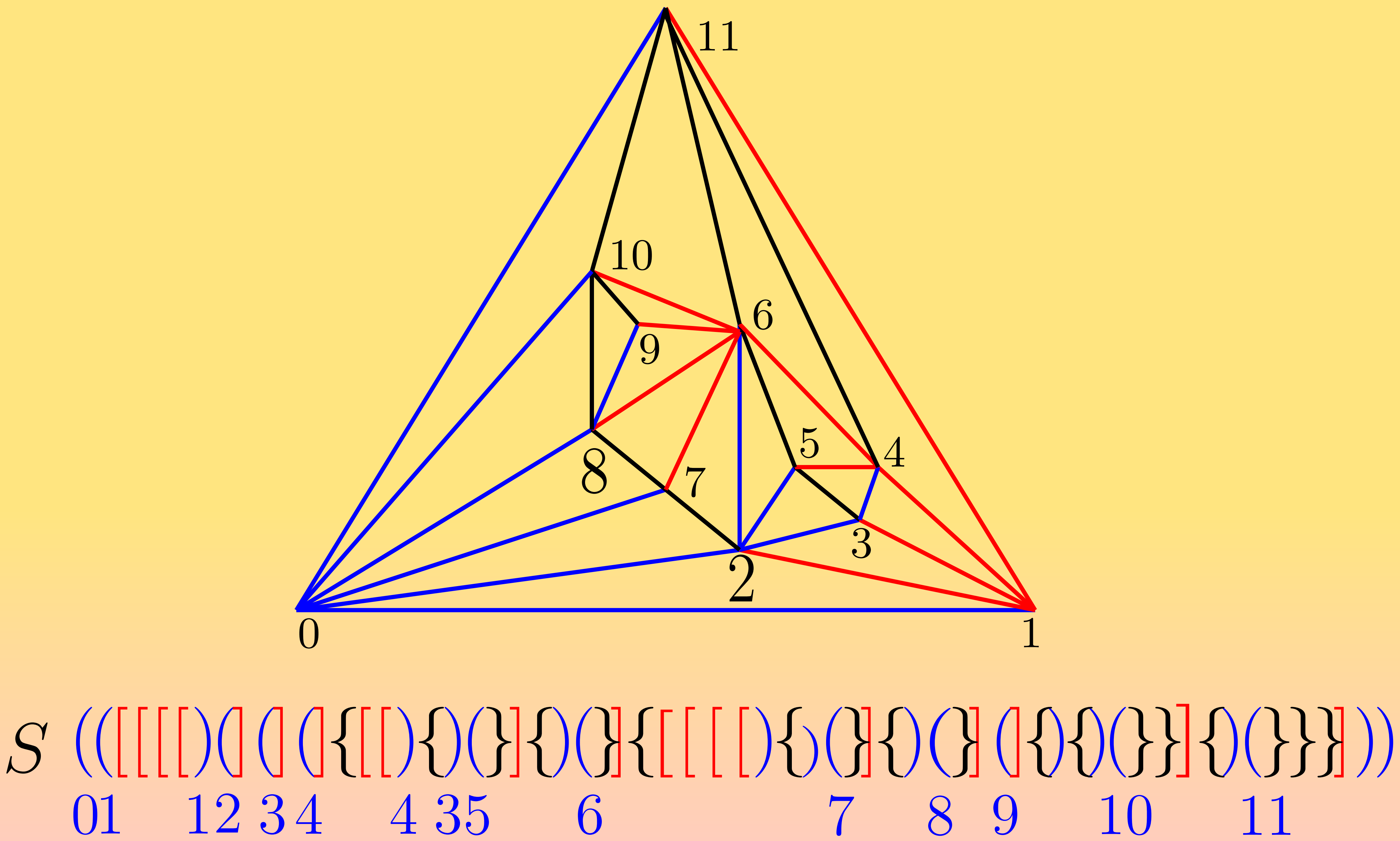
S (([[[[)(](](]{[[){)(}]{)(}]{[[[[){)(}]{)(}](]{){)(}}]{)(}}}]))

01 12 3 4 4 35 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 a triangulation (with multiple parenthesis strings)

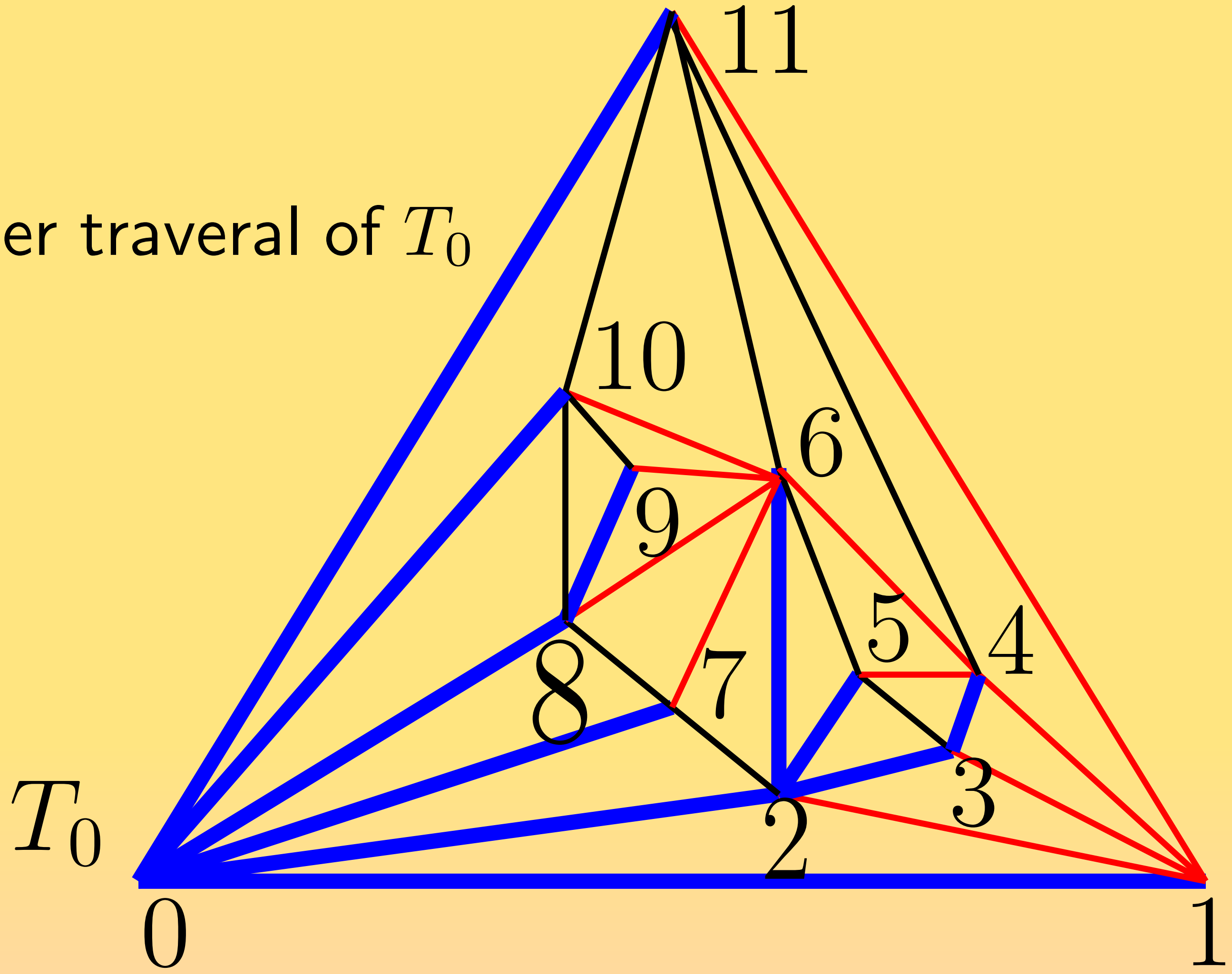
New orders on the vertices and multiple parenthesis encoding



First order

New orders on the vertices and multiple parenthesis encoding

ccw preorder traveral of T_0

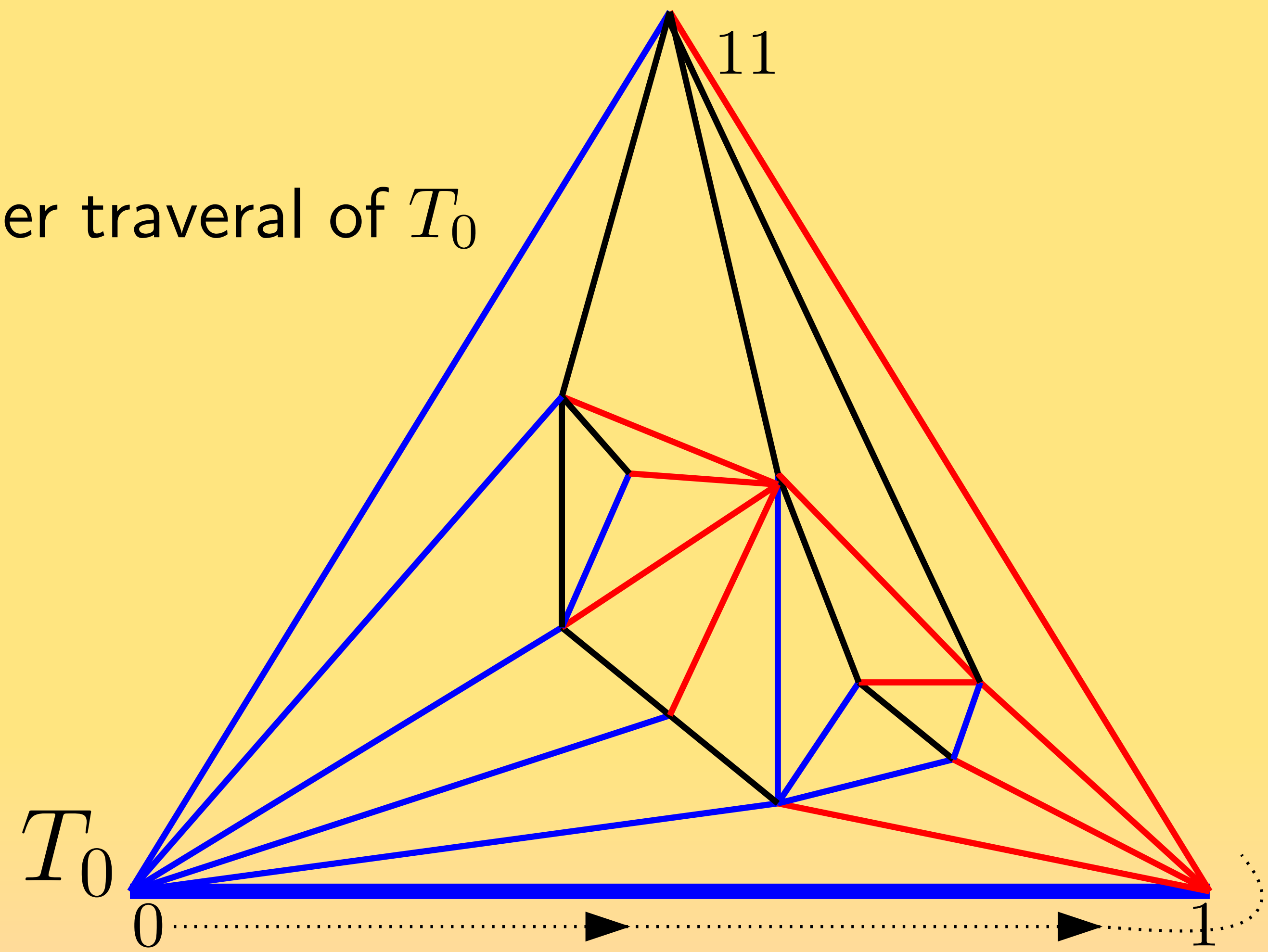


$$\pi_0 : v_0 < v_1 < v_2 < \ldots < v_{10} < v_{11}$$

Second order

Constructing the parenthesis representation

ccw preorder traveral of T_0



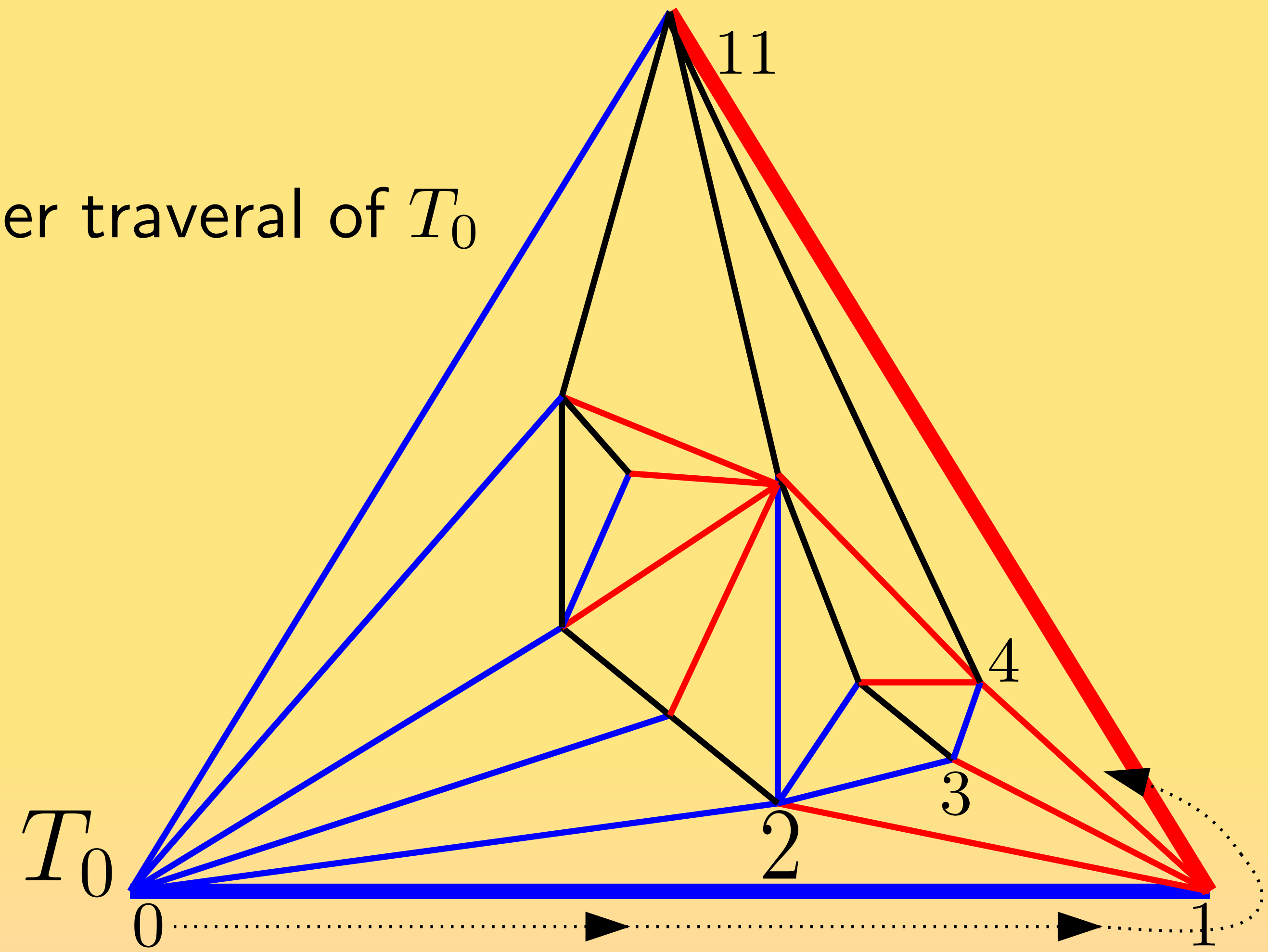
S ((

01

Second order

Constructing the parenthesis representation

ccw preorder traveral of T_0



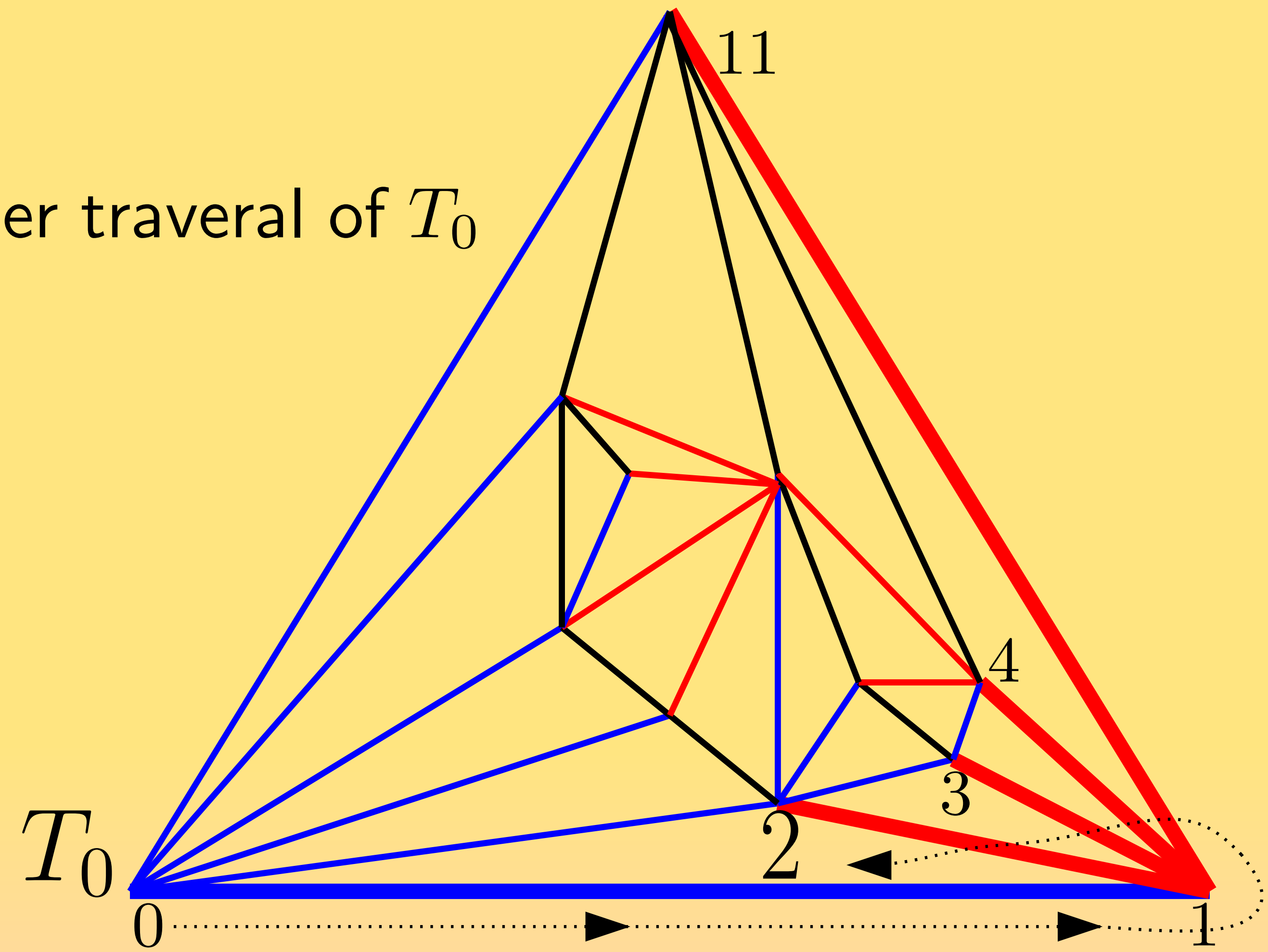
S $(([$
01

$\pi_1 : v_1 < v_{11}$

Second order

Constructing the parenthesis representation

ccw preorder traveral of T_0



S (([[[[

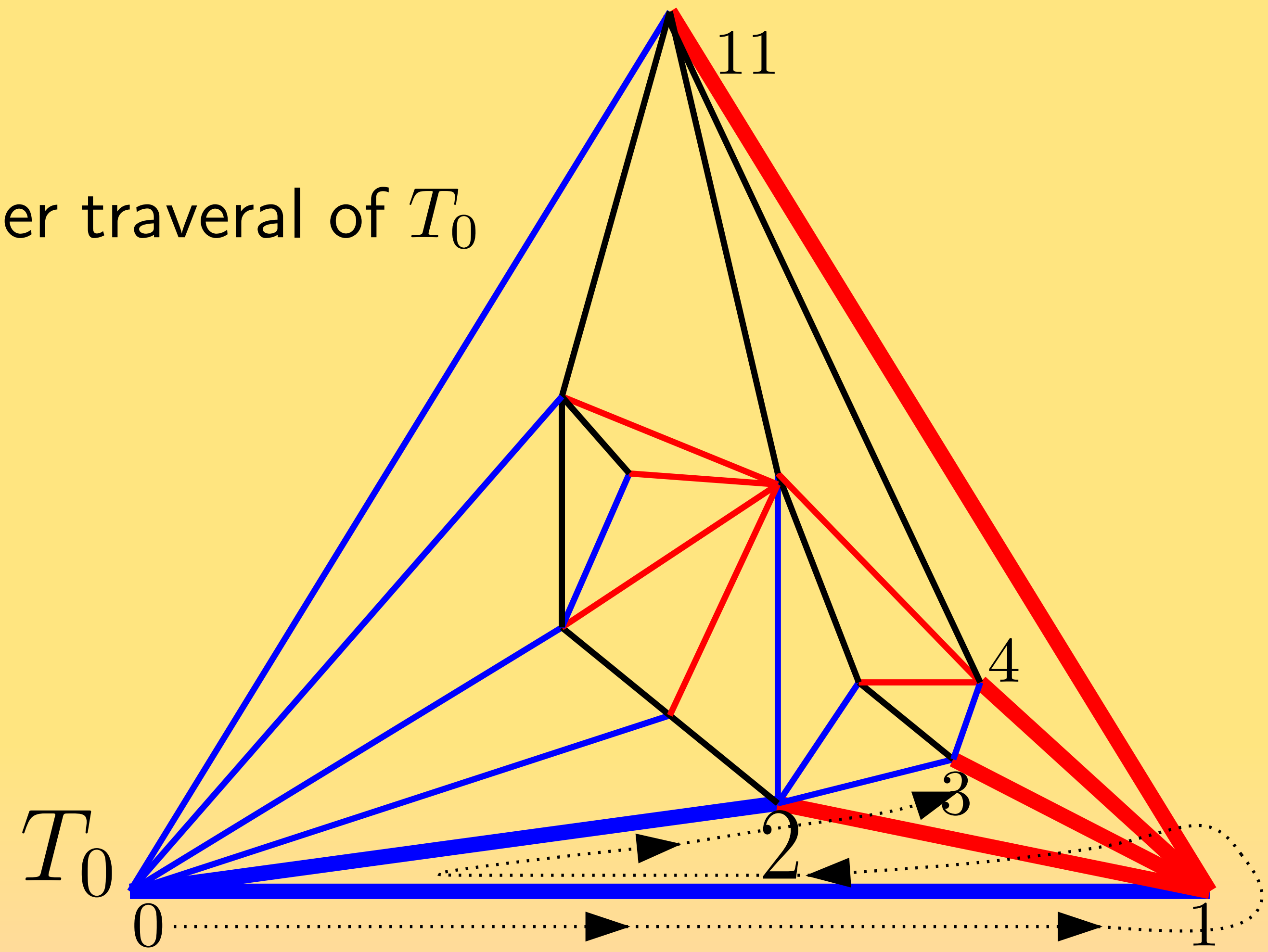
01

$$\pi_1 : v_1 < v_{11} < v_4 < v_3 < v_2$$

Second order

Constructing the parenthesis representation

ccw preorder traveral of T_0



S (([[[[)(]

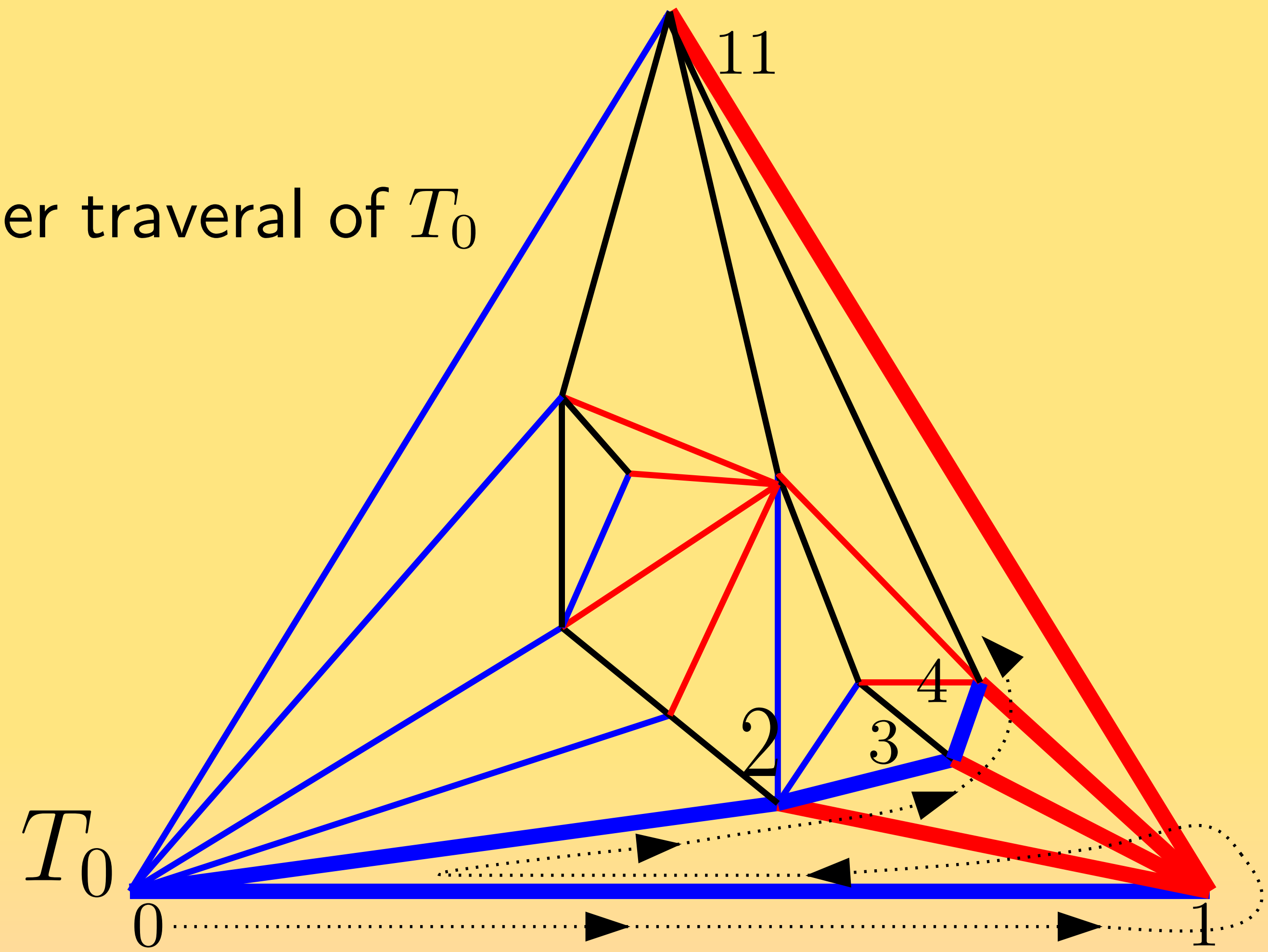
01 12

$$\pi_1 : v_1 < v_{11} < v_4 < v_3 < v_2$$

Second order

Constructing the parenthesis representation

ccw preorder traveral of T_0



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

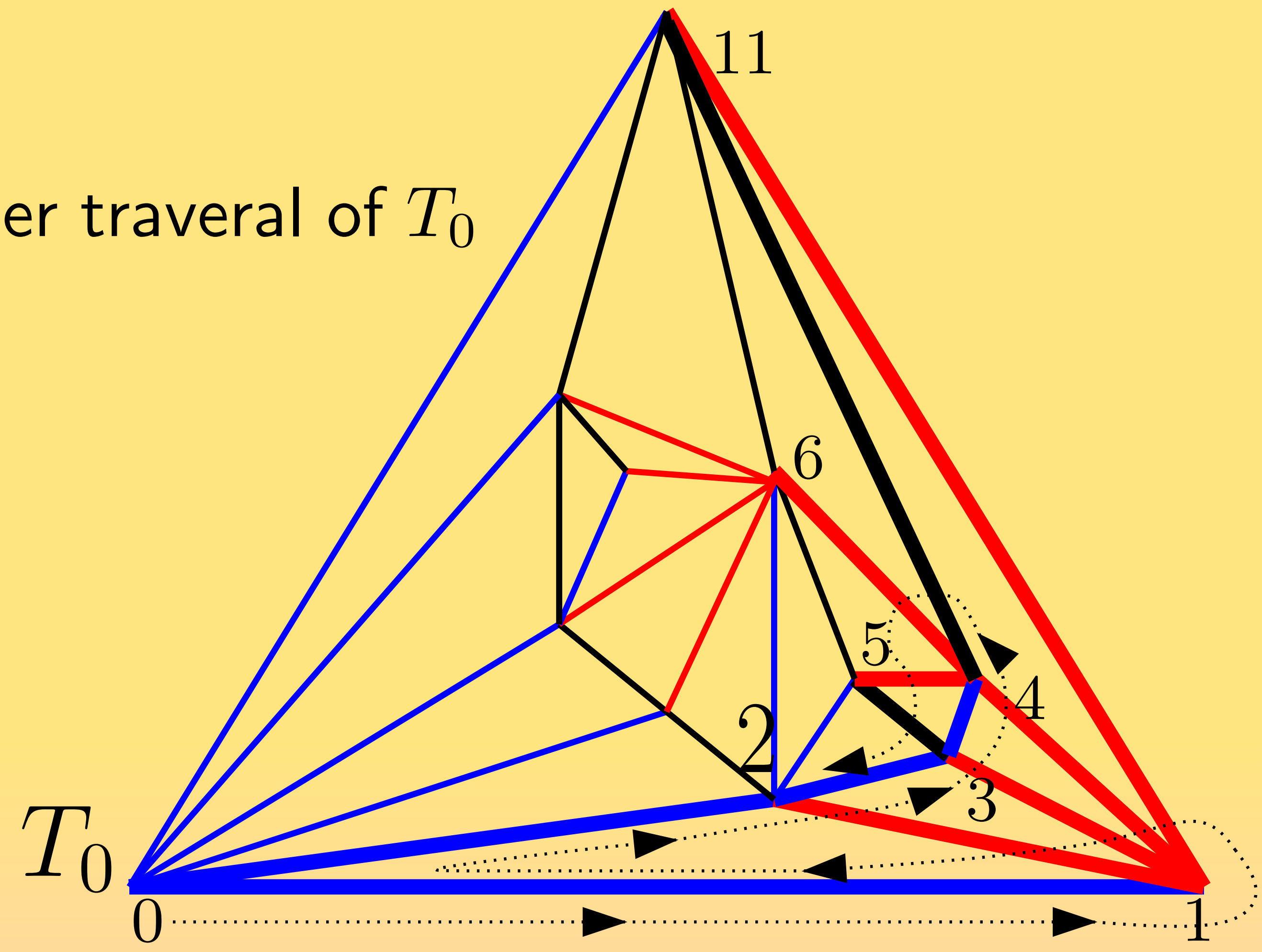
01 12 3 4

$\pi_1 : v_1 < v_{11} < v_4 < v_3 < v_2$

Second order

Constructing the parenthesis representation

ccw preorder traveral of T_0



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

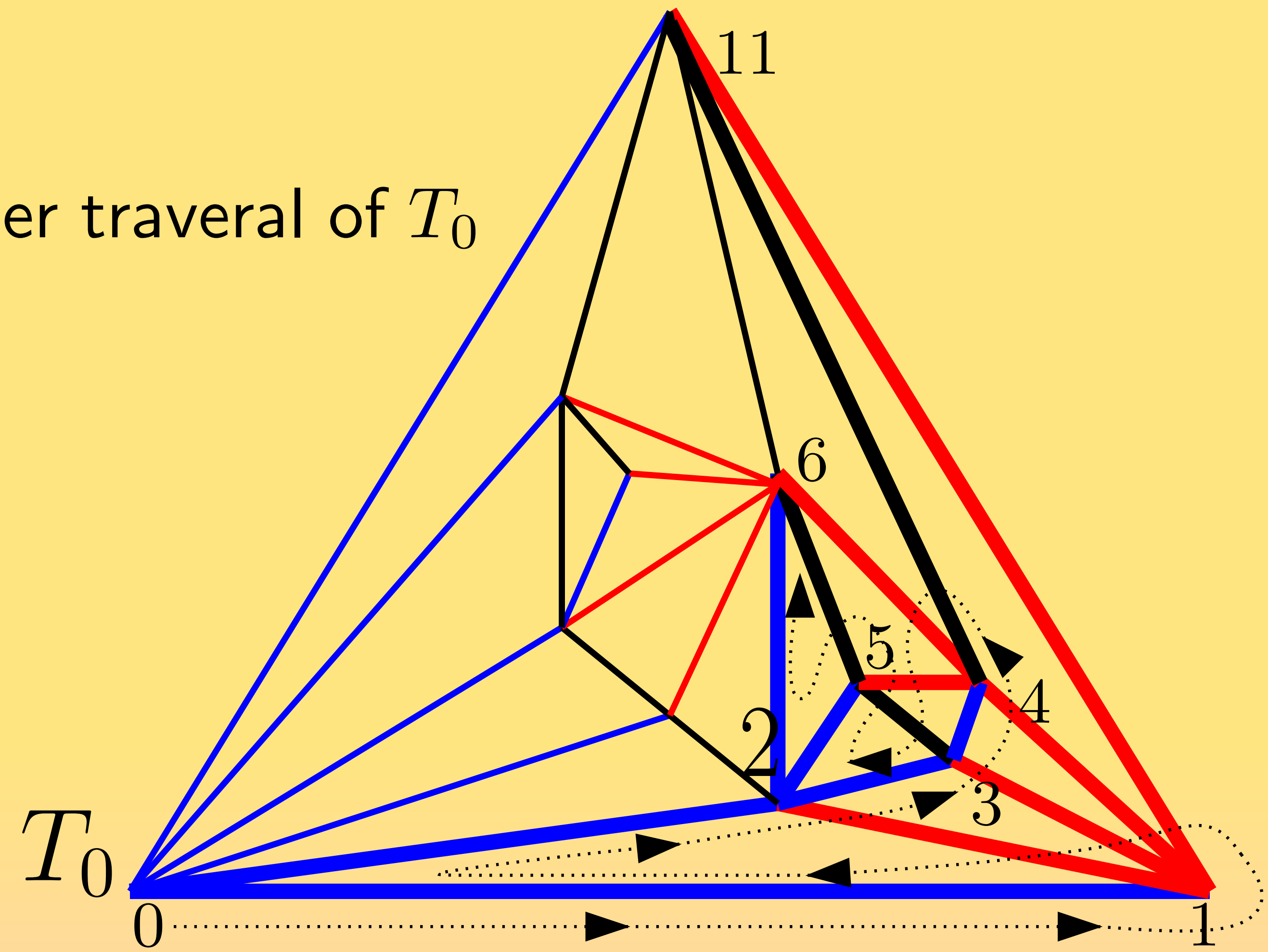
$0\,1 \qquad 1\,2\;3\;4 \qquad 4\;3$

$\pi_1 : v_1 < v_{11} < v_4 < v_3 < v_2 < v_6 < v_5$

Second order

Constructing the parenthesis representation

ccw preorder traveral of T_0



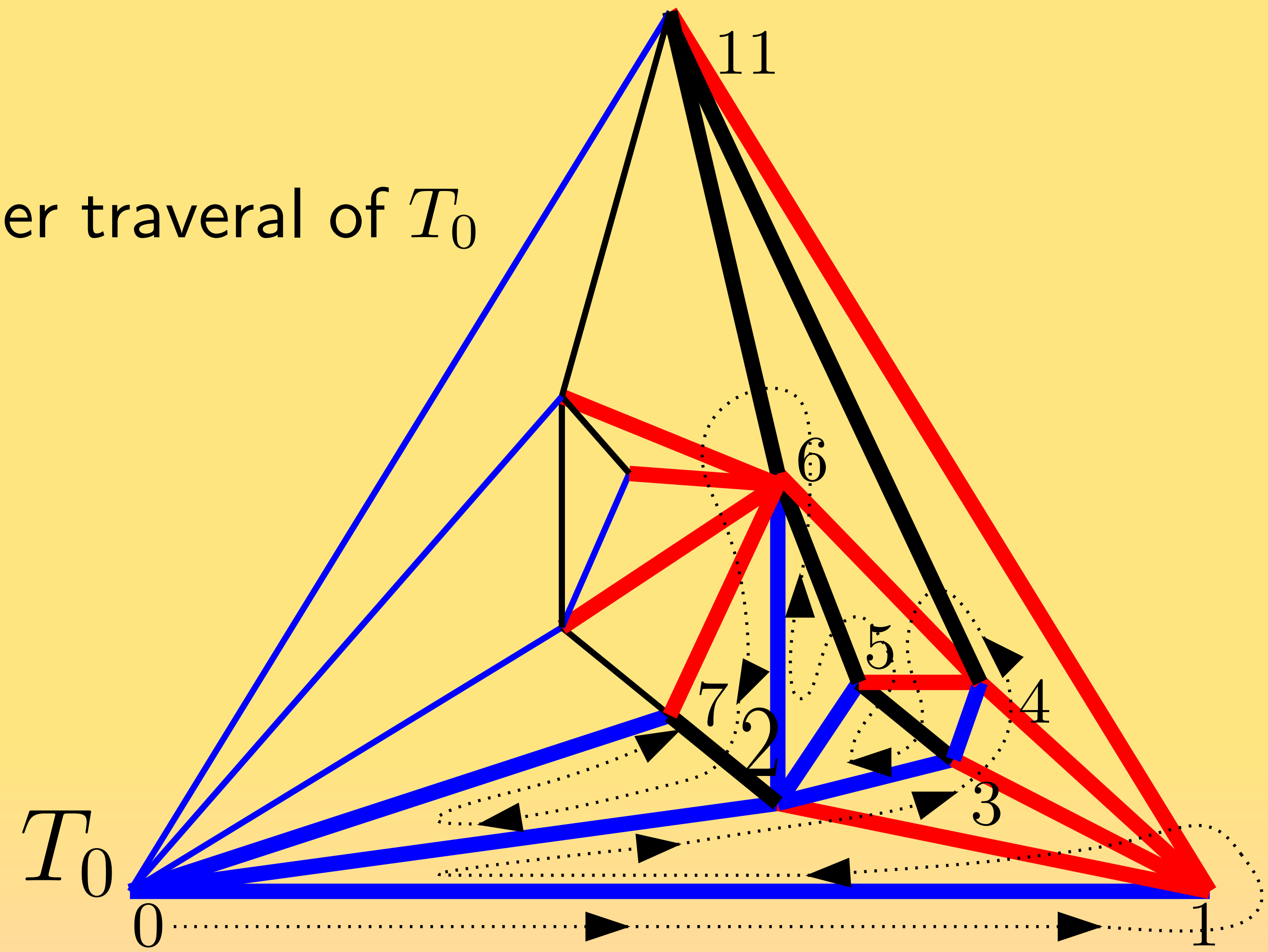
S $(([[[[)(](](]\{[[)\{)(\}]\{)($ $\pi_1 : v_1 < v_{11} < v_4 < v_3 < v_2 < v_6 < v_5$

01 12 3 4 4 35 6

Second order

Constructing the parenthesis representation

ccw preorder traveral of T_0



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

01 12 3 4 4 35 6 7

Second order

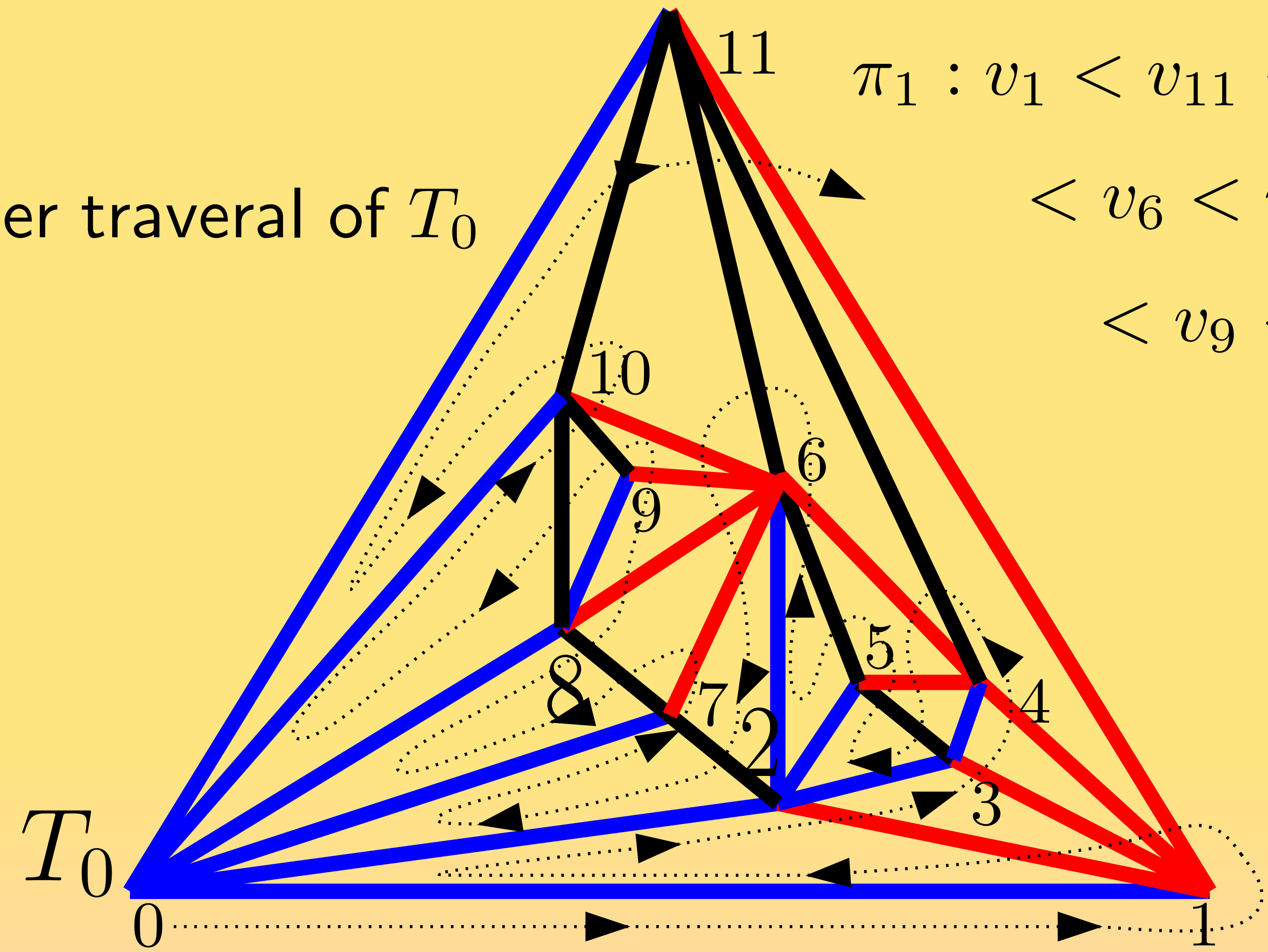
Constructing the parenthesis representation

$\boxed{\text{ccw}}$ preorder traveral of T_0

$\pi_1 : v_1 < v_{11} < v_4 < v_3 < v_2 <$

$< v_6 < v_5 < v_{10} <$

$< v_9 < v_8 < v_7$



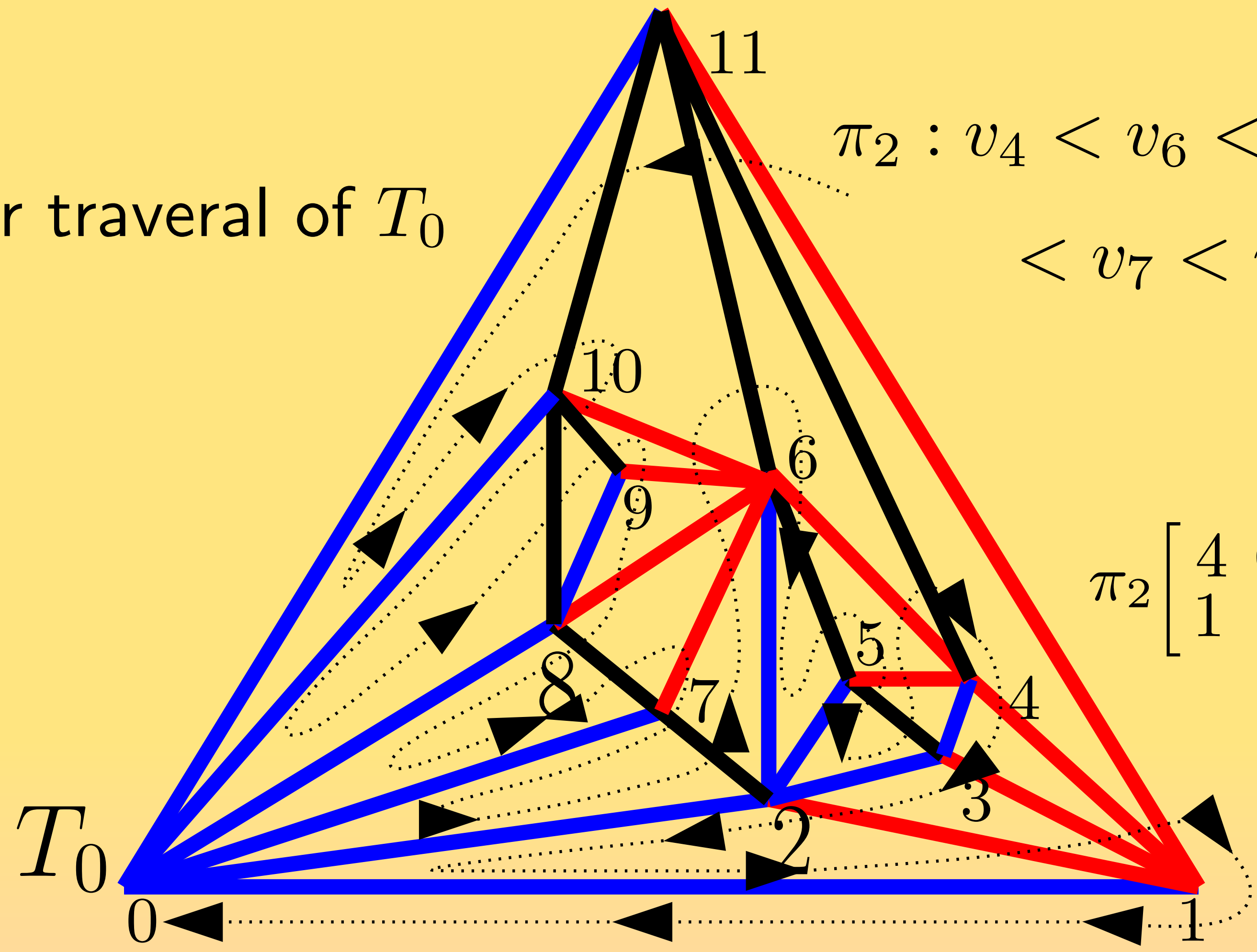
S (([[[[)(](](]{[[){)(}]{)(}]{[[[){)(}]{)(}](]{){)(}}]{)(}}}]))

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

Third order

Constructing the parenthesis representation

cw preorder traveral of T_0



$\pi_2 : v_4 < v_6 < v_{10} < v_9 < v_8 <$

$< v_7 < v_2 < v_5 < v_3 <$

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

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

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

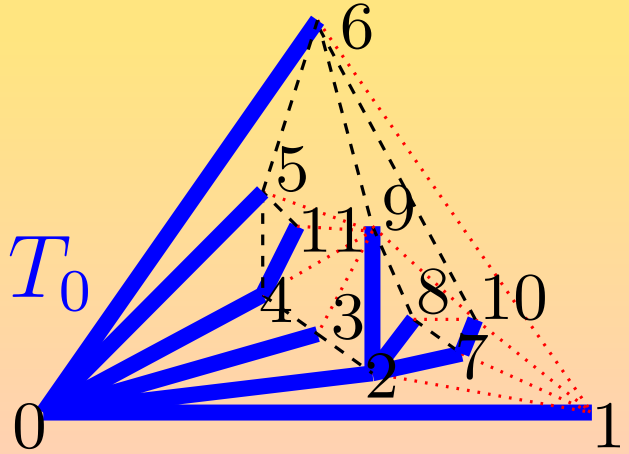
Important facts about π_0, π_1 and π_2

Lemma

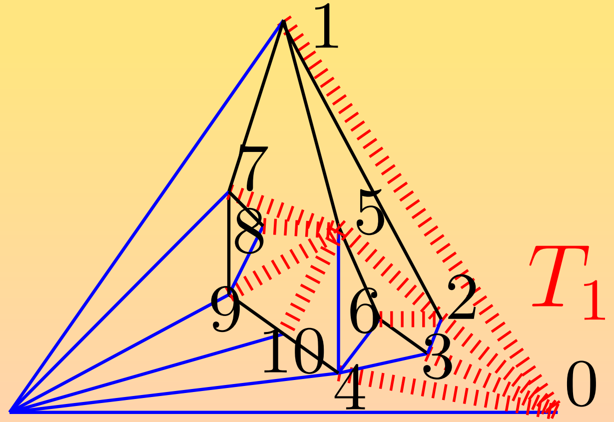
For any node x, its children in T_1 (resp. T_2), listed in ccw order (resp. cw order) have consecutive numbers in π_1 (resp. π_2).
In the case of T_0, children are listed consecutively by a `DFUDS` traversal of T_0

Lemma

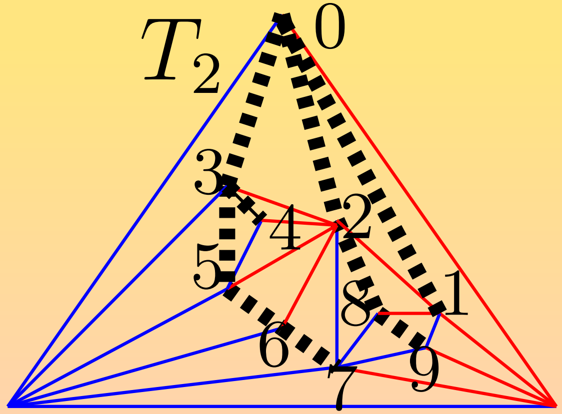
For any node x, the transformations between π_0, π_1 and π_2 can be computed in $O(1)$ time, with a combination of `Rank`/`Select` operations on the string S



`DFUDS` on T_0



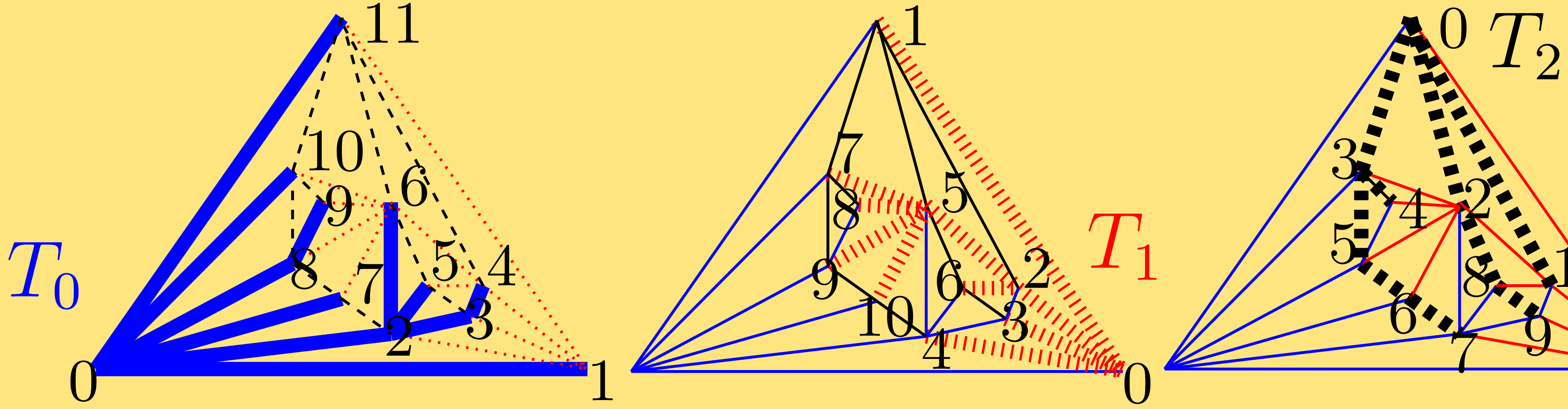
$$\pi_1 \begin{bmatrix} 11 & 4 & 3 & 2 & 6 & 5 & 10 & 9 & 8 & 7 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \end{bmatrix}$$



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

Last step

Multiple parenthesis encoding



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

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

S_2 { { } { } { { } { } { { } } { } } }

3 5 2 7 8 9 10 6 4

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

A succinct representation of (vertex) labeled graphs

(Barbay, Castelli Aleardi, He and Munro, Isaac '07)

Theorem
For labeled planar triangulations with e edges and n vertices , with σ labels associated with vertices, there exists a succinct representation which requires

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

while supporting labeled based navigation in

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

For large σ

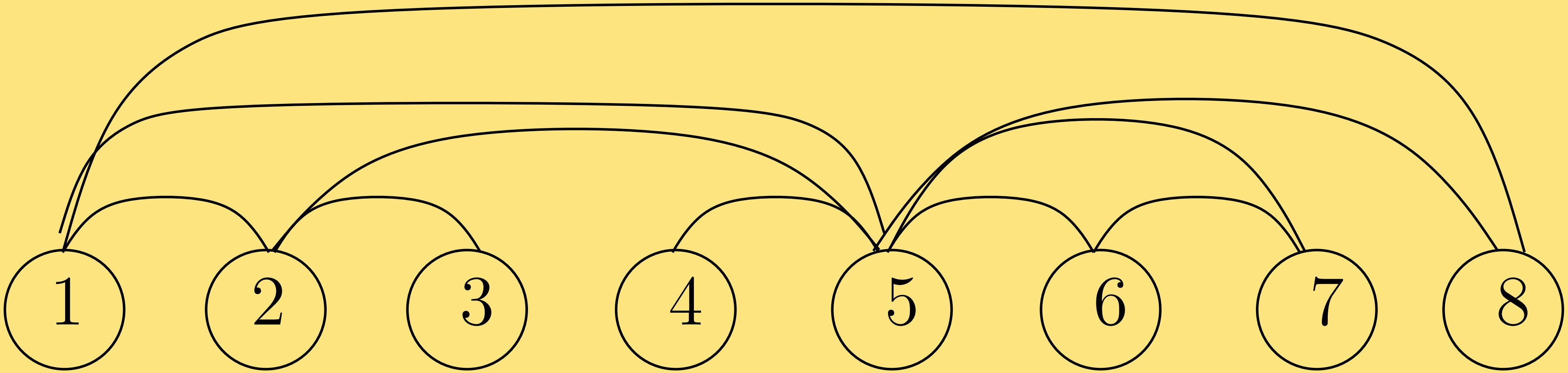
$$(2\log_2 6)e \approx 3(2\log_2 6)n << n \cdot \lg\sigma$$

Another result

(for edge labeled graphs)

Book embedding and planar graphs

One page graph (outerplanar graph)



$(((\quad)((\quad) \quad (\quad)))(((\quad)(\quad)) \quad))$

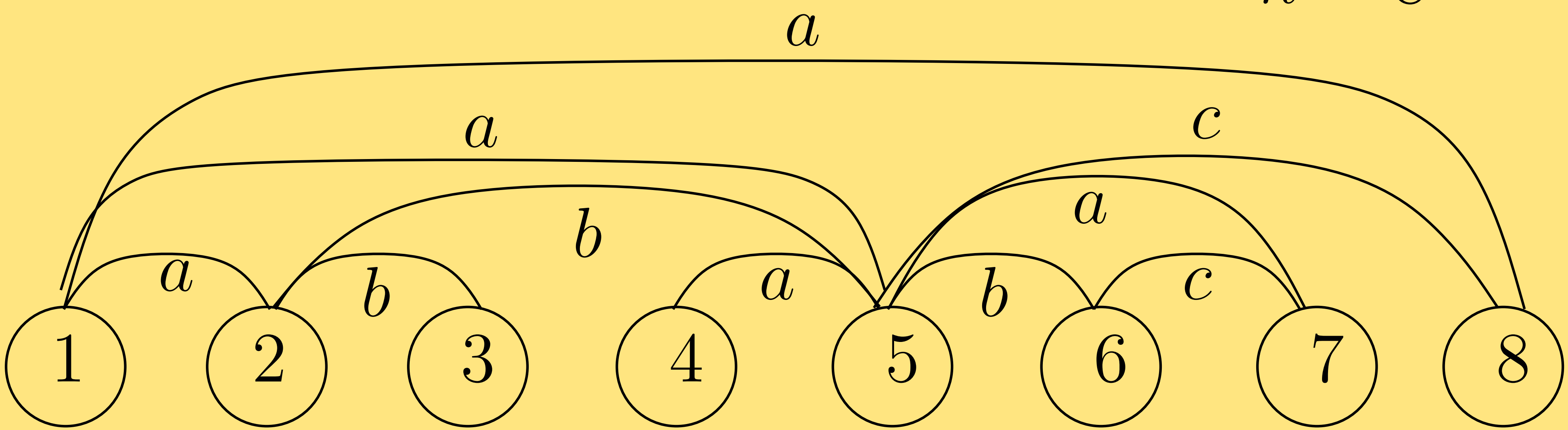
Planar graphs admit a 4-pages book embedding

Book embedding and planar graphs

One page (edge) labeled graph

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

$n = 8$

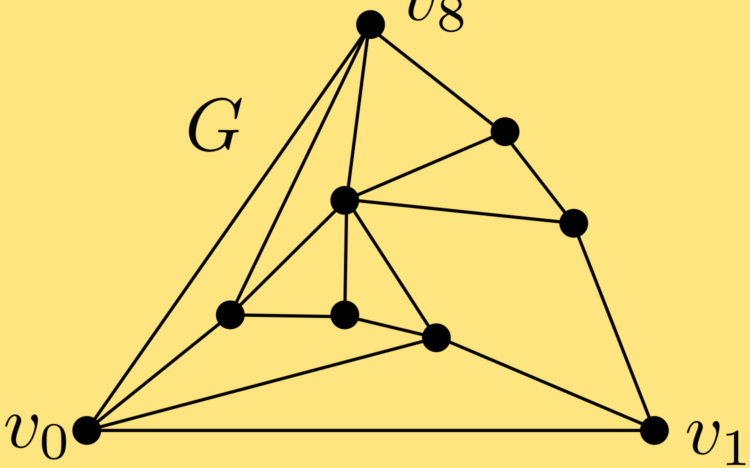


(((	)((	)	(	)))(((	)(	))	))
1000	1000	10	10	1000000	100	100	100
aaa	bb		a	cab	c		
	a	b		aba	b	ca	ca

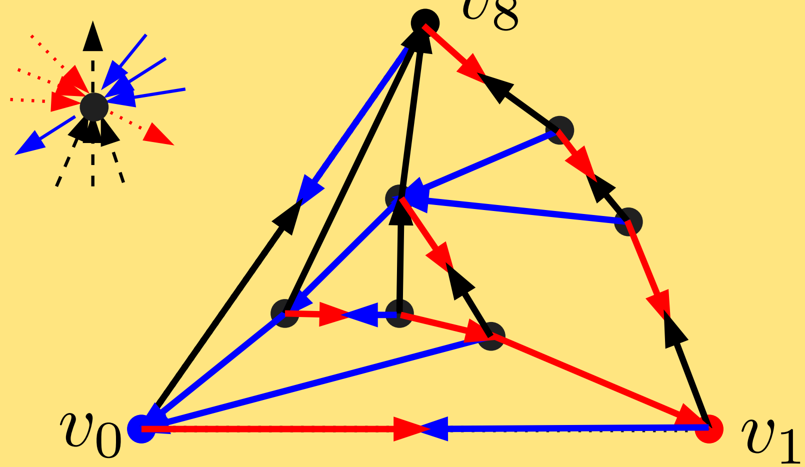
future work

Future work

- Extension to polygonal meshes



3-connected planar graph



Schnyder wood

- 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