

# Faire de la recherche en Géométrie algorithmique

Olivier Devillers



# Faire de la recherche en Géométrie algorithmique

Olivier Devillers





[CV \(in french\)](#)

## Olivier Devillers

Research Director / Directeur de recherche à *L'Inria*

INRIA, Centre de Recherche Nancy - Grand est,

CNRS, Loria,

Université de Lorraine

projet [Gamble](#), Tranche B, 615 rue du Jardin Botanique, B.P. 101, 54602 Villers-lès-Nancy cedex, FRANCE ,

Phone : (33) 3 54 95 85 25 Fax : (33) 3 83 27 83 19

email : [Olivier.Devillers\(at\)inria.fr](mailto:Olivier.Devillers(at)inria.fr)

## Research topics in computational geometry

Delaunay  
Randomisation  
Exact arithmetic  
Degeneracies  
Geometric compression  
Geometric probability

## Current Duties:

Head of [Gamble](#) project-team at Inria (2017- )

Head of [ANR Aspag](#) (2018-2022)

### Past duties:

Loria elected member at AM2I council (2017-2022)  
Head of [Associated team TRIP](#) (2018-2020),  
Local head of [ANR Présage](#) (2011-2015),  
Head of [ANR Triangles](#) (2008-2011),  
Vice-head of Inria project-teams [Prisme](#) and [Geometrica](#) (1998-2007),  
Member of the hiring committee for [détachements](#) (2000-2004),  
Head of CUMI at Inria-Sophia (2001-2004),  
Elected member at the lab committee at Inria-Sophia (2004-2008),  
Member of the council of the [Ecole doctorale](#) (2003-2007).

## PhD students

[Pascal Desnoguès](#), 1993-1996

[Pierre-Marie Gandoïn](#) ±, 1998-2001

[Philippe Guigue](#), 2000-2003

[Luca Castelli Aleardi](#), 2003-2006

[Abdelkrim Mebarki](#) ±, 2004-2008

[Pedro Machado Manhães de Castro](#) ±, 2007-2010

[Ross Hemsley](#), 2011-2014

[Rémy Thomasse](#), 2012-2015

[Charles Duménil](#), 2016-2022

## Master students

[Leo Donati](#), 1991

[Sylvain Lazard](#), 1992

[Pascal Desnoguès](#), 1993

[Pierre Alliez](#), 1997

[Pierre-Marie Gandoïn](#), 1998

[Philippe Guigue](#), 2000

[Luca Castelli Aleardi](#), 2003

François Collet, 2015

Guillermo Reyes, 2017

## Teaching

--- M2 AVR (Apprentissage, Vision, Robotique) (Université de Lorraine)

RECHERCHE  
UNIVERSITÉ  
CLASSE  
CHICHE

## Publications

[PUBLICATIONS](#) [dissemin](#), [hal](#), [google scholar](#) [HAL](#) [ACM Portal](#) [Citeseer](#) [DBLP](#) [MathSciNet](#) [zbMATH](#) [orcid](#)

[Collaborations](#), [google profile](#) [ResearchGate](#) [MathSciNet](#)

Demos: [Point location \[SoCG2011\]](#) [Trees](#), and [Stars](#).

(Demos run faster with safari or chrome than firefox)

[Invited talk](#) at [EuroCG 2012](#)

[Talk](#) at [Séminaire Francilien de Géométrie Algorithmique et Combinatoire](#)

## Links

--- [Computational geometry pages](#)

--- [Journées de géométrie algorithmique](#)

--- Diamond open journals where I already published (free for readers and authors):

[JoCG](#), [JGAA](#), [ECP](#), [DMTCS](#), [JCGT](#).

**I do not review for commercial journals**



[CV \(in french\)](#)

## Olivier Devillers

Research Director / Directeur de recherche à *L'Inria*

[INRIA](#), Centre de Recherche [Nancy - Grand est](#),

CNRS, [Loria](#),

Université de Lorraine

projet [Gamble](#), Tranche B, 615 rue du Jardin Botanique, B.P. 101, 54602 Villers-lès-Nancy cedex, FRANCE ,

Phone : (33) 3 54 95 85 25 Fax : (33) 3 83 27 83 19

email : [Olivier.Devillers\(at\)inria.fr](mailto:Olivier.Devillers(at)inria.fr)

## Research topics in computational geometry

Delaunay  
Randomisation  
Exact arithmetic  
Degeneracies  
Geometric compression  
Geometric probability

## Current Duties:

Head of [Gamble](#) project-team at Inria (2017- )

Head of [ANR Aspag](#) (2018-2022)

### Past duties:

Loria elected member at [AM2I](#) council (2017-2022)  
Head of [Associated team TRIP](#) (2018-2020),  
Local head of [ANR Présage](#) (2011-2015),  
Head of [ANR Triangles](#) (2008-2011),  
Vice-head of Inria project-teams [Prisme](#) and [Geometrica](#) (1998-2007),  
Member of the hiring committee for [détachements](#) (2000-2004),  
Head of CUMI at Inria-Sophia (2001-2004),  
Elected member at the lab committee at Inria-Sophia (2004-2008),  
Member of the council of the [Ecole doctorale](#) (2003-2007).

## PhD students

[Pascal Desnoguès](#), 1993-1996

[Pierre-Marie Gandoïn](#) ±, 1998-2001

[Philippe Guigue](#), 2000-2003

[Luca Castelli Aleardi](#), 2003-2006

[Abdelkrim Mebarki](#) ±, 2004-2008

[Pedro Machado Manhães de Castro](#) ±, 2007-2010

[Ross Hemsley](#), 2011-2014

[Rémy Thomasse](#), 2012-2015

[Charles Duménil](#), 2016-2022

## Master students

[Leo Donati](#), 1991

[Sylvain Lazard](#), 1992

[Pascal Desnoguès](#), 1993

[Pierre Alliez](#), 1997

[Pierre-Marie Gandoïn](#), 1998

[Philippe Guigue](#), 2000

[Luca Castelli Aleardi](#), 2003

François Collet, 2015

Guillermo Reyes, 2017

## Teaching



## Publications

[PUBLICATIONS](#) [dissemin](#), [hal](#), [google scholar](#) [HAL](#) [ACM Portal](#) [Citeseer](#) [DBLP](#) [MathSciNet](#) [zbMATH](#) [orcid](#)

[Collaborations](#), [google profile](#) [ResearchGate](#) [MathSciNet](#)

Demos: [Point location \[SoCG2011\]](#) [Trees](#), and [Stars](#).

(Demos run faster with safari or chrome than firefox)

[Invited talk](#) at [EuroCG 2012](#)

[Talk](#) at [Séminaire Francilien de Géométrie Algorithmique et Combinatoire](#)

## Links

--- [Computational geometry pages](#)

--- [Journées de géométrie algorithmique](#)

--- Diamond open journals where I already published (free for readers and authors):

[JoCG](#), [JGAA](#), [ECP](#), [DMTCS](#), [JCGT](#).

**I do not review for commercial journals**

# Olivier Devillers

Lycée Callot (1974-1981)

classe prépa (Lycée du Parc, Lyon)

École Normale Supérieure (Paris)

Service militaire (scientifique)

*Inria*

Sophia-Antipolis (1989)

Nancy (2014)

# Olivier Devillers

## Lycée Callot (1974-1981)



lyon)

ris)

## MITRA 15

antipolis (1989)

014)



Sans masque !

# Géométrie algorithmique



# Géométrie algorithmique

Concevoir des algorithmes

pour organiser des données géométriques

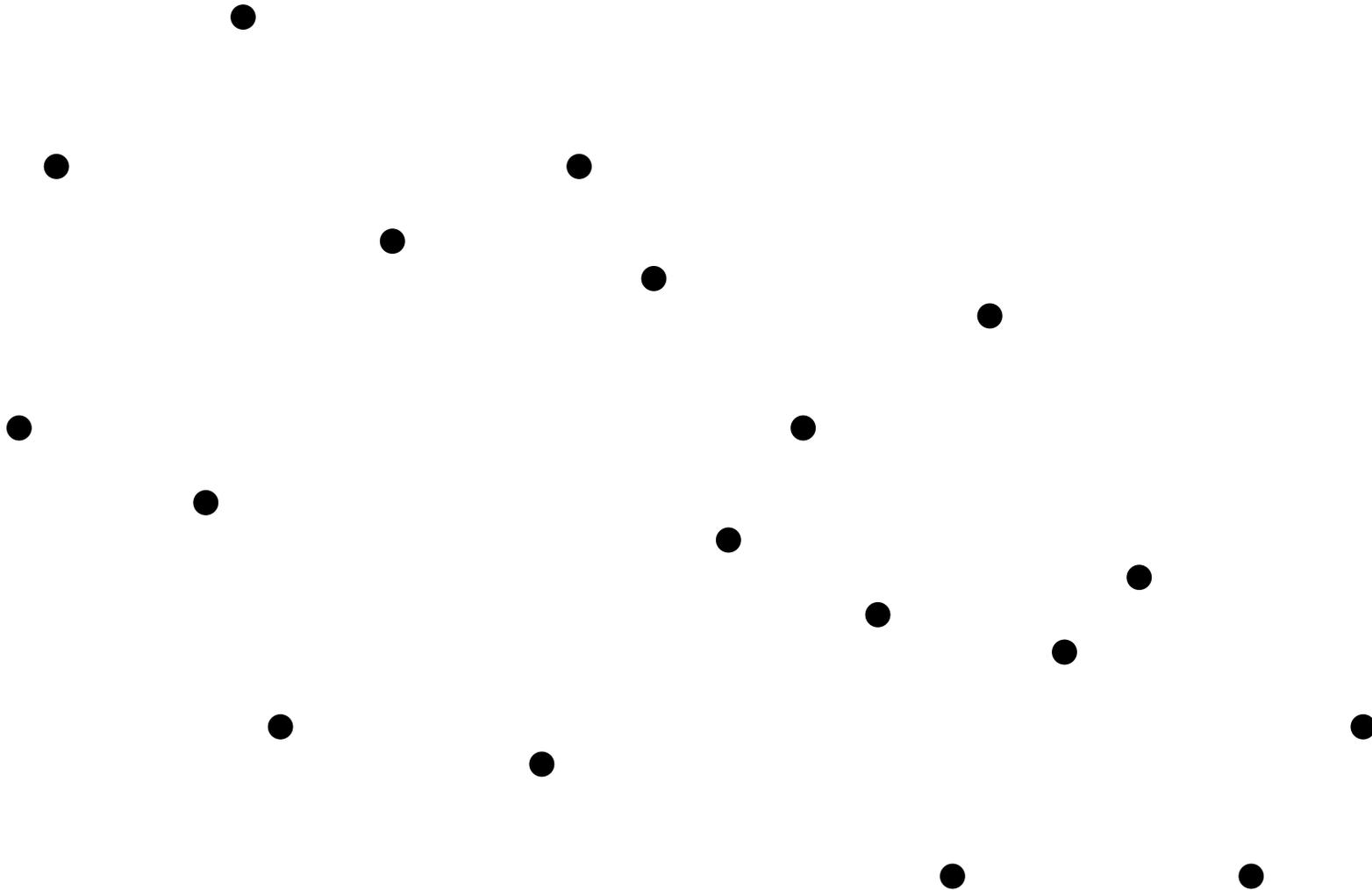
# Géométrie algorithmique

Concevoir des algorithmes

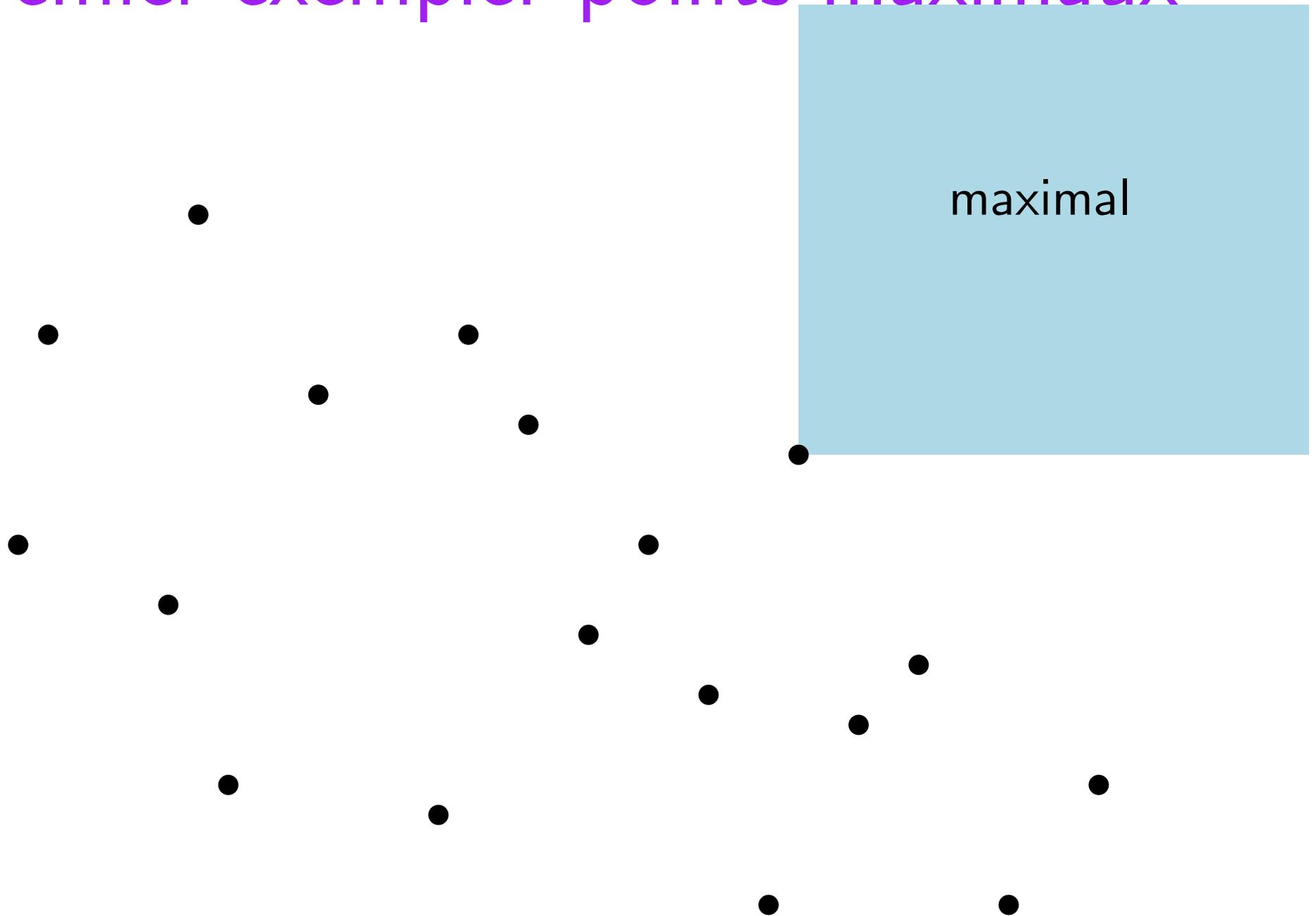
pour organiser des données géométriques

Étudier leur complexité

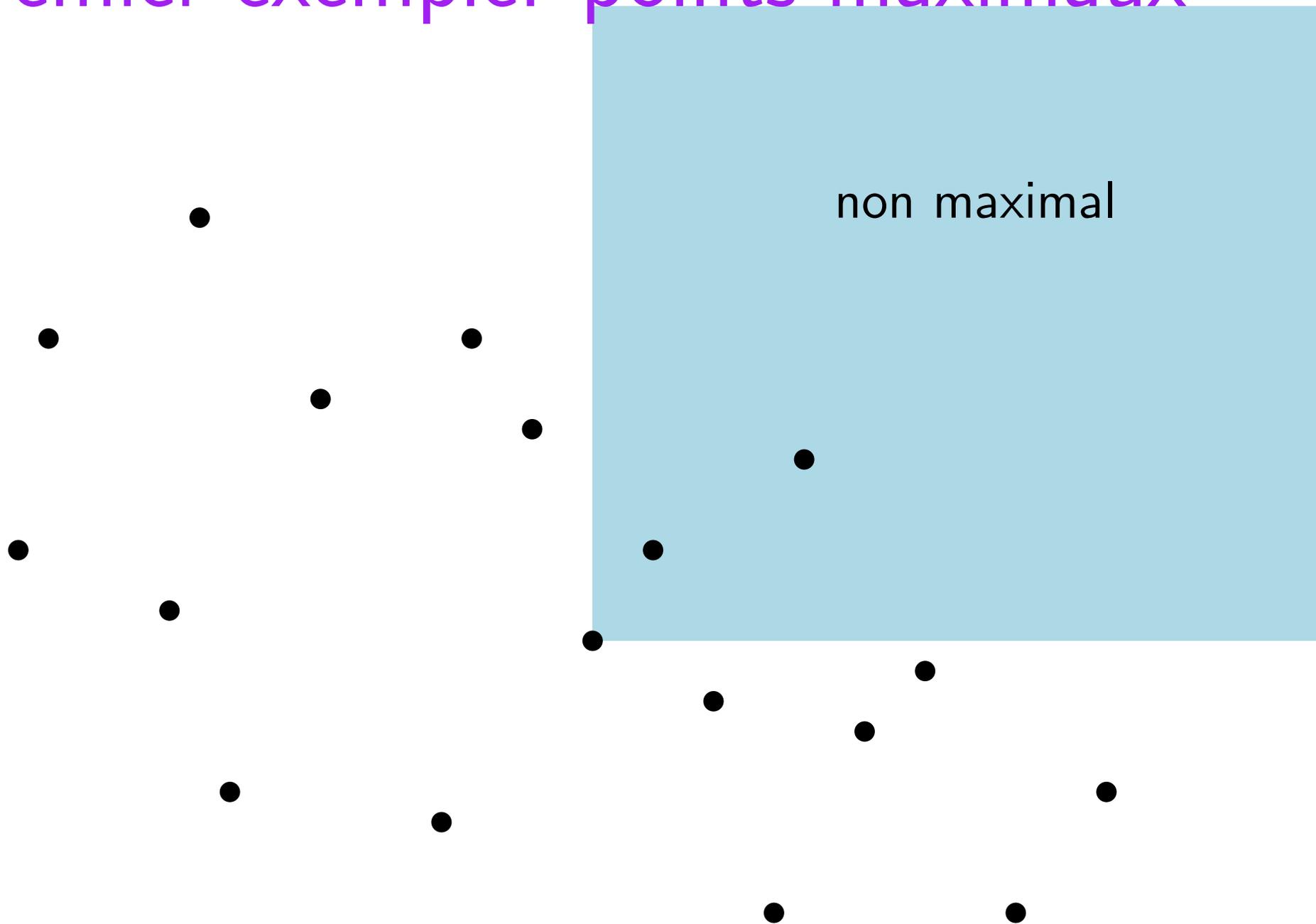
# Premier exemple: points maximaux



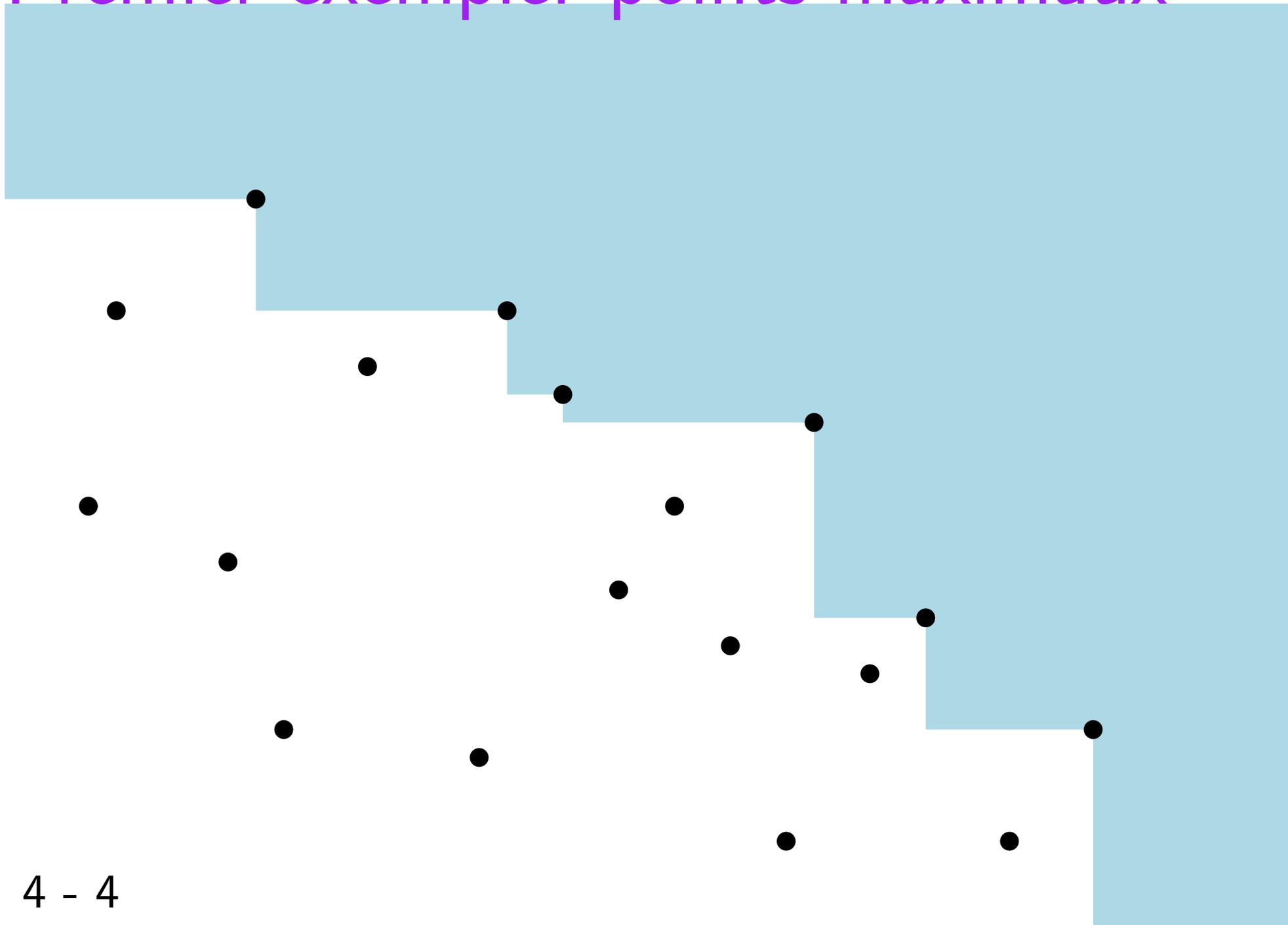
# Premier exemple: points maximaux



# Premier exemple: points maximaux

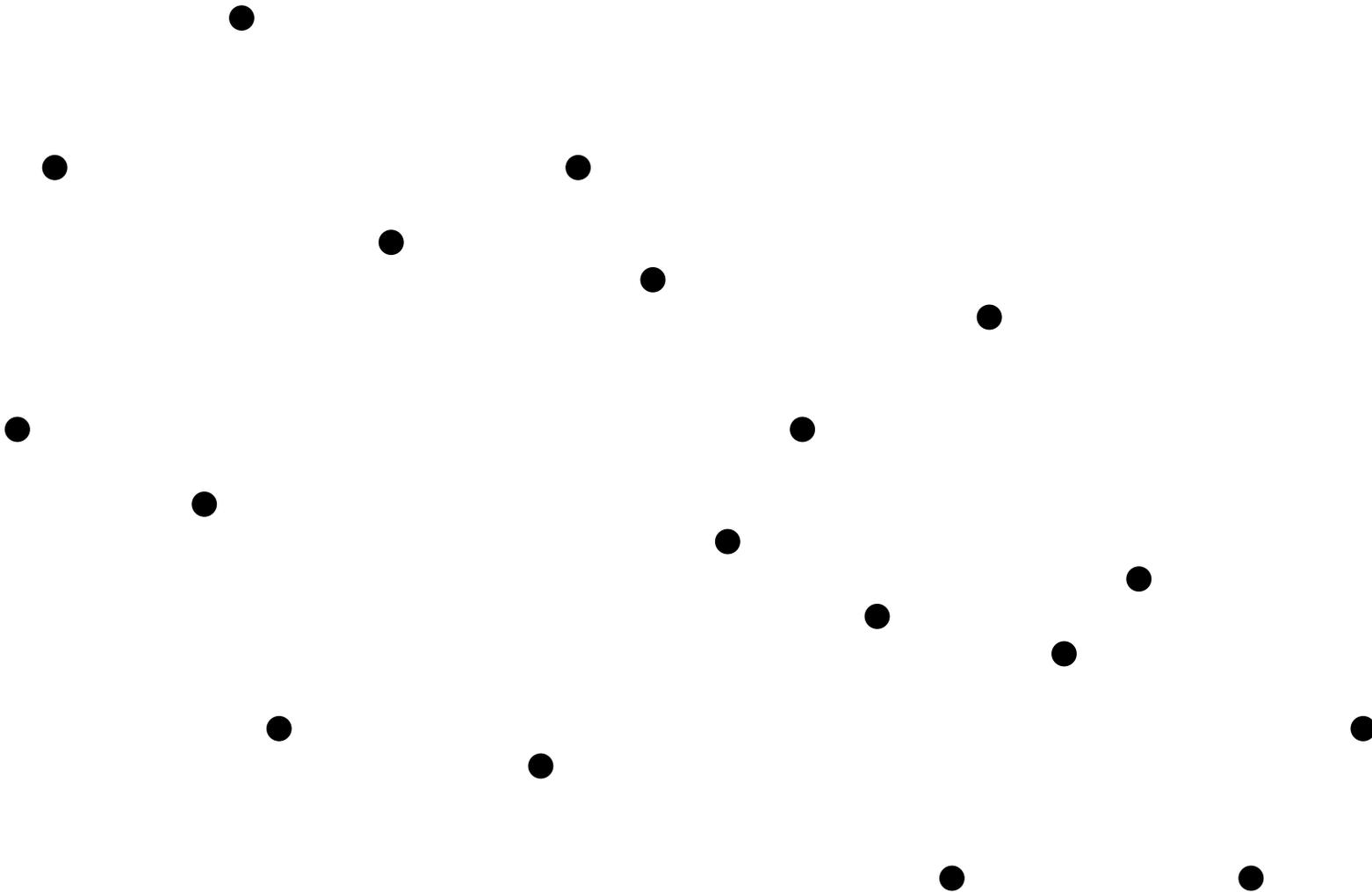


# Premier exemple: points maximaux



# Premier exemple: points maximaux

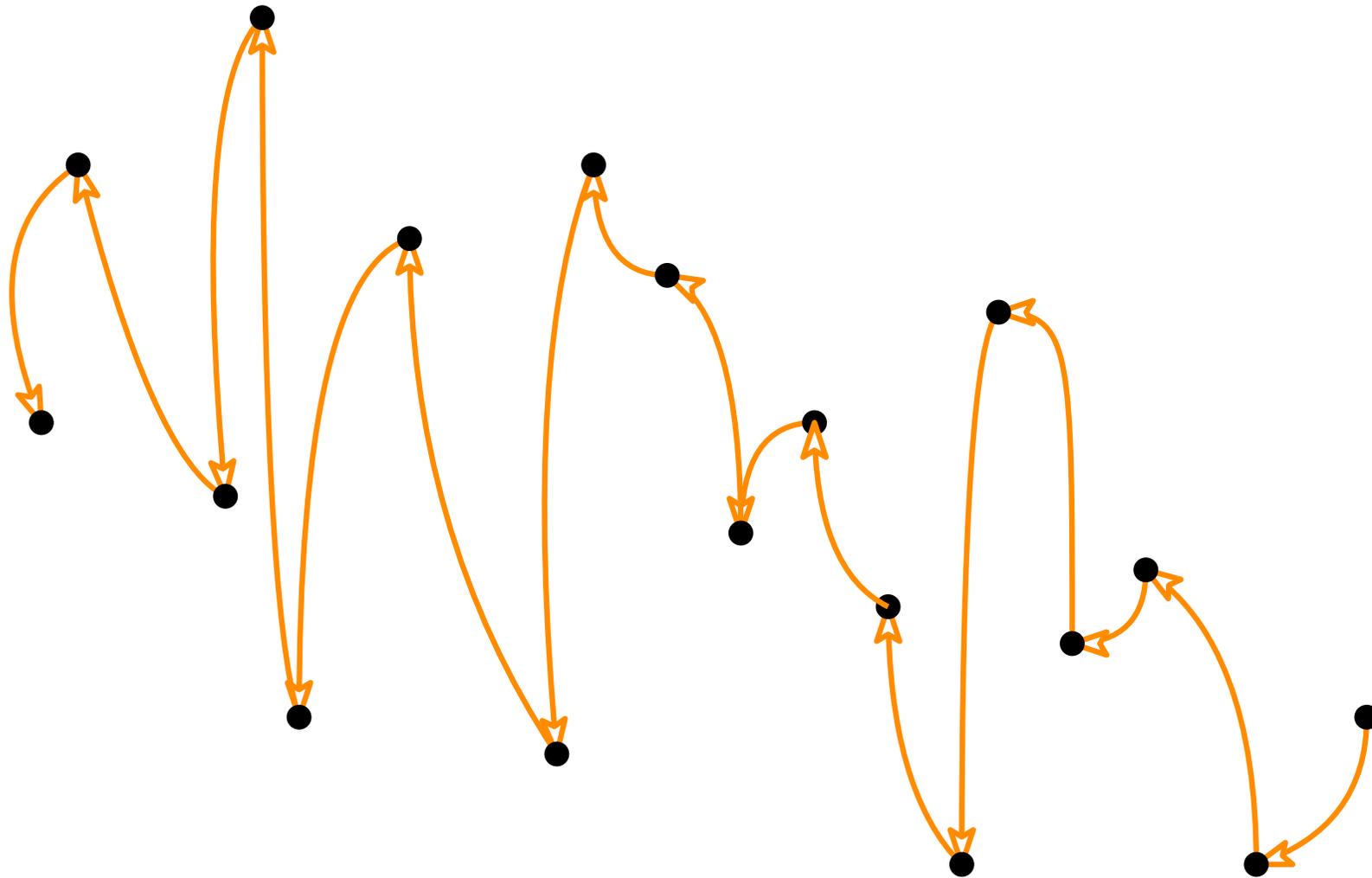
Algorithme



# Premier exemple: points maximaux

Algorithme

Trier



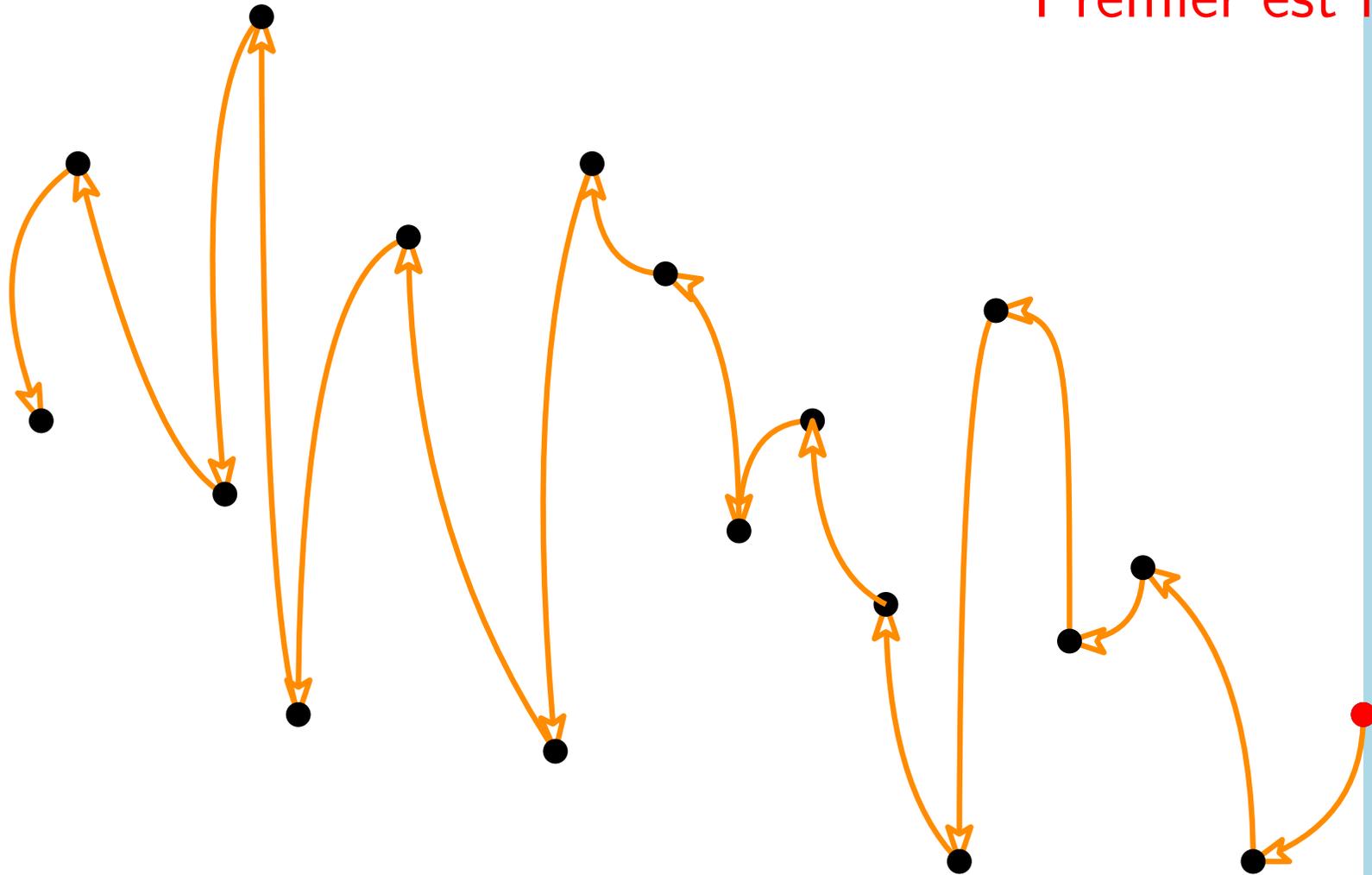
# Premier exemple: points maximaux

Algorithme

Trier



Premier est maximal







# Premier exemple: points maximaux

Algorithme

Trier ←

Premier est maximal

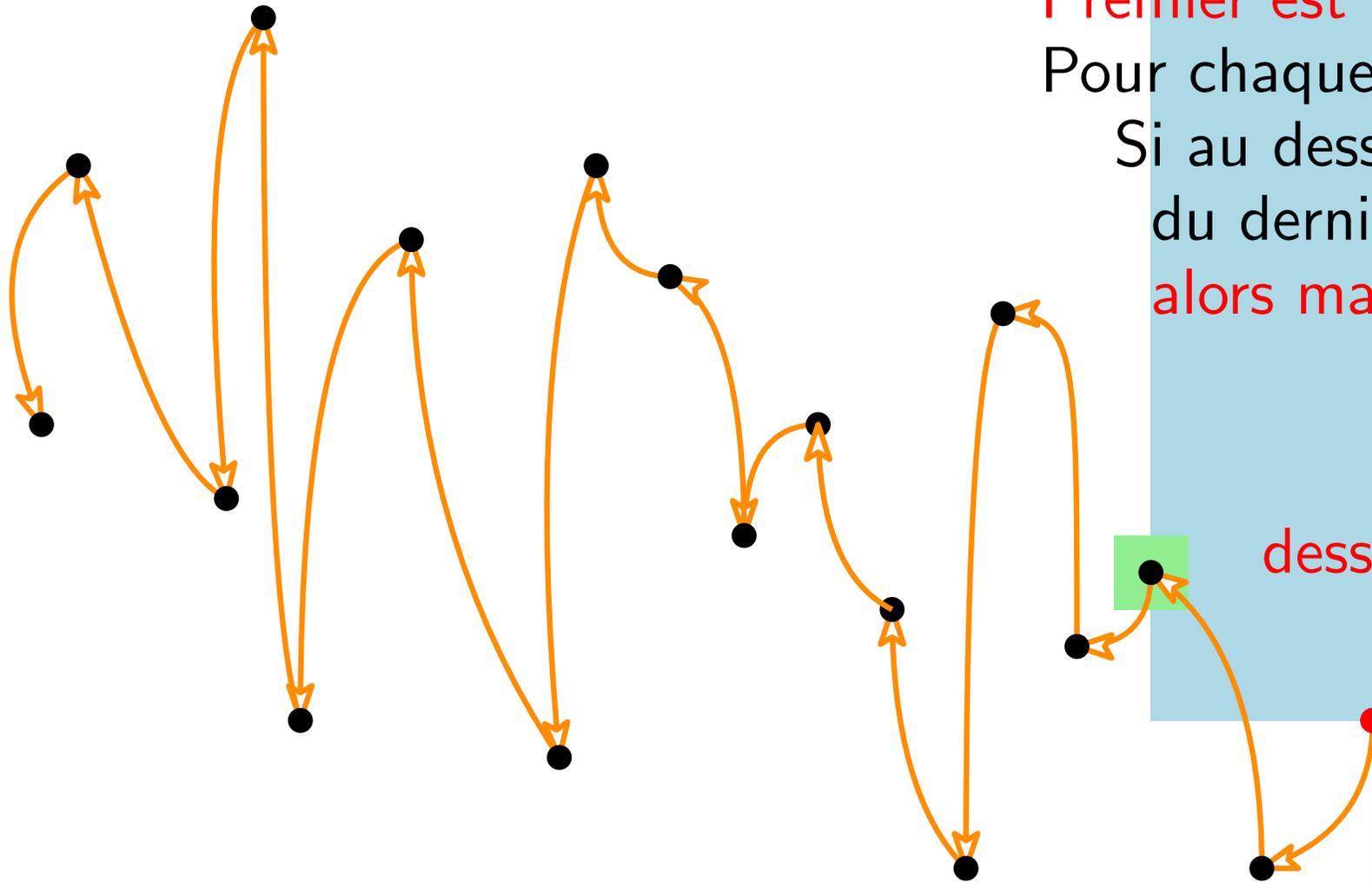
Pour chaque point

Si au dessus

du dernier maximal

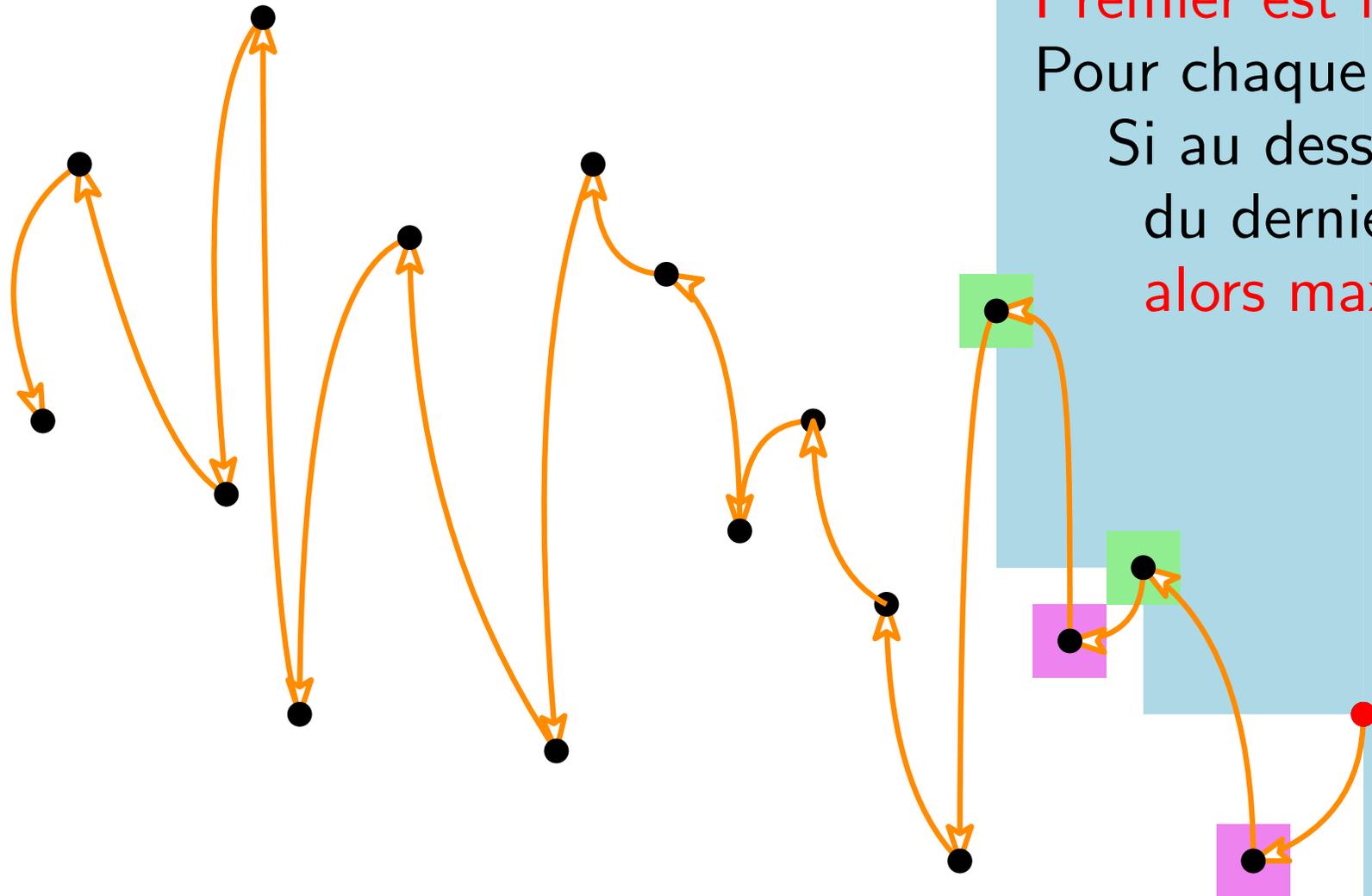
alors maximal

dessus





# Premier exemple: points maximaux



Algorithme

Trier ←

Premier est maximal

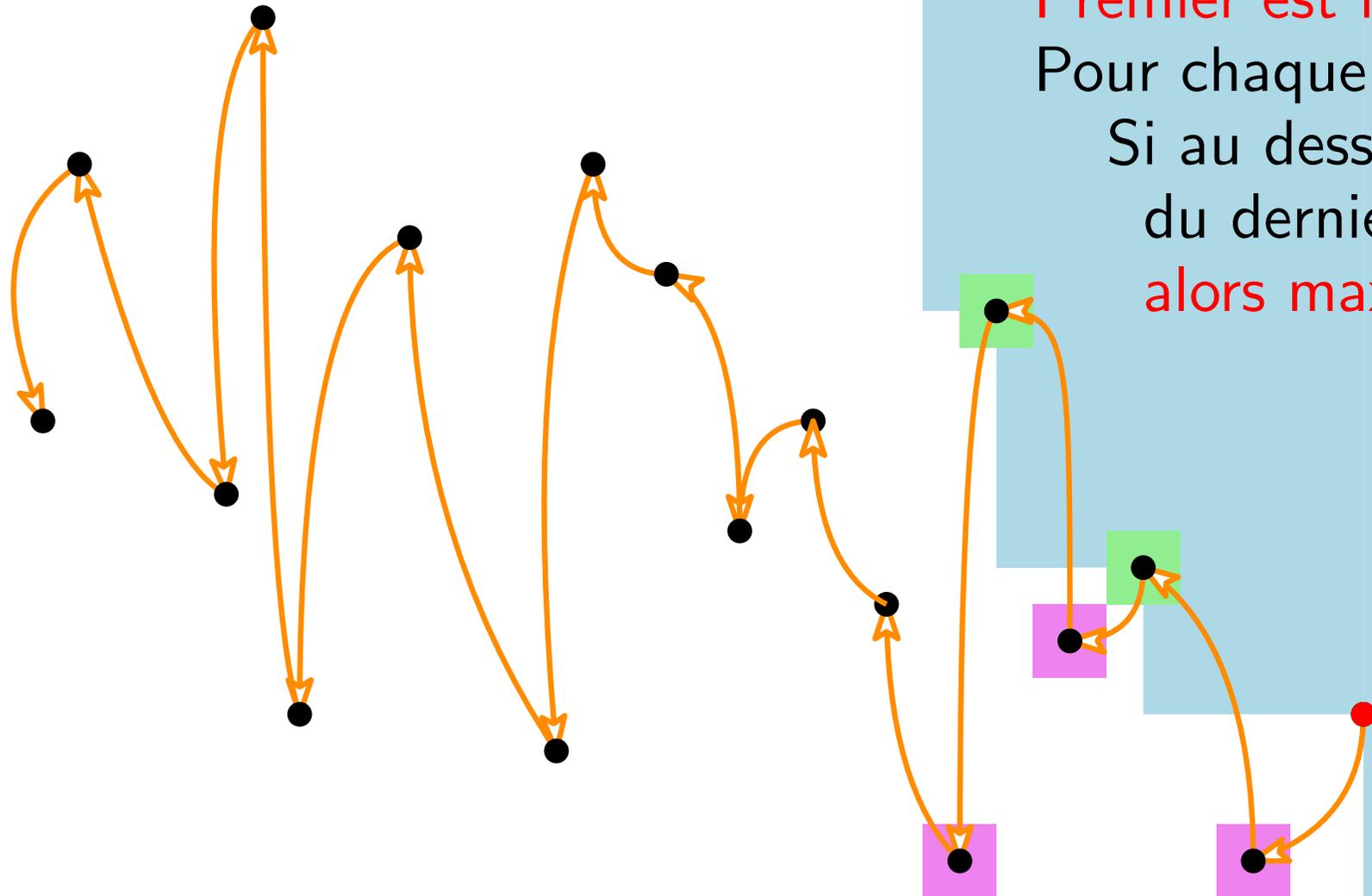
Pour chaque point

Si au dessus

du dernier maximal

alors maximal

# Premier exemple: points maximaux



Algorithme

Trier ←

Premier est maximal

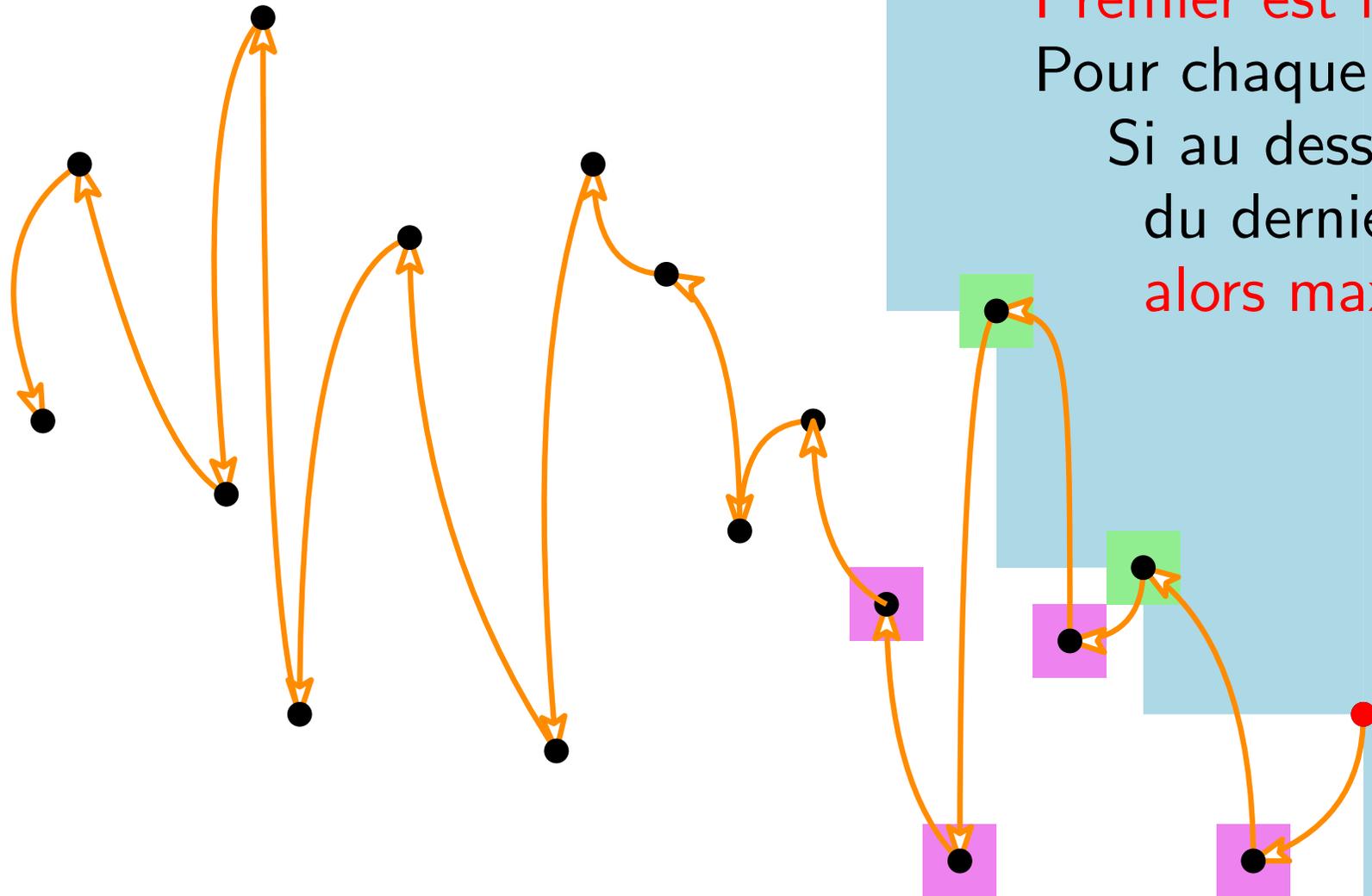
Pour chaque point

Si au dessus

du dernier maximal

alors maximal

# Premier exemple: points maximaux



Algorithme

Trier ←

Premier est maximal

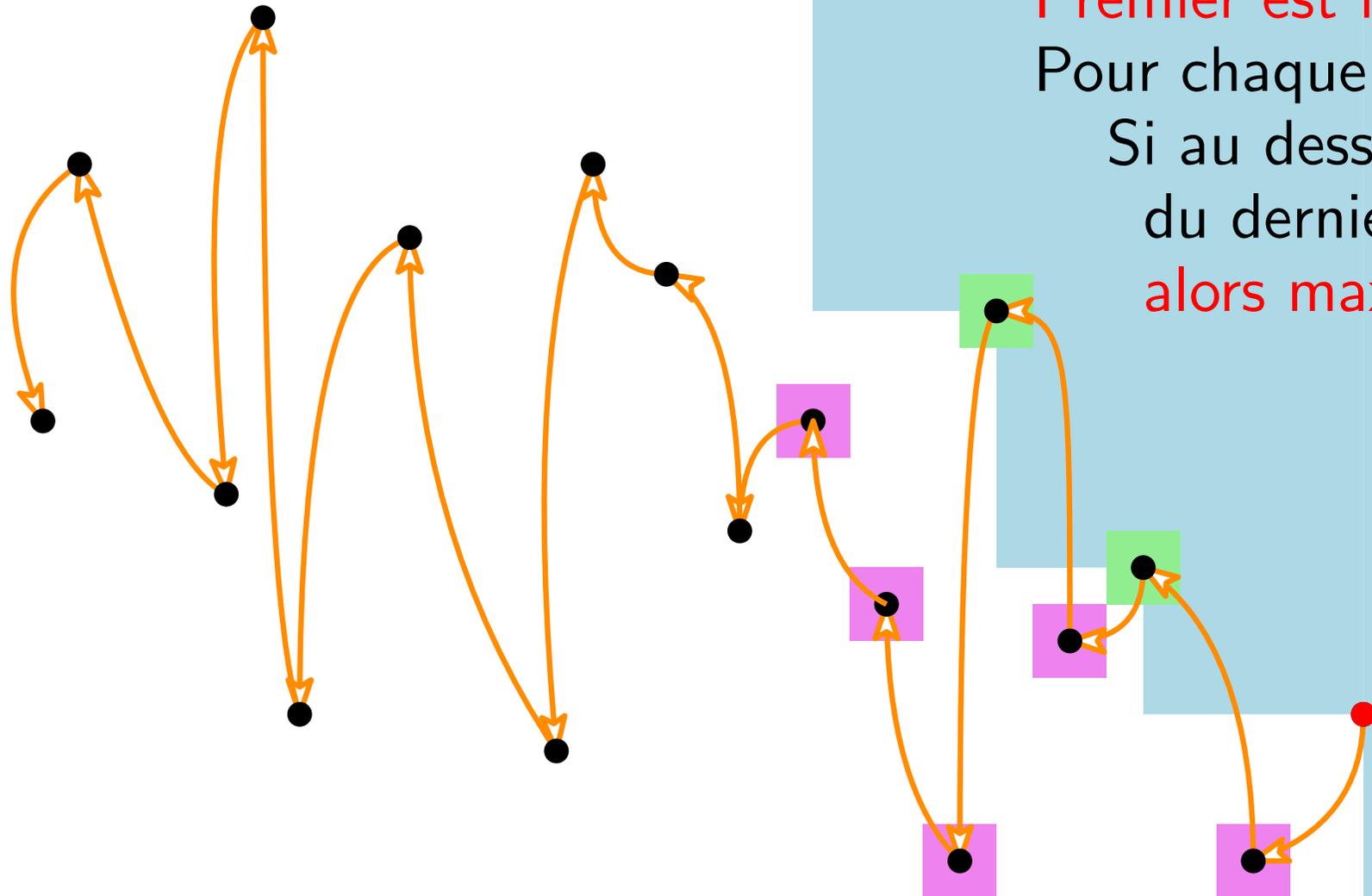
Pour chaque point

Si au dessus

du dernier maximal

alors maximal

# Premier exemple: points maximaux



Algorithme

Trier ←

Premier est maximal

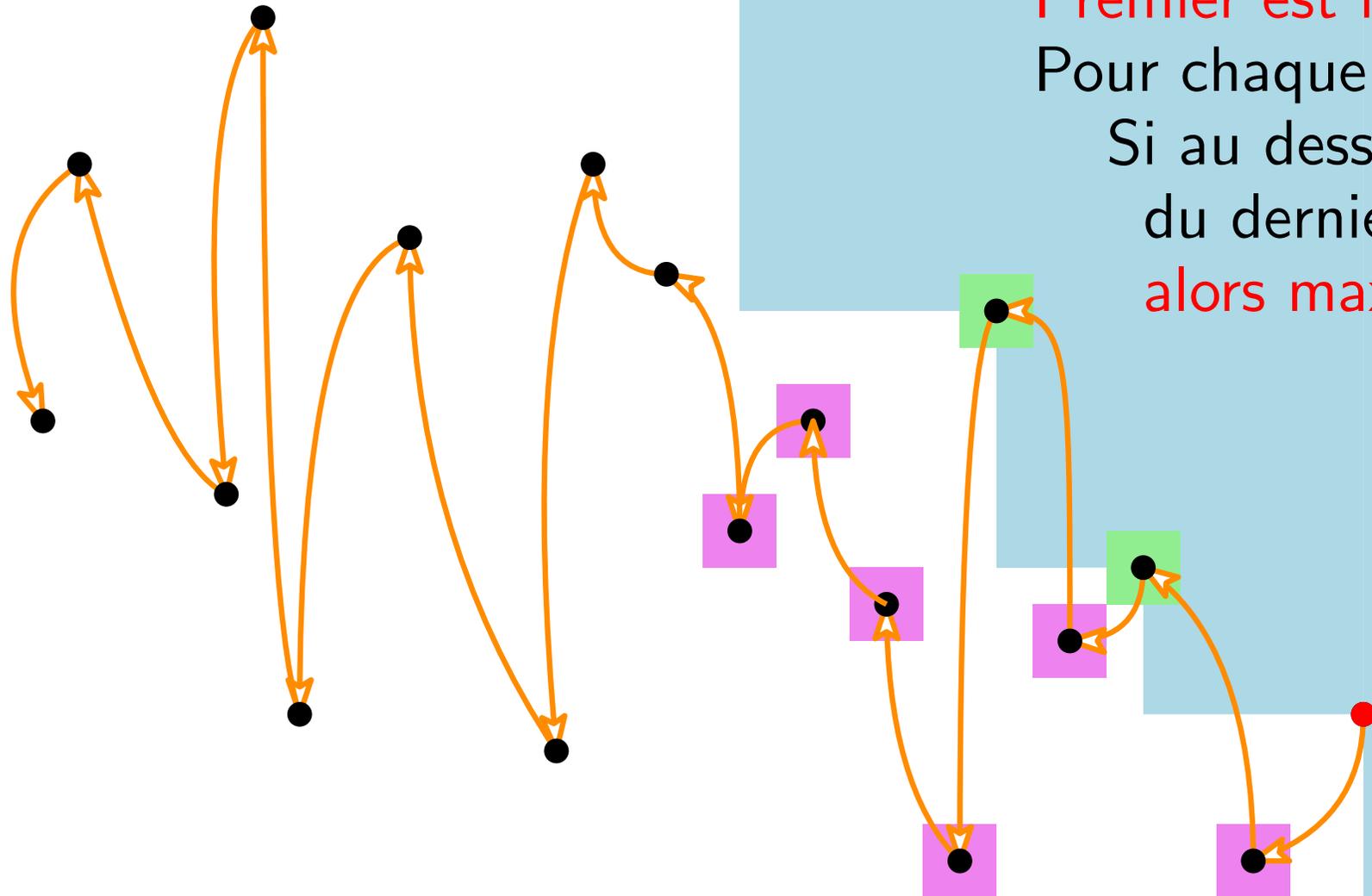
Pour chaque point

Si au dessus

du dernier maximal

alors maximal

# Premier exemple: points maximaux



Algorithme

Trier ←

Premier est maximal

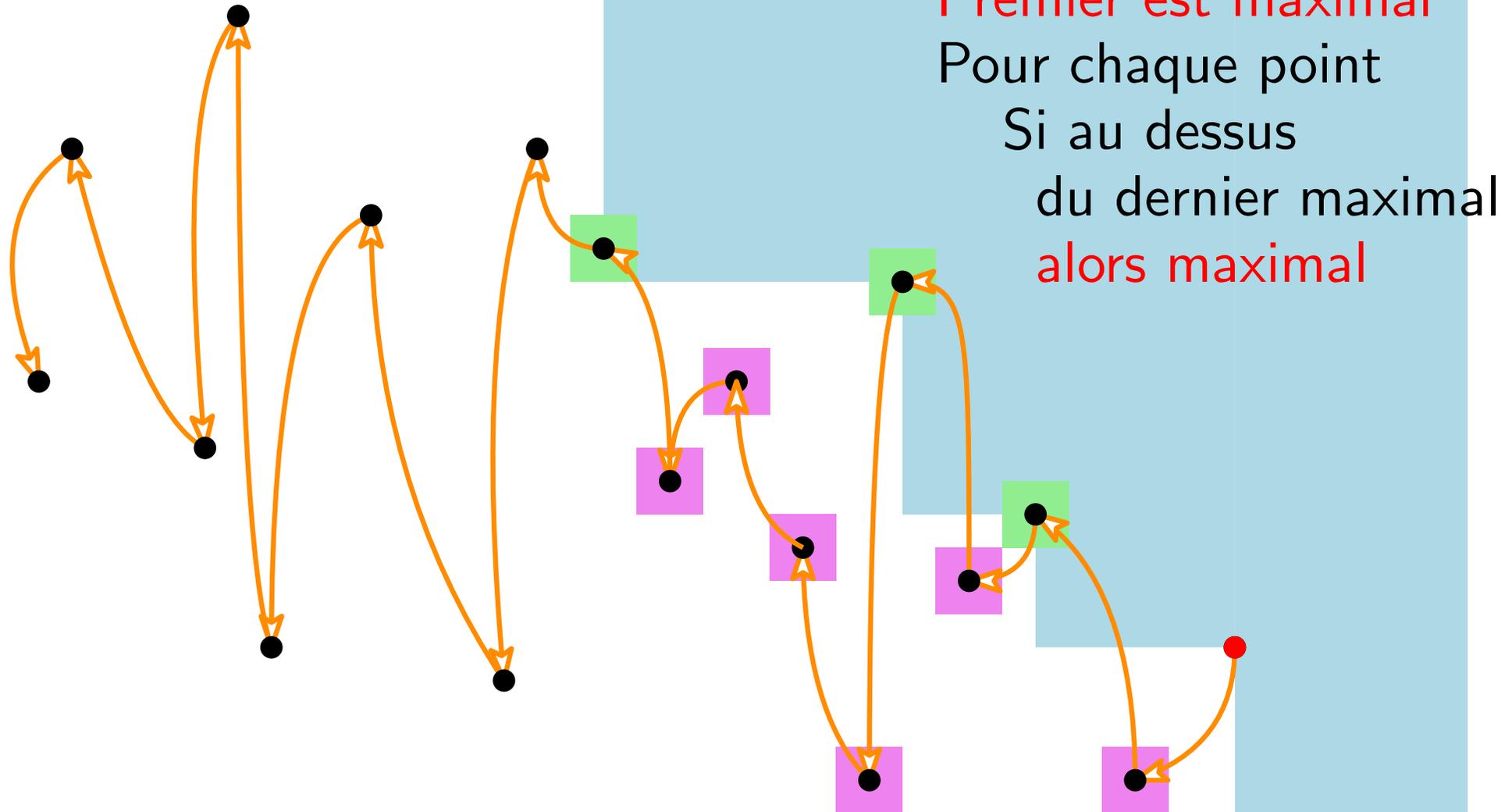
Pour chaque point

Si au dessus

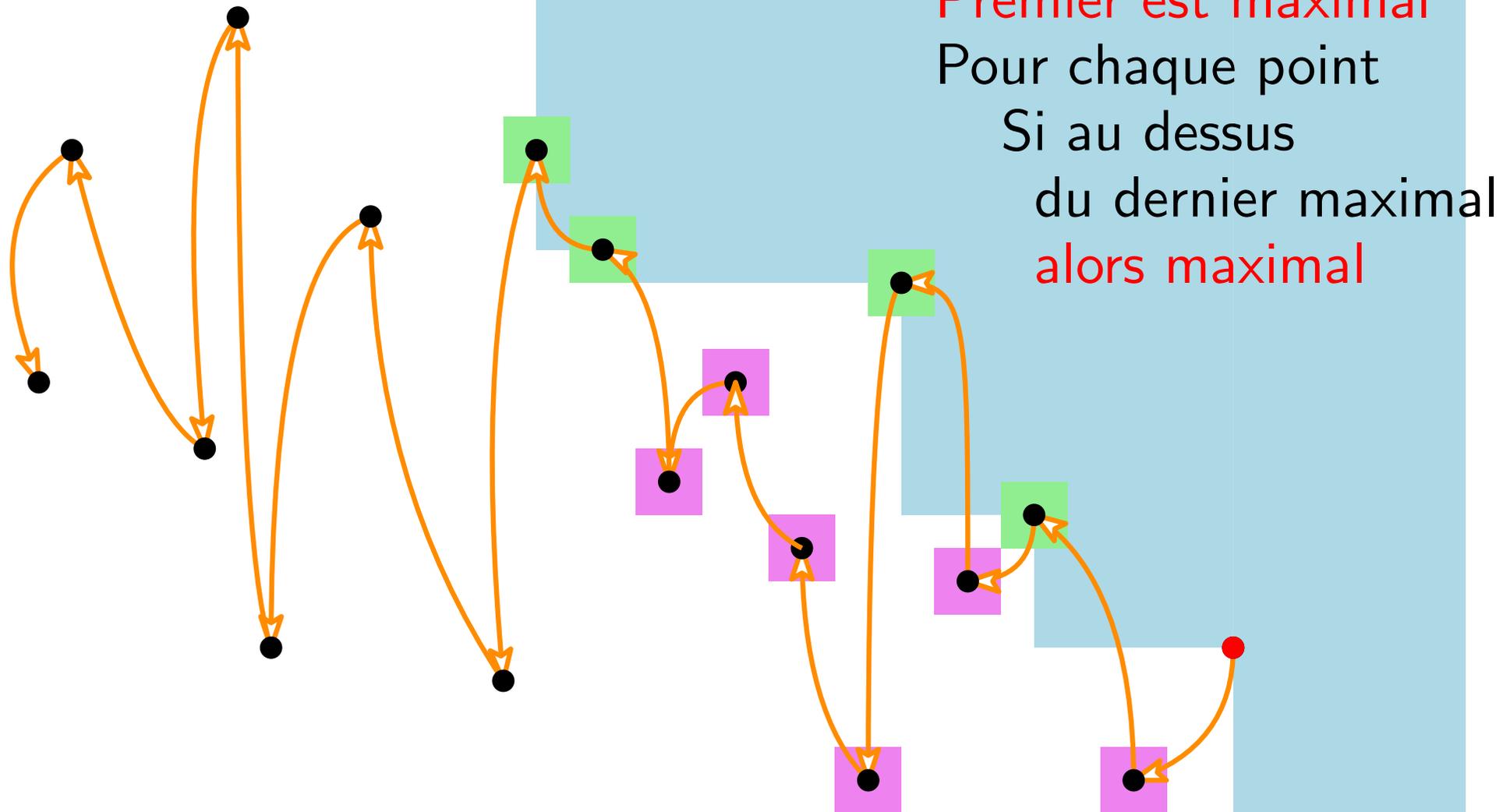
du dernier maximal

alors maximal

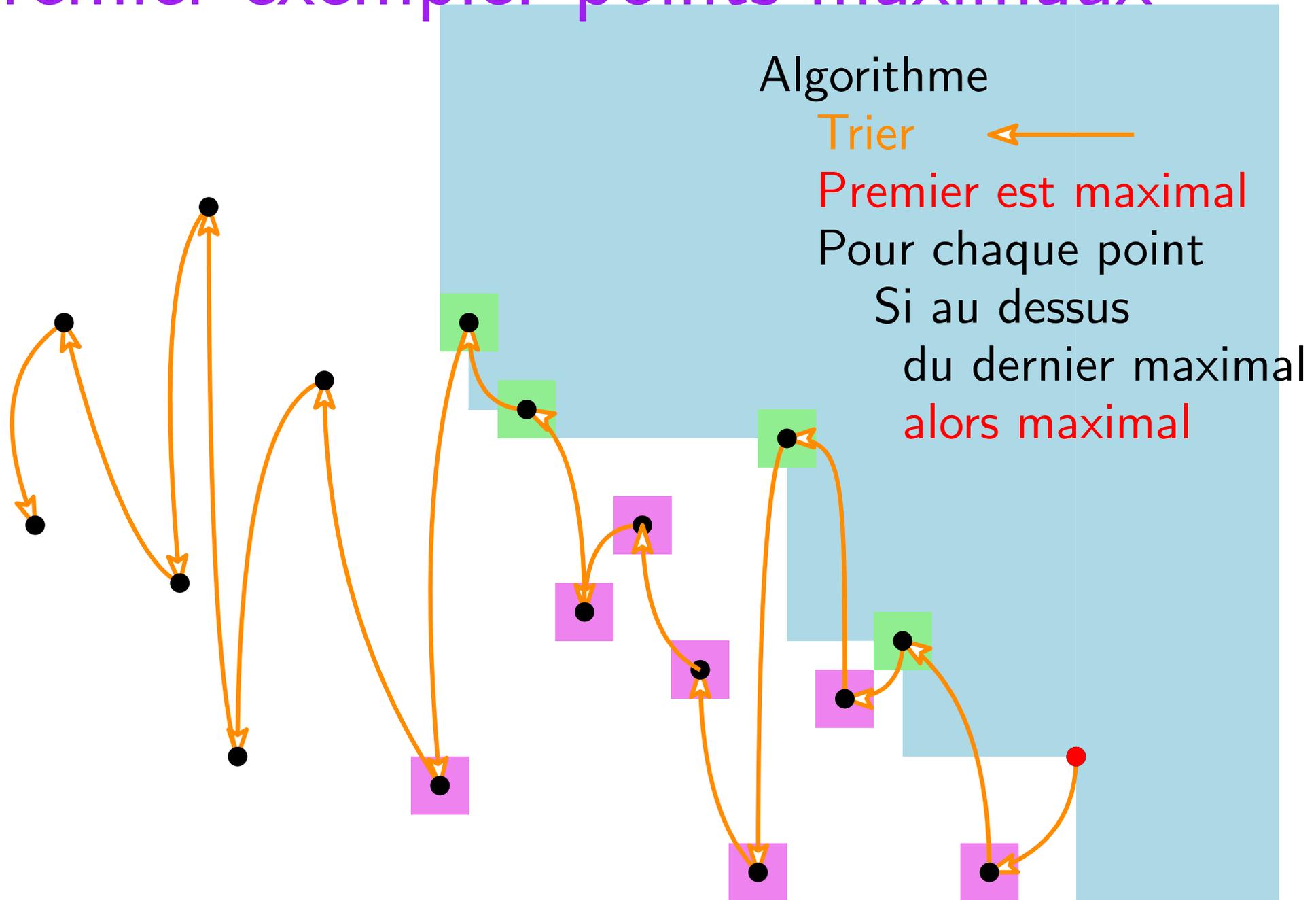
# Premier exemple: points maximaux



# Premier exemple: points maximaux

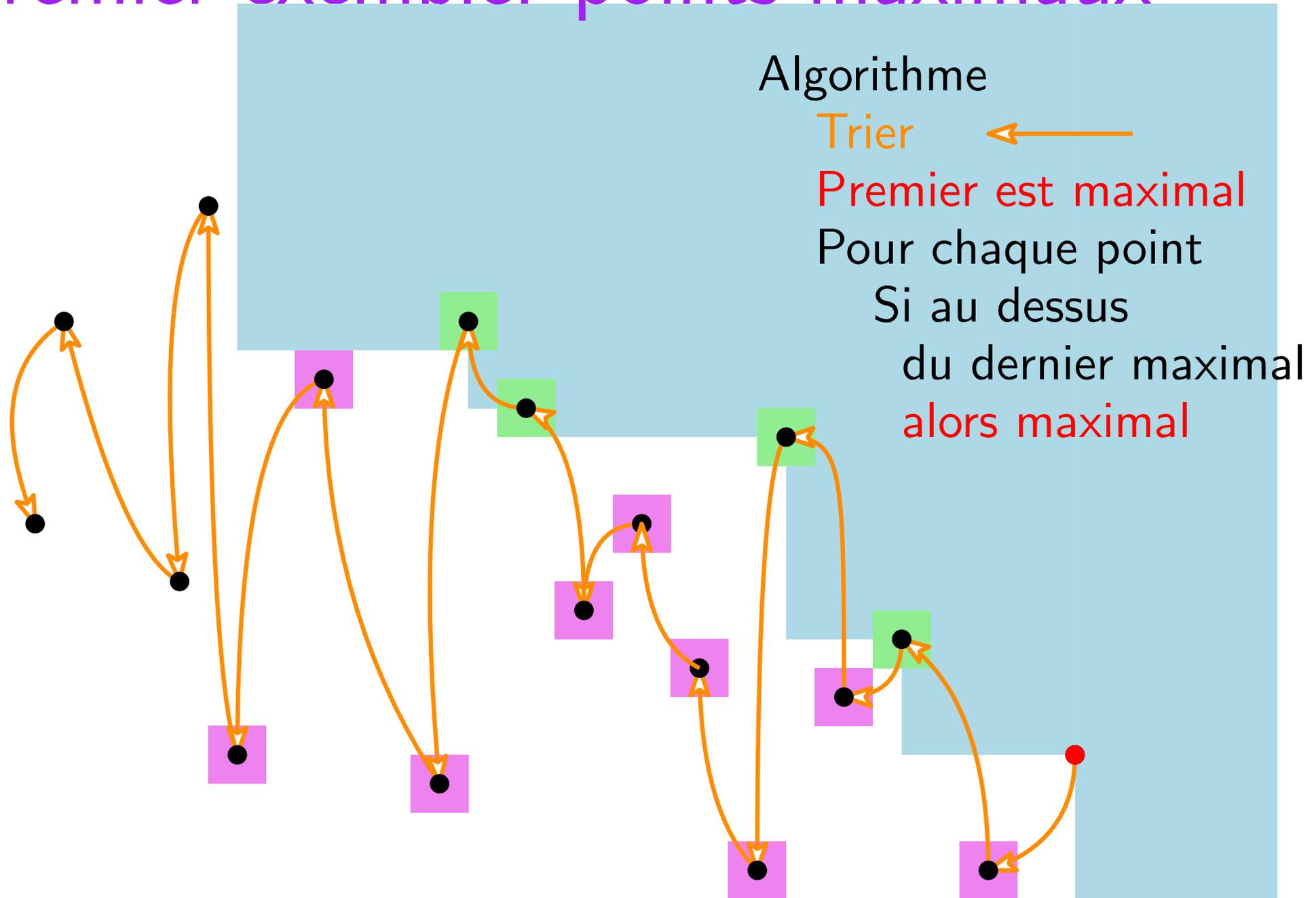


# Premier exemple: points maximaux

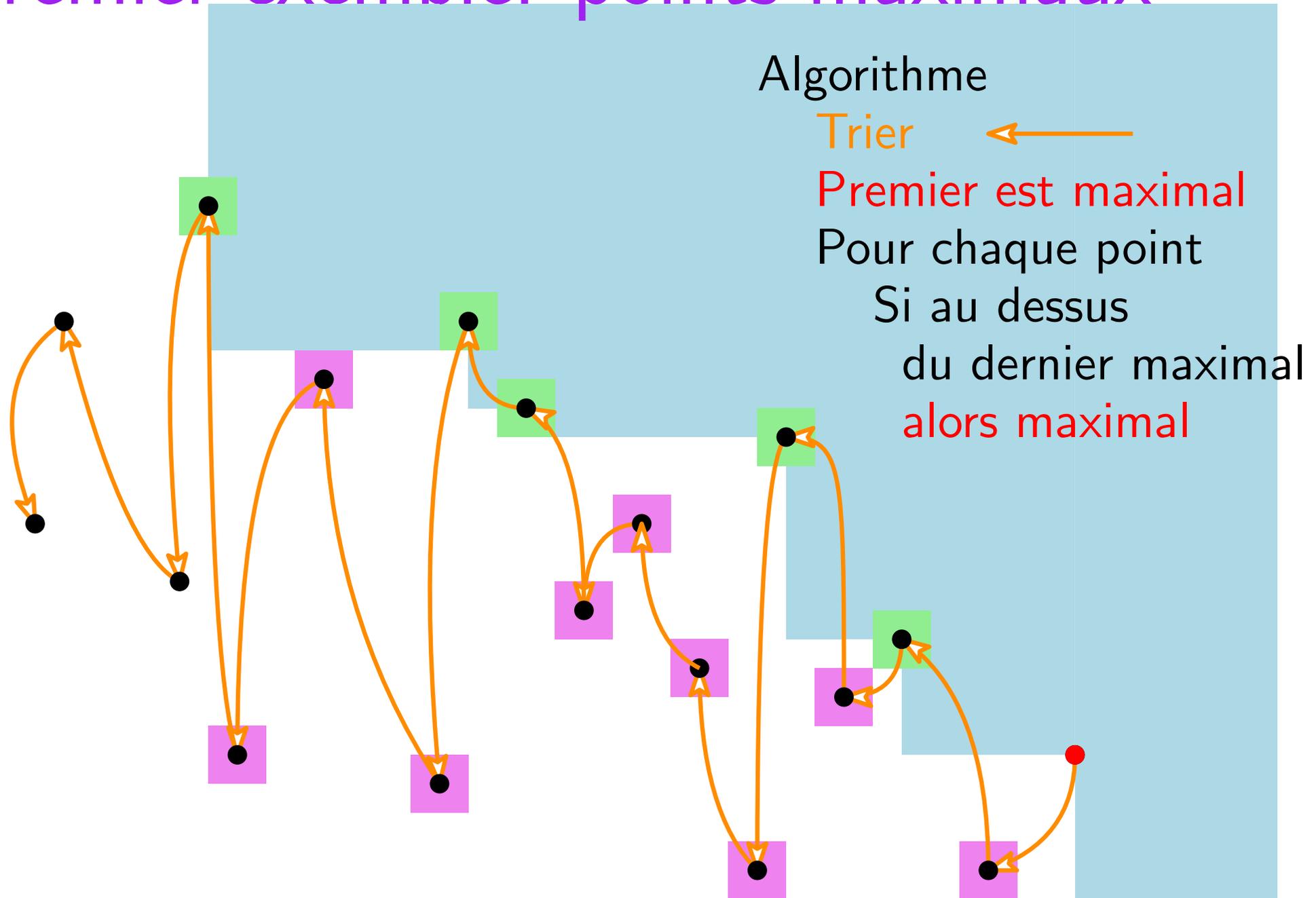




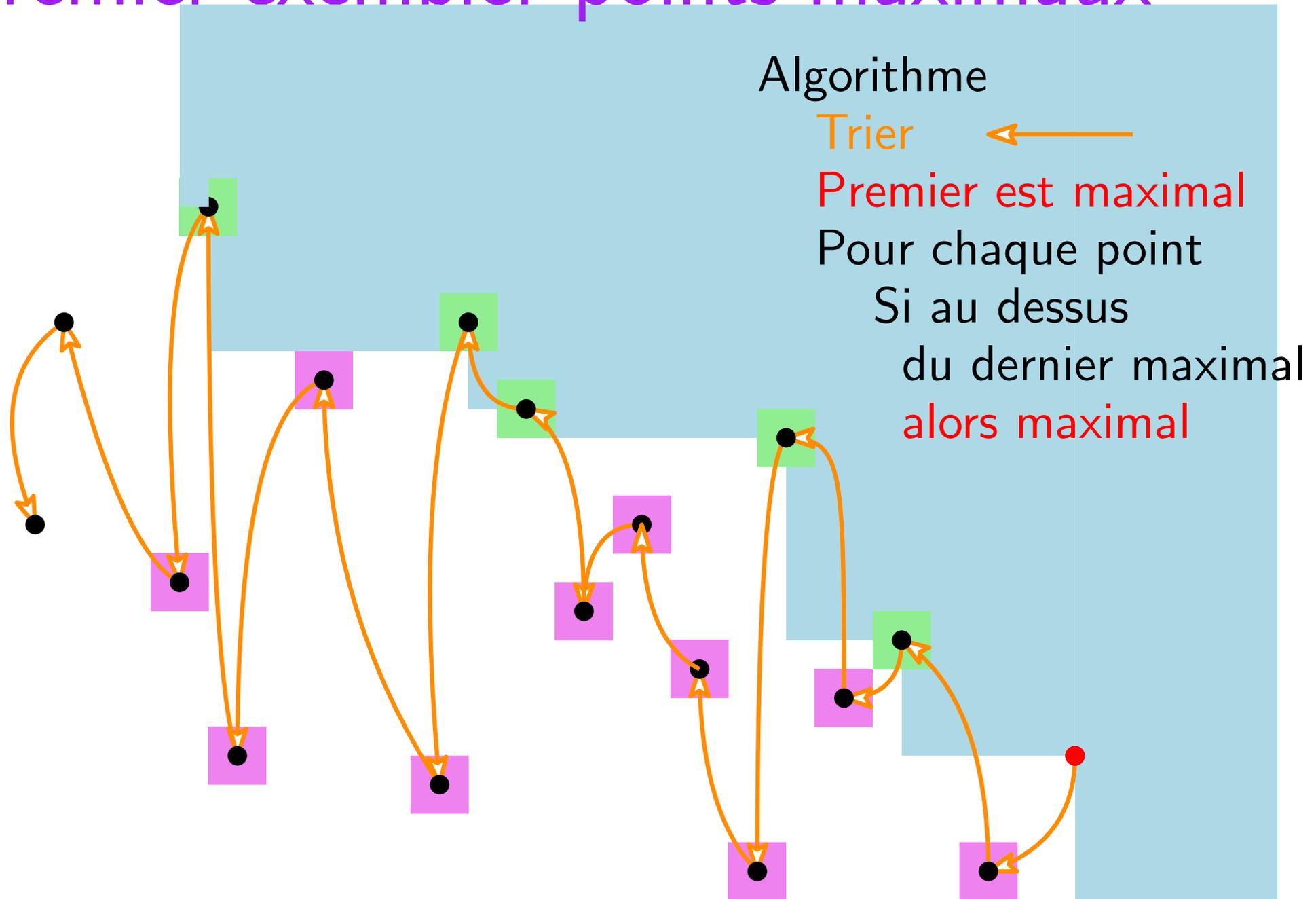
# Premier exemple: points maximaux



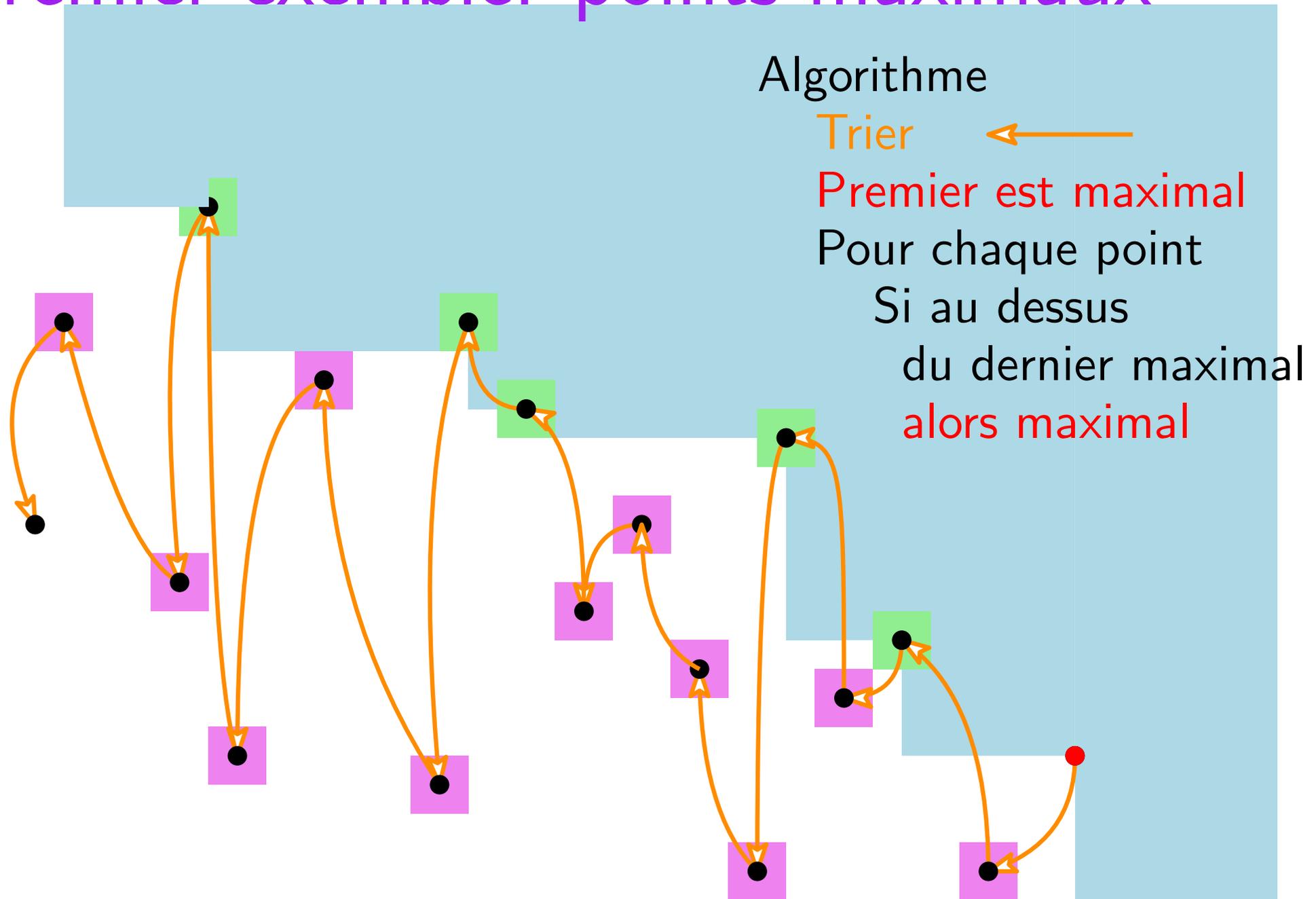
# Premier exemple: points maximaux



# Premier exemple: points maximaux



# Premier exemple: points maximaux



# Premier exemple: points maximaux

Algorithme

Trier ←

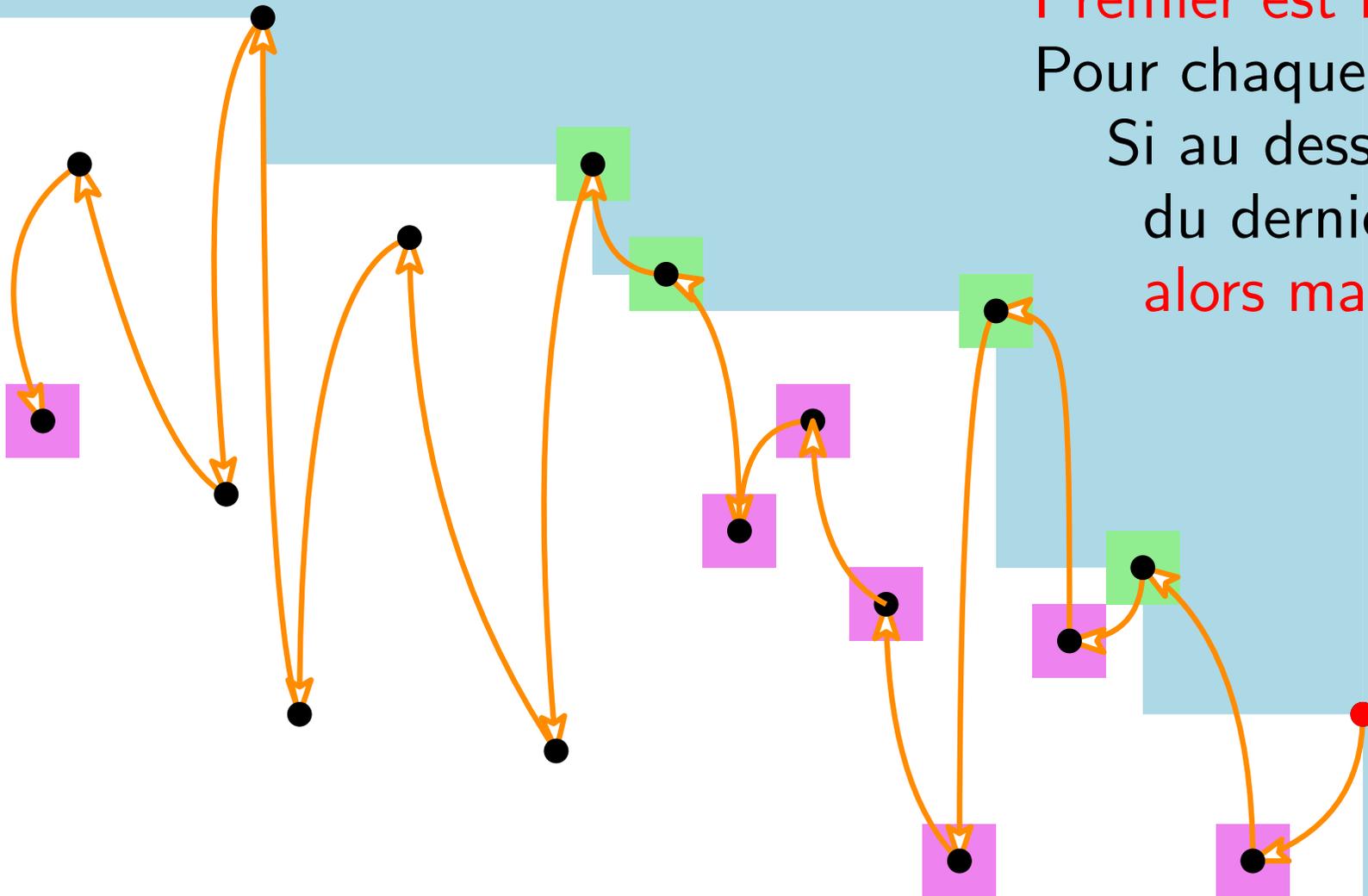
Premier est maximal

Pour chaque point

Si au dessus

du dernier maximal

alors maximal



# Premier exemple: points maximaux

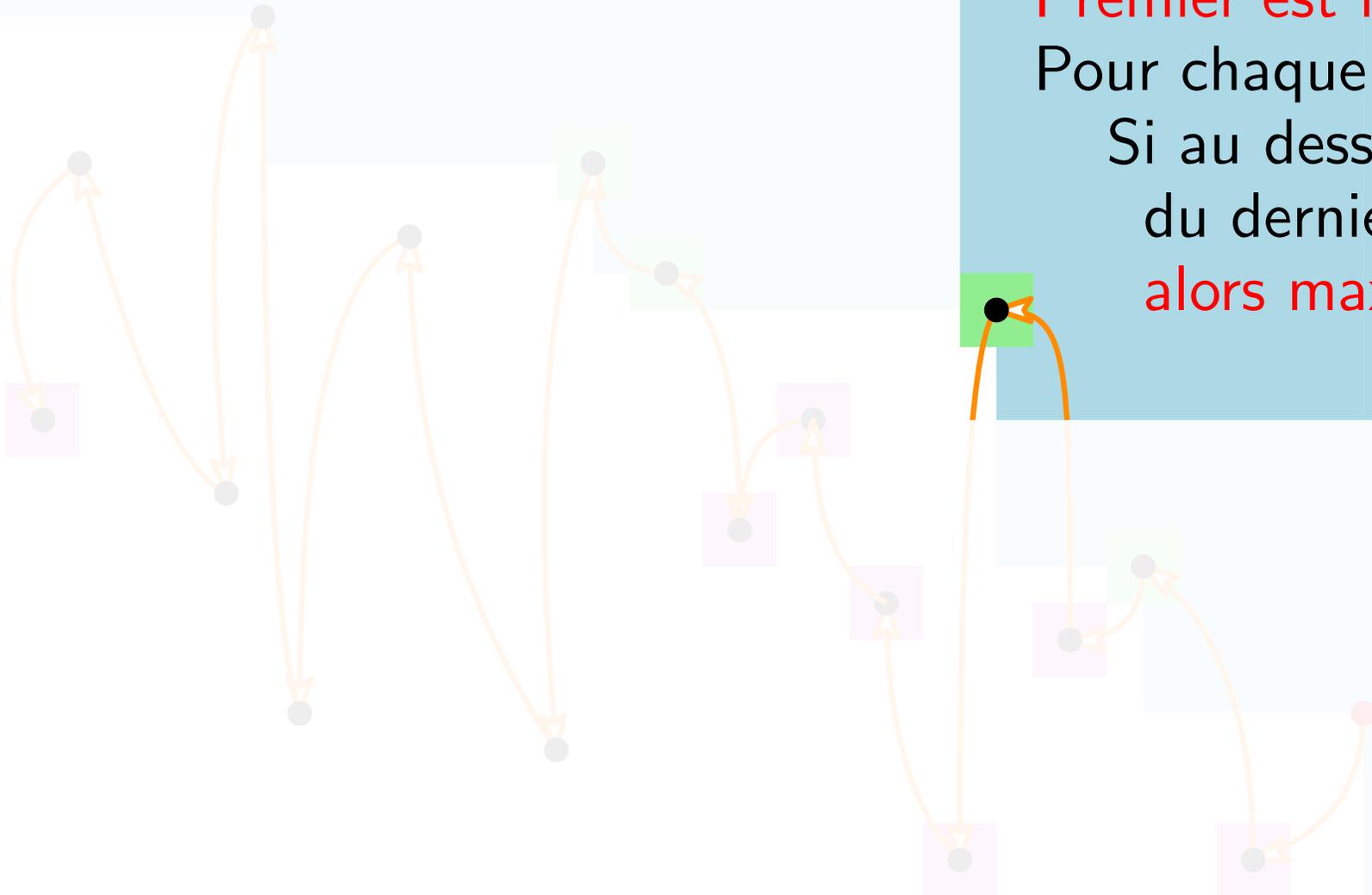
Complexité (nombre de points maximaux)

Algorithme

Trier ←

Premier est maximal

Pour chaque point  
Si au dessus  
du dernier maximal  
alors maximal



# Premier exemple: points maximaux

Complexité (nombre de points maximaux)

$$O(n)$$

Algorithme

Trier ←

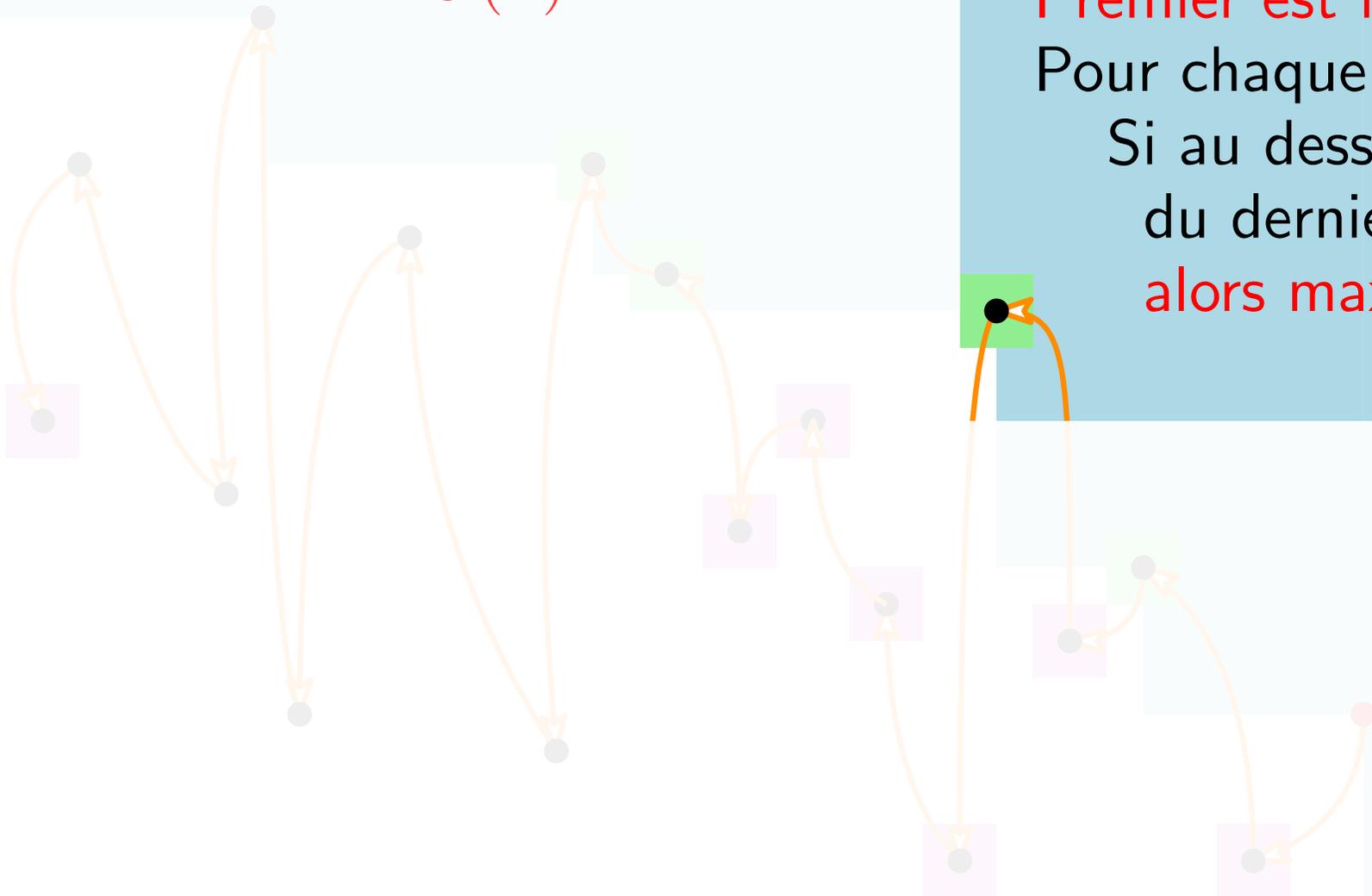
Premier est maximal

Pour chaque point

Si au dessus

du dernier maximal

alors maximal



# Premier exemple: points maximaux

Complexité (nombre de points maximaux)

$$O(n)$$

Complexité pour des points aléatoires dans un rectangle

Algorithme

Trier ←

Premier est maximal

Pour chaque point

Si au dessus

du dernier maximal

alors maximal

# Premier exemple: points maximaux

Complexité (nombre de points maximaux)

$$O(n)$$

Complexité pour des points aléatoires dans un rectangle

le  $k^{\text{ième}}$  est maximal si c'est le plus haut des  $k$  le plus à droite

Algorithme

Trier ←

Premier est maximal

Pour chaque point  
Si au dessus  
du dernier maximal  
alors maximal

# Premier exemple: points maximaux

Complexité (nombre de points maximaux)

$$O(n)$$

Complexité pour des points aléatoires dans un rectangle

le  $k^{\text{ième}}$  est maximal si c'est le plus haut des  $k$  le plus à droite

$$\text{probabilité} = \frac{1}{k}$$

Algorithme

Trier ←

Premier est maximal

Pour chaque point  
Si au dessus  
du dernier maximal  
alors maximal

# Premier exemple: points maximaux

Complexité (nombre de points maximaux)

$$O(n)$$

Complexité pour des points aléatoires dans un rectangle

le  $k^{\text{ième}}$  est maximal si c'est le plus haut des  $k$  le plus à droite

$$\text{probabilité} = \frac{1}{k}$$

$$\mathbb{E}(\#\text{points max}) = \sum_{1 \leq i \leq n} \frac{1}{k} \simeq \log n$$

Algorithme

Trier ←

Premier est maximal

Pour chaque point

Si au dessus

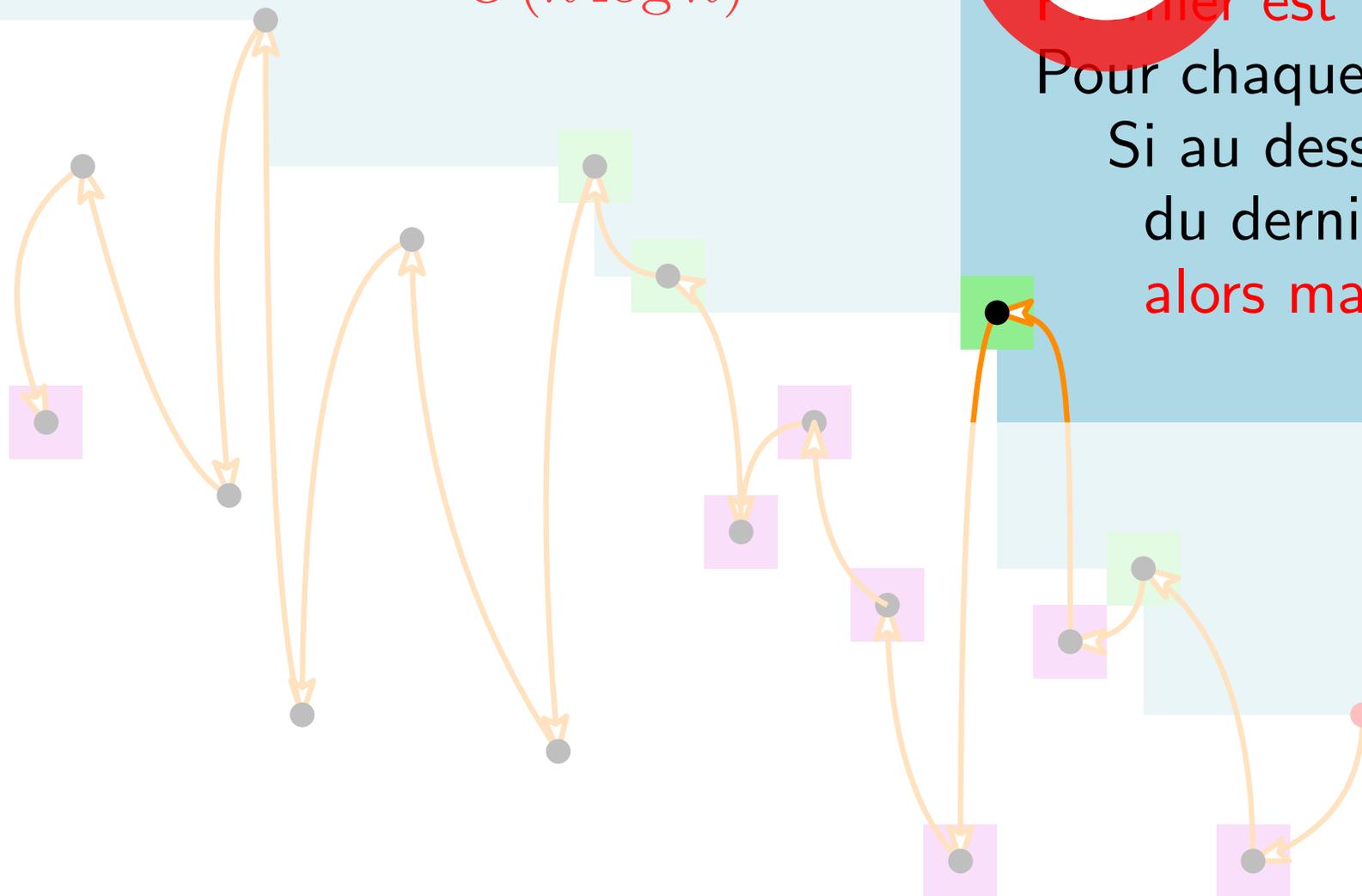
du dernier maximal

alors maximal

# Premier exemple: points maximaux

Complexité de l'algorithme

$$O(n \log n)$$



Algorithme

Trier

Premier est maximal

Pour chaque point

Si au dessus

du dernier maximal

alors maximal



# Premier exemple: points maximaux

Complexité de l'algorithme

$$O(n \log n)$$

$$\Theta(n \log n)$$

Complexité pour des points aléatoires dans un rectangle

$$\Theta(n)$$

Algorithme

Trier

Premier est maximal

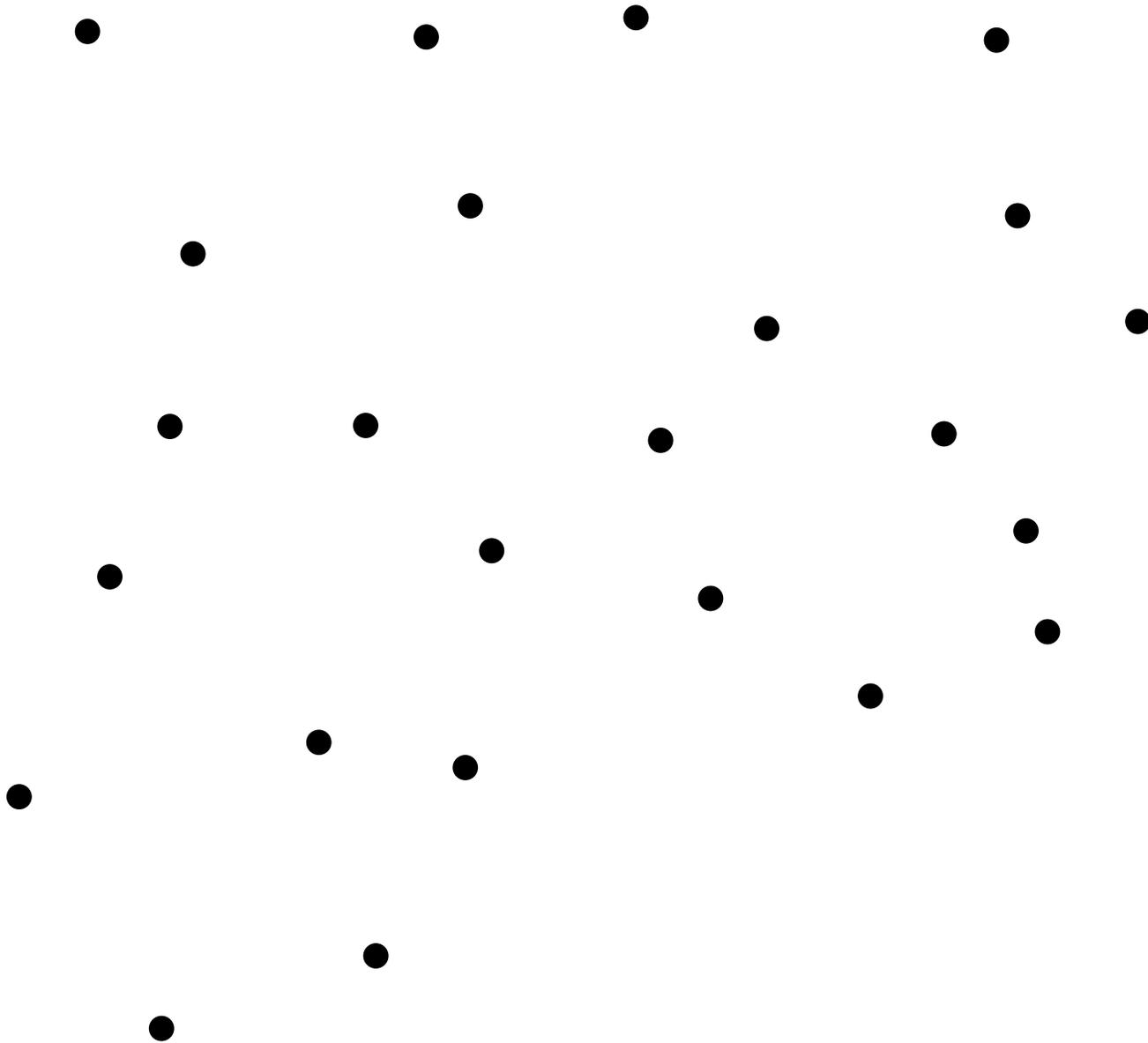
Pour chaque point

Si au dessus

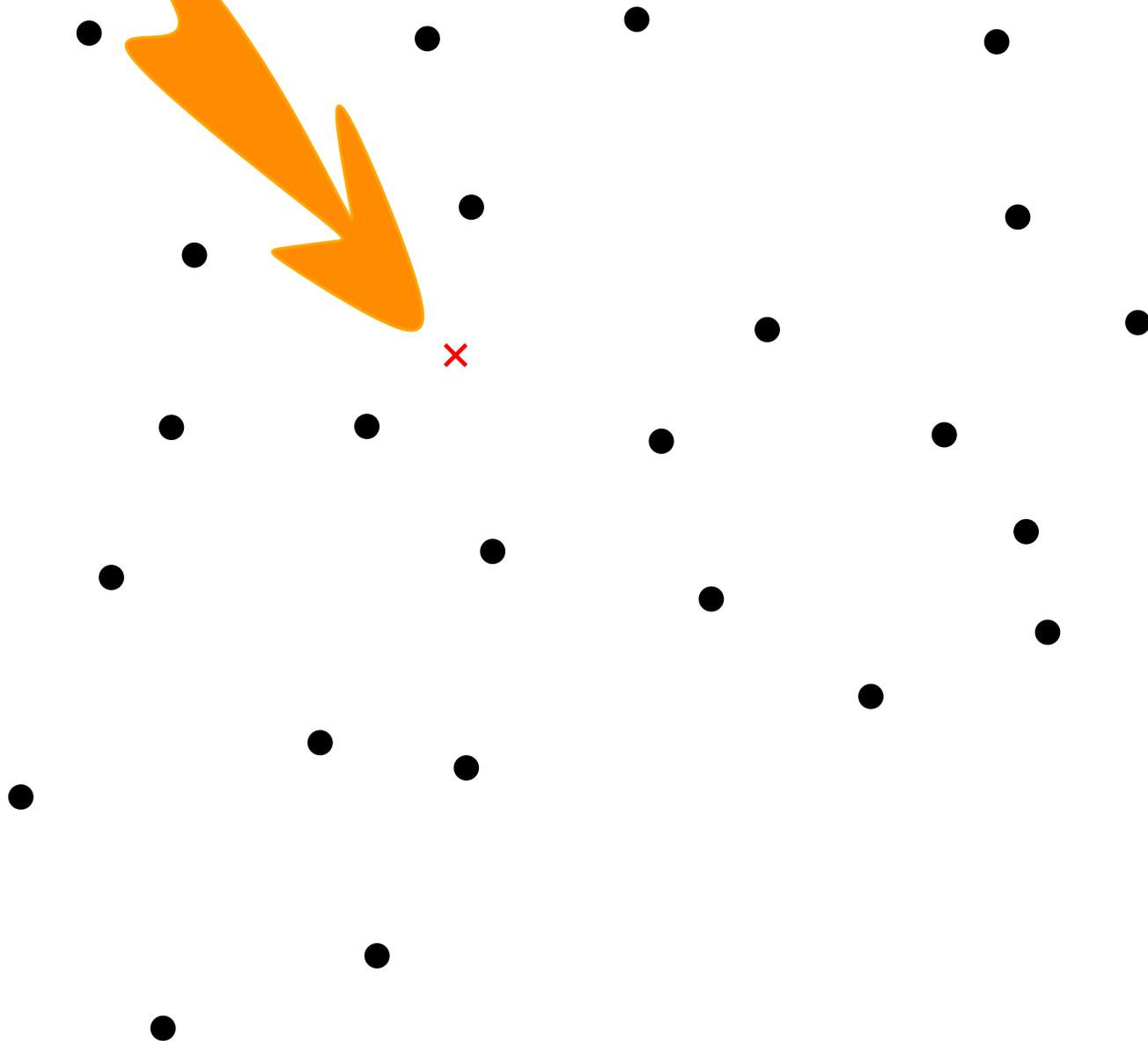
du dernier maximal

alors maximal

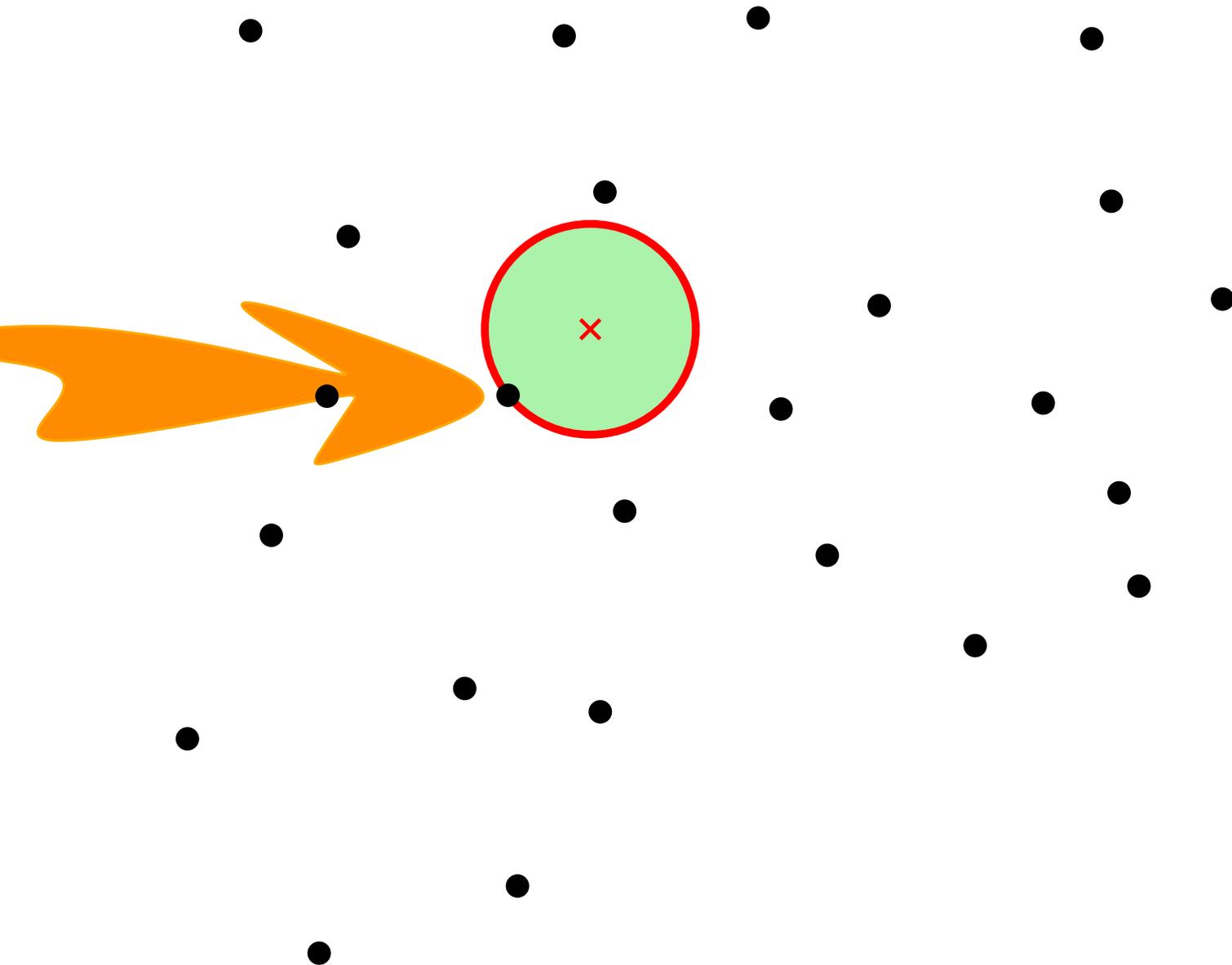
# Mon problème favori



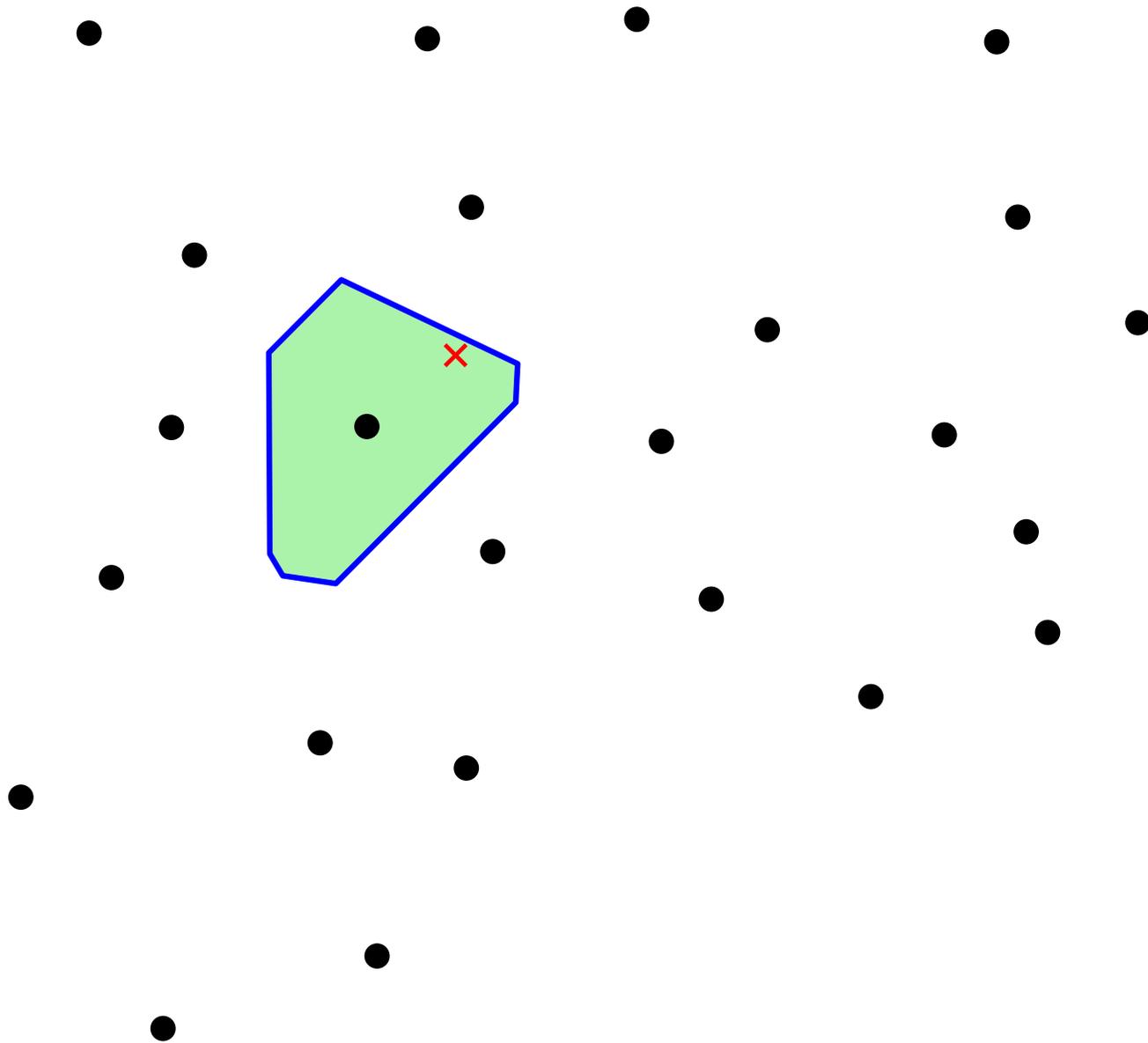
# Mon problème favori



# Mon problème favori



# Mon problème favori



# Mon problème favori

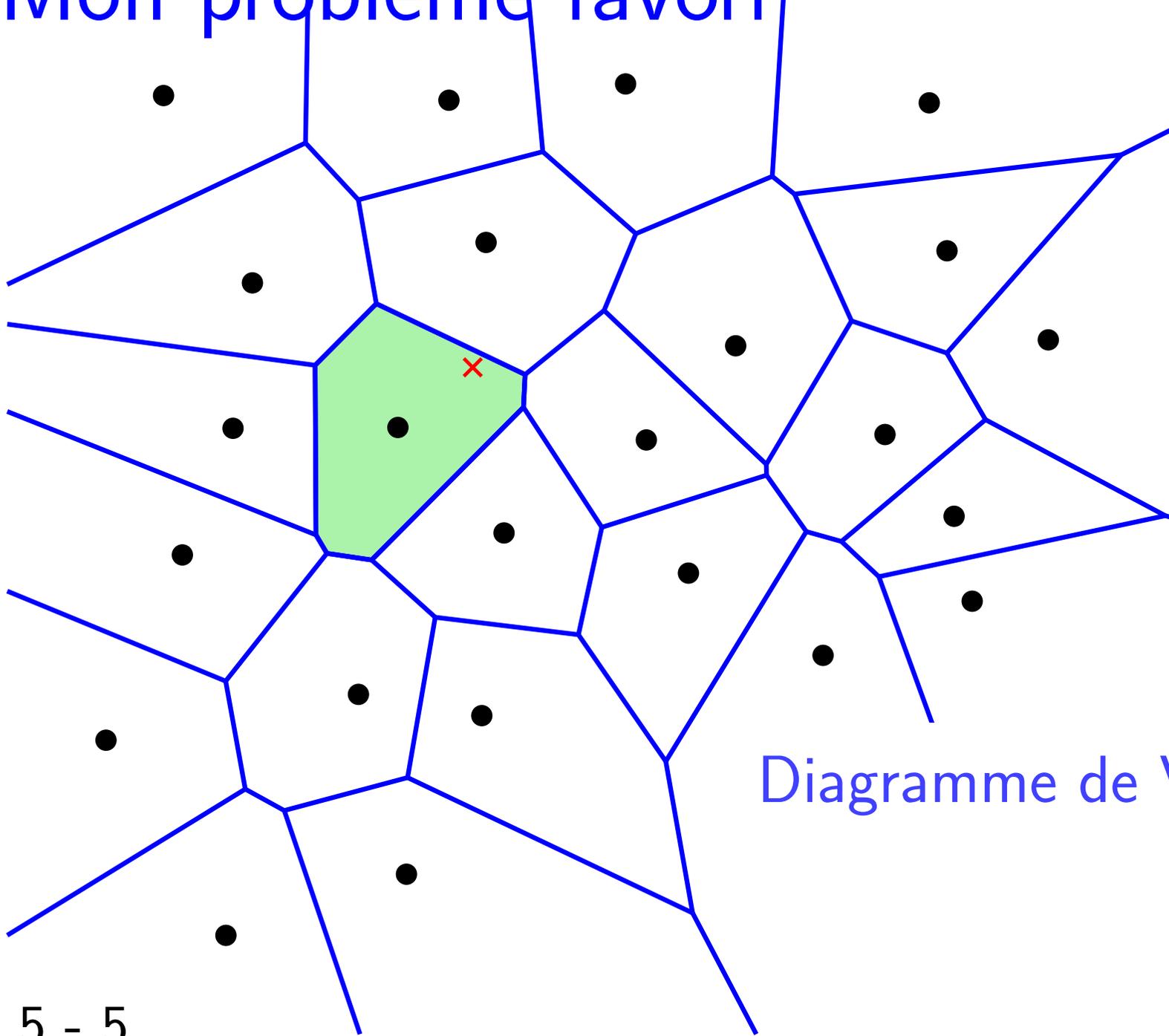


Diagramme de Voronoï

# Mon problème favori

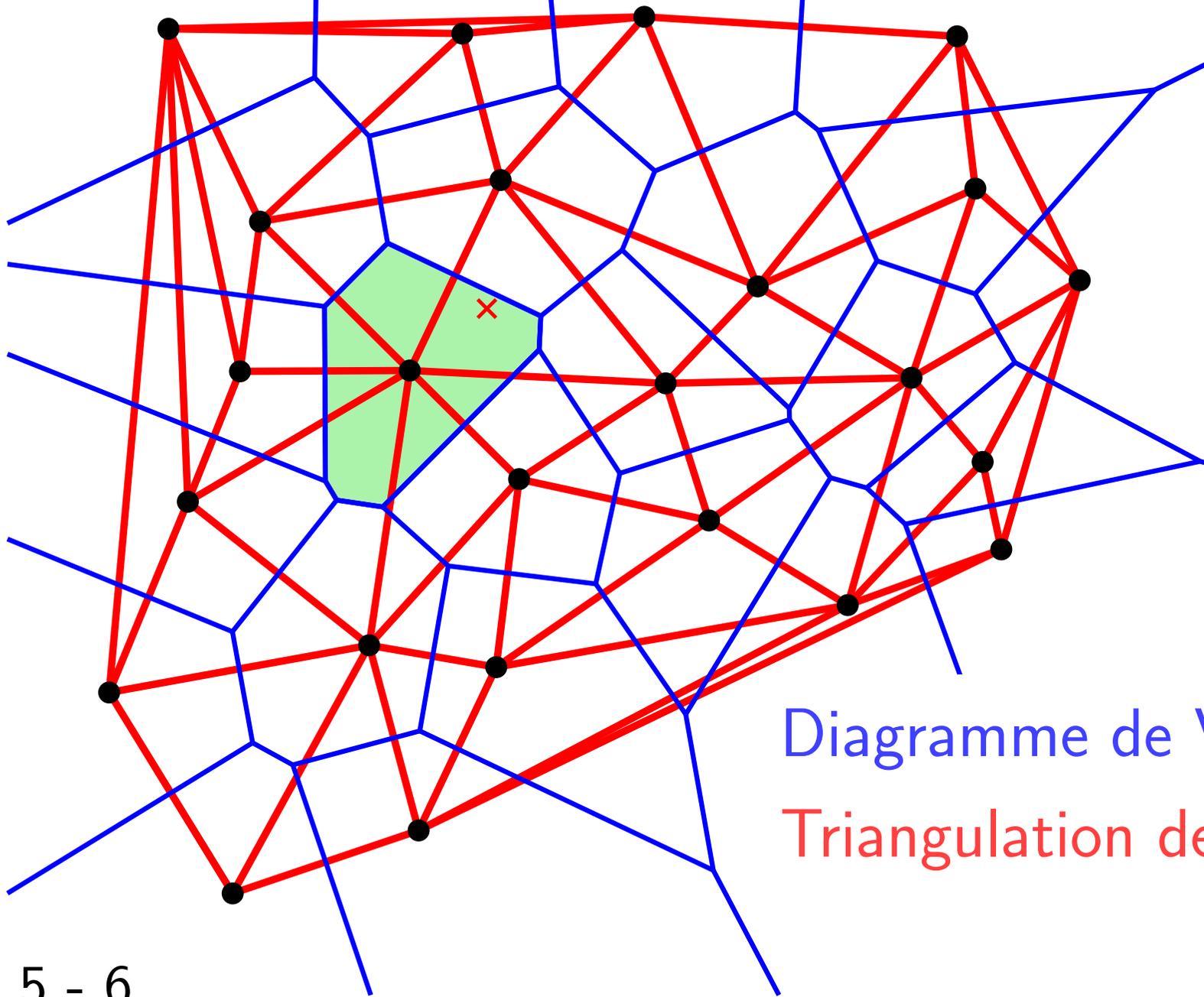


Diagramme de Voronoï

Triangulation de Delaunay

# Mon problème favori

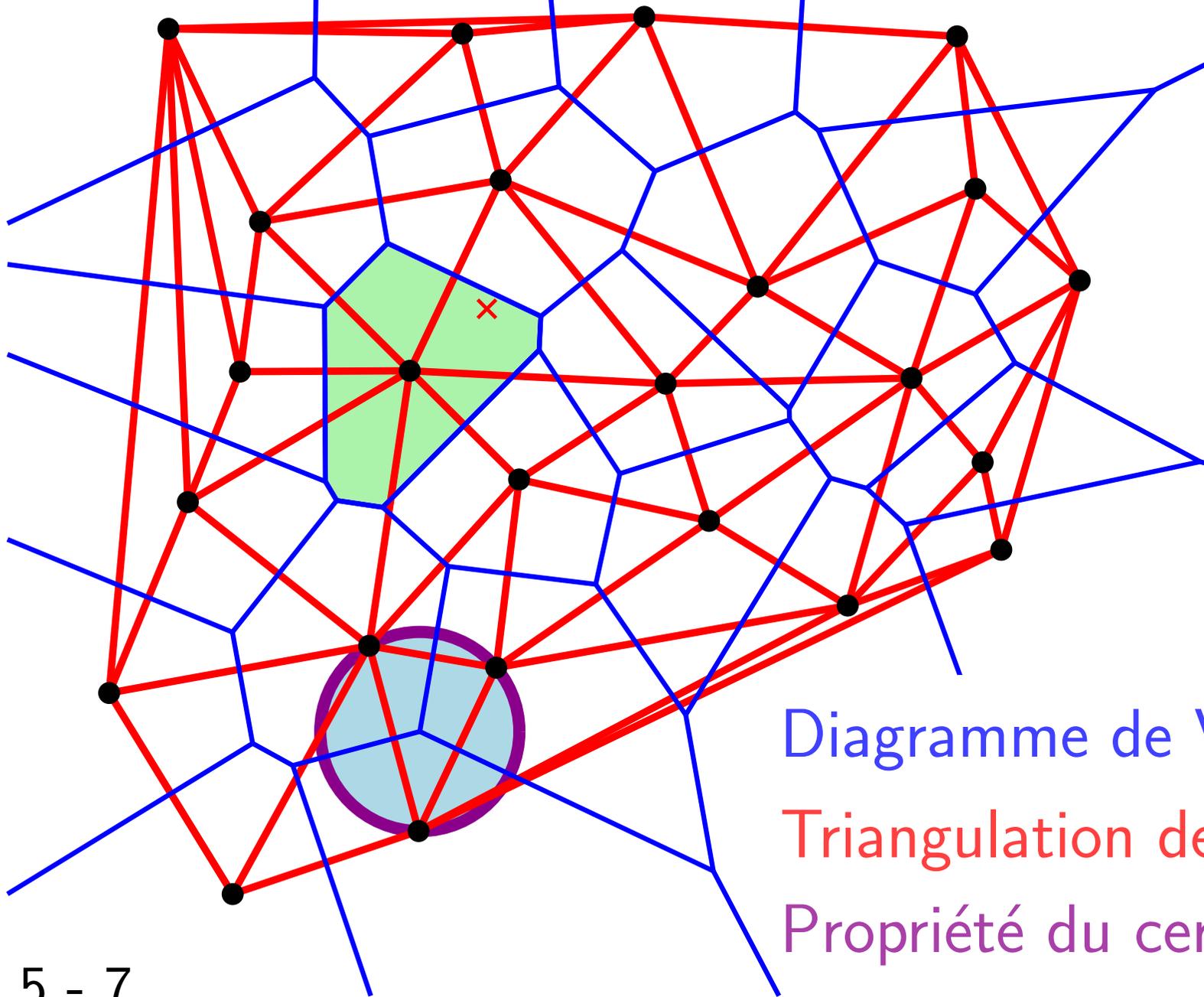


Diagramme de Voronoï  
Triangulation de Delaunay  
Propriété du cercle vide

# Taille d'un graphe planaire

$$\# \text{faces} - \# \text{arêtes} + \# \text{sommets} = \# \text{comp. connexes}$$

# Taille d'un graphe planaire

$$\# \text{faces} - \# \text{arêtes} + \# \text{sommets} = \# \text{comp. connexes}$$

par induction

initialement

$$0=0$$

# Taille d'un graphe planaire

$$\# \text{faces} - \# \text{arêtes} + \# \text{sommets} = \# \text{comp. connexes}$$

par induction

$$\text{initialement} \quad 0=0$$

$$\text{ajout d'un sommet} \quad +0-0+1=+1$$



# Taille d'un graphe planaire

$$\# \text{faces} - \# \text{arêtes} + \# \text{sommets} = \# \text{comp. connexes}$$

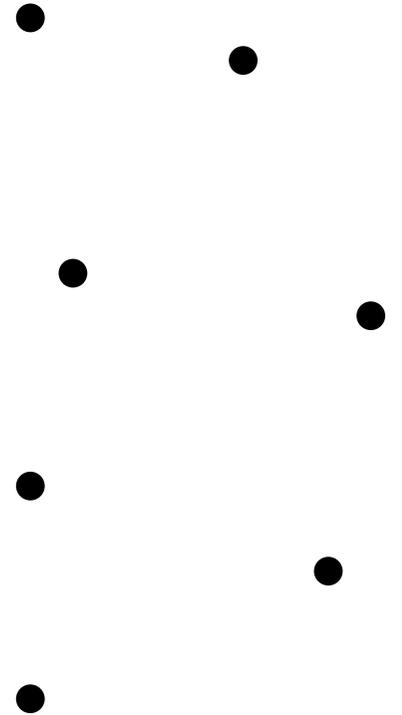
par induction

initialement

$$0=0$$

ajout d'un sommet

$$+0-0+1=+1$$



# Taille d'un graphe planaire

$$\# \text{faces} - \# \text{arêtes} + \# \text{sommets} = \# \text{comp. connexes}$$

par induction

initialement

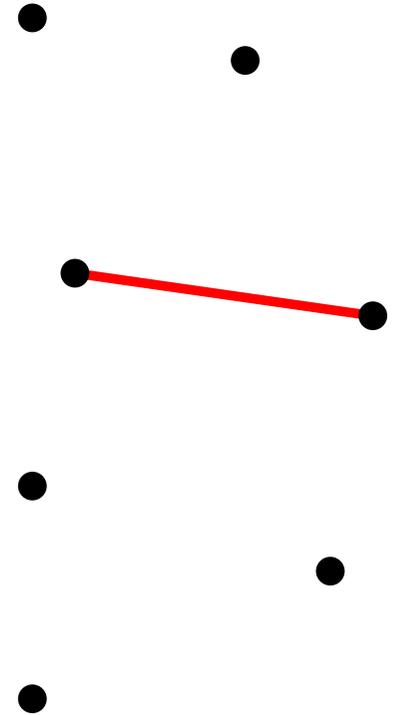
$$0=0$$

ajout d'un sommet

$$+0-0+1=+1$$

ajout d'une arête

$$+0-1+0=-1$$



# Taille d'un graphe planaire

$$\#faces - \#arêtes + \#sommets = \#comp. \text{ connexes}$$

par induction

initialement

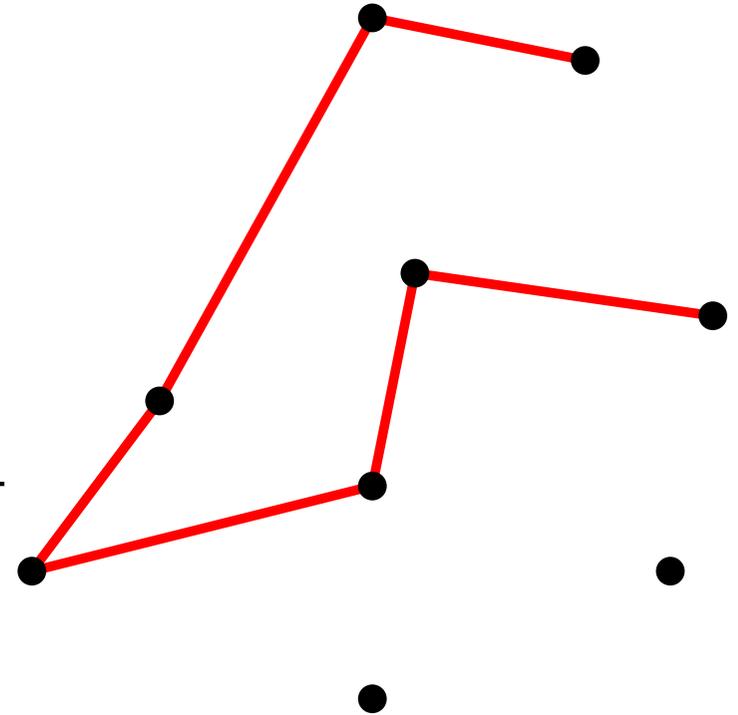
$$0=0$$

ajout d'un sommet

$$+0-0+1=+1$$

ajout d'une arête

$$+0-1+0=-1$$



# Taille d'un graphe planaire

$$\# \text{faces} - \# \text{arêtes} + \# \text{sommets} = \# \text{comp. connexes}$$

par induction

initialement

$$0=0$$

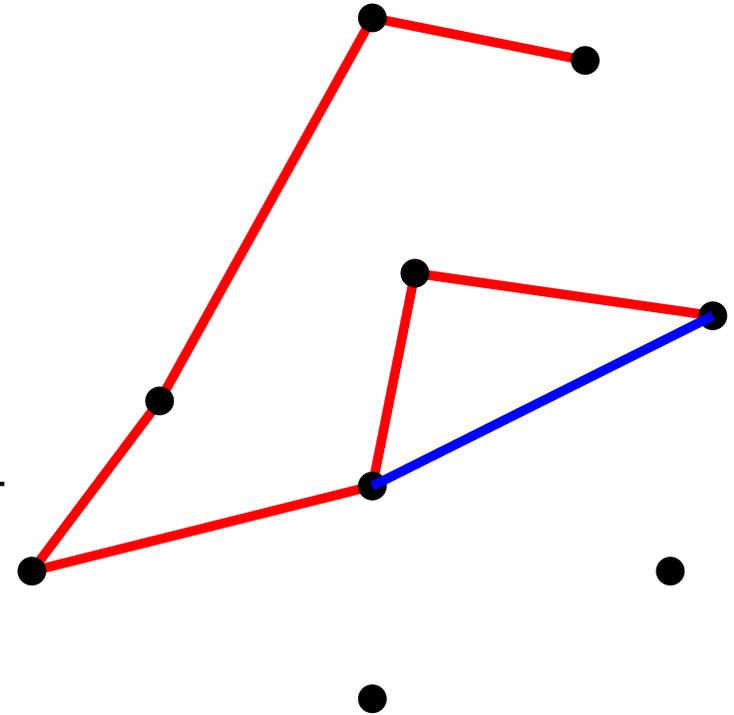
ajout d'un sommet

$$+0-0+1=+1$$

ajout d'une arête

$$+0-1+0=-1$$

$$+1-1+0=+0$$



# Taille d'un graphe planaire

$$\# \text{faces} - \# \text{arêtes} + \# \text{sommets} = \# \text{comp. connexes}$$

par induction

initialement

$$0=0$$

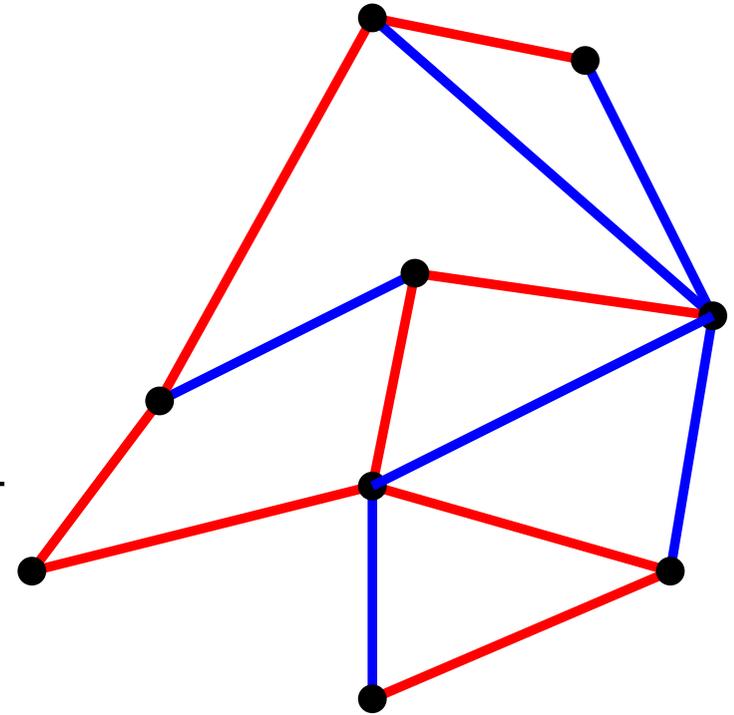
ajout d'un sommet

$$+0-0+1=+1$$

ajout d'une arête

$$+0-1+0=-1$$

$$+1-1+0=+0$$



# Taille d'un graphe planaire

$$\# \text{faces} - \# \text{arêtes} + \# \text{sommets} = \# \text{comp. connexes}$$

par induction

initialement

$$0 = 0$$

ajout d'un sommet

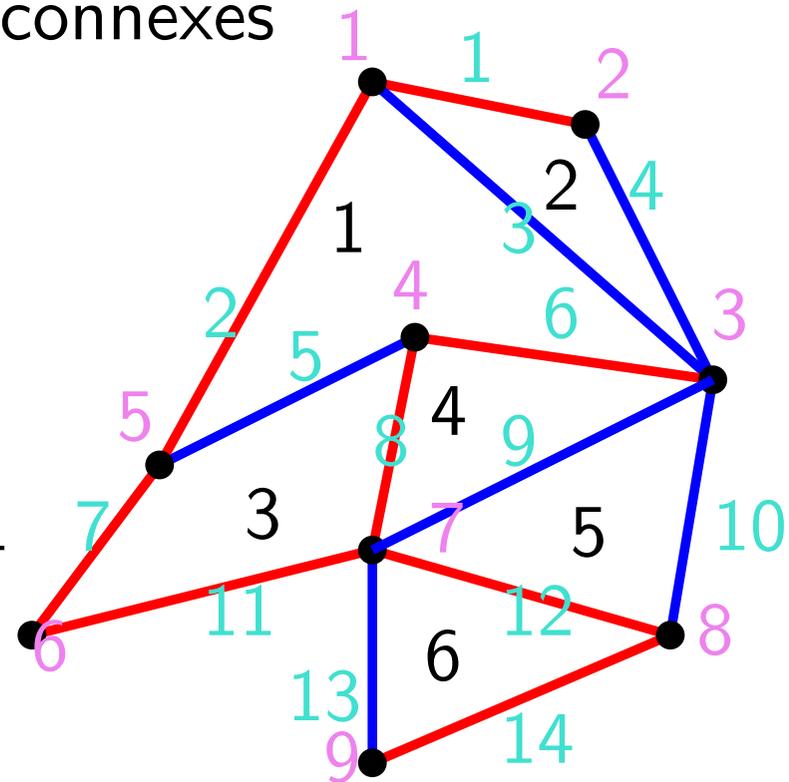
$$+0 - 0 + 1 = +1$$

ajout d'une arête

$$+0 - 1 + 0 = -1$$

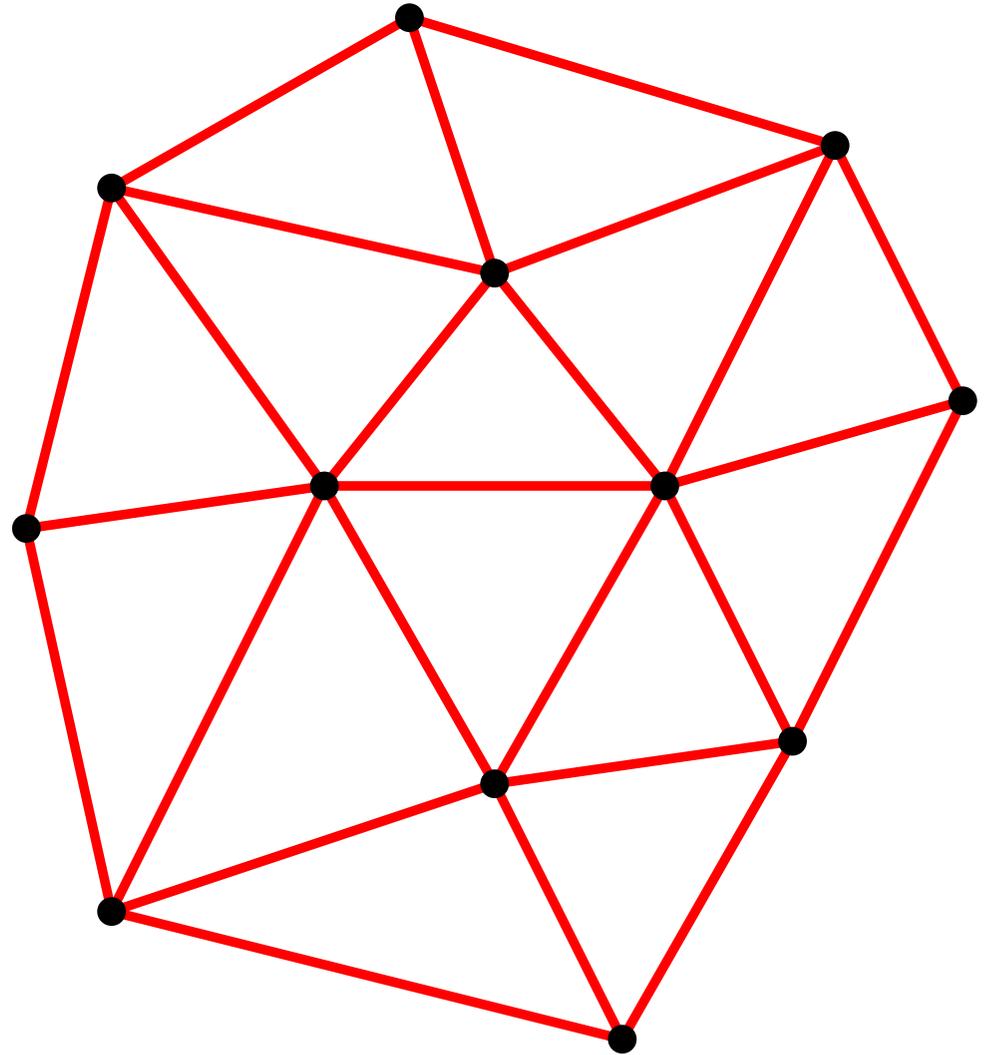
$$+1 - 1 + 0 = +0$$

$$+6 - 14 + 9 = 1$$



# Taille d'une triangulation

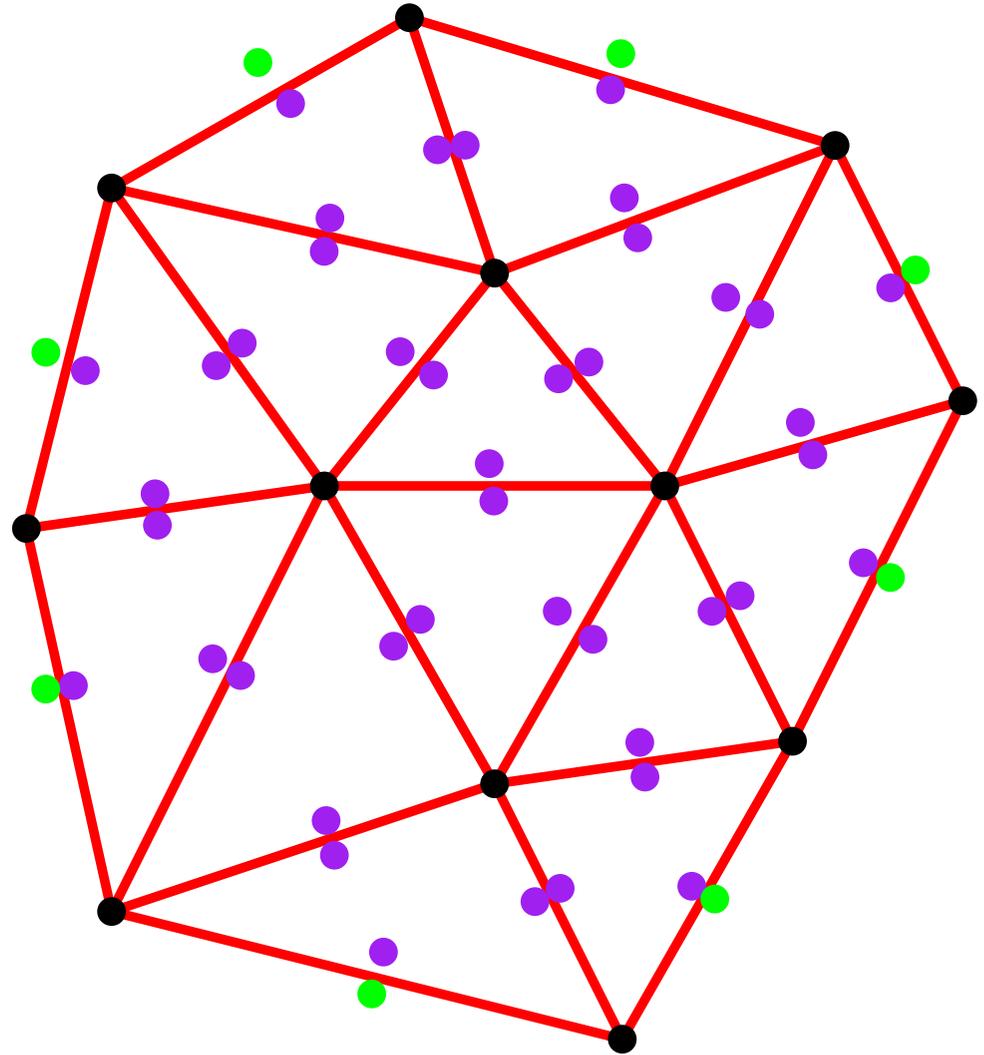
$$\#t - \#a + \#s = 1$$



# Taille d'une triangulation

$$\#t - \#a + \#s = 1$$

$$3\#t + \#bord = 2\#a$$



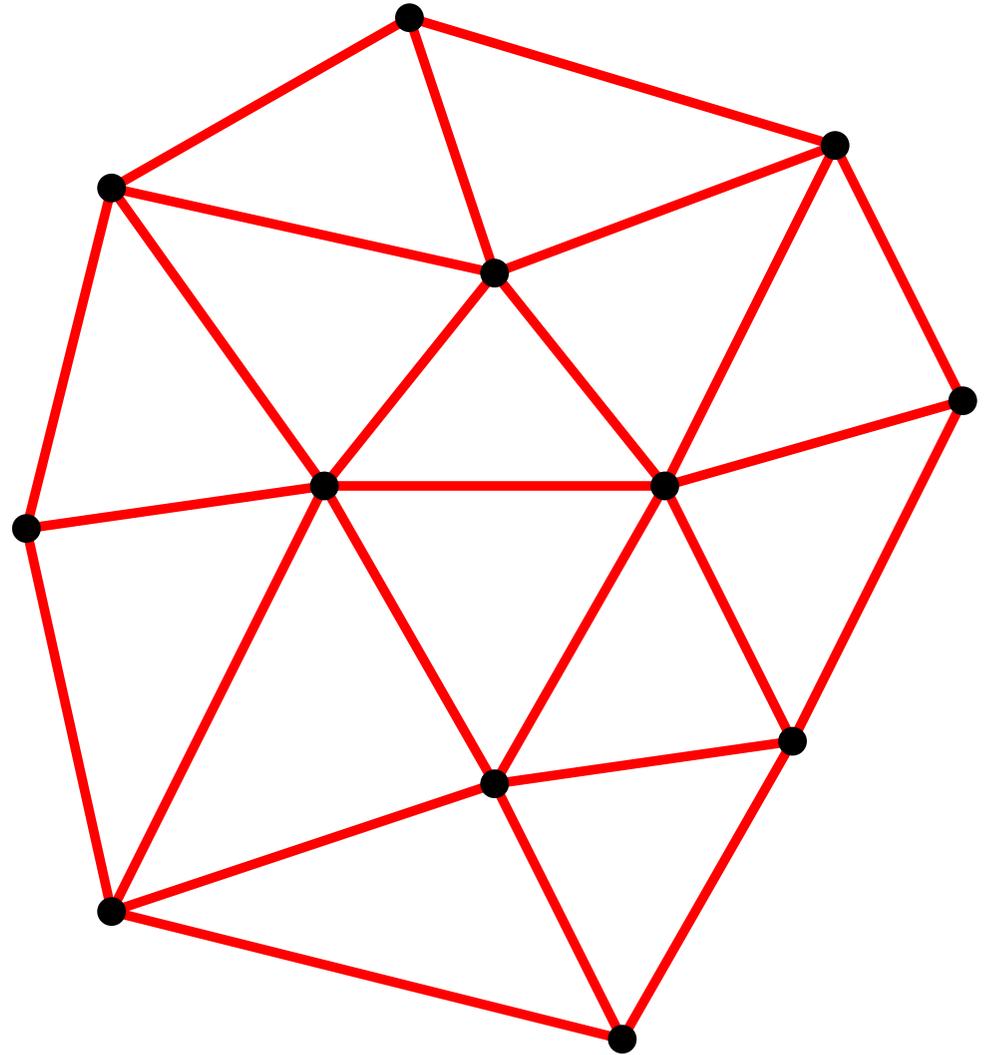
# Taille d'une triangulation

$$\#t - \#a + \#s = 1$$

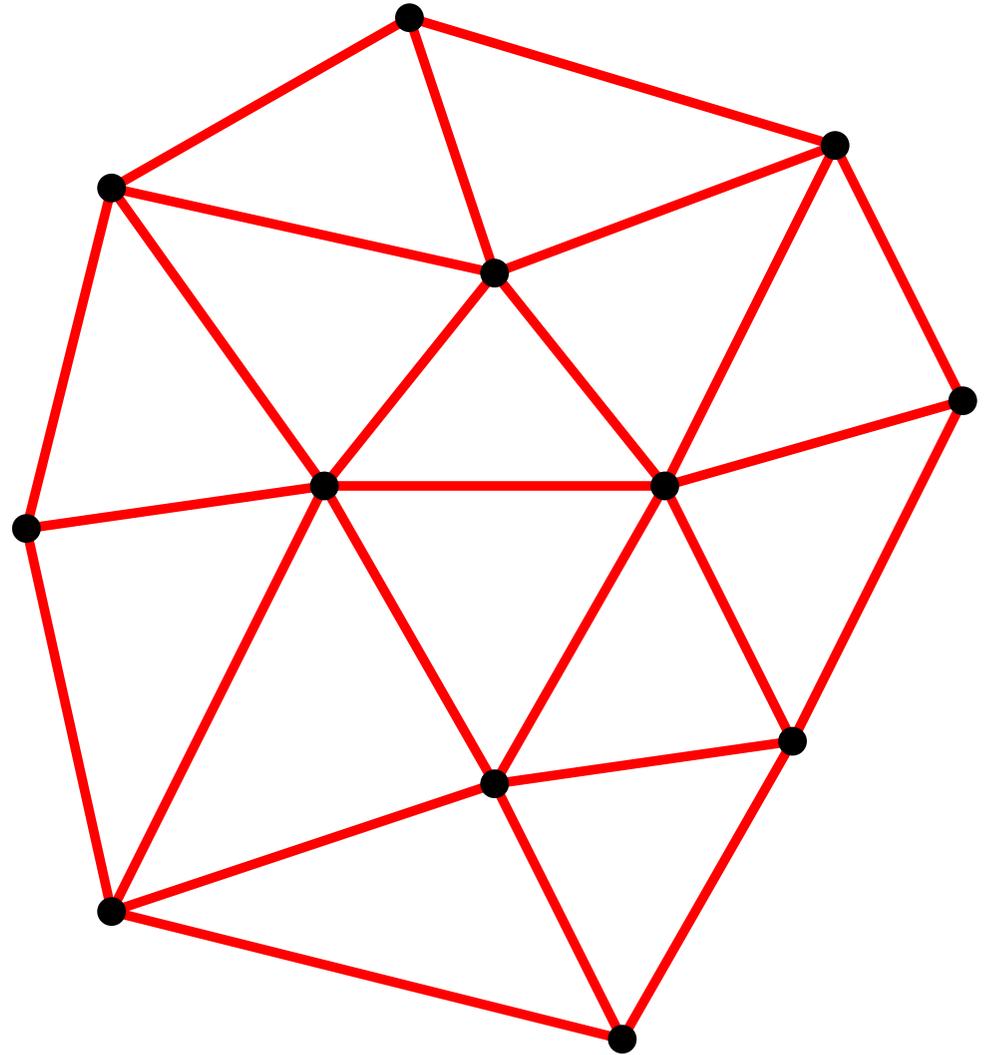
$$3\#t + \#bord = 2\#a$$

$$\#t = 2\#s - 2 - \#b$$

$$\#a = 3\#s - 3 - \#b$$

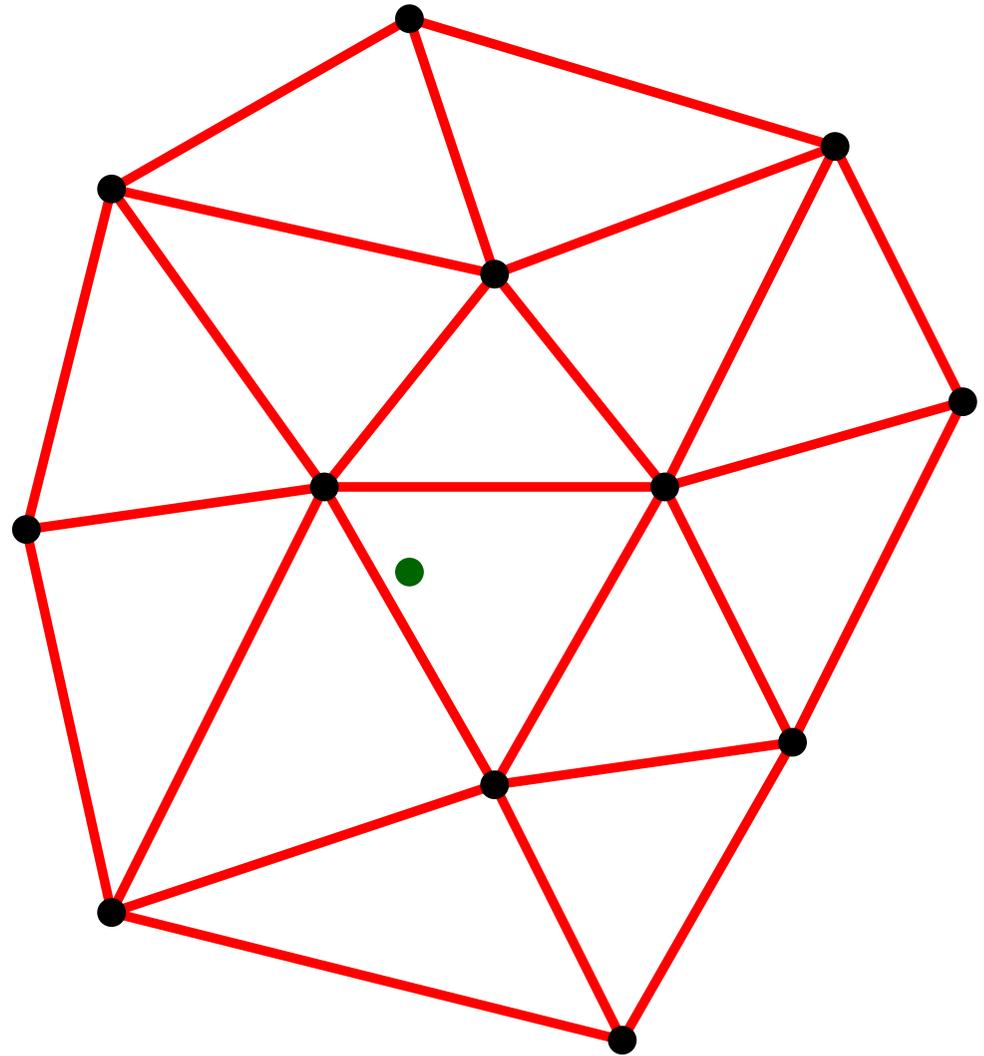


# Construction incrémentale de Delaunay



# Construction incrémentale de Delaunay

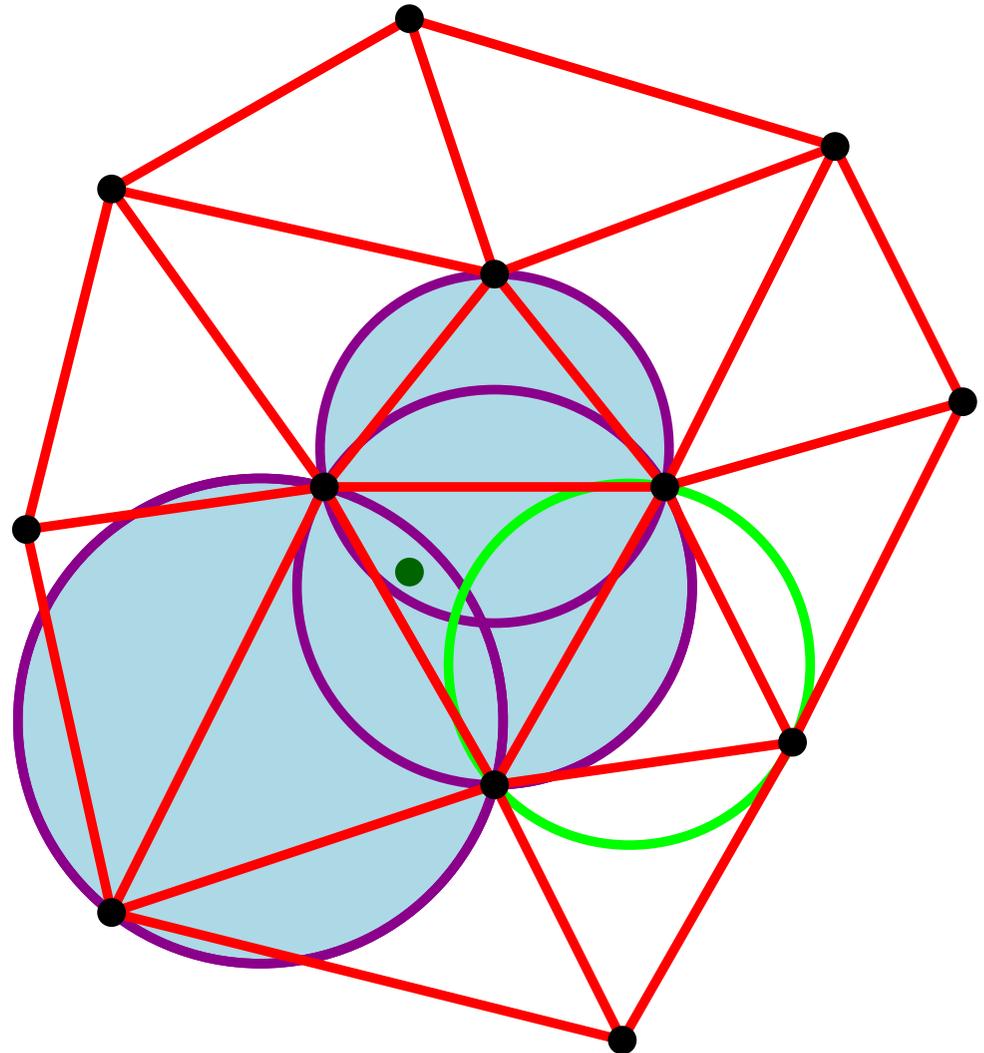
nouveau point



# Construction incrémentale de Delaunay

nouveau point

triangles en conflits

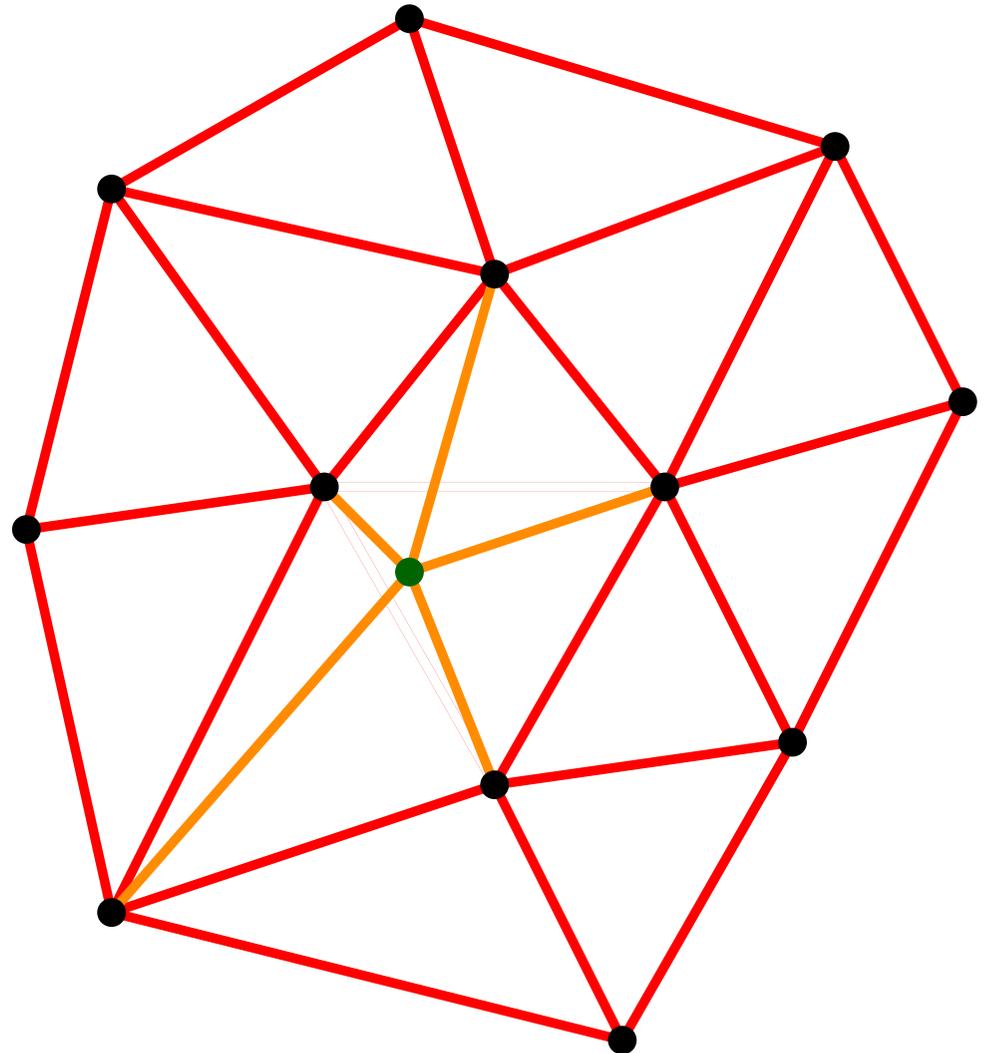


# Construction incrémentale de Delaunay

nouveau point

triangles en conflits

retriangler



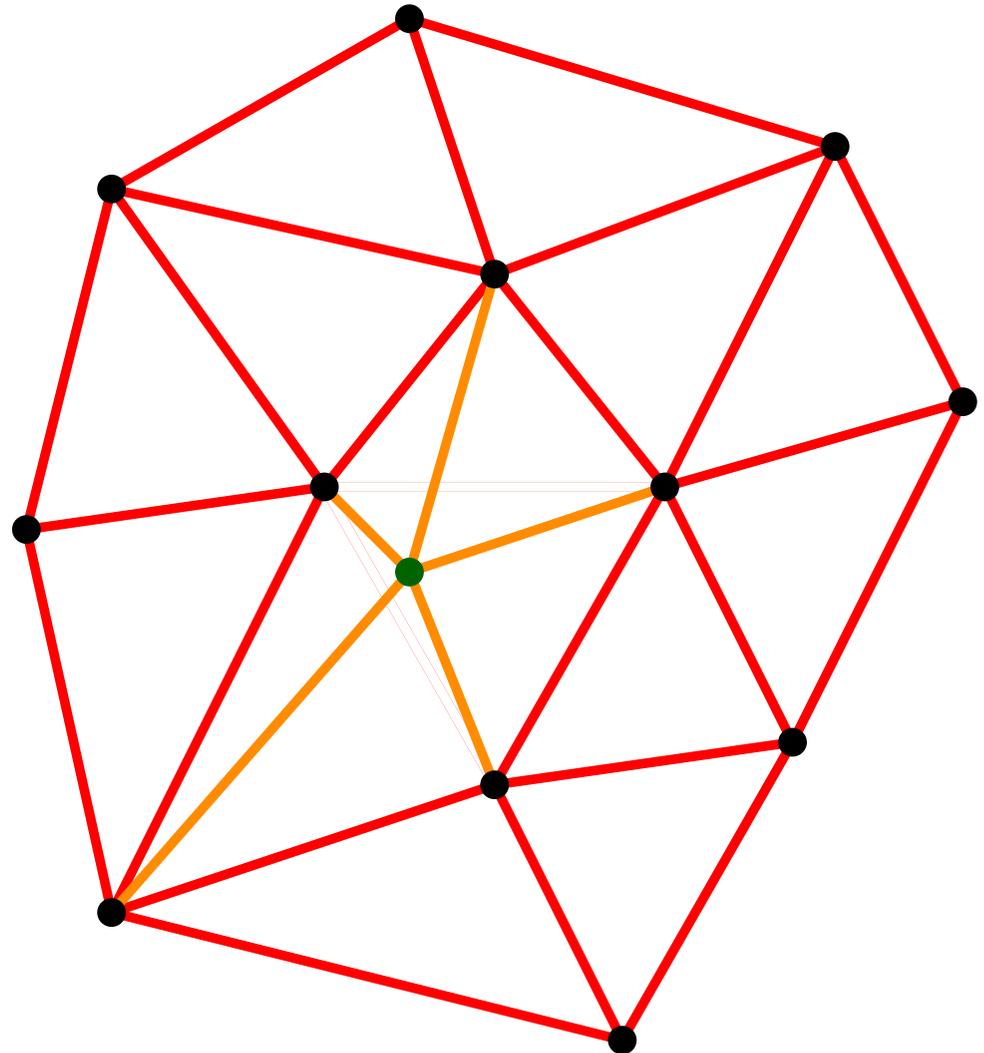
# Construction incrémentale de Delaunay

nouveau point

triangles en conflits

retriangler

Combien de triangles ?



# Construction incrémentale de Delaunay

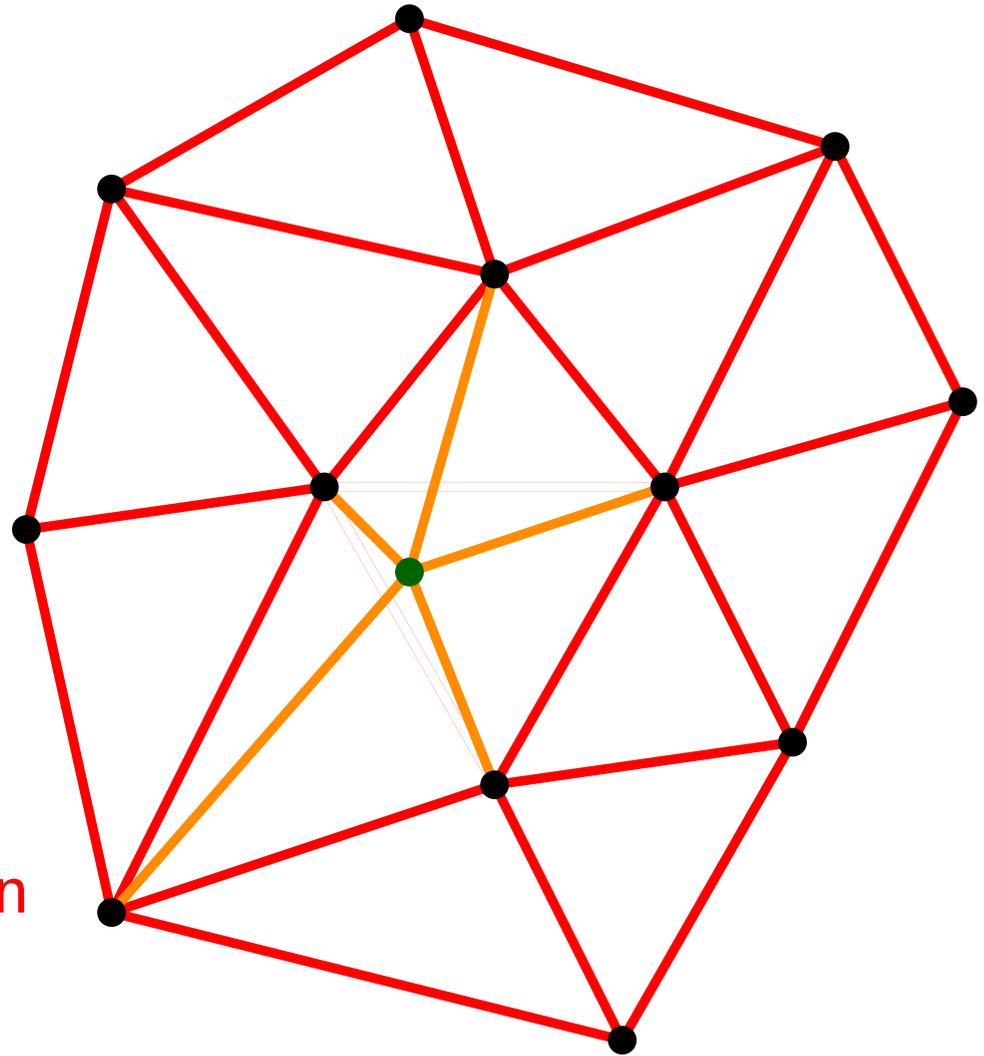
nouveau point

triangles en conflits

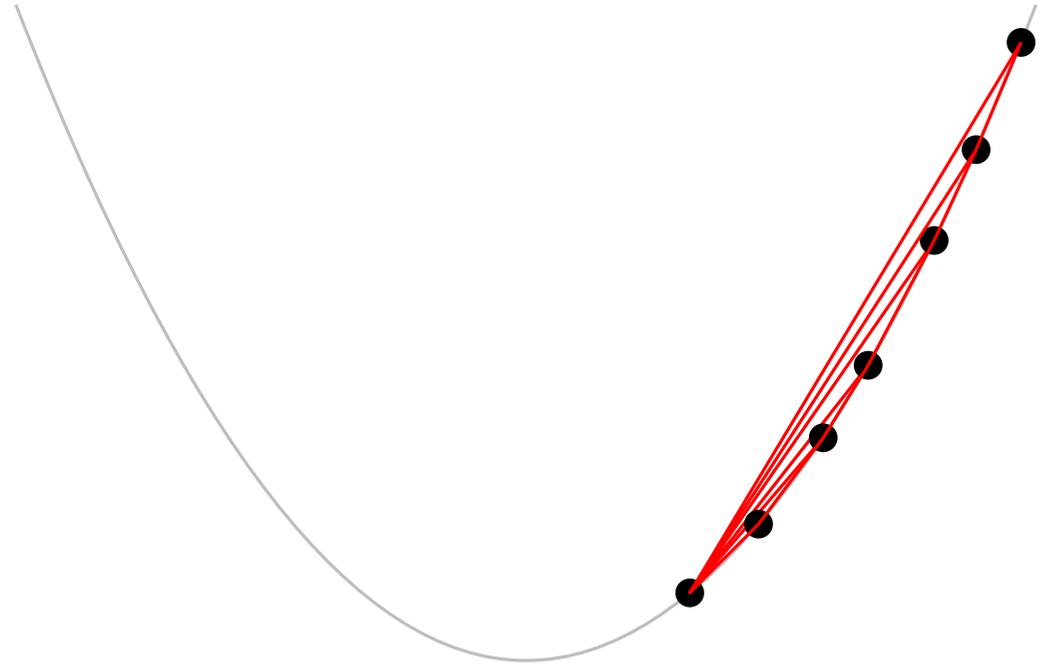
retriangler

Combien de triangles ?

$$\#t = 2\#s - 2 - \#b \text{ à la fin}$$

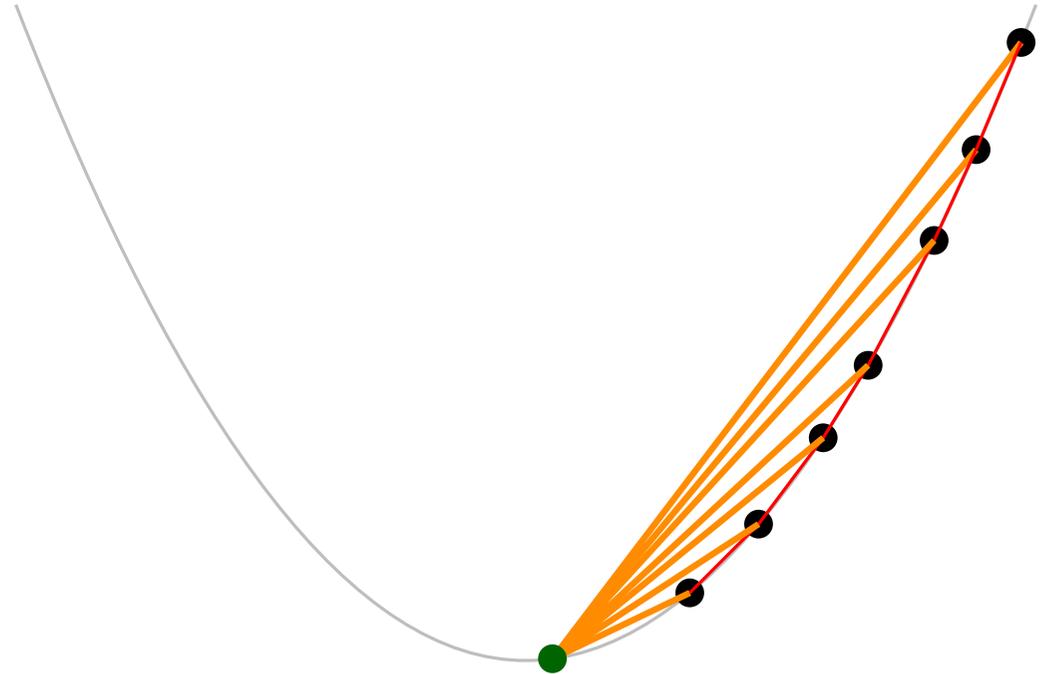


# Construction incrémentale de Delaunay



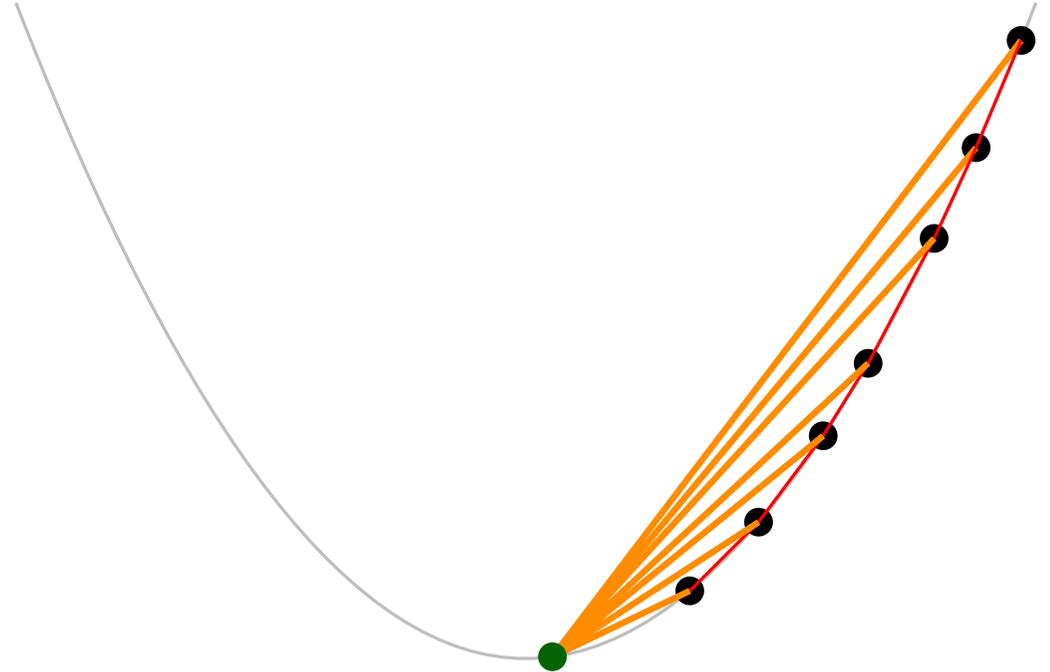
Combien de triangles ?  
au total pendant la construction ?

# Construction incrémentale de Delaunay



Combien de triangles ?  
au total pendant la construction ?

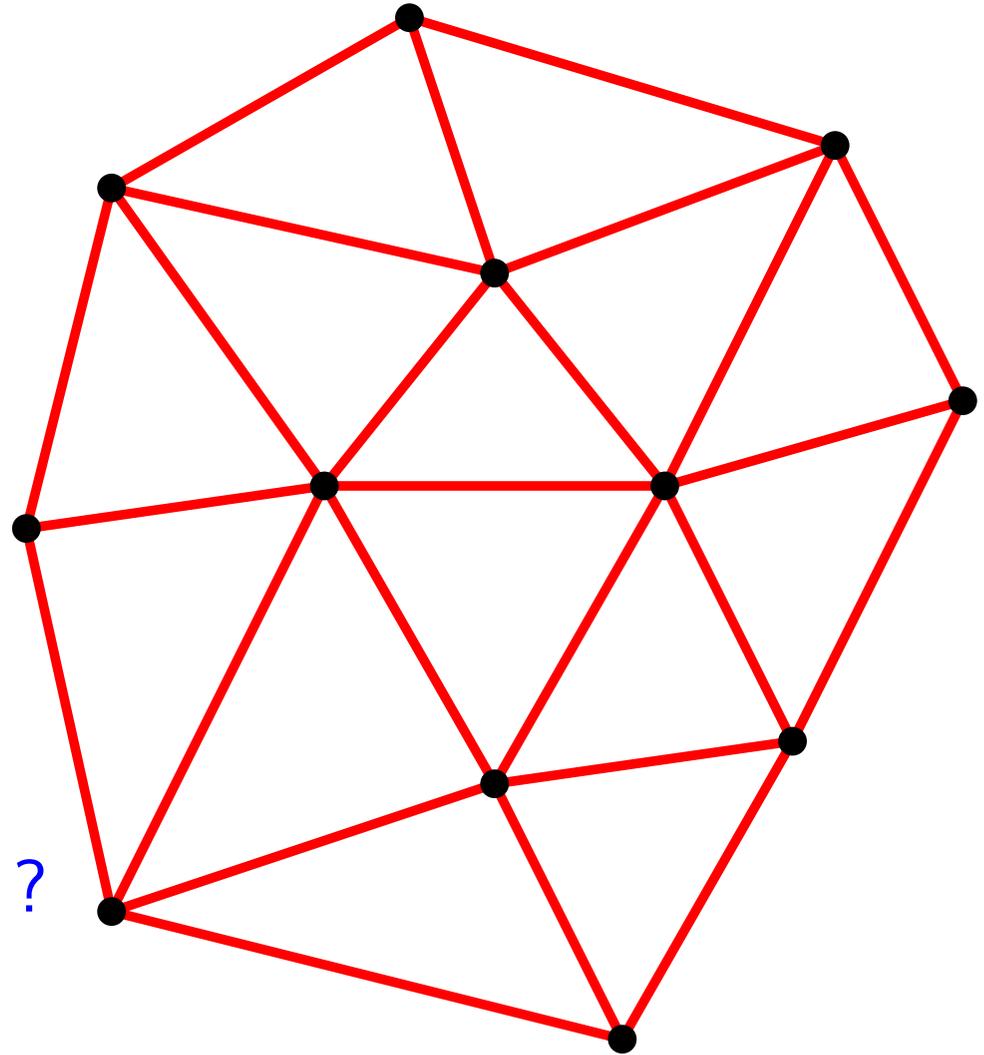
# Construction incrémentale de Delaunay



Combien de triangles ?  
au total pendant la construction ?

$$1 + 2 + 3 + \dots = \frac{(n-2)(n-1)}{2} \text{ dans le cas le pire}$$

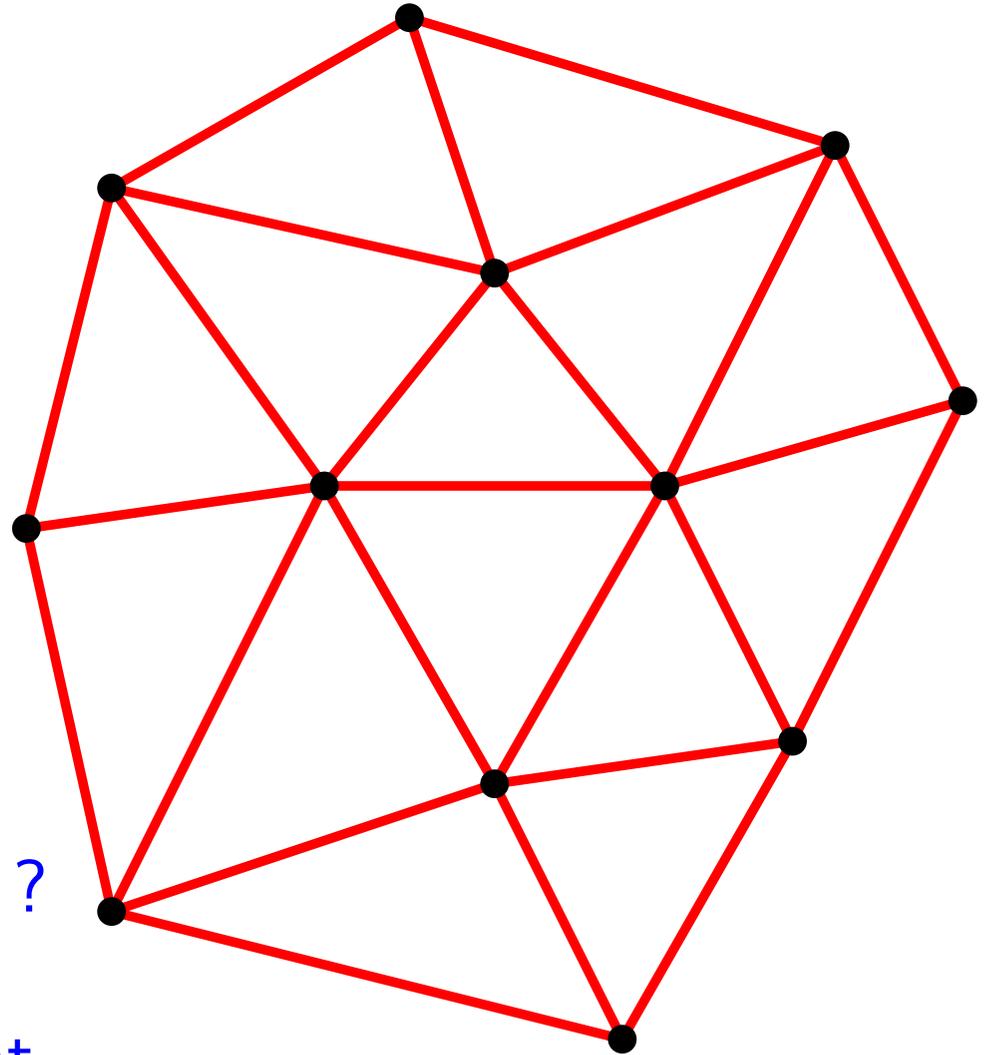
# Construction incrémentale de Delaunay



Combien de triangles ?

créés par le dernier point ?

# Construction incrémentale de Delaunay



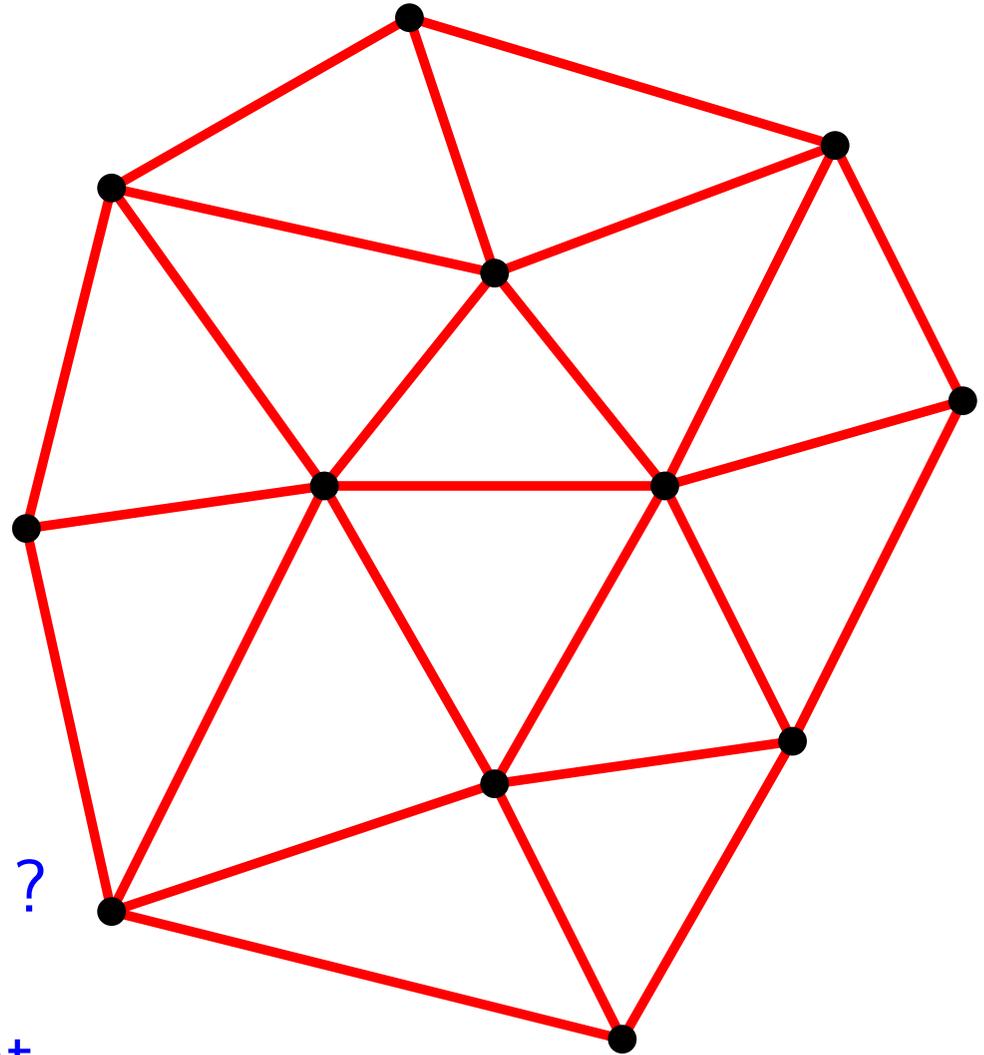
Combien de triangles ?

créés par le dernier point ?

degré du dernier point

# Construction incrémentale de Delaunay

$$\sum_{k \leq n} d^{\circ}(k^{\text{ème}} \text{ point})$$



Combien de triangles ?

créés par le dernier point ?

degré du dernier point

# Construction incrémentale de Delaunay

$$\sum_{k \leq n} d^{\circ}(k^{\text{ème}} \text{ point})$$

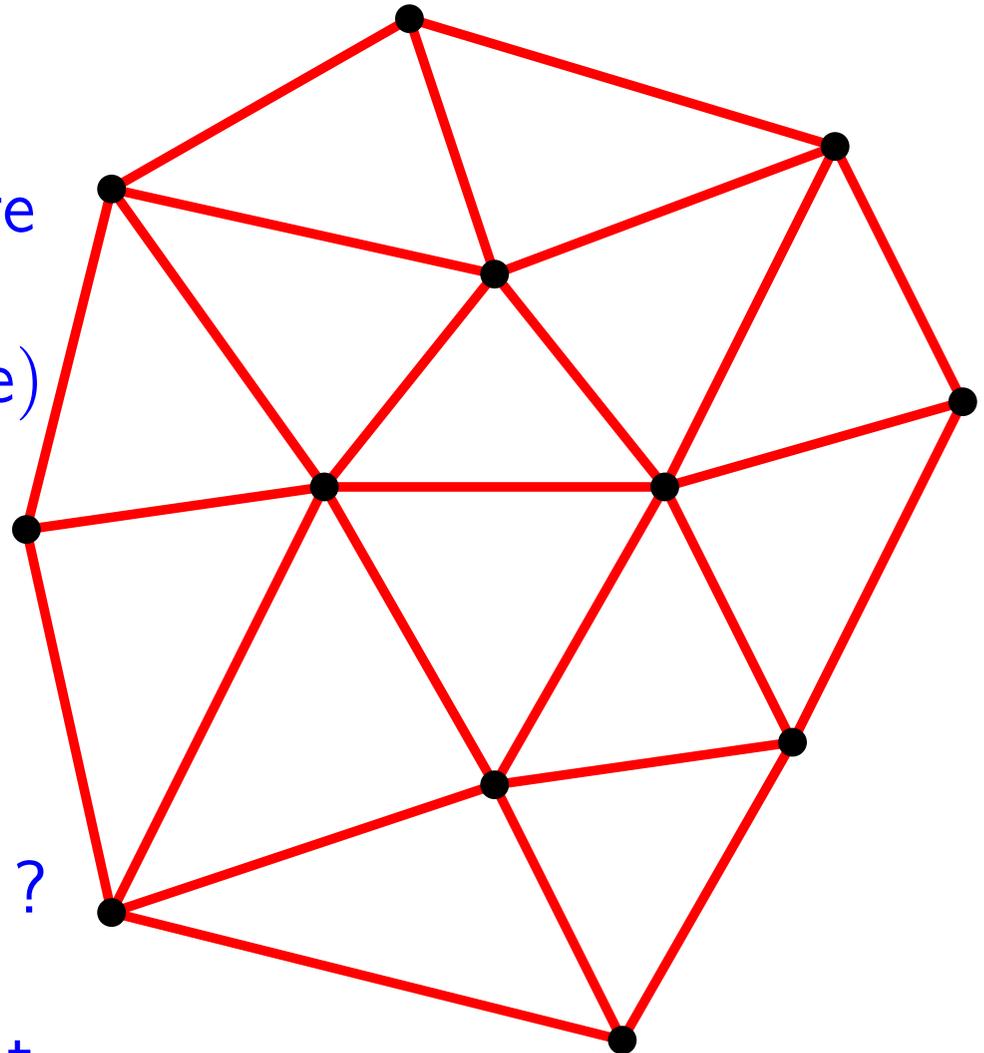
si ordre d'insertion aléatoire

$$= \sum_{k \leq n} d^{\circ}(\text{point aléatoire})$$

Combien de triangles ?

créés par le dernier point ?

degré du dernier point

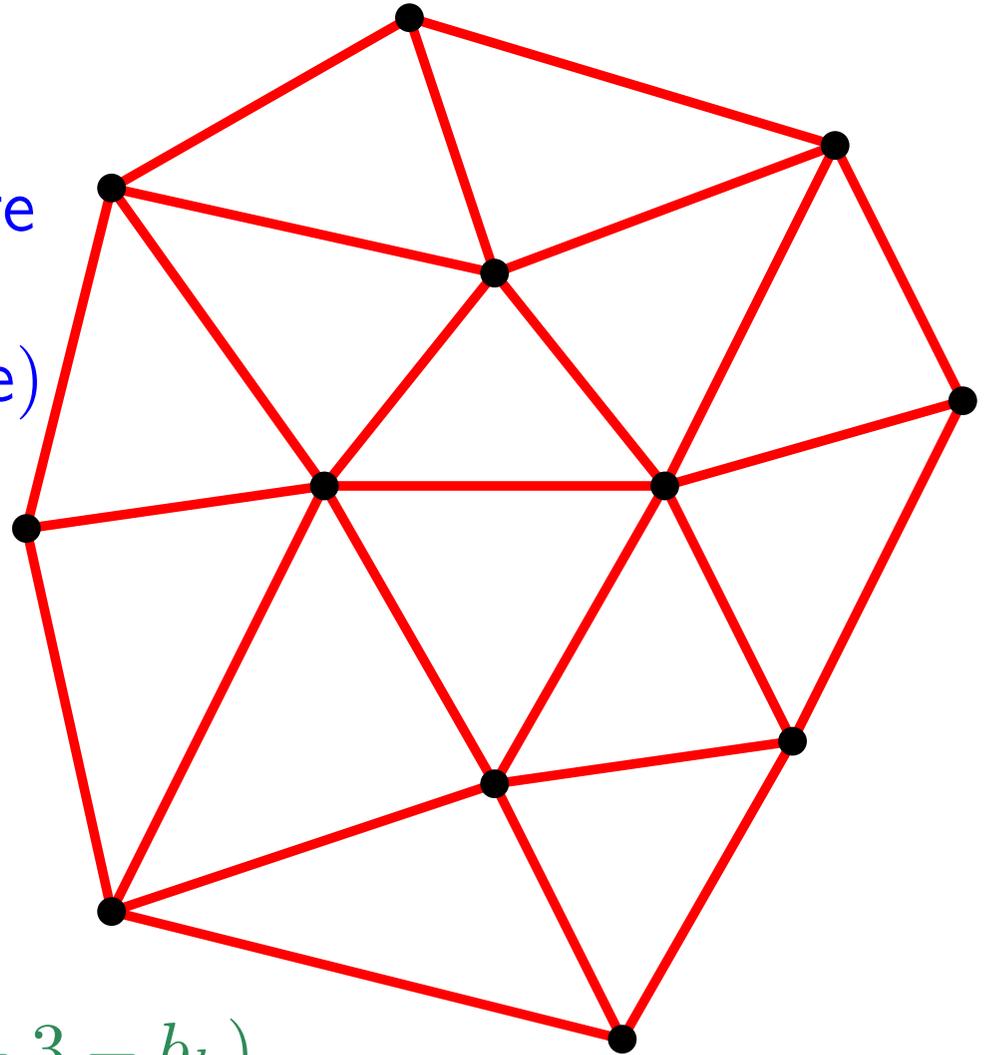


# Construction incrémentale de Delaunay

$$\sum_{k \leq n} d^\circ(k^{\text{ème}} \text{ point})$$

si ordre d'insertion aléatoire

$$= \sum_{k \leq n} d^\circ(\text{point aléatoire})$$



$$d^\circ(\text{point aléatoire}) = \frac{1}{k} \sum d^\circ = \frac{2(3k - 3 - b_k)}{k} < 6$$

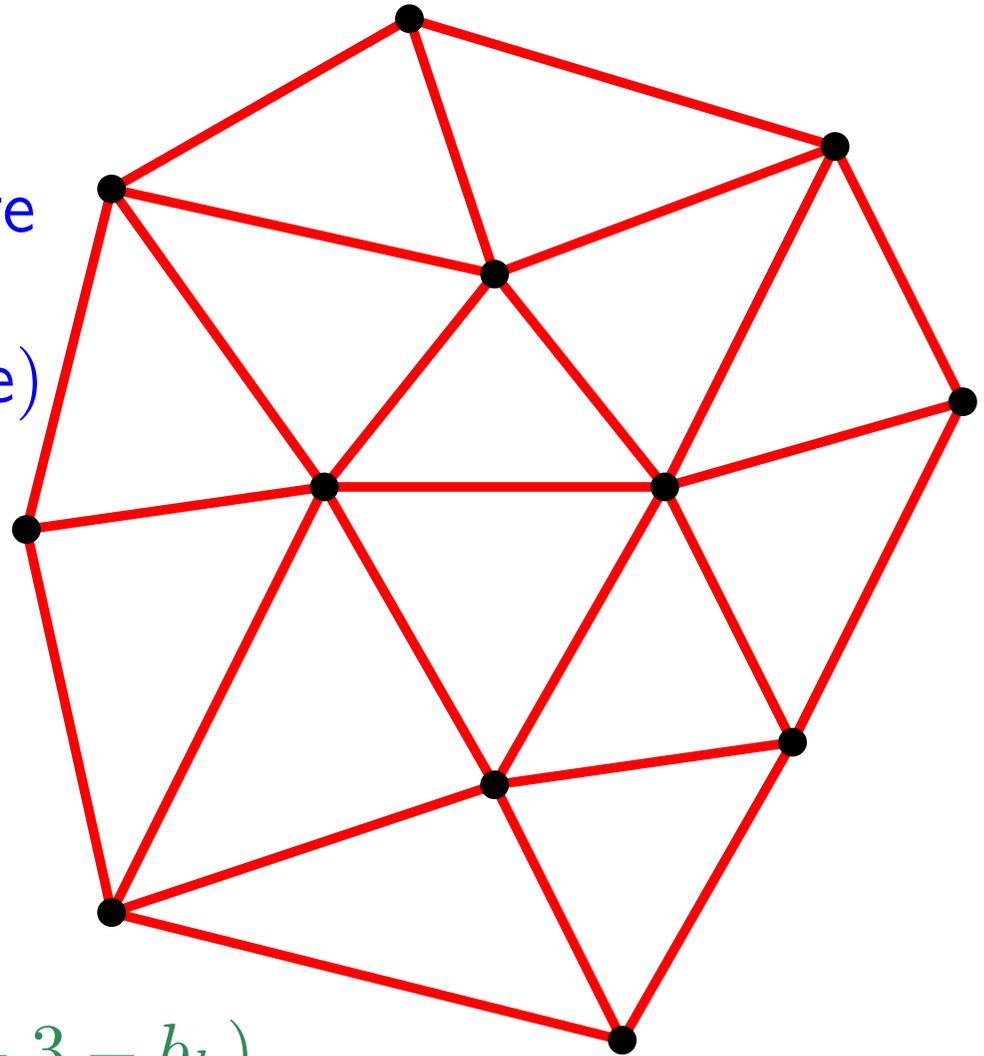
# Construction incrémentale de Delaunay

$$\sum_{k \leq n} d^\circ(k^{\text{ème}} \text{ point})$$

si ordre d'insertion aléatoire

$$= \sum_{k \leq n} d^\circ(\text{point aléatoire})$$

$$< \sum_{k \leq n} 6 = 6n$$



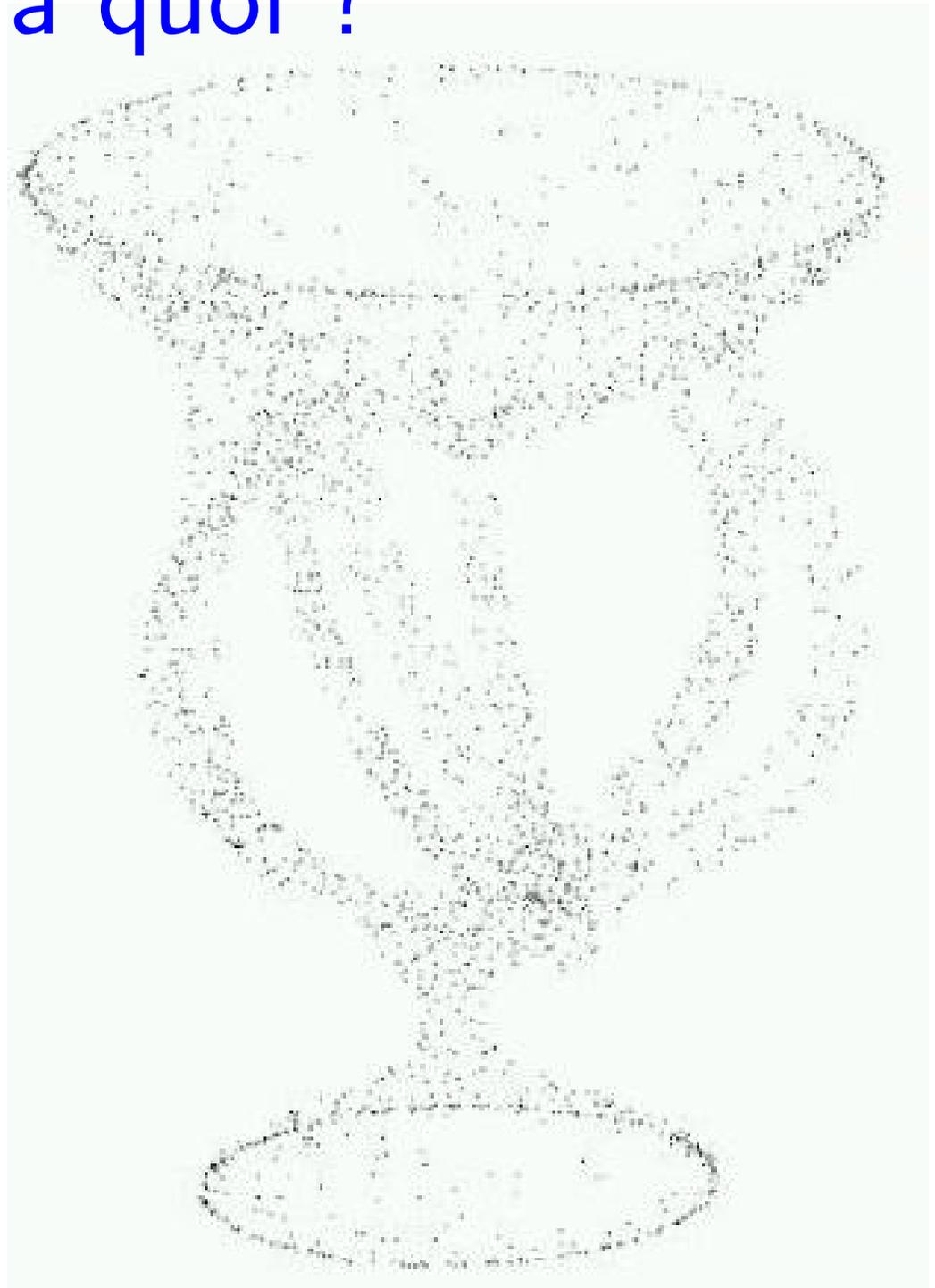
$$d^\circ(\text{point aléatoire})$$

$$= \frac{1}{k} \sum d^\circ = \frac{2(3k - 3 - b_k)}{k} < 6$$

# Delaunay, ça sert à quoi ?

## Reconstruction

Des points

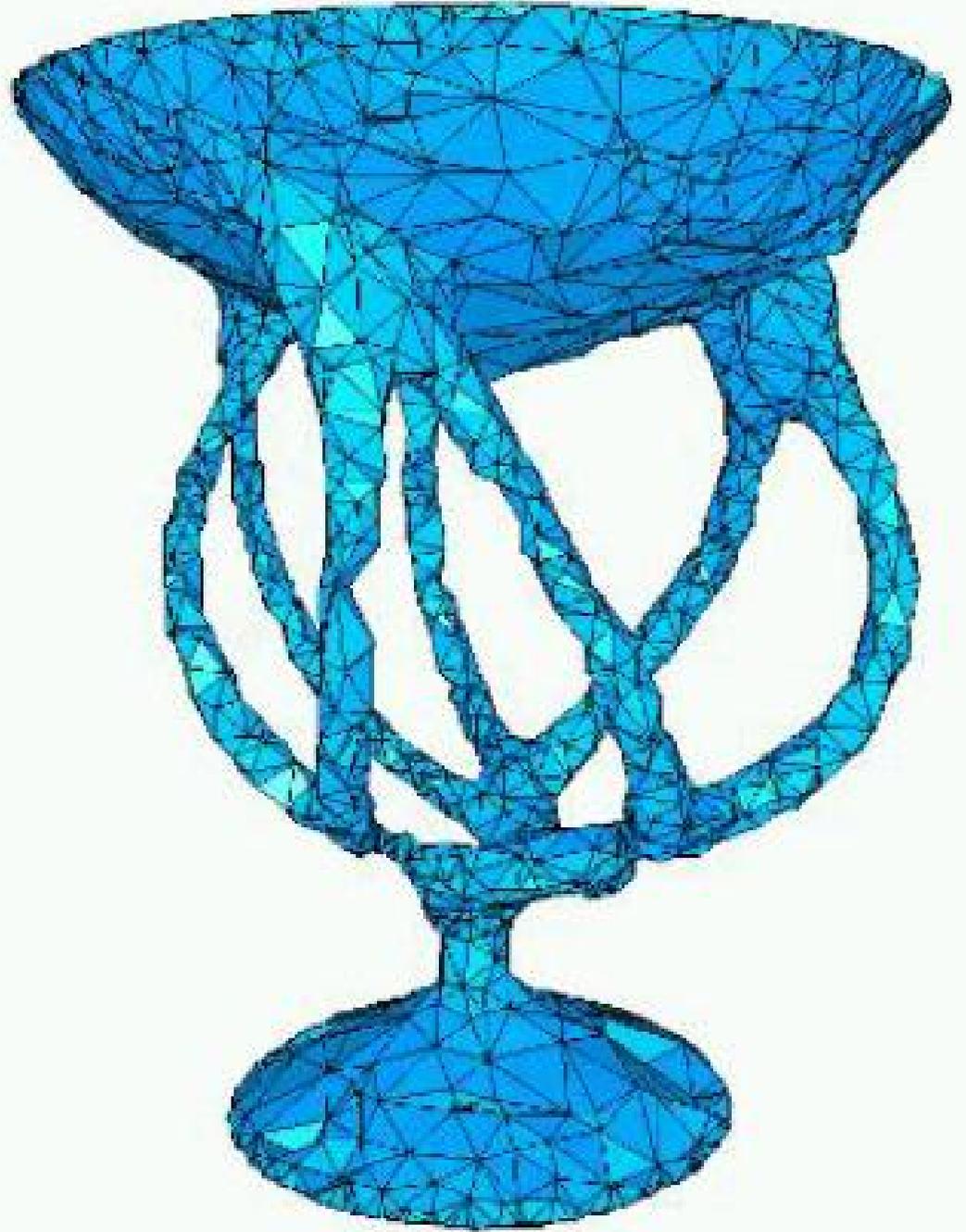


# Delaunay, ça sert à quoi ?

## Reconstruction

Des points

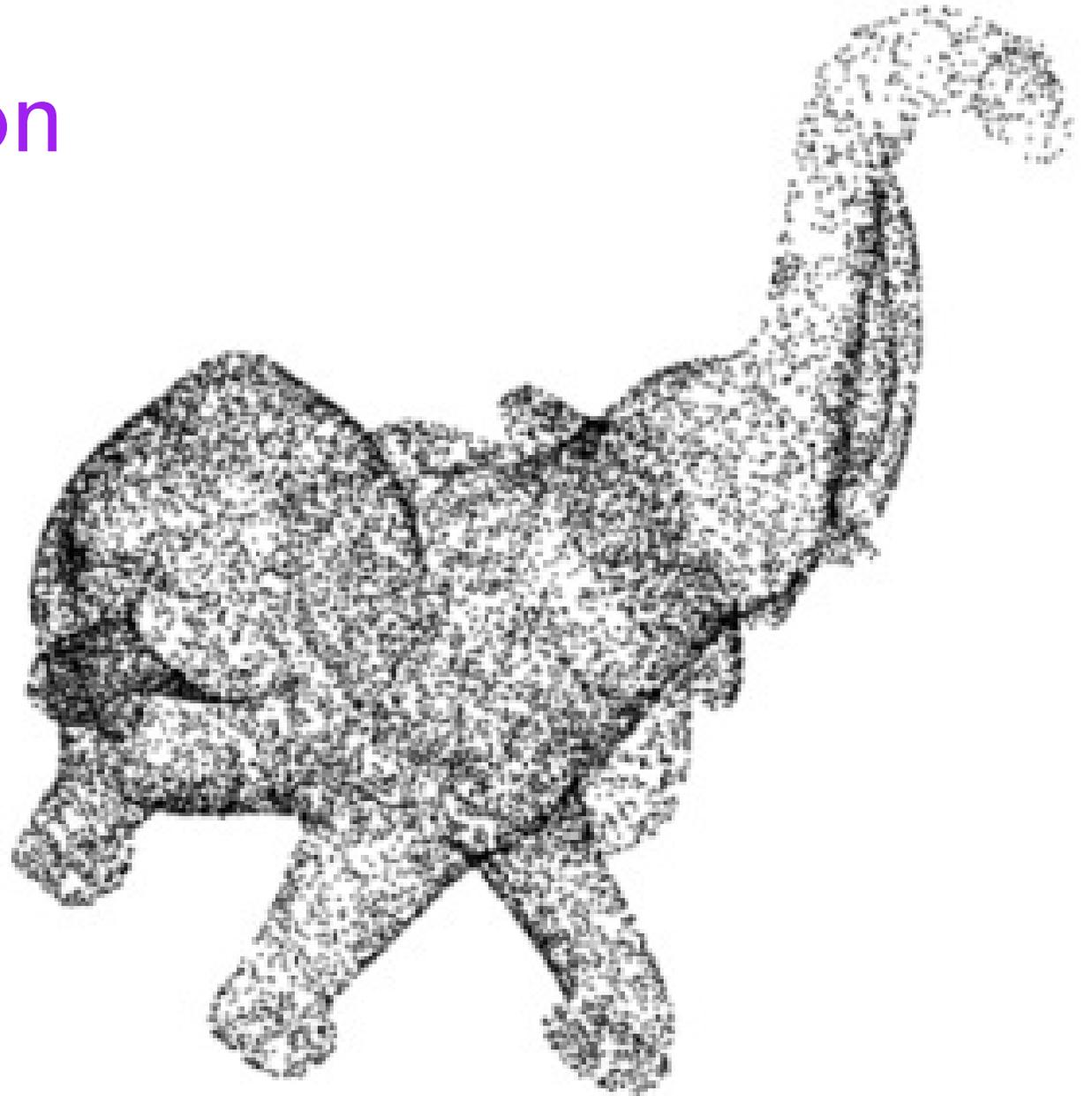
à la forme



# Delaunay, ça sert à quoi ?

## Reconstruction

Des points

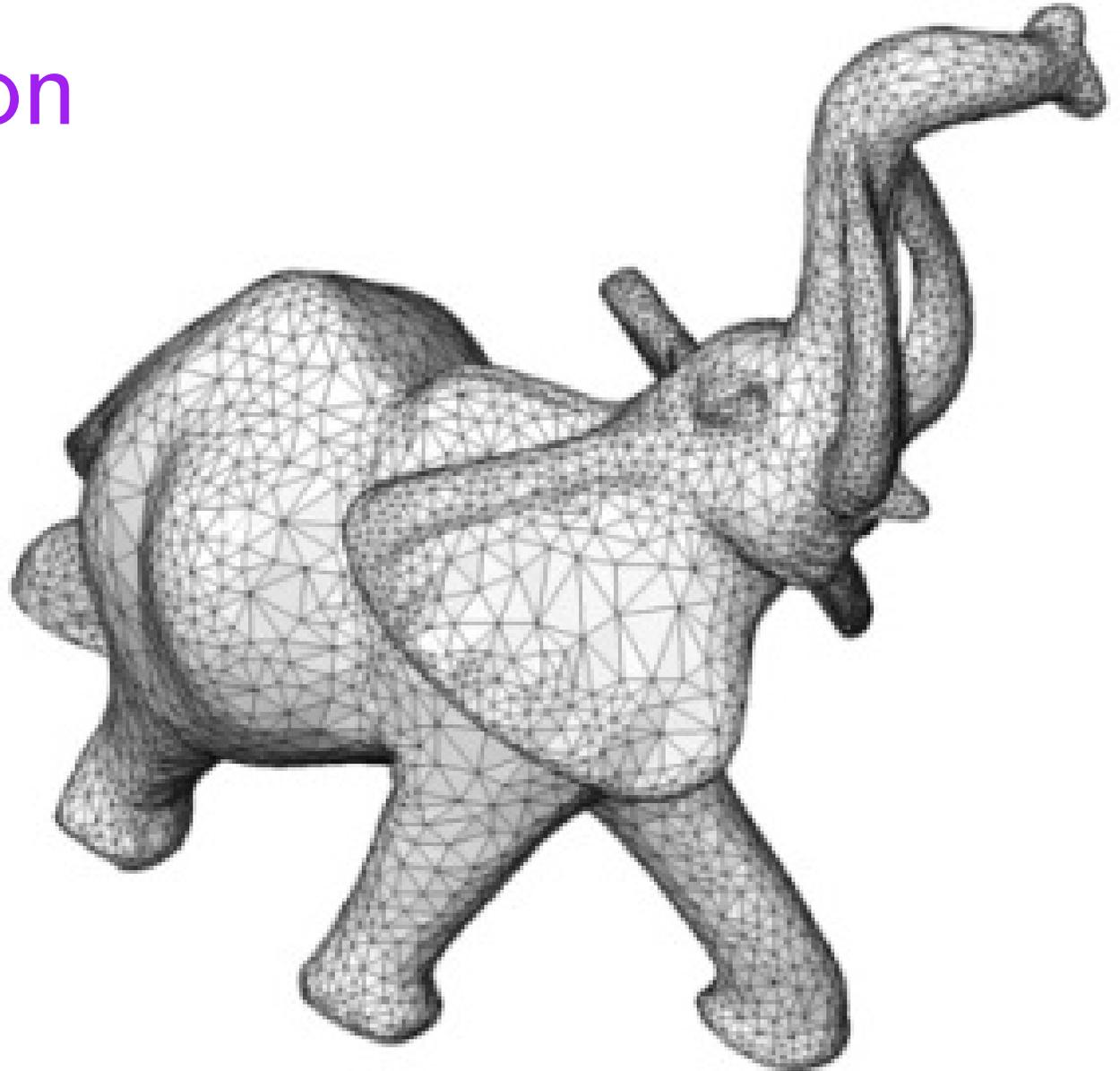


# Delaunay, ça sert à quoi ?

## Reconstruction

Des points

à la forme



# Delaunay, ça sert à quoi ?

## Reconstruction

Des points

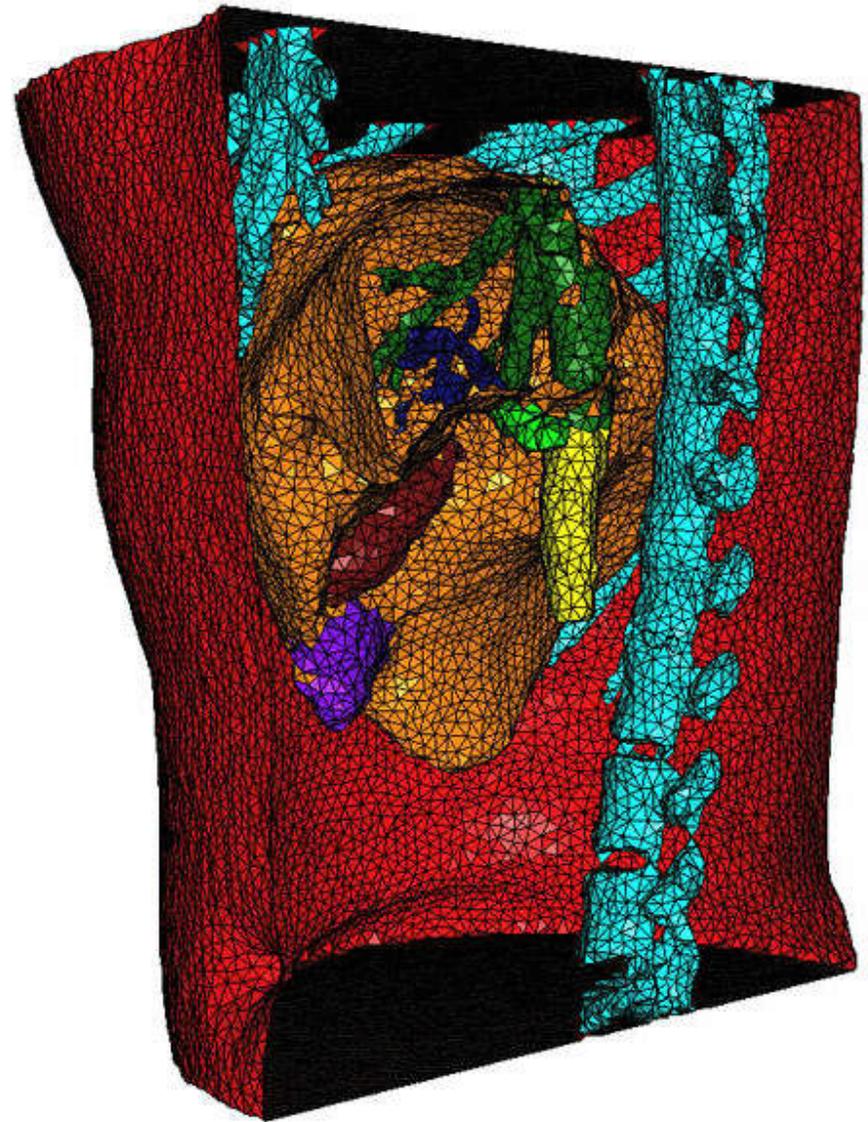


# Delaunay, ça sert à quoi ?

## Reconstruction

Des points

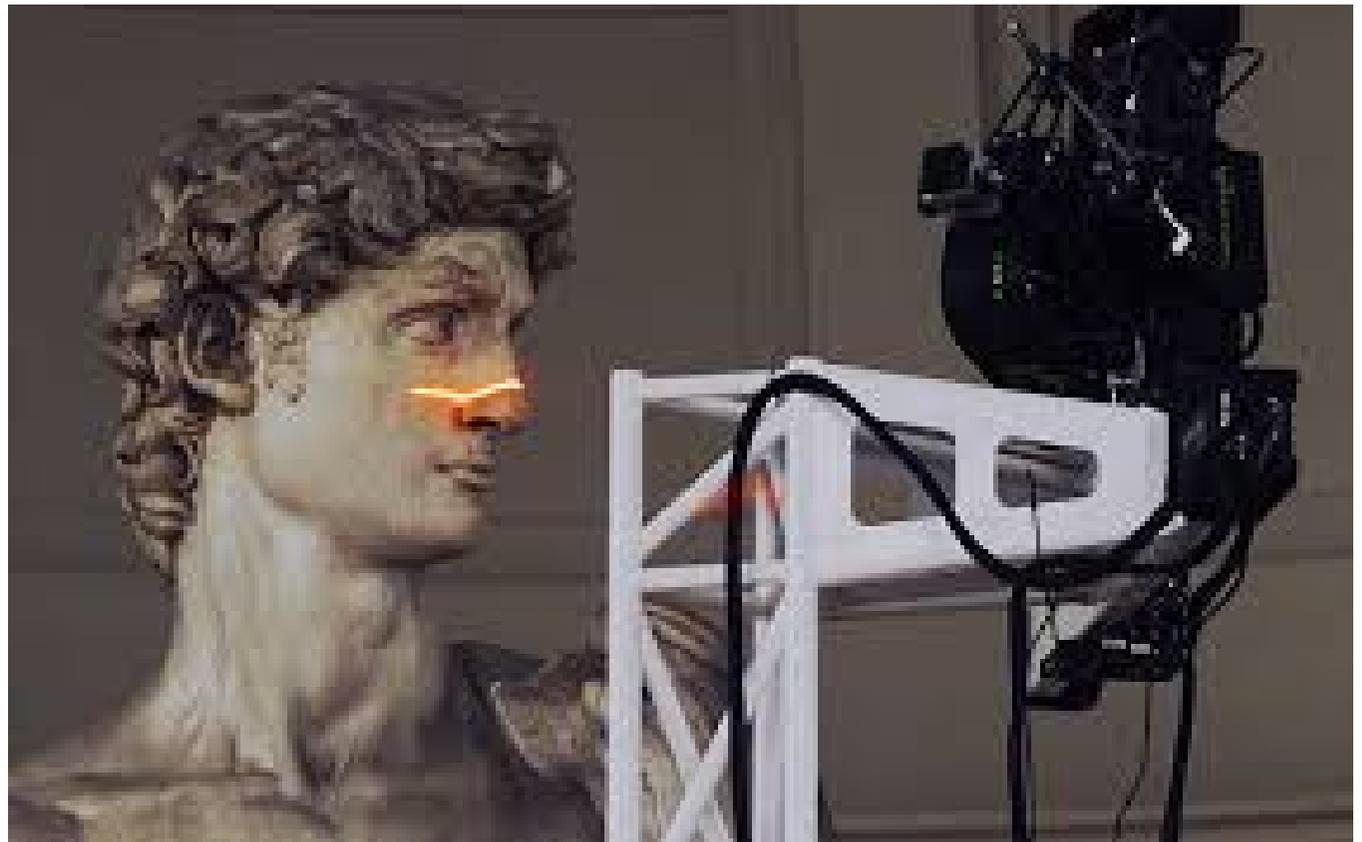
à la forme



# Delaunay, ça sert à quoi ?

## Reconstruction

Des points

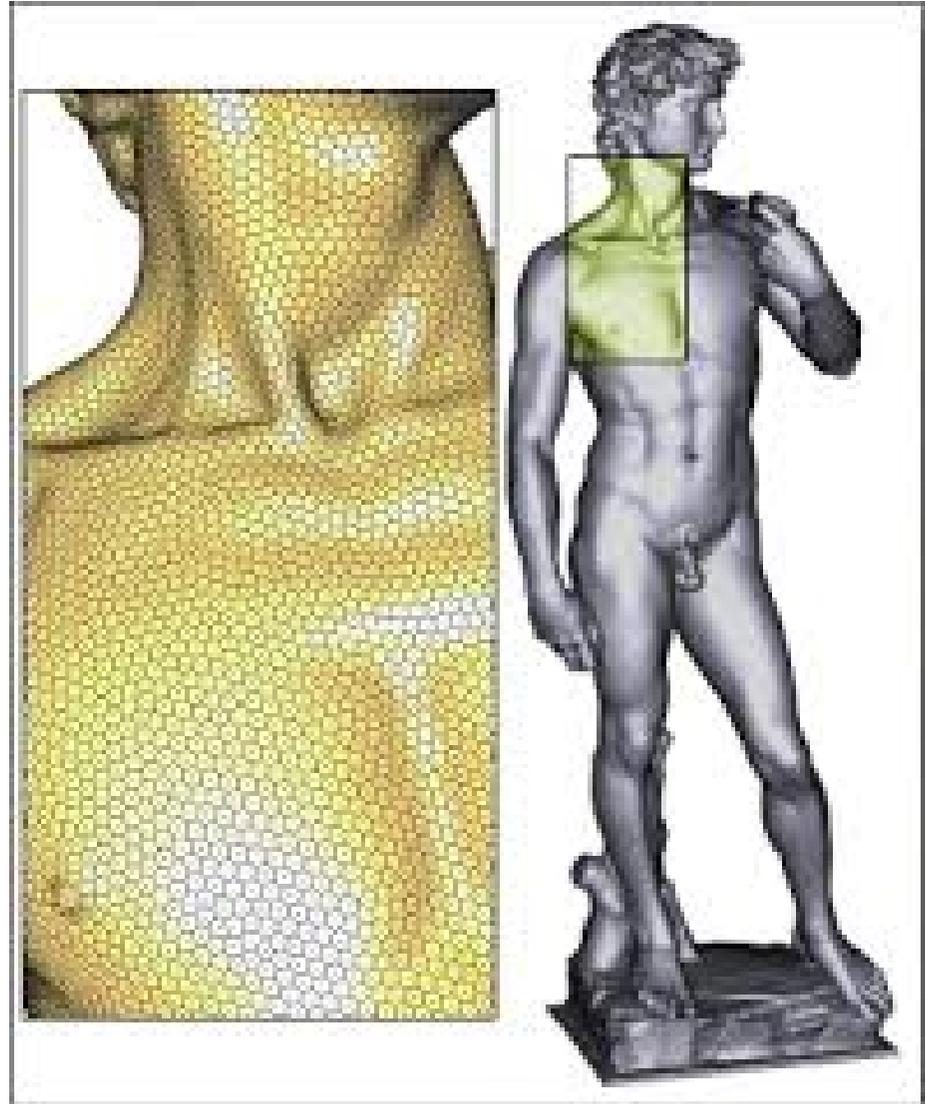


# Delaunay, ça sert à quoi ?

## Reconstruction

Des points

à la forme



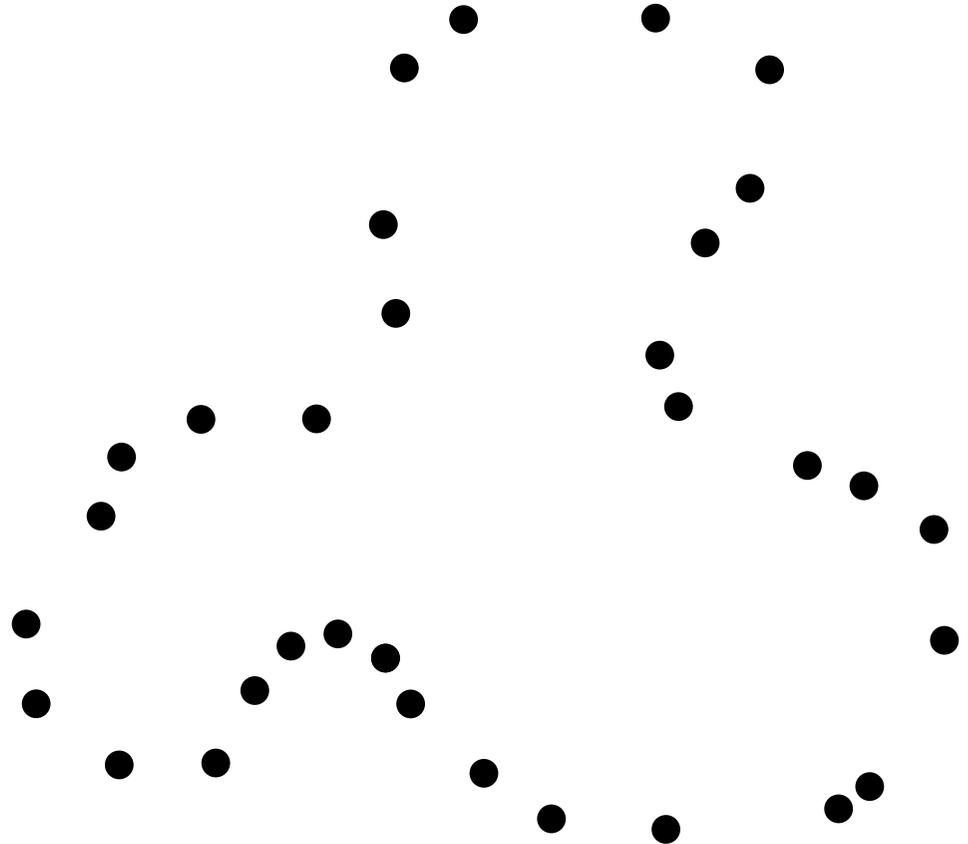
# Delaunay, ça sert à quoi ?

## Reconstruction

Des points

à la forme

Un algorithme (2D)



# Delaunay, ça sert à quoi ?

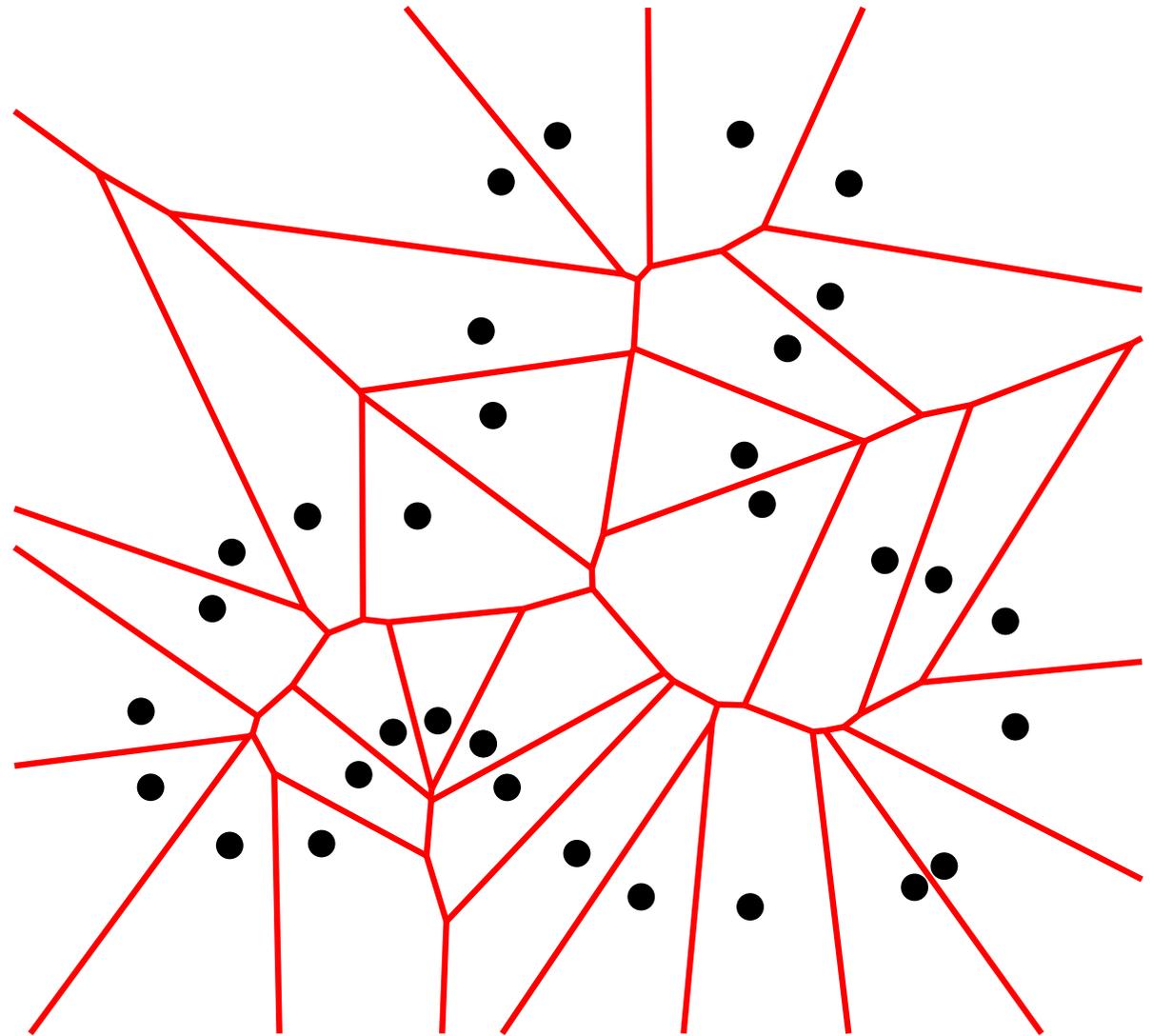
## Reconstruction

Des points

à la forme

Un algorithme (2D)

Voronoi



# Delaunay, ça sert à quoi ?

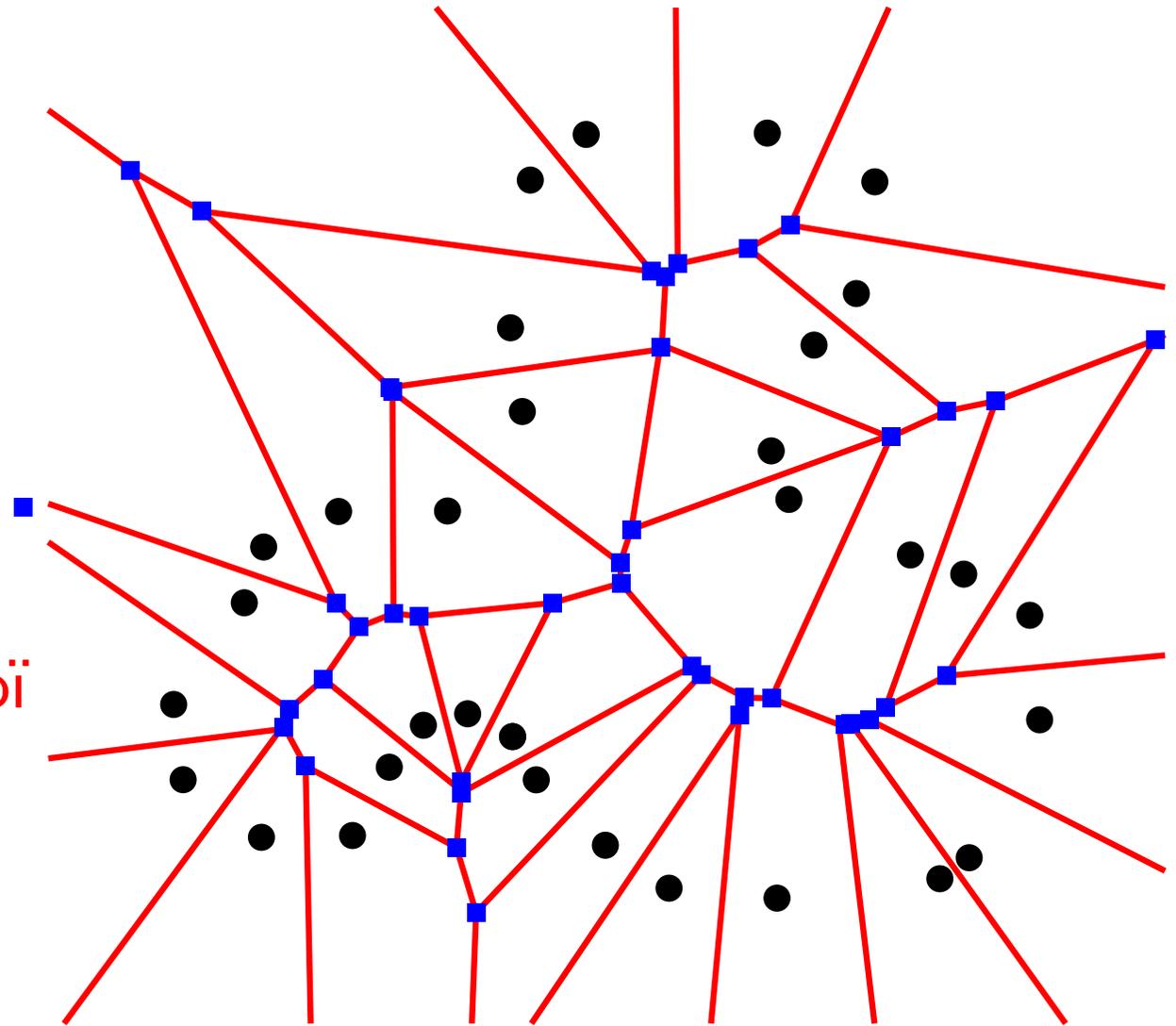
## Reconstruction

Des points

à la forme

Un algorithme (2D)

Sommets de Voronoï



# Delaunay, ça sert à quoi ?

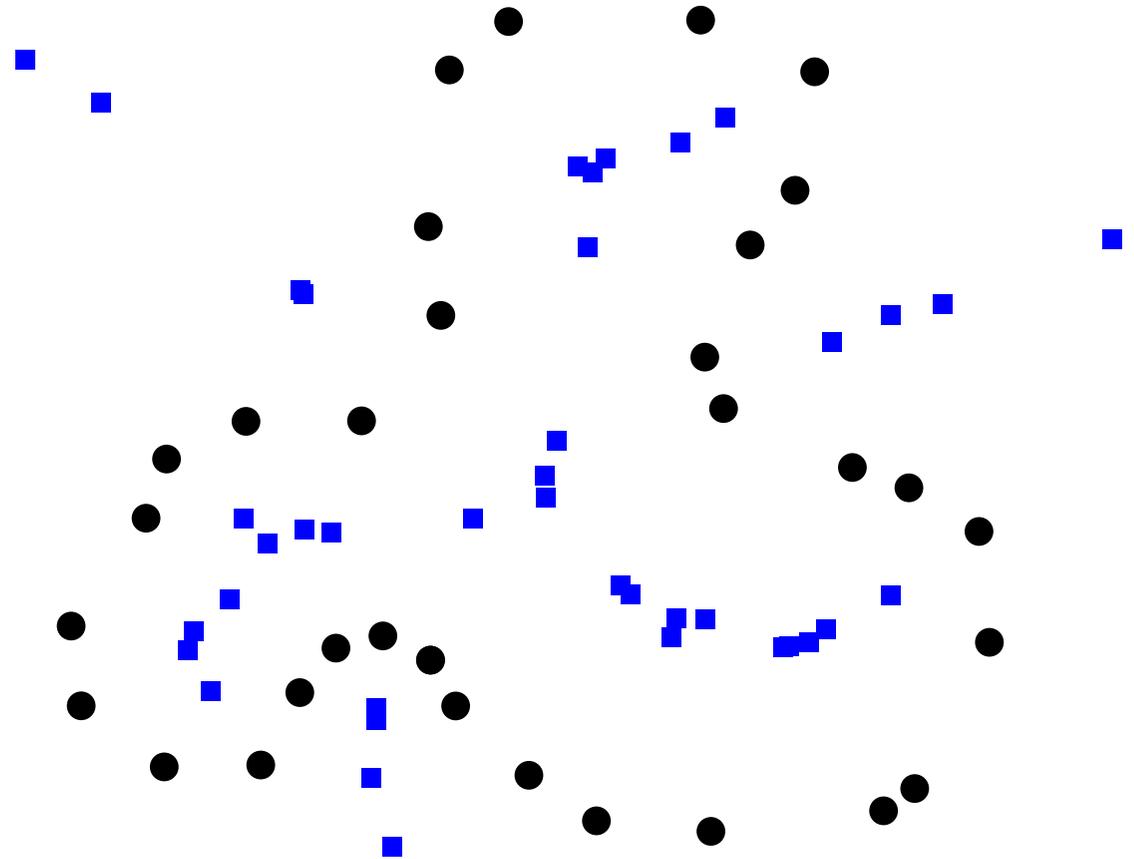
## Reconstruction

Des points

à la forme

Un algorithme (2D) 

Sommets de Voronoï 



# Delaunay, ça sert à quoi ?

## Reconstruction

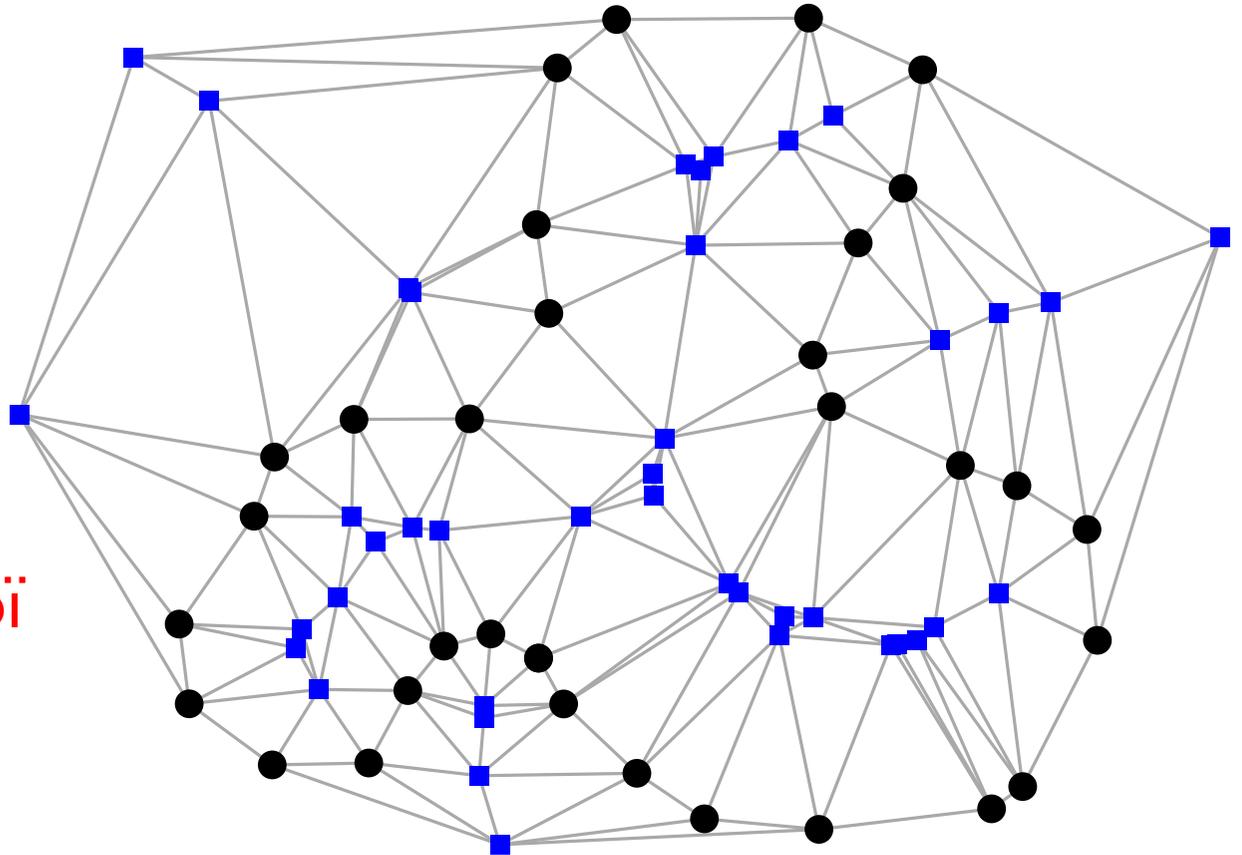
Des points

à la forme

Un algorithme (2D)

Sommets de Voronoï

Delaunay



# Delaunay, ça sert à quoi ?

## Reconstruction

Des points

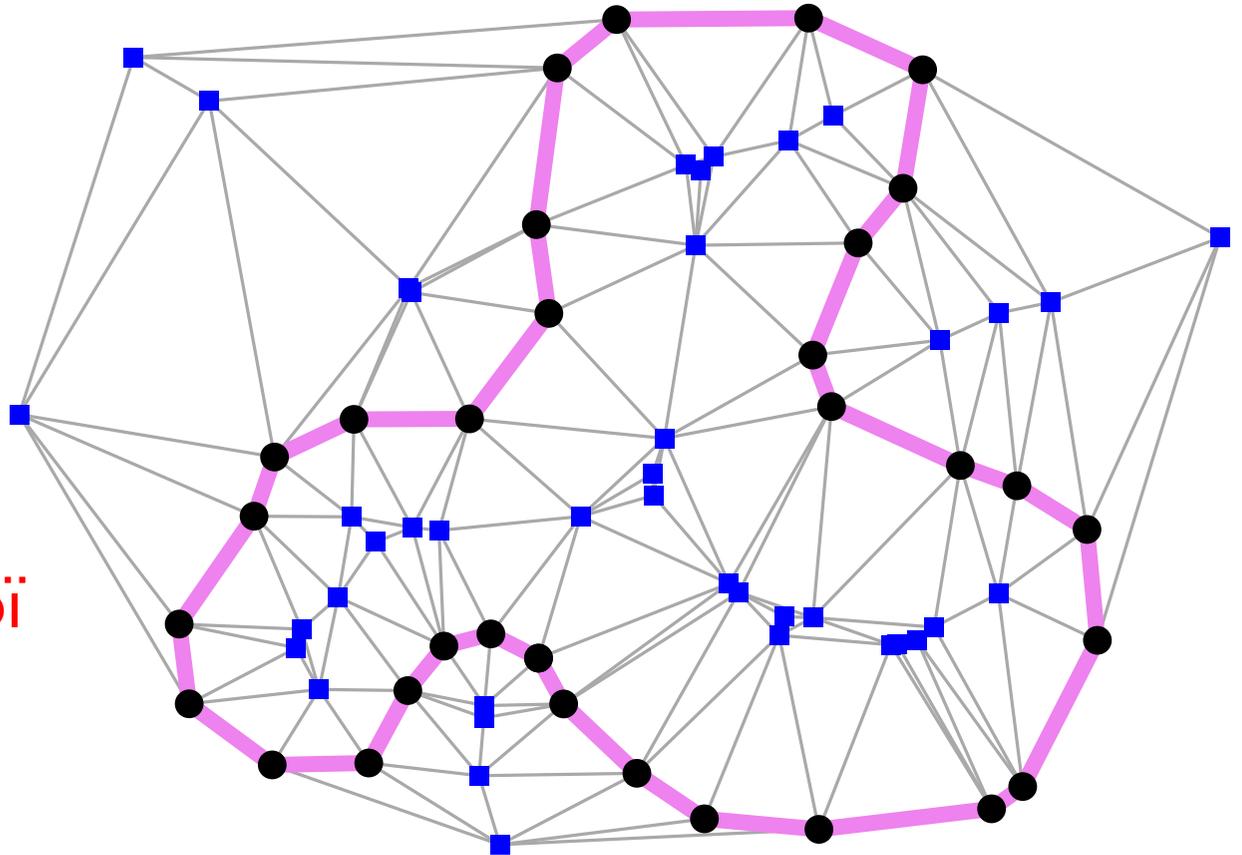
à la forme

Un algorithme (2D)

Sommets de Voronoï

Delaunay

Sélection des arêtes noir-noir



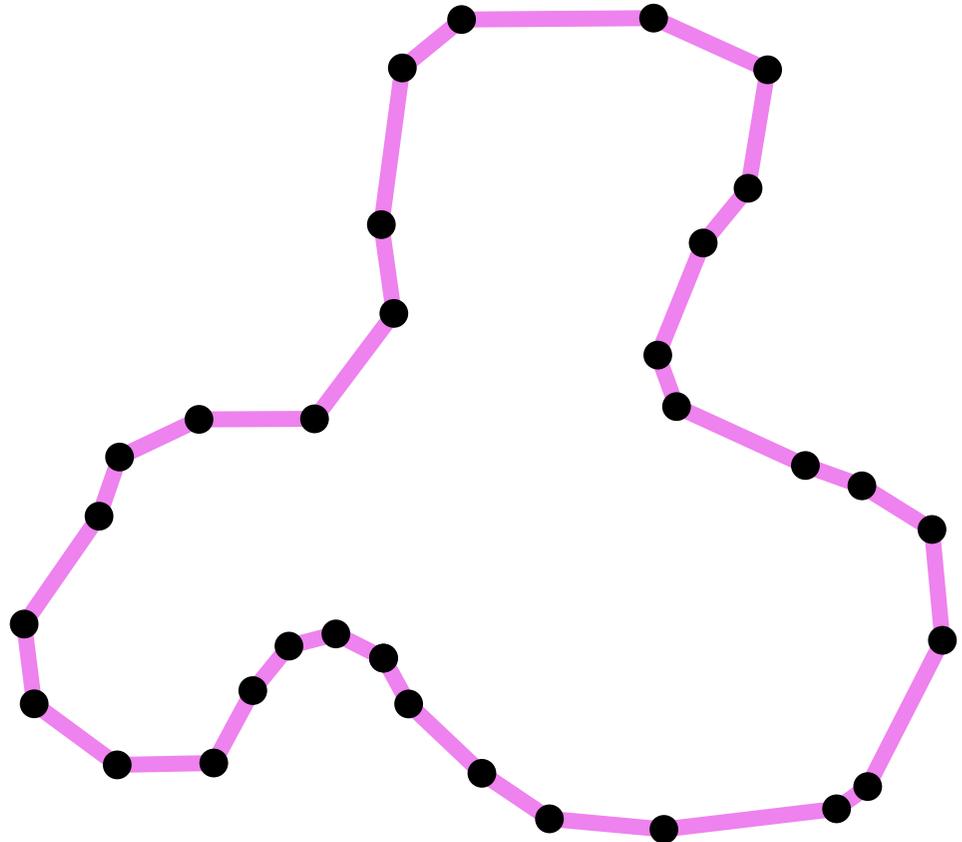
# Delaunay, ça sert à quoi ?

## Reconstruction

Des points

à la forme

Un algorithme (2D)



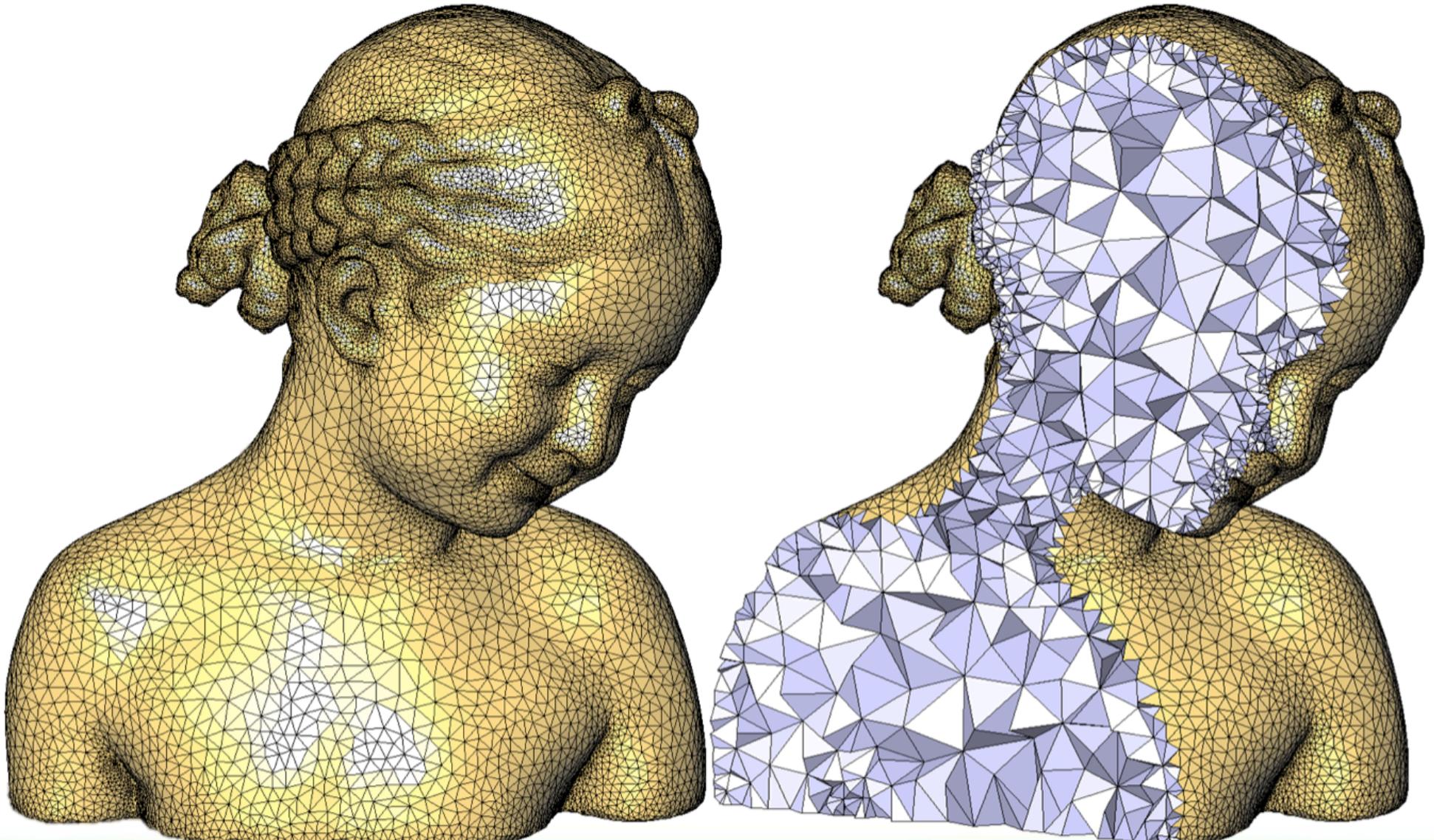
Sélection des arêtes noir-noir

Delaunay, ça sert à quoi ?

Maillage

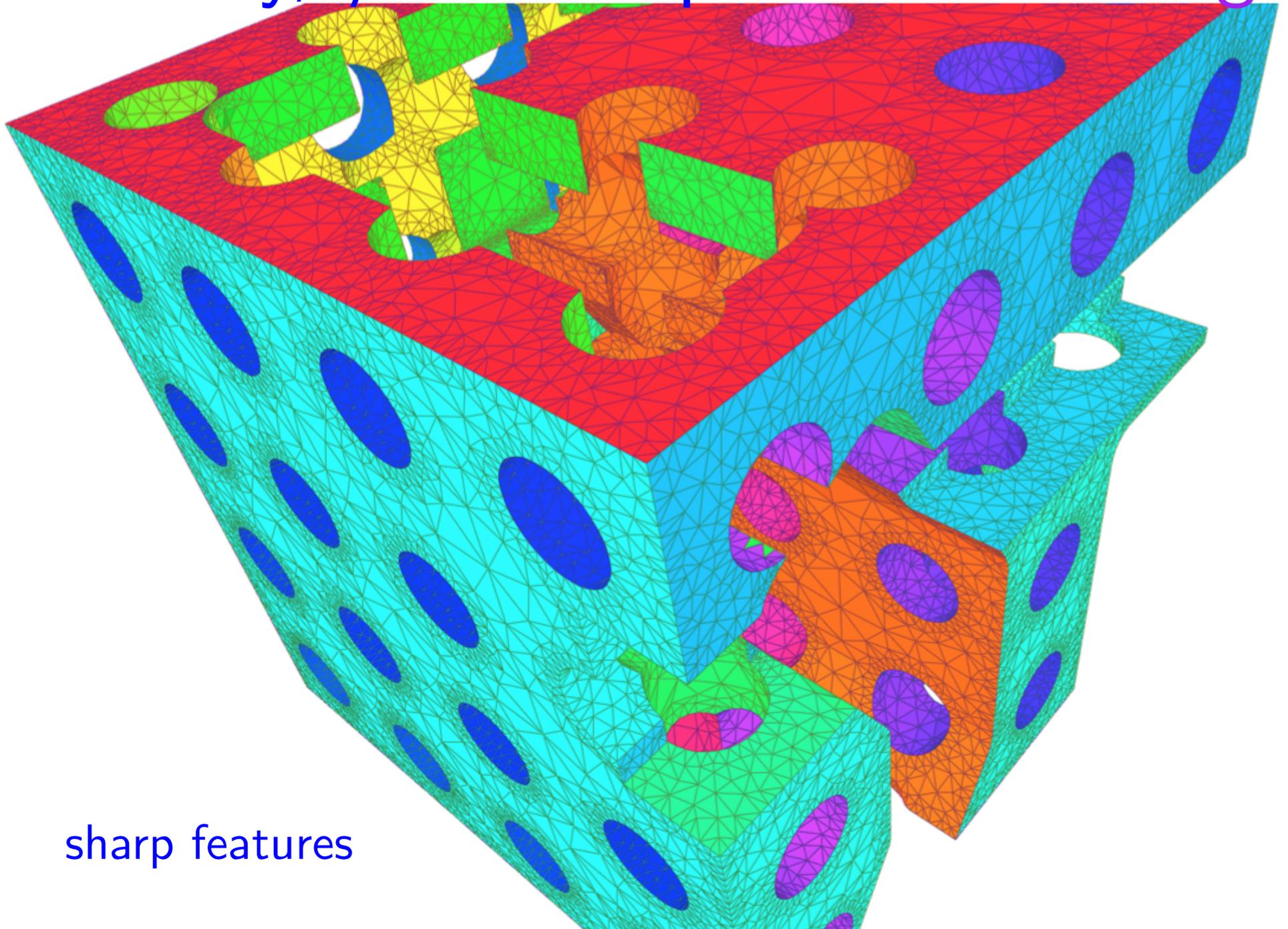
Delaunay, ça sert à quoi ?

Maillage



Delaunay, ça sert à quoi ?

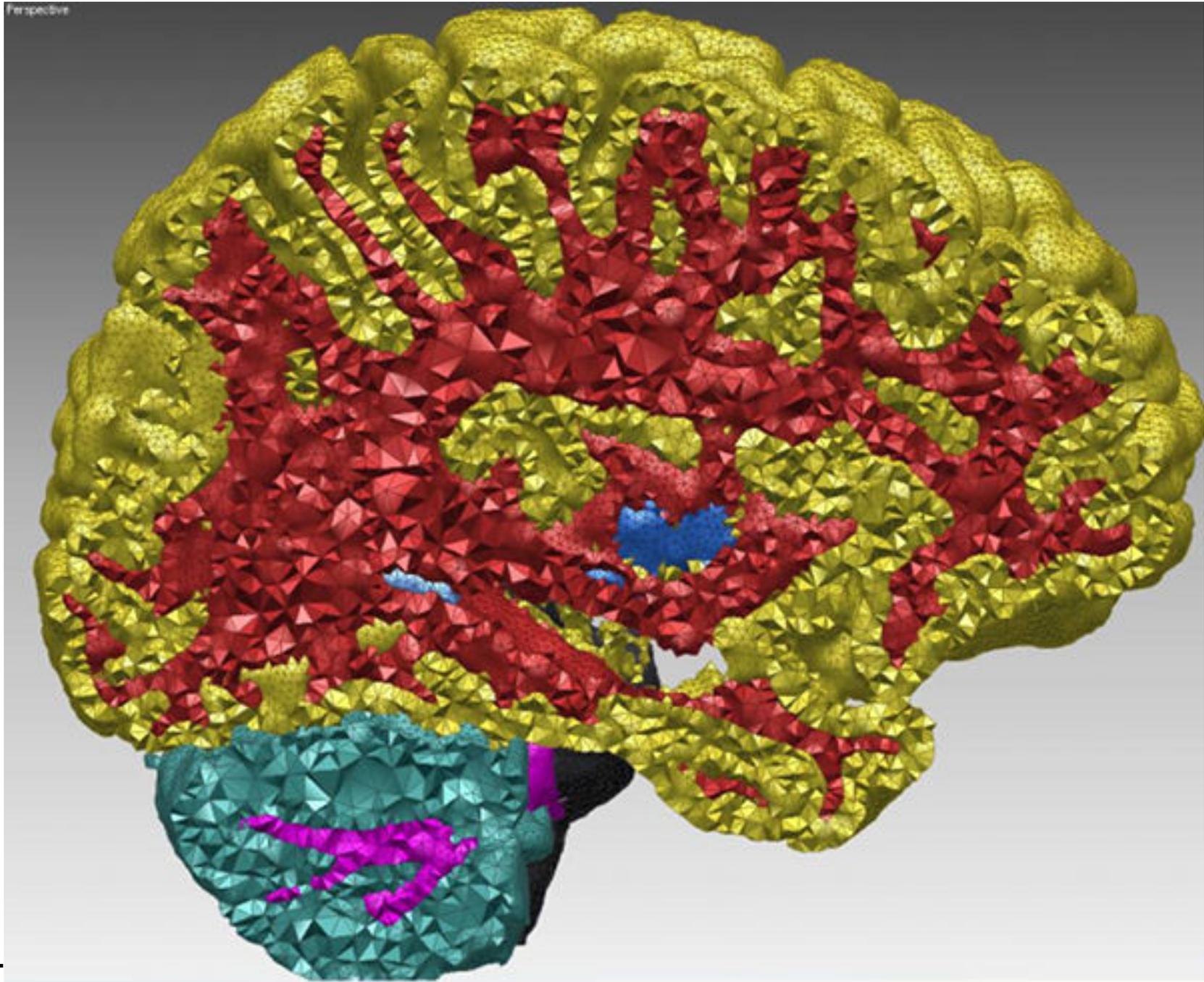
Maillage



sharp features

Delaunay, ça sert à quoi ?

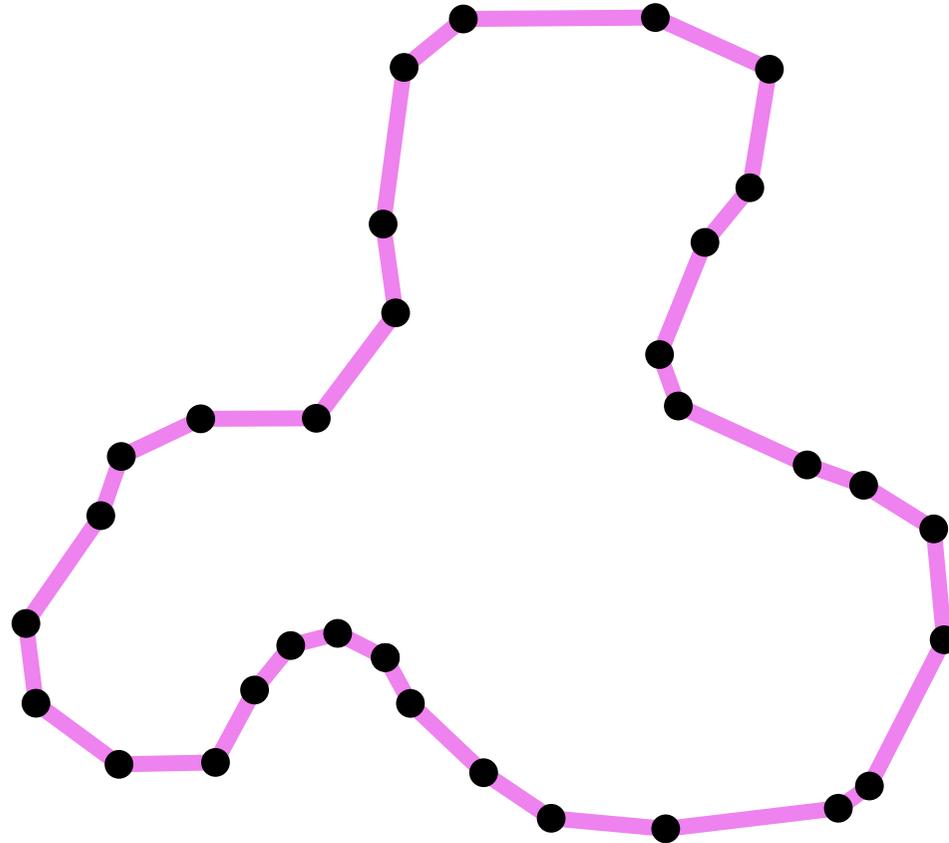
Maillage



# Delaunay, ça sert à quoi ?

# Maillage

Donnée: une forme

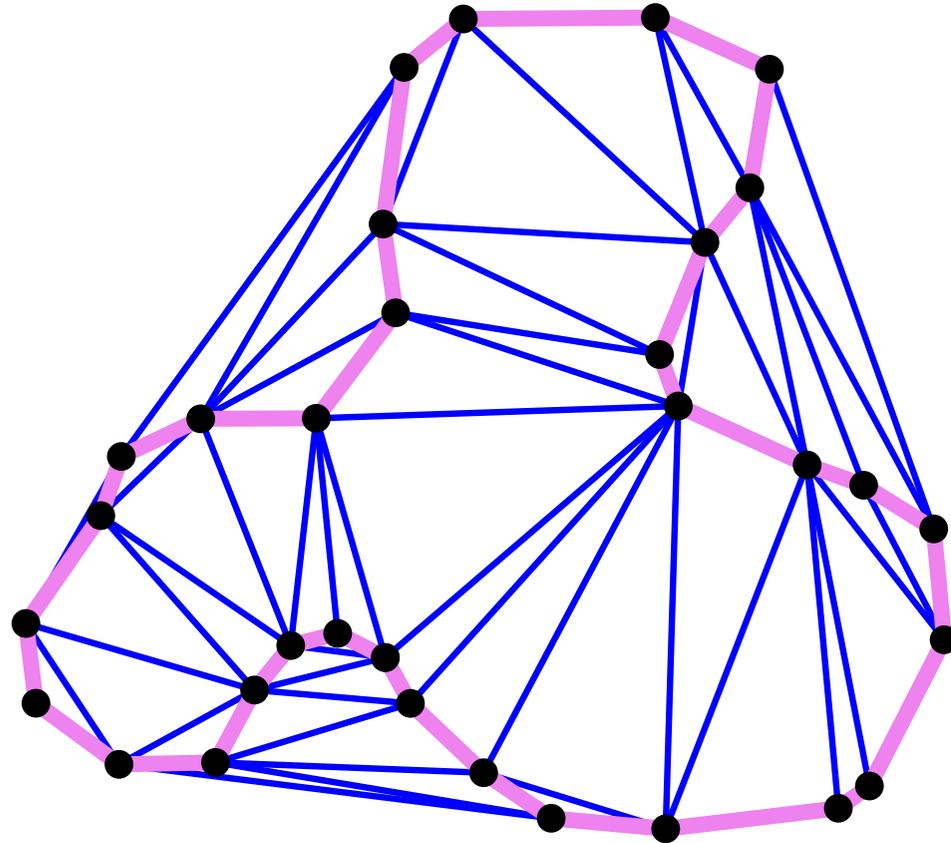


# Delaunay, ça sert à quoi ?

# Maillage

Donnée: une forme

Delaunay



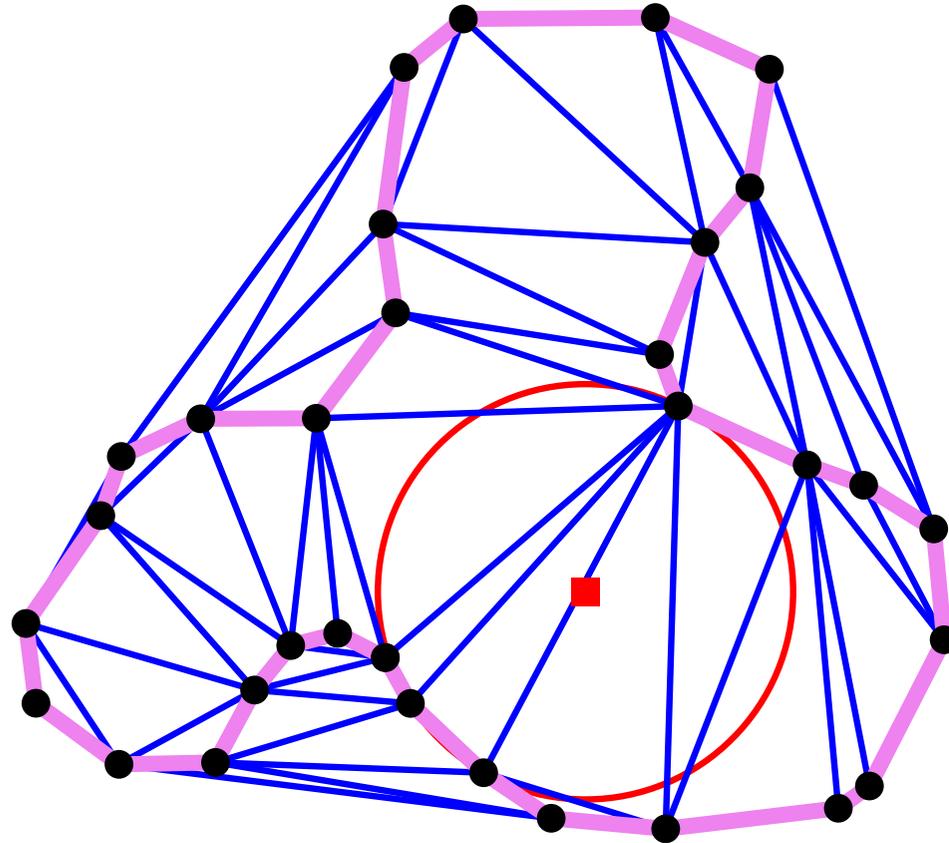
# Delaunay, ça sert à quoi ?

# Maillage

Donnée: une forme

Delaunay

Ajouter des centres  
de cercle pour  
casser les mauvais  
triangles



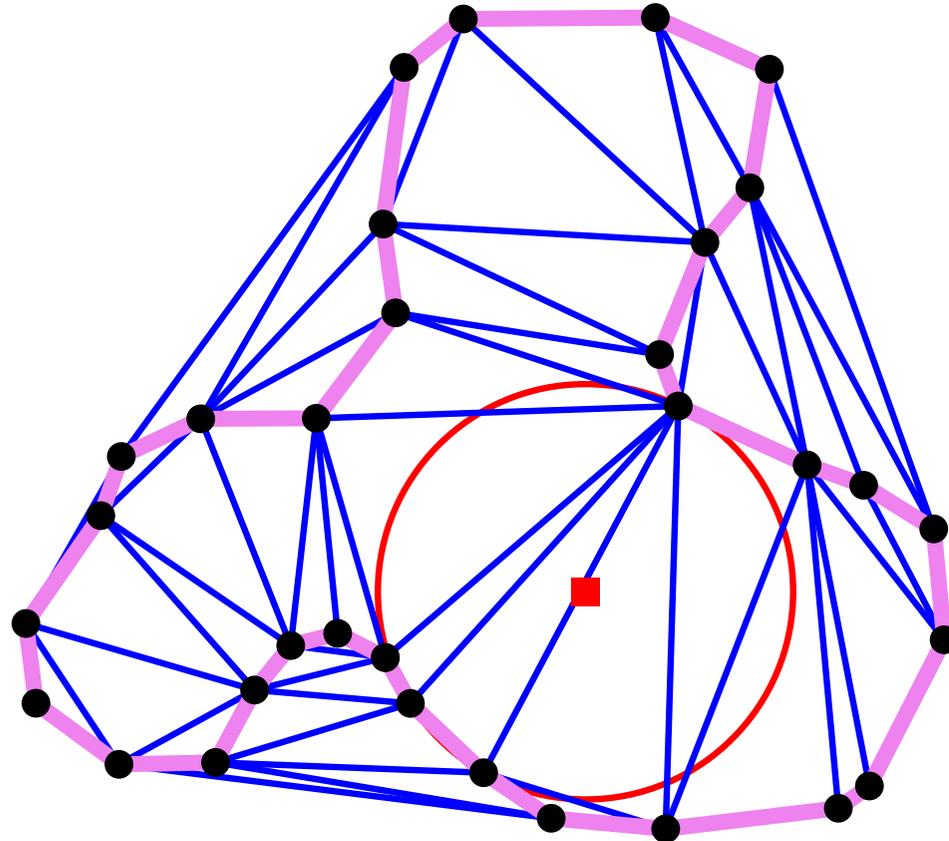
# Delaunay, ça sert à quoi ?

# Maillage

Donnée: une forme

Delaunay

Ajouter des centres de cercle pour casser les mauvais triangles



Mauvais triangle = trop grand, trop aplati

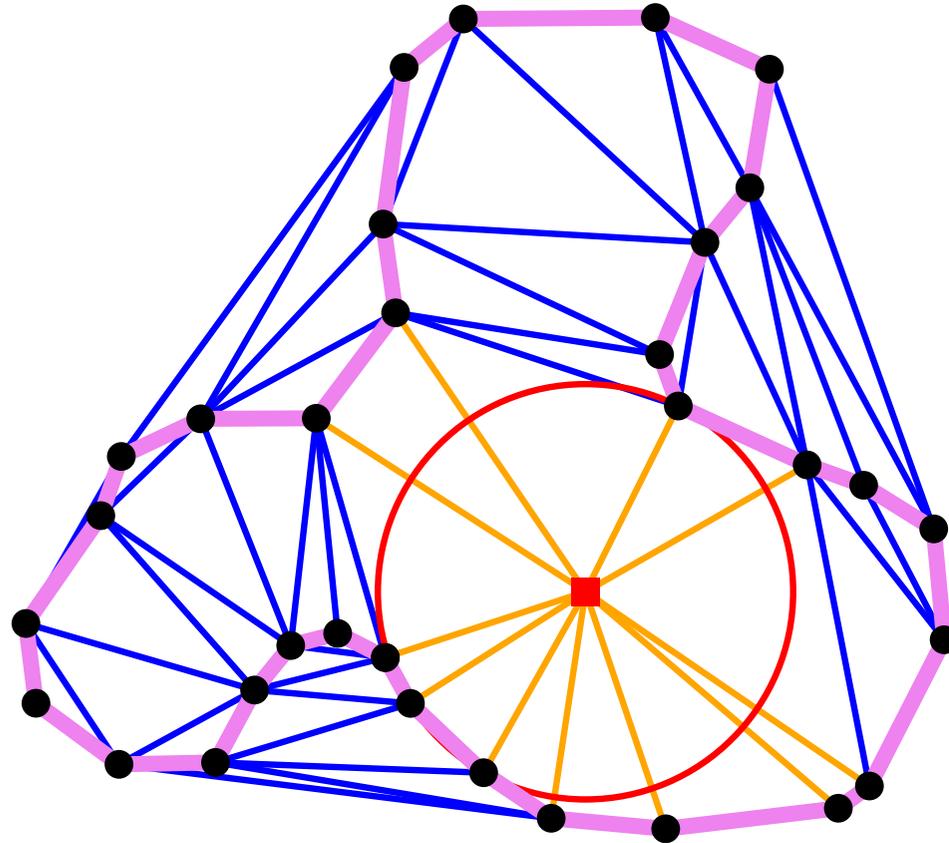
# Delaunay, ça sert à quoi ?

# Maillage

Donnée: une forme

Delaunay

Ajouter des centres  
de cercle pour  
casser les mauvais  
triangles



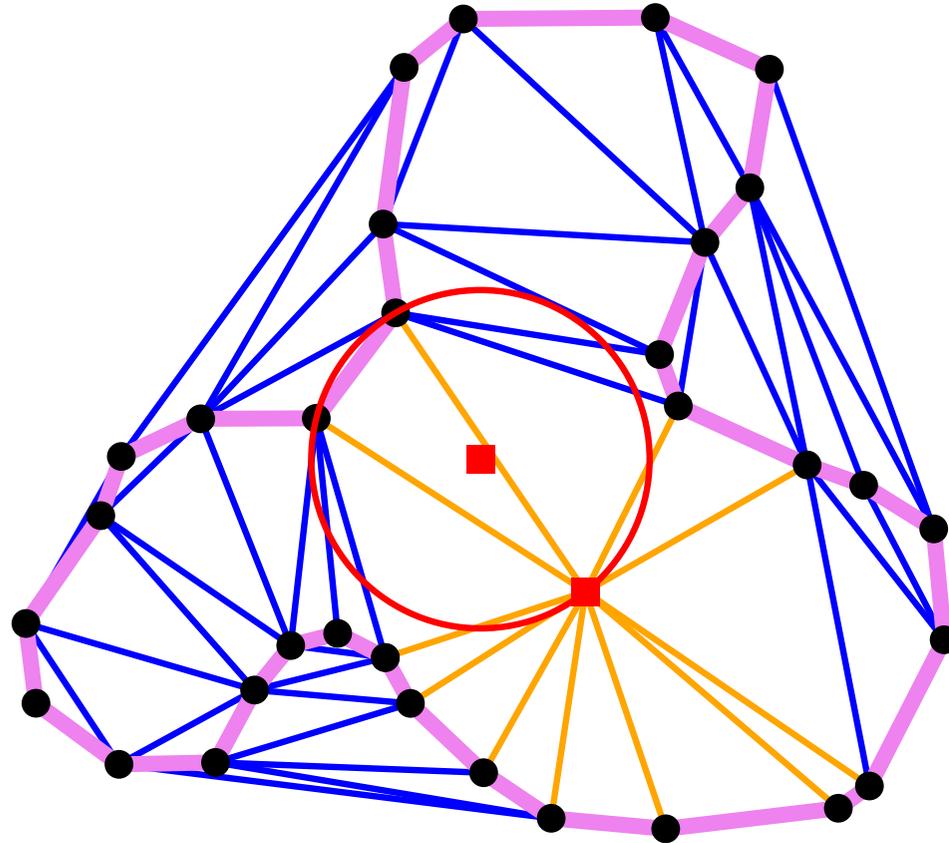
# Delaunay, ça sert à quoi ?

# Maillage

Donnée: une forme

Delaunay

Ajouter des centres  
de cercle pour  
casser les mauvais  
triangles



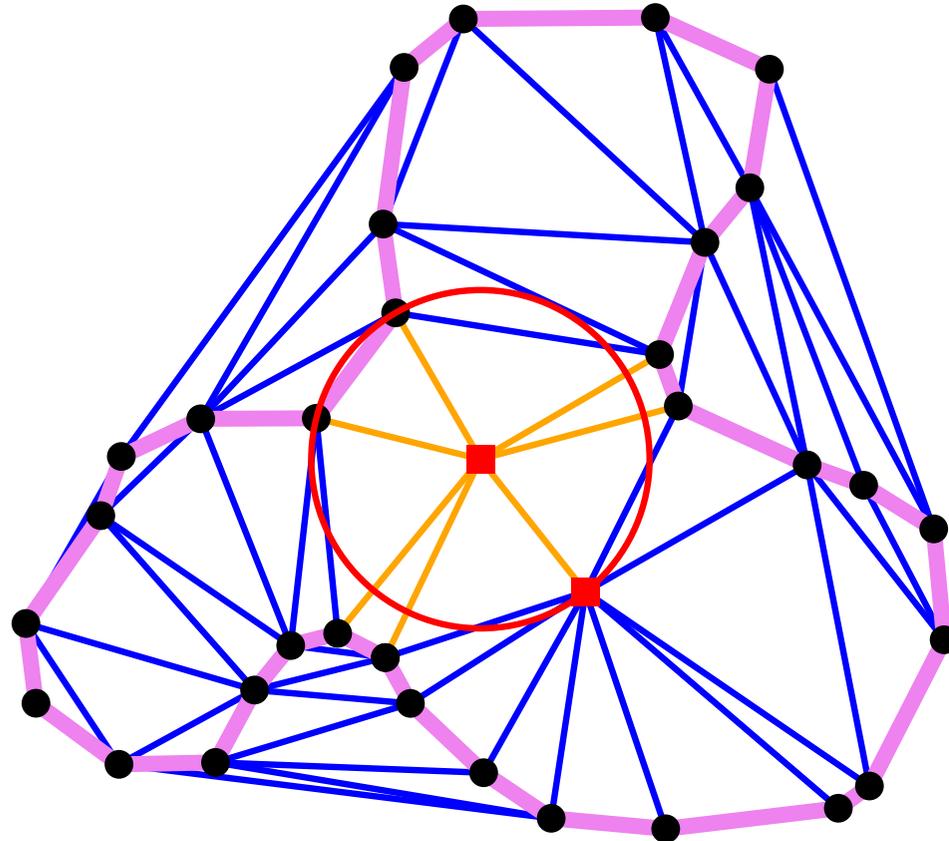
# Delaunay, ça sert à quoi ?

# Maillage

Donnée: une forme

Delaunay

Ajouter des centres  
de cercle pour  
casser les mauvais  
triangles



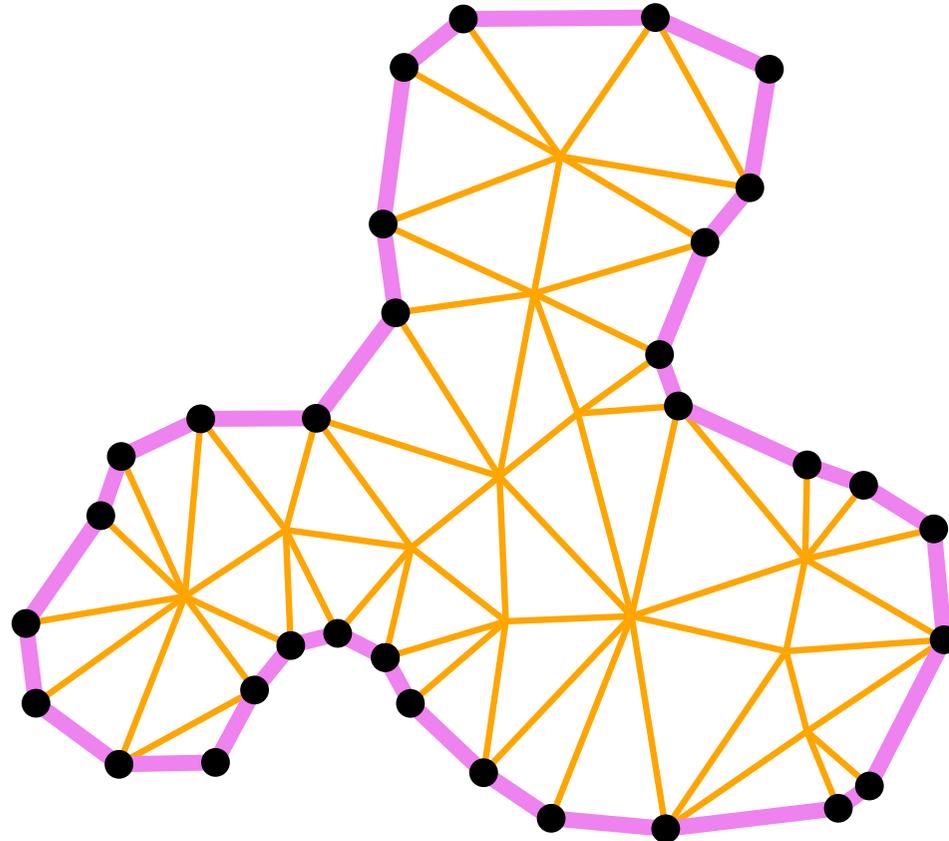
# Delaunay, ça sert à quoi ?

# Maillage

Donnée: une forme

Delaunay

Ajouter des centres  
de cercle pour  
casser les mauvais  
triangles



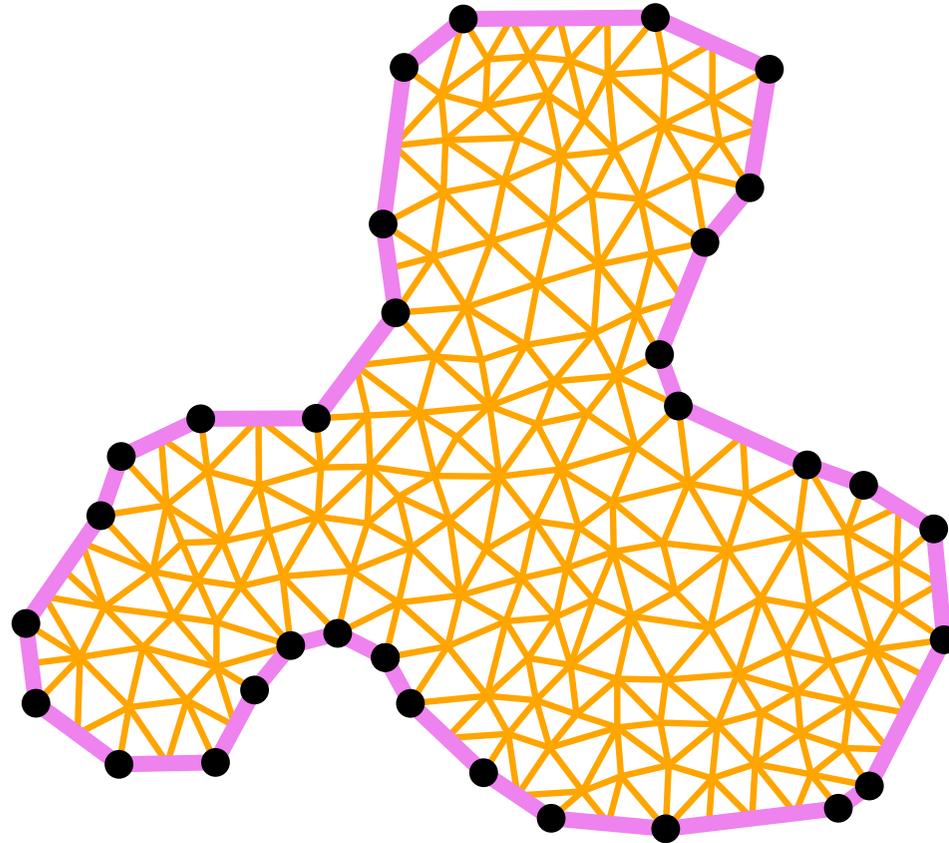
# Delaunay, ça sert à quoi ?

# Maillage

Donnée: une forme

Delaunay

Ajouter des centres  
de cercle pour  
casser les mauvais  
triangles



# Un historique du domaine 1975-1985

Des algorithmes compliqués

Complexité dans le cas le pire

Complexités asymptotiques

Modèle Real RAM

Bornes inférieure

Hypothèses de position générales

# Un historique du domaine

## 1975-1985

Des algorithmes compliqués

programmable?

Complexité dans le cas le pire

réaliste?

Complexités asymptotiques

$n$  "assez" grand?

Modèle Real RAM

réaliste?

Bornes inférieure

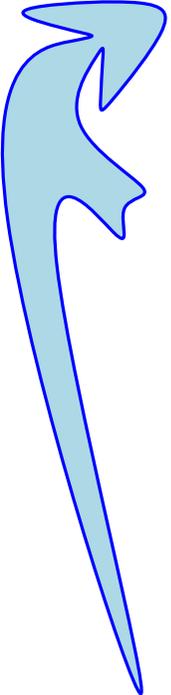
exemples pathologiques?

Hypothèses de position générales

ça existe en vrai?

# Un historique du domaine

## 1975-1985



Des algorithmes compliqués

programmable?

Complexité dans le cas le pire

réaliste?

Complexités asymptotiques

$n$  "assez" grand?

Modèle Real RAM

réaliste?

Bornes inférieure

exemples pathologiques?

Hypothèses de position générales

ça existe en vrai?

# Un historique du domaine

## 1975-1985

pas utilisé en pratique

Des algorithmes compliqués

programmable?

Complexité dans le cas le pire

réaliste?

Complexités asymptotiques

$n$  "assez" grand?

Modèle Real RAM

réaliste?

Bornes inférieure

exemples pathologiques?

Hypothèses de position générales

ça existe en vrai?

# Un historique du domaine

1975-1985      1985-2000

Des algorithmes compliqués

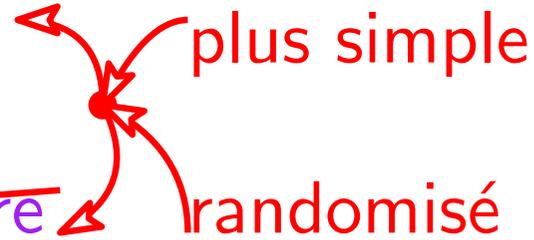
Complexité dans le cas le pire

Complexités asymptotiques

Modèle Real RAM

Bornes inférieure

Hypothèses de position générales



# Un historique du domaine

1975-1985      1985-2000

Des algorithmes compliqués

plus simple

Complexité dans le cas le pire

randomisé

Complexités asymptotiques

études plus pratiques

Modèle Real RAM

Bornes inférieure

Hypothèses de position générales

# Un historique du domaine

1975-1985      1985-2000

Des algorithmes compliqués

plus simple

Complexité dans le cas le pire

randomisé

Complexités asymptotiques

études plus pratiques

Modèle Real RAM

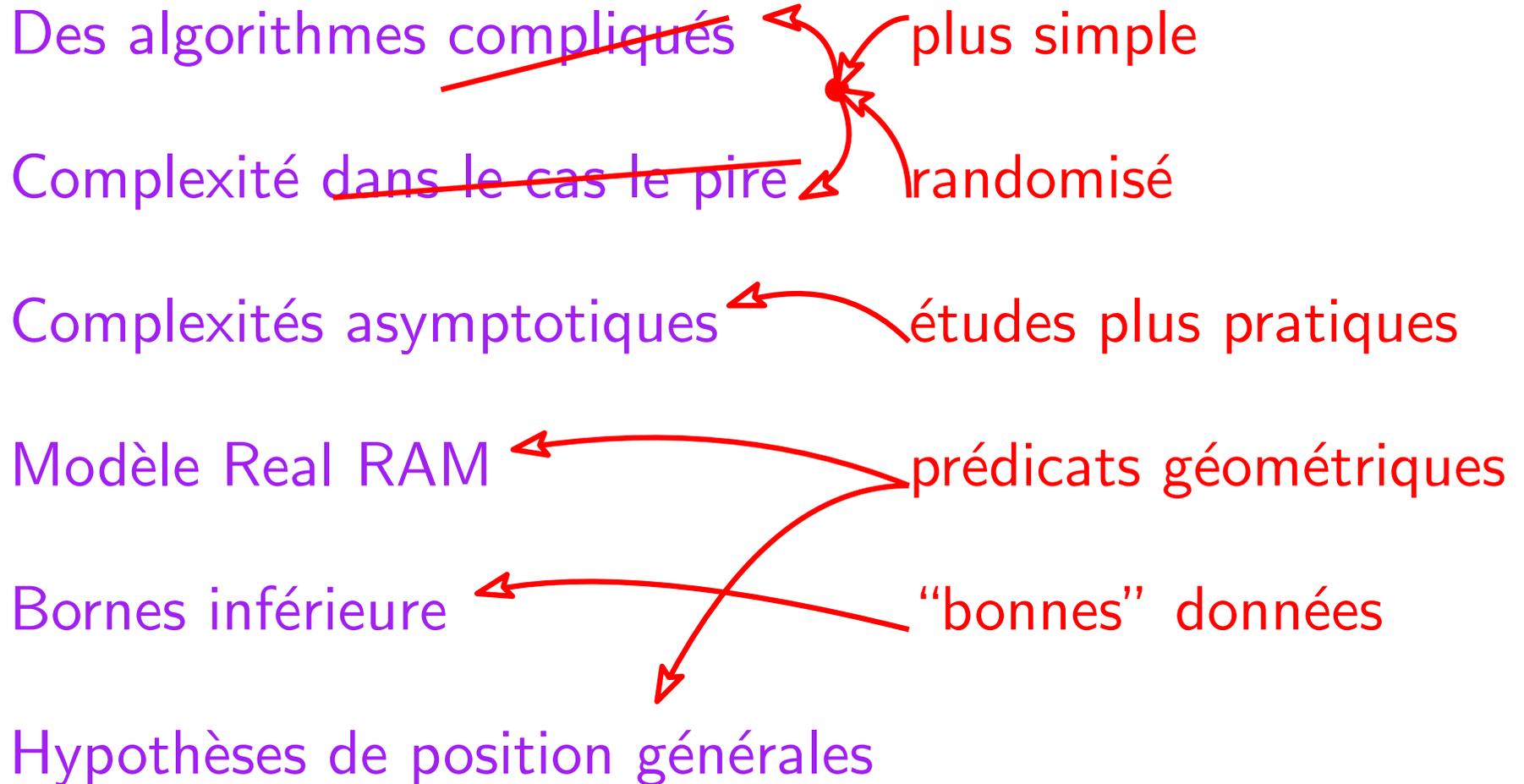
prédicats géométriques

Bornes inférieure

Hypothèses de position générales

# Un historique du domaine

1975-1985      1985-2000

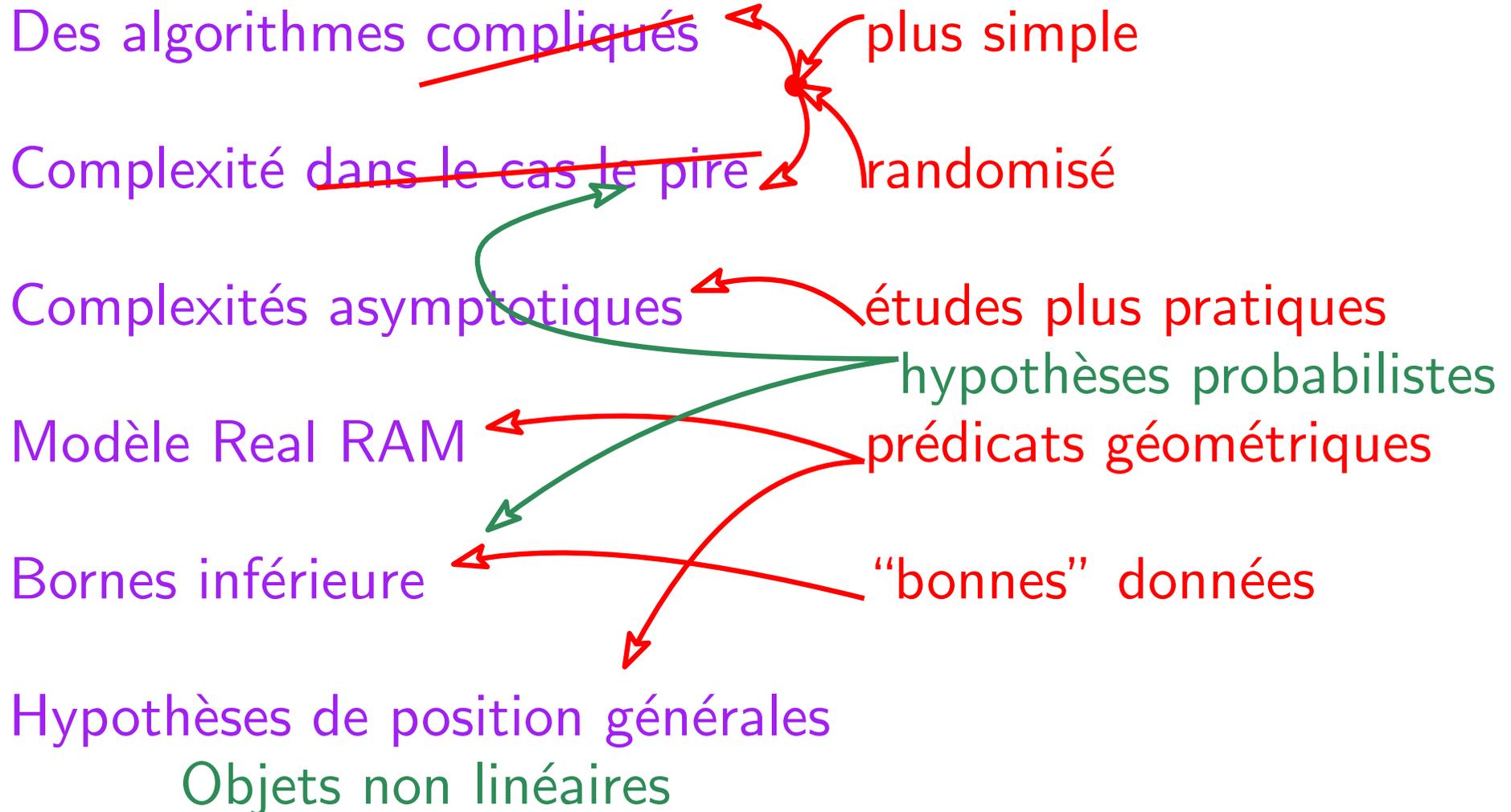


# Un historique du domaine

1975-1985

1985-2000

2000-2020



# Triangulation de Delaunay

# Triangulation de Delaunay quelques résultats

Calcul efficace  $O(n \log n)$  incremental randomisé

des millions de points par seconde

Maintenance dynamique (insertion et suppression)

Algo non randomisé  $O(n \log n)$  (compliqué)

# Triangulation de Delaunay quelques résultats

3D

Calcul efficace

$O(\#\text{tetras})$

incremental randomisé

de points par seconde

Maintenance dynamique (insertion et suppression)

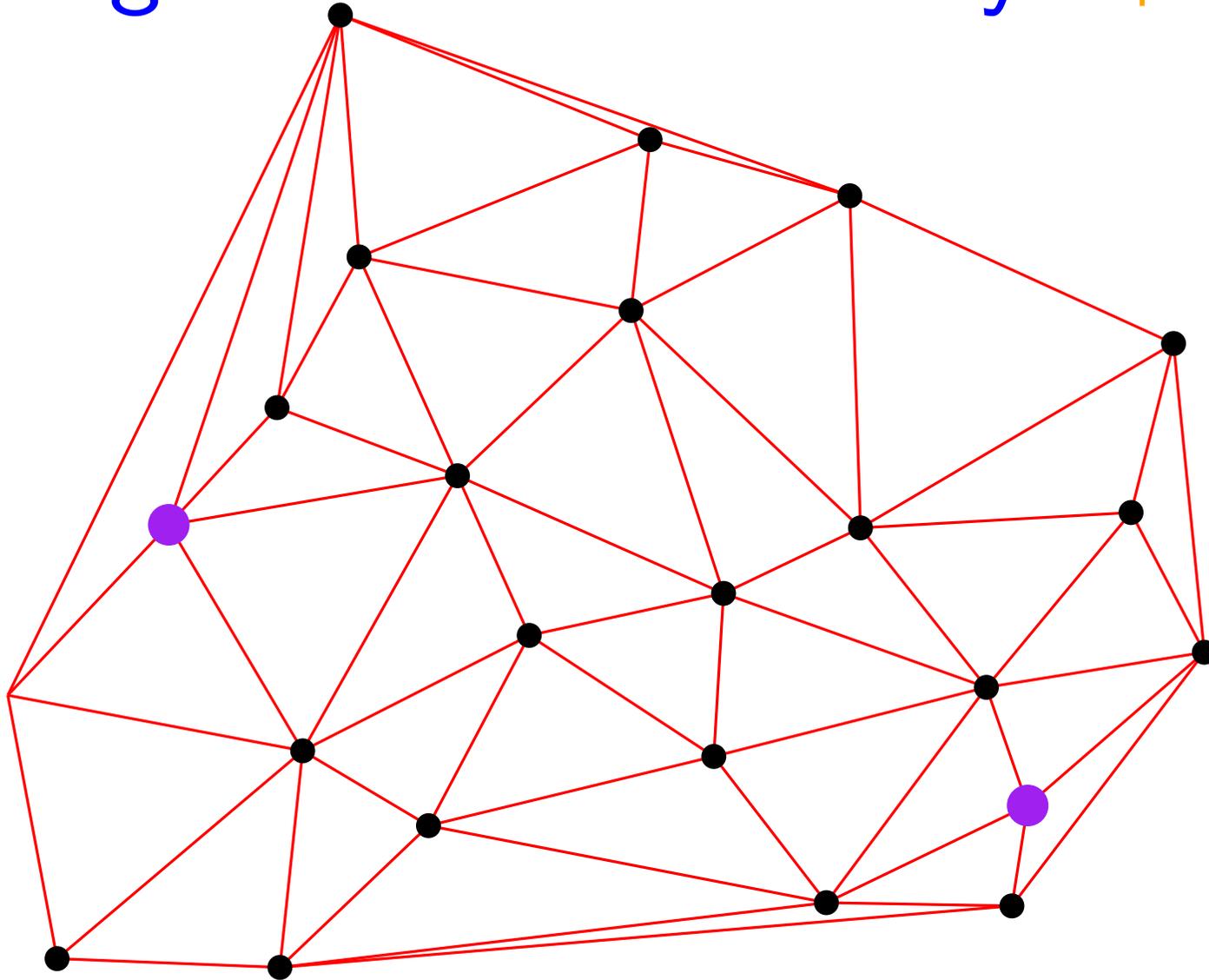
~~Algo non randomisé  $O(n \log n)$~~

(compliqué)

$$\#\text{tetras} = O(n)$$

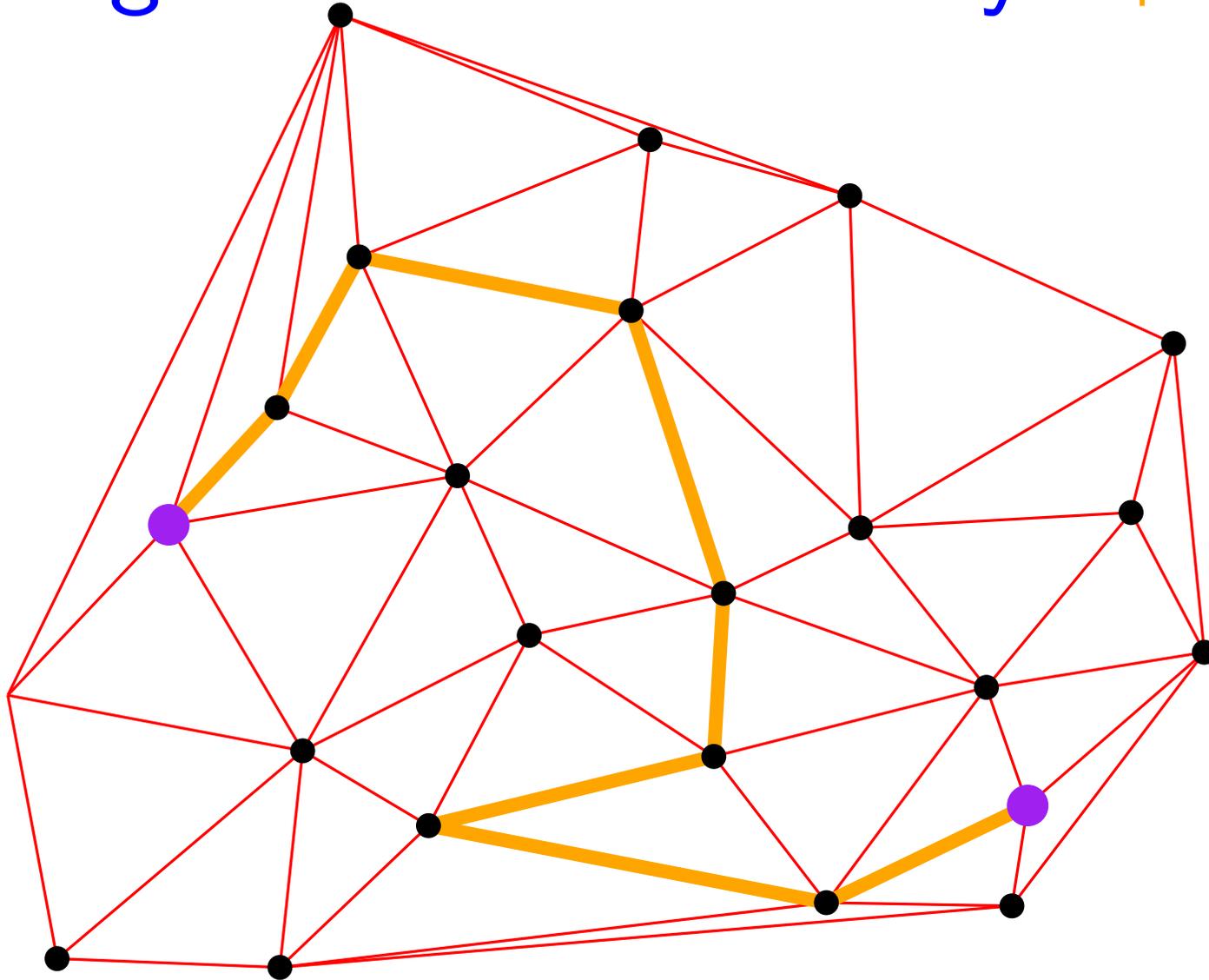
pour un échantillon aléatoire d'une surface sympa

# Triangulation de Delaunay un problème ouvert



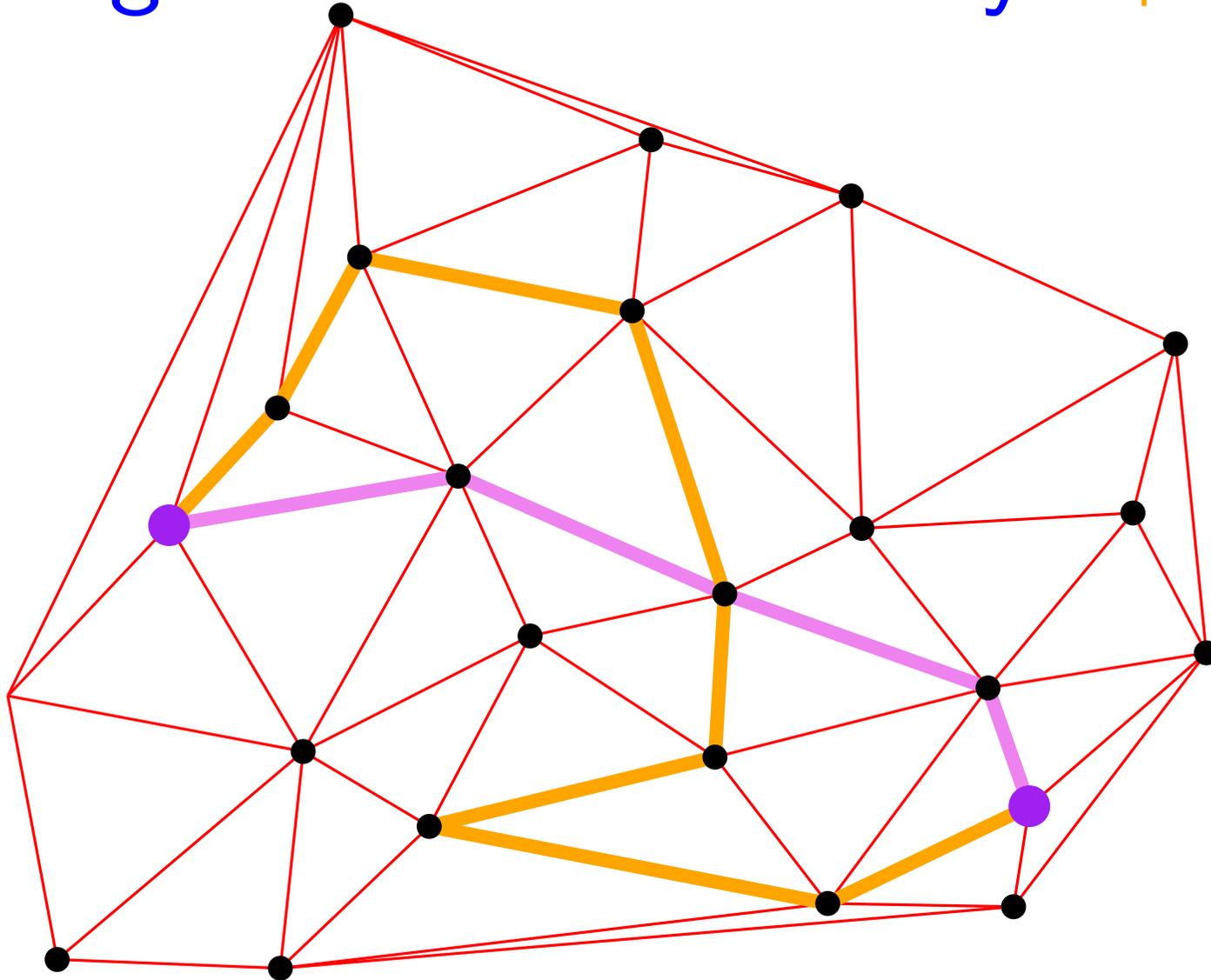
chemin dans Delaunay ?

# Triangulation de Delaunay un problème ouvert



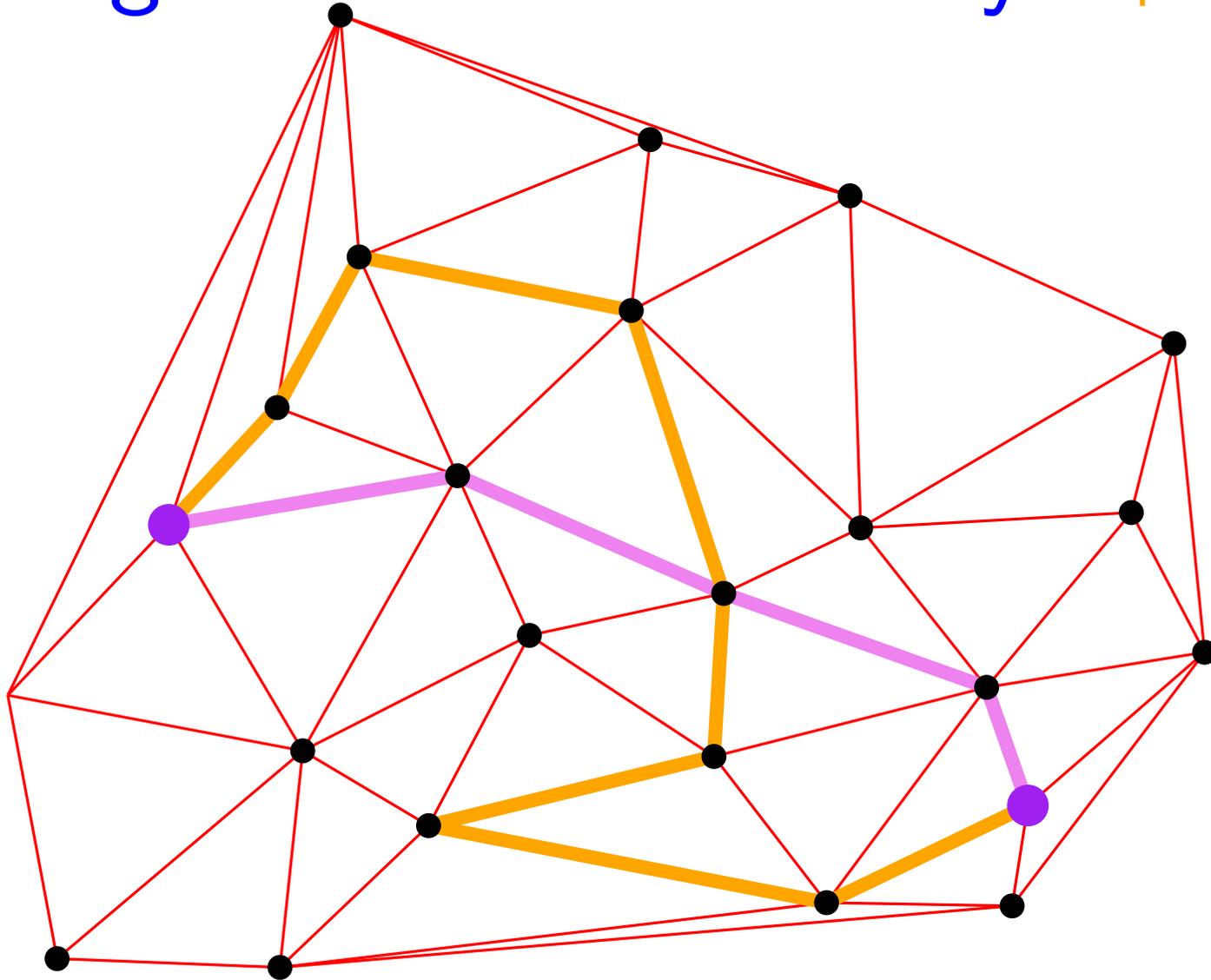
chemin dans Delaunay ?

# Triangulation de Delaunay un problème ouvert



plus court chemin dans Delaunay ?

# Triangulation de Delaunay un problème ouvert



Longueur du plus court chemin dans Delaunay ?

# Triangulation de Delaunay un problème ouvert

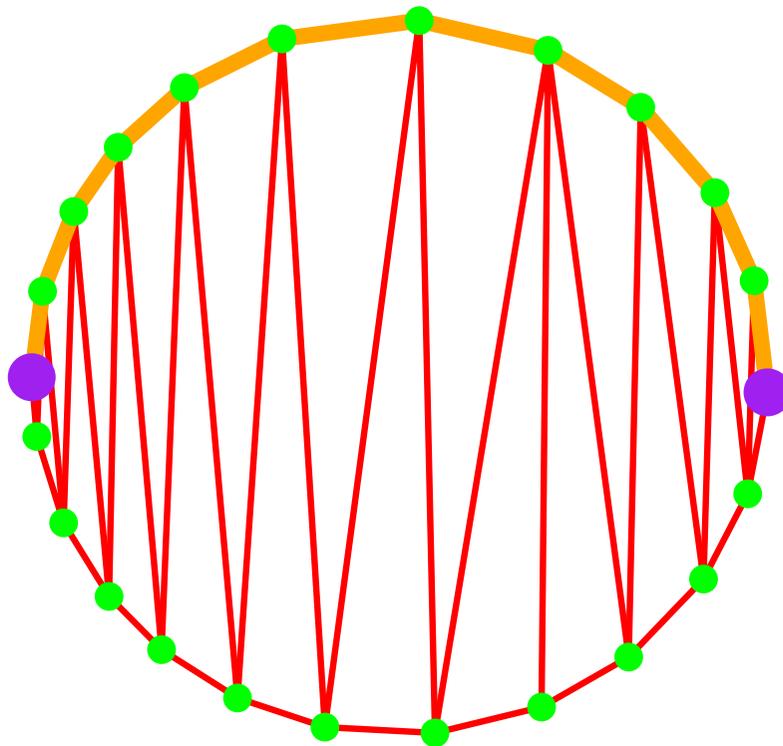
Longueur du plus court chemin dans Delaunay ?

# Triangulation de Delaunay un problème ouvert

Longueur du plus court chemin dans Delaunay ?

Borne sup  $< 1.998$

Borne inf  $> 1.5932$



$$\frac{\pi}{2} = 1.5708$$

# Triangulation de Delaunay un problème ouvert

Longueur du plus court chemin dans Delaunay ?

Borne sup  $< 1.998$

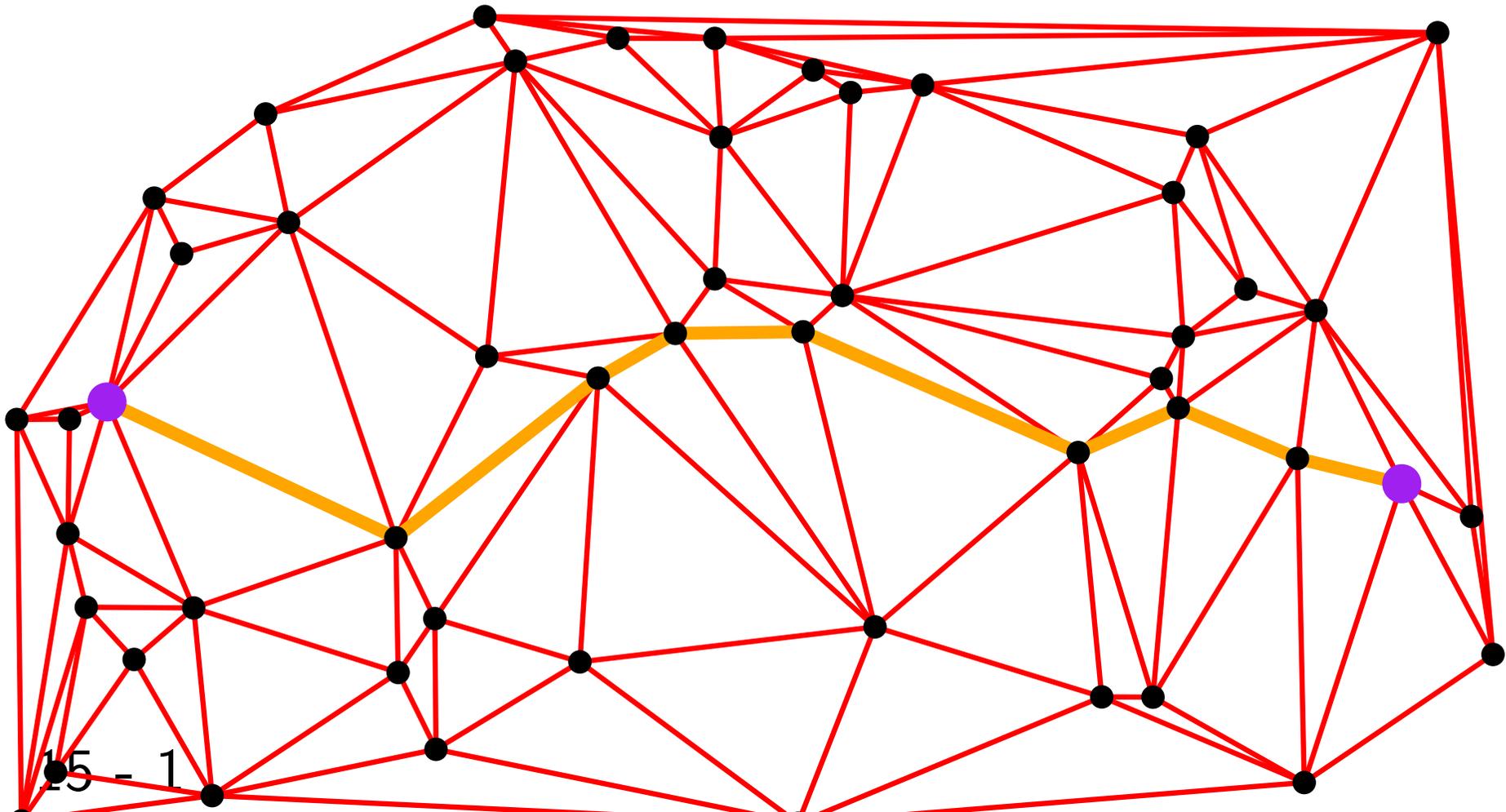
Borne inf  $> 1.5932$

Et pour des points aléatoires ?

# Triangulation de Delaunay un problème ouvert

Longueur du plus court chemin dans Delaunay ?

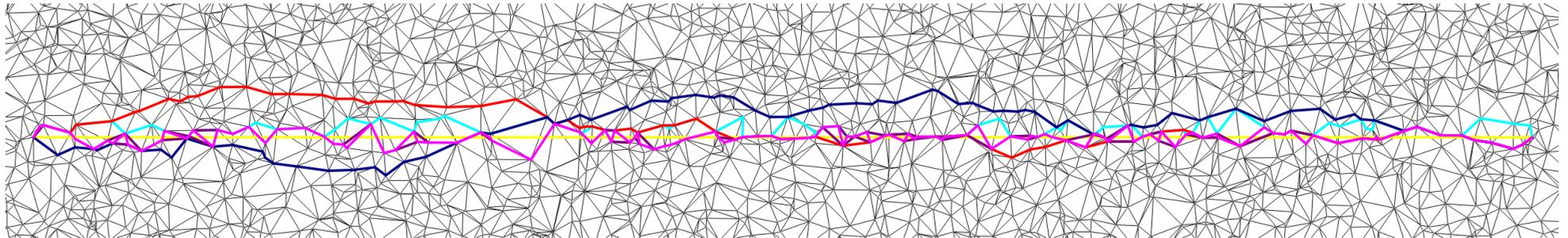
Et pour des points aléatoires ?



# Triangulation de Delaunay un problème ouvert

Longueur du plus court chemin dans Delaunay ?

Et pour des points aléatoires ?



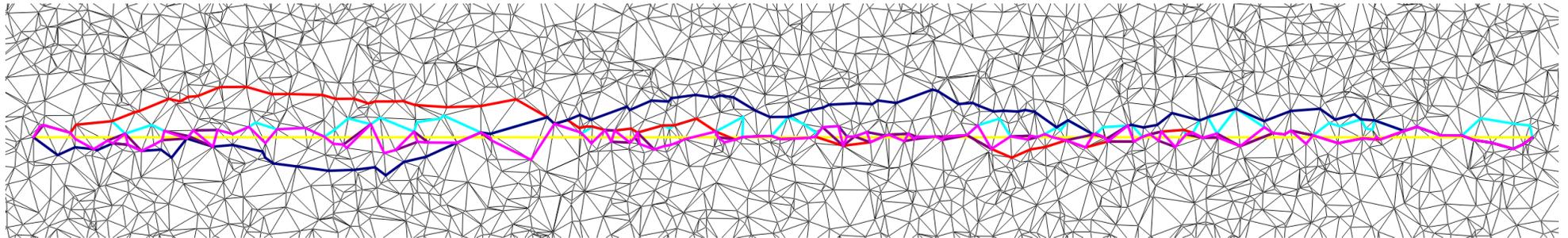
Expérimentalement

1.04

# Triangulation de Delaunay un problème ouvert

Longueur du plus court chemin dans Delaunay ?

Et pour des points aléatoires ?



Expérimentalement

1.04

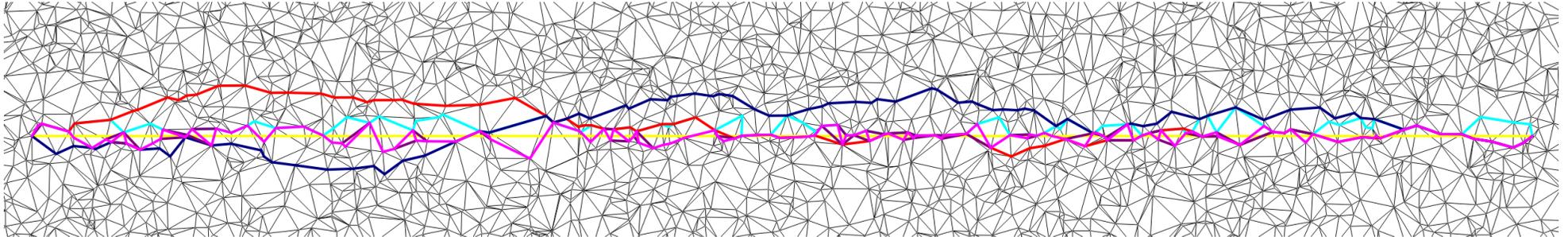
Pour un algo particulier

1.18

# Triangulation de Delaunay un problème ouvert

Longueur du plus court chemin dans Delaunay ?

Et pour des points aléatoires ?

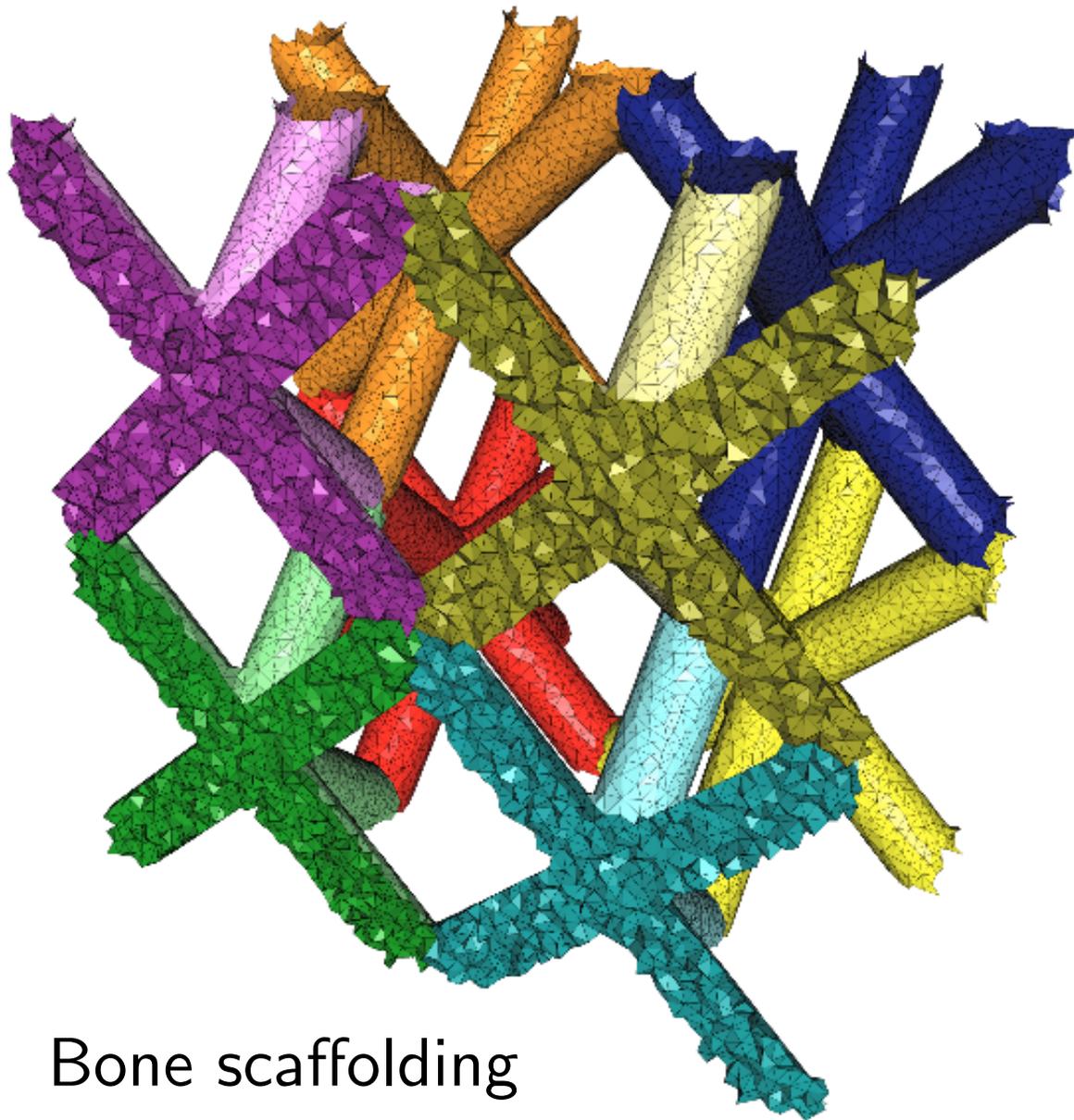


Expérimentalement **1.04**

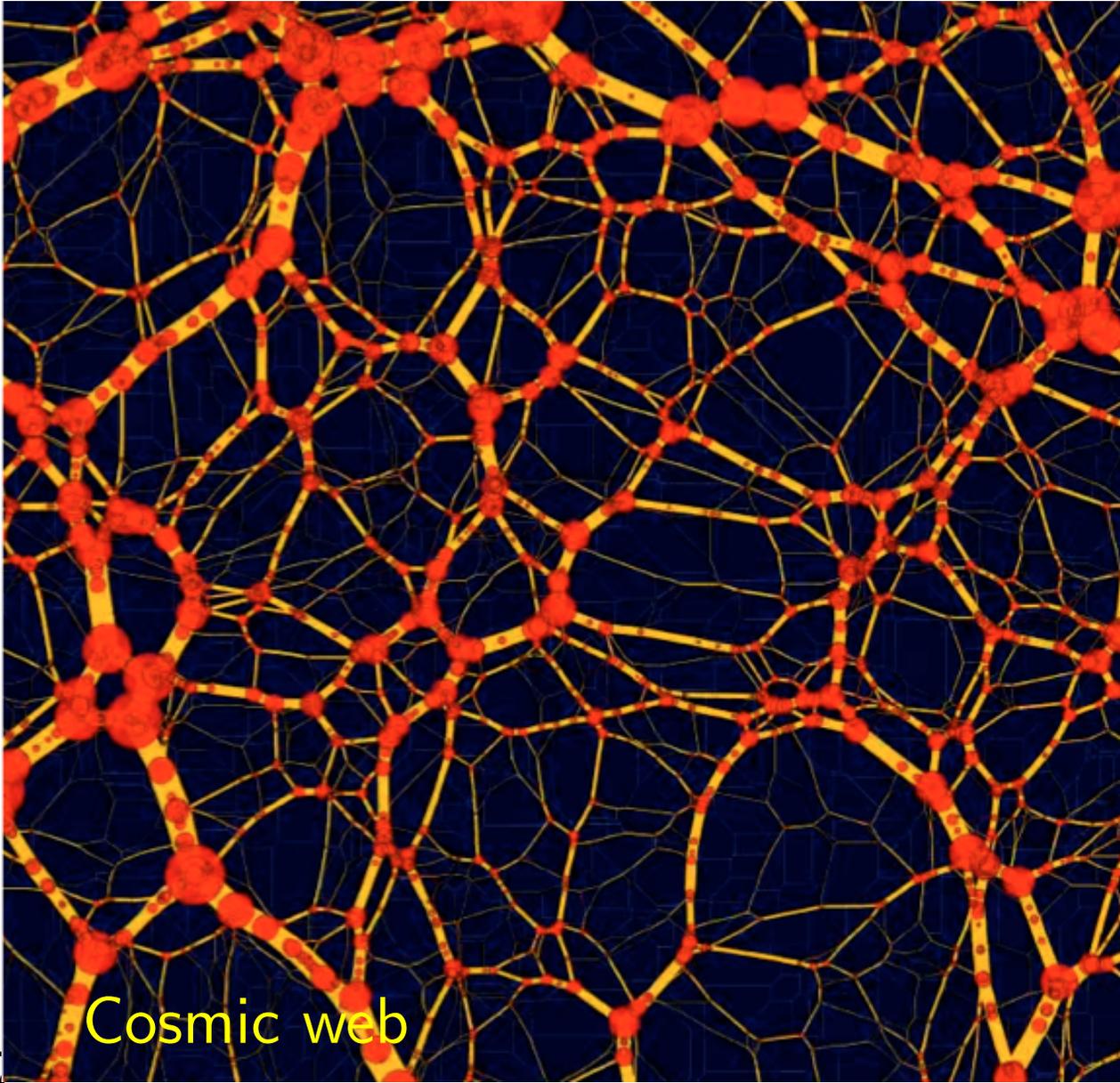
Pour un algo particulier **1.18**

Borne inf démontrée **1.0000000001**

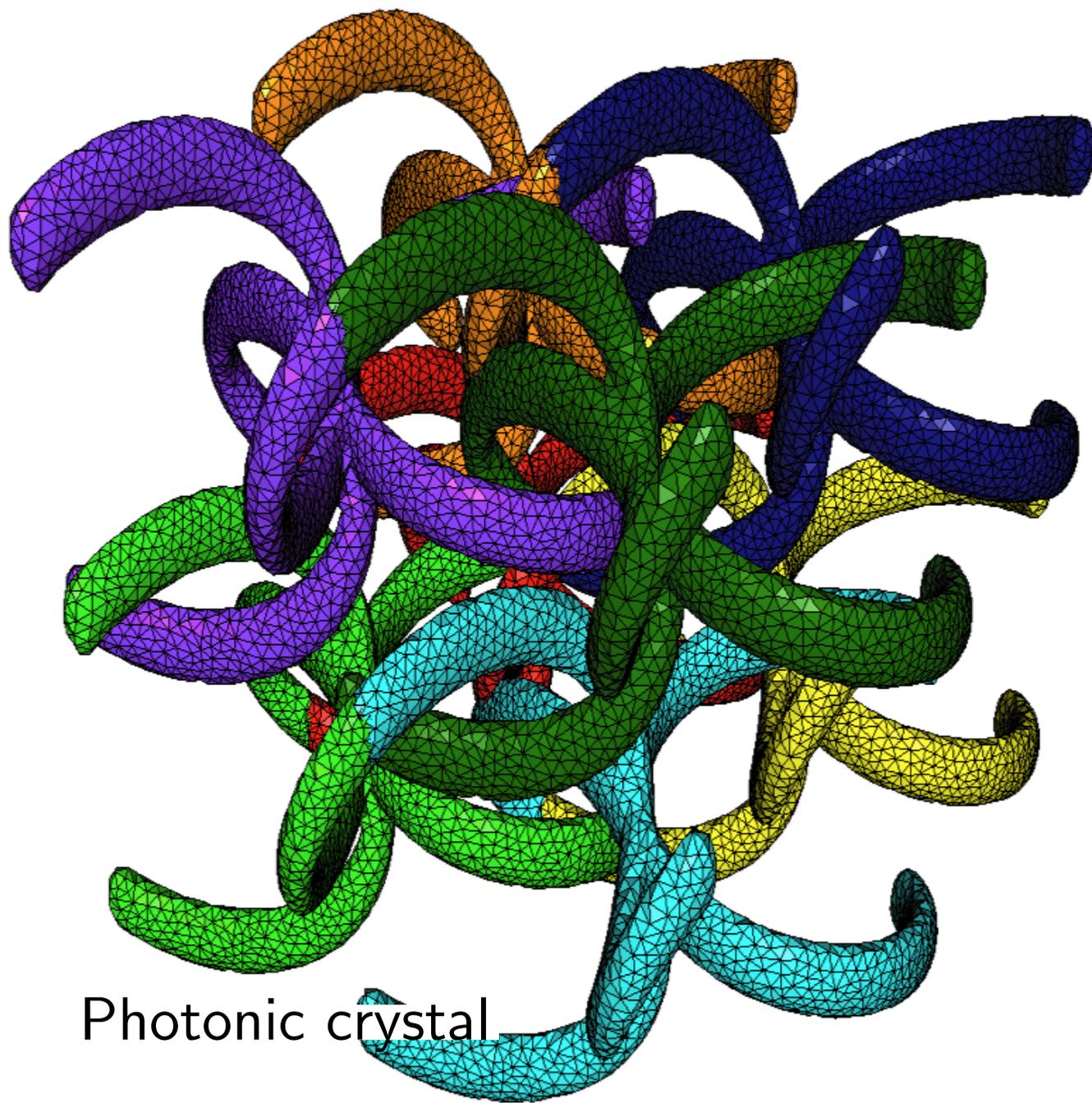
Merci



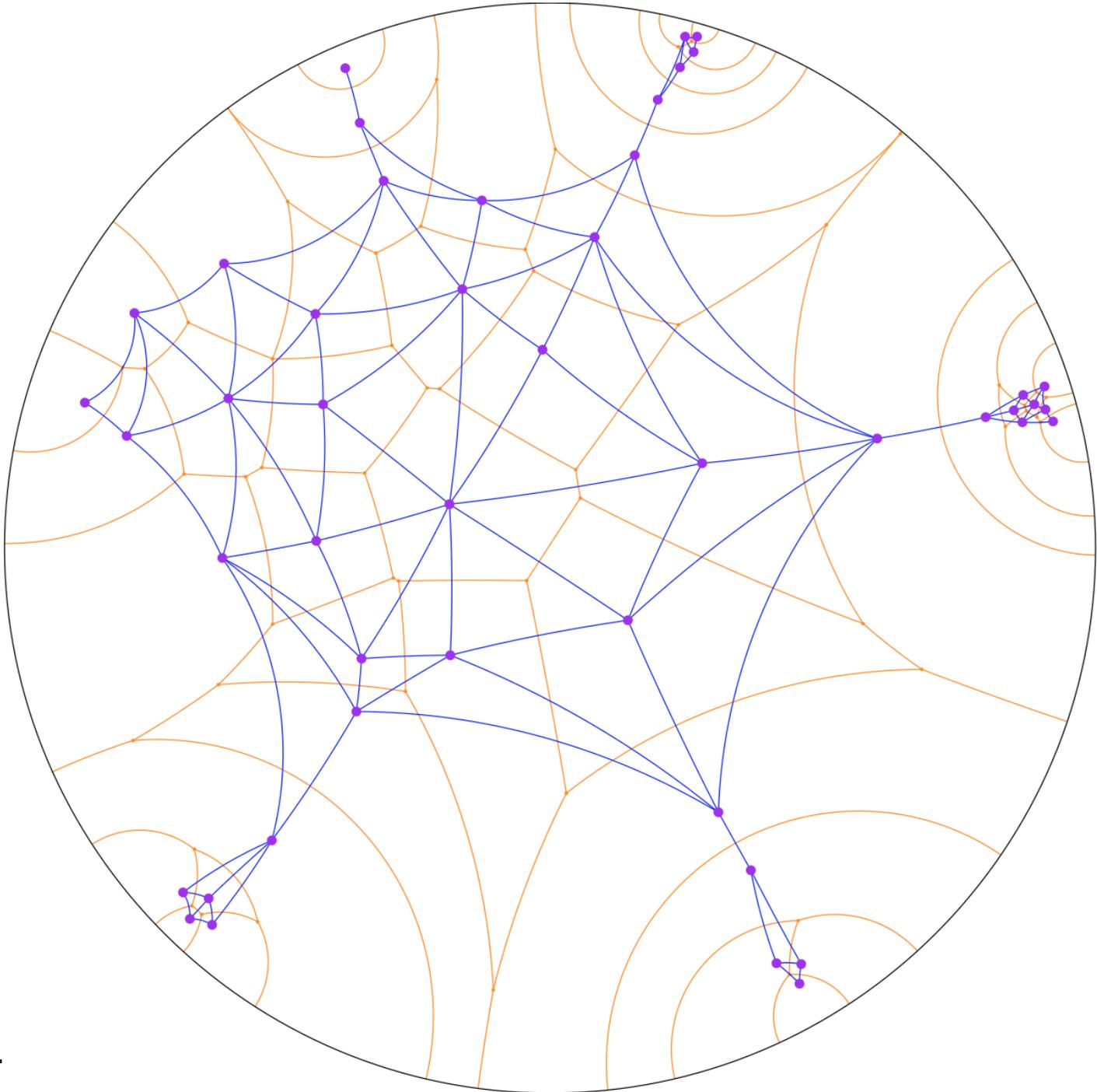
Bone scaffolding

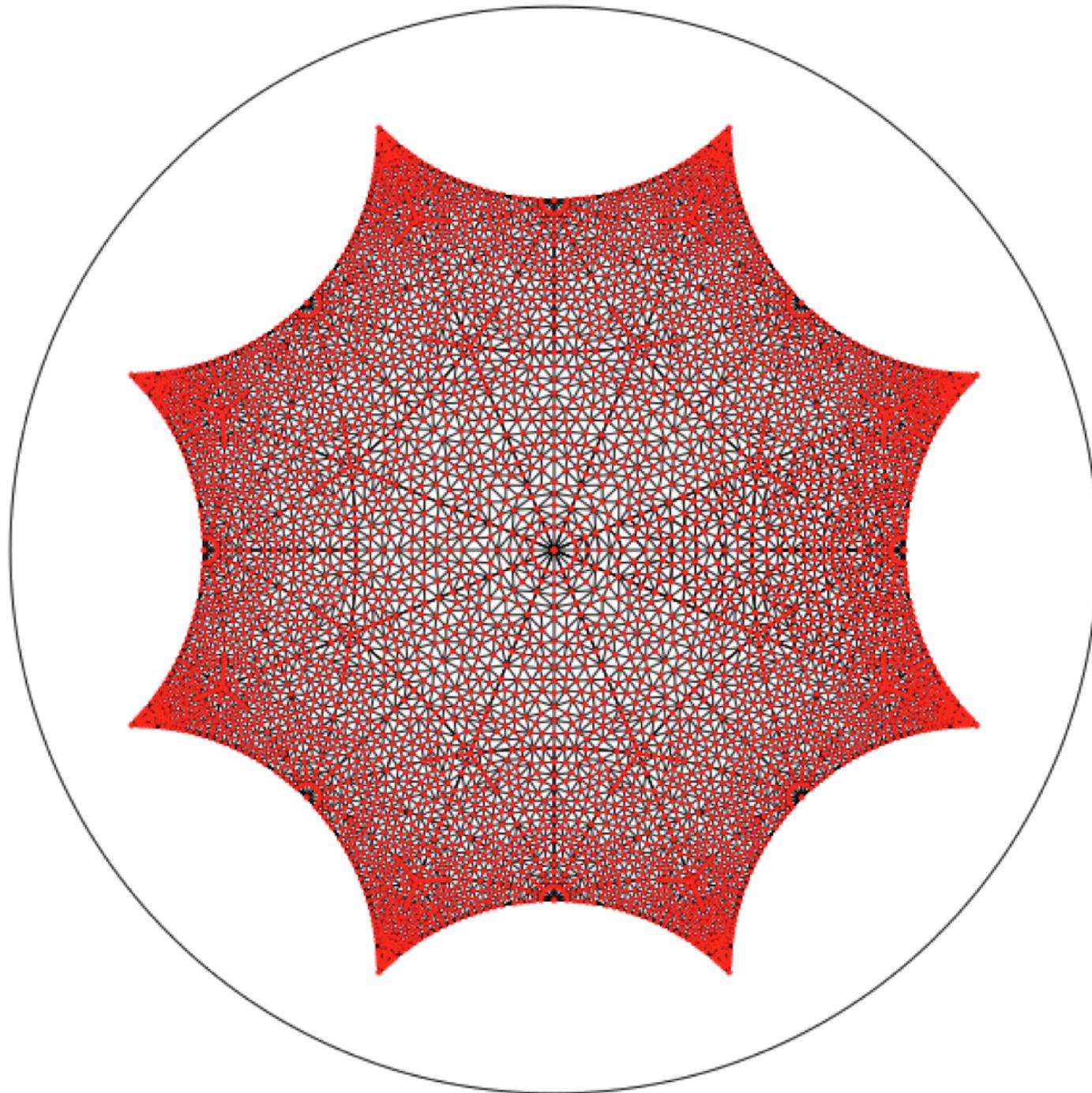


Cosmic web



Photonic crystal





CGAL