Dessin de triangulations: algorithmes, combinatoire, et analyse

Éric Fusy

Projet ALGO, INRIA Rocquencourt et LIX, École Polytechnique
Motivations

Display of large structures on a planar surface
Planar maps

- A **planar map** is obtained by embedding a planar graph in the plane **without edge crossings**.
- A planar map is defined **up to continuous deformation**

![Planar map](image1)

**Planar map**

![The same planar map](image2)

**The same planar map**

![Quadrangulation](image3)

**Quadrangulation**

![Triangulation](image4)

**Triangulation**
Planar maps

- Planar maps are combinatorial objects
- They can be encoded without dealing with coordinates

Planar map

Choice of labels

Encoding: to each vertex is associated the (cyclic) list of its neighbours in clockwise order

1: (2, 4)
2: (1, 7, 6)
3: (4, 6, 5)
4: (1, 3)
5: (3, 7)
6: (3, 2, 7)
7: (2, 5, 6)
Combinatorics of maps

Triangulations

Quadrangulations

Tetravalent

3 spanning trees

2 spanning trees

eulerian orientation

\[ |T_n| = \frac{2(4n-3)!}{n!(3n-2)!} \]

\[ |Q_n| = \frac{2(3n-3)!}{n!(2n-2)!} \]

\[ |E_n| = \frac{2 \cdot 3^n (2n)!}{n!(n+2)!} \]

⇒ Tutte, Schaeffer, Schnyder, De Fraysseix et al...
A particular family of triangulations

- We consider triangulations of the 4-gon (the outer face is a quadrangle)
- Each 3-cycle delimits a face (irreducibility)
Transversal structures

We define a transversal structure using local conditions. Inner edges are colored blue or red and oriented:

- Inner vertex
- Border vertices

Example:
**Link with bipolar orientations**

bipolar orientation = **acyclic** orientation with a unique minimum and a unique maximum
The blue (resp. red) edges give a bipolar orientation
The two bipolar orientations are **transversal**
bipolar orientation = acyclic orientation with a unique minimum and a unique maximum
The blue (resp. red) edges give a bipolar orientation
The two bipolar orientations are transversal
Definition and properties of transversal structures on triangulations
Planar maps

- A planar map is obtained by drawing a planar graph in the plane \textit{without edge crossings}.
- A planar map is defined \textit{up to continuous deformation}.
Planar maps

- Planar maps are **combinatorial** objects
- They can be encoded **without dealing with coordinates**

Encoding: to each vertex is associated the (cyclic) list of its neighbours in clockwise order:

1: (2, 4)
2: (1, 7, 6)
3: (4, 6, 5)
4: (1, 3)
5: (3, 7)
6: (3, 2, 7)
7: (2, 5, 6)
A particular family of triangulations

- We consider **triangulations of the 4-gon** (the outer face is a quadrangle)
- Each **3-cycle delimits a face** (irreducibility)
Transversal structures

We define a **transversal structure** using local conditions. Inner edges are colored **blue** or **red** and oriented:

Inner vertex ⇒

Border vertices ⇒

Example:

cf Regular edge 4-labelings (Kant, He)
bipolar orientation = \textbf{acyclic} orientation with a unique minimum and a unique maximum
The \textcolor{blue}{blue} (resp. \textcolor{red}{red}) edges give a bipolar orientation
The two bipolar orientations are \textcolor{red}{transversal}
Overview

- Transversal structures
- Existence Properties
- Analysis
- Applications
- Drawing
- Lattice
- Grid size: \( \frac{11}{27} n \times \frac{11}{27} n \)

Bijection:

\[ T_n = \frac{4}{2n+2} \frac{(3n)!}{n!(2n+1)!} \]
Definition and properties of transversal structures on triangulations
Existence of transversal structures

- For each triangulation, there exists a transversal structure (Kant, He 1997)
- There exists linear time iterative algorithms to compute transversal structures
Existence of transversal structures

- For each triangulation, there exists a transversal structure (Kant, He 1997)
- There exists linear time iterative algorithms to compute transversal structures
Existence of transversal structures

- For each triangulation, there exists a transversal structure (Kant, He 1997)
- There exists linear time iterative algorithms to compute transversal structures
Existence of transversal structures

• For each triangulation, there exists a transversal structure (Kant, He 1997)
• There exists linear time iterative algorithms to compute transversal structures
Existence of transversal structures

- For each triangulation, there exists a transversal structure (Kant, He 1997)
- There exists linear time iterative algorithms to compute transversal structures
Existence of transversal structures

• For each triangulation, there exists a transversal structure (Kant, He 1997)

• There exists linear time iterative algorithms to compute transversal structures
Existence of transversal structures

- For each triangulation, there exists a transversal structure (Kant, He 1997)
- There exists linear time iterative algorithms to compute transversal structures
Existence of transversal structures

- For each triangulation, there exists a transversal structure (Kant, He 1997)
- There exists linear time iterative algorithms to compute transversal structures
The structure of transversal structures

- For each triangulation $T$, such transversal structures are not unique
- Let $X_T$ be the set of transversal bicolorations of $T$
- What is the structure of $X_T$?
Without orientations

The orientation of edges are not necessary. The local conditions can be defined just with the bicoloration:

Inner vertex ⇒

Border vertices ⇒

Example:
The two definitions are equivalent

The orientations of edges can be recovered in an unique way
⇒ bijection between the two structures

Local conditions
The two definitions are equivalent

The orientations of edges can be recovered in a unique way
\[ \Rightarrow \text{bijection} \] between the two structures

Local conditions
The two definitions are equivalent

The orientations of edges can be recovered in an unique way
⇒ bijection between the two structures

Local conditions
The two definitions are equivalent

The orientations of edges can be recovered in an unique way
⇒ bijection between the two structures

Local conditions
The two definitions are equivalent

The orientations of edges can be recovered in an unique way

⇒ bijection between the two structures

Local conditions
The two definitions are equivalent

The orientations of edges can be recovered in an unique way
⇒ bijection between the two structures

Local conditions
The two definitions are equivalent

The orientations of edges can be recovered in an unique way
⇒ bijection between the two structures
The set $X_T$ is a distributive lattice.
The set $X_T$ is a distributive lattice.
Angular graph of $T$

We associate to $T$ an angular graph $Q(T)$:

- The black vertices of $Q(T)$ are the vertices of $T$
Angular graph of $T$

We associate to $T$ an angular graph $Q(T)$:

- There is a white vertex of $Q(T)$ in each face of $T$
Angular graph of $T$

We associate to $T$ an angular graph $Q(T)$:

- To each angle of $T$ corresponds an edge of $Q(T)$
Angular graph of $T$

We associate to $T$ an angular graph $Q(T)$:
Angular graph of $T$

We associate to $T$ an angular graph $Q(T)$:
Induced orientation on $Q(T)$

**Vertex-condition:**

**Face-condition:**
Bijective correspondences

Transversal with orientations

Remove orientations

Transversal without orientations

Iterative algorithm

Local rule: angle of $T \to$ edge of $Q(T)$

Orientation of $Q(T)$
- outdegree 4
- outdegree 1
Bijective correspondences

Transversal with orientations → Remove orientations → Transversal without orientations

Iterative algorithm

Local rule
angle of \( T \) → edge of \( Q(T) \)

Orientation of \( Q(T) \):
- ● outdegree 4
- ○ outdegree 1

DISTRIBUTIVE LATTICE
(Ossona de Mendez, Felsner)
Flip on $Q(T)$ and then on $T$
Flip on $Q(T)$ and then on $T$
Flip on $Q(T)$ and then on $T$
Flip on $Q(T)$ and then on $T$
Flip on $Q(T)$ and then on $T$
Flip on $Q(T)$ and then on $T$
The set $X_T$ is a distributive lattice

We distinguish:

**left** alternating 4-cycles

**right** alternating 4-cycles

Flip operation: switch colors inside a right alternating 4-cycle

The unique transversal bicoloration of $T$ without right alternating 4-cycle is said **minimal**
Straight-line drawing algorithm from the transversal structures
Application to graph drawing

The transversal structure can be used to produce a planar drawing on a regular grid.
The red map and the blue map of $T$
The red map gives abscissas (1)

Let $v$ be an inner vertex of $T$

Let $\mathcal{P}_r(v)$ be the unique path passing by $v$ which is:
- the \textbf{rightmost} one before arriving at $v$
- the \textbf{leftmost} one after leaving $v$
The red map gives abscissas (2)

The absciss of \( v \) is the number of faces of the red map on the left of \( \mathcal{P}_r(v) \)

\[ \Rightarrow \text{A has absciss 2} \]
The blue map gives ordinates (1)

Similarly we define $\mathcal{P}_b(v)$ the unique blue path which is:

- the **rightmost** one before arriving at $v$
- the **leftmost** one after leaving $v$
The blue map gives ordinates (2)

The ordinate of $v$ is the number of faces of the blue map below $\mathcal{P}_b(v)$

$\Rightarrow B$ has ordinate 4
Execution of the algorithm
Execution of the algorithm

Let $f_r$ be the number of faces of the red map

$$f_r = 8$$
Execution of the algorithm

Let \( f_b \) be the number of faces of the blue map

\[
\begin{align*}
  f_r &= 8 \\
  f_b &= 7
\end{align*}
\]
Execution of the algorithm

Take a regular grid of width $f_r$ and height $f_b$ and place the 4 border vertices of $T$ at the 4 corners of the grid.
Execution of the algorithm

Place all other points using the **red path for absciss and the blue path for ordinate**
Execution of the algorithm

Place all other points using the red path for absciss and the blue path for ordinate.

4 faces on the left
Execution of the algorithm

Place all other points using the **red path for absciss** and the **blue path for ordinate**

3 faces below
Execution of the algorithm

Place all other points using the red path for absciss and the blue path for ordinate
Execution of the algorithm

Link each pair of adjacent vertices by a segment
Execution of the algorithm
Results

• The obtained drawing is a **straight line embedding**
• The drawing **respects the transversal structure**:
  • **Red edges** are oriented from bottom-left to top-right
  • **Blue edges** are oriented from top-left to bottom-right
• If $T$ has $n$ vertices, the width $W$ and height $H$ verify \( W + H = n - 1 \)

similar grid size as He (1996) and Miura et al (2001)
Compaction step

- Some abscissas and ordinates are not used
- The deletion of these unused coordinates keeps the drawing planar
Compaction step

- Some abscissas and ordinates are not used
- The *deletion* of these unused coordinates keeps the drawing planar
Compaction step

- Some abscissas and ordinates are not used
- The *deletion* of these unused coordinates keeps the drawing planar
Size of the grid after deletion

- If the transversal structure is the minimal one, the number of deleted coordinates can be analyzed:
- After deletion, the grid has size $\frac{11}{27}n \times \frac{11}{27}n$ “almost surely”
- Reduction of $\frac{5}{27} \approx 18\%$ compared to He and Miura et al
Bijection between triangulations and ternary trees
A ternary tree is a plane tree with:

- Vertices of degree 4 called inner nodes
- Vertices of degree 1 called leaves
- An edge connected two inner nodes is called inner edge
- An edge incident to a leaf is called a stem

A ternary tree can be endowed with a transversal structure
From a ternary tree to a triangulation

Local operations to “close” triangular faces
From a ternary tree to a triangulation

Local operations to “close” triangular faces
From a ternary tree to a triangulation

Local operations to “close” triangular faces
From a ternary tree to a triangulation

Local operations to “close” triangular faces
From a ternary tree to a triangulation

Local operations to “close” triangular faces
From a ternary tree to a triangulation

Local operations to “close” triangular faces
From a ternary tree to a triangulation

Local operations to “close” triangular faces
From a ternary tree to a triangulation

Draw a quadrangle outside of the figure
From a ternary tree to a triangulation

Merge remaining stems to form triangles
From a ternary tree to a triangulation

Merge remaining stems to form triangles
From a ternary tree to a triangulation

Merge remaining stems to form triangles
From a ternary tree to a triangulation

Merge remaining stems to form triangles
From a ternary tree to a triangulation

Merge remaining stems to form triangles
From a ternary tree to a triangulation

Merge remaining stems to form triangles
From a ternary tree to a triangulation

Merge remaining stems to form triangles
From a ternary tree to a triangulation

Merge remaining stems to form triangles
From a ternary tree to a triangulation

Merge remaining stems to form triangles
From a ternary tree to a triangulation

Merge remaining stems to form triangles
From a ternary tree to a triangulation

Merge remaining stems to form triangles
Properties of the closure-mapping

• The closure mapping is a **bijection** between ternary trees with \( n \) inner nodes and triangulations with \( n \) inner vertices.

• The closure transports the transversal structure

• The obtained transversal structure on \( T \) is **minimal**
Observation to find the inverse mapping

The original 4 incident edges of each inner vertex of $T$ remain the clockwise-most edge in each bunch

Tree
Recover the tree

Compute the minimal transversal structure
Recover the tree
Recover the tree

Remove quadrangle
Recover the tree
Recover the tree

Keep the clockwisemost edge in each bunch
Recover the tree
Applications of the bijection

• Enumeration: \( T_n = \frac{4}{2n+2} \frac{(3n)!}{n!(2n+1)!} \)

• Random generation: linear-time uniform random sampler of triangulations with \( n \) vertices

• Analysis of the grid size: almost surely \( \frac{5n}{27} \) deleted coordinates for a random triangulation with \( n \) vertices
Analysis of the size of the grid using the bijection
Size of the compact drawing?

Let $T$ be a triangulation with $n$ vertices endowed with its minimal transversal structure

- Unoptimized drawing: $W + H = n - 1$
- Delete unused coordinates $\Rightarrow$ Compact drawing:
  $$W_c + H_c = n - 1 - \#(\text{unused coord.})$$

Theorem: $\#(\text{unused coord.}) \sim \frac{5n}{27}$ almost surely
Rule to give abscissa

The absciss of $v$ is the number of faces of the red map on the left of $\mathcal{P}_r(v)$

$\Rightarrow$ A has absciss 2
Absciss ↔ face of the red-map

• Let $f_r$ be the number of faces of the red-map

• Let $i \in [1, f_r]$ be an absciss-candidate

• There exists a face $f_i$ of the red-map such that:

\[ \text{Abs}(v) \geq i \iff f_i \text{ is on the left of } \mathcal{P}_r(v) \]

Example: $i = 6$
An absciss-candidate $i \in [1, f_r]$ is unused iff:

\[
\text{Abs}(v) \geq i \Rightarrow \text{Abs}(v) \geq i + 1
\]

$\Rightarrow$ Faces $f_i$ and $f_{i+1}$ can not be separated by a path $\mathcal{P}_r(v)$

$\Rightarrow f_i$ and $f_{i+1}$ are contiguous

BottomRight($f_i$)$=$TopLeft($f_{i+1}$)
Unused abscissa and opening

Triangulation ↔ Ternary tree

Unused abscissa

opening

Internal edge such that cw–consecutive edge at each extremity is an internal edge
Reduction to a tree-problem

Width of the grid of the compact drawing?

How many unused abscissas in a random triangulation?

How many in a random triangulation?

How many in a random ternary tree?

$\Rightarrow \sim \frac{5n}{27}$ (using generating functions)
Analysis of the tree parameter

Ternary trees

- **One-variable grammar** ⇒ \[ T(z) = \sum_n T_n z^n \]
  \[ T = \mathcal{Z} \times (1 + T)^3 \] ⇒ \[ T(z) = z(1 + T(z))^3 \]

- **Two-variables grammar** ⇒ \[ T(z, u) = \sum_{n,k} T_{n,k} z^n u^k \]

  node marked by \( z \)

  marked by \( u \)

- Use quasi-power theorem (Hwang, Flajolet Sedgewick)

  \[ \rho(u) := \text{Sing}(u \to T(z, u)) \quad -\frac{\rho'(1)}{\rho(1)} = \frac{5}{27} \]

  ⇒ The number of \( \mathcal{Z} \) is \( \sim \frac{5n}{27} \) almost surely