

# Internships in Gamble team

<http://gamble.loria.fr/positions.html>

# Delaunay triangulation on surfaces by flips

Proposed by Monique Teillaud & Vincent Despré

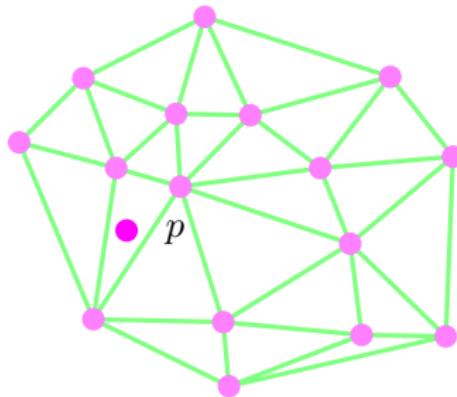
PhD on the same topic

# Delaunay triangulation on surfaces by flips

Vincent Despré and Monique Teillaud

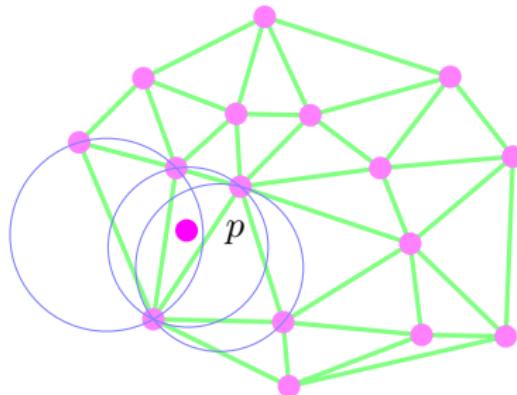
INRIA Nancy - Grand Est, LORIA

# “Star hole” incremental construction



For each new point  $p$

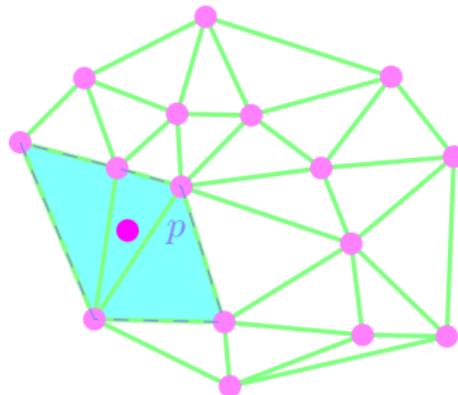
# “Star hole” incremental construction



For each new point  $p$

- locate  $p$  = find triangles in conflict

# “Star hole” incremental construction

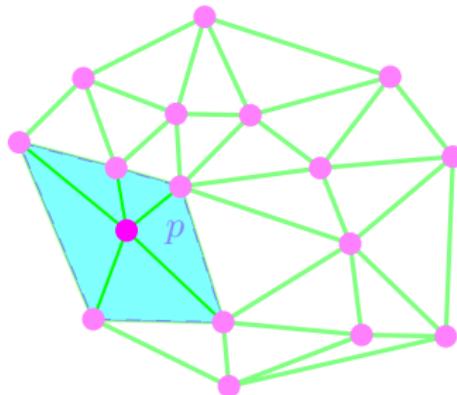


For each new point  $p$

- locate  $p$  = find triangles in conflict

the hole is a topological disk

# “Star hole” incremental construction

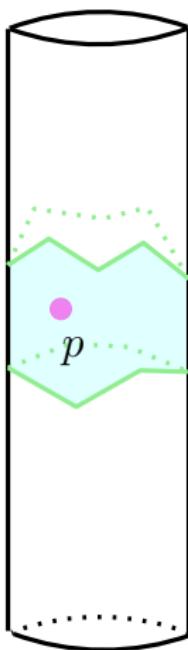


For each new point  $p$

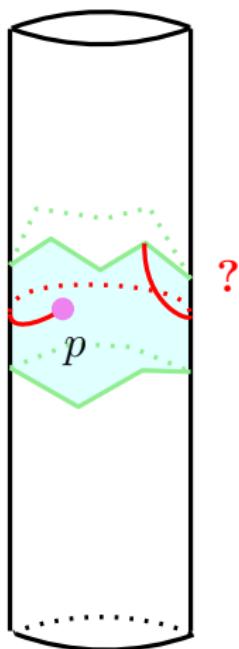
- locate  $p$  = find triangles in conflict
- star the hole from  $p$

the hole is a topological disk

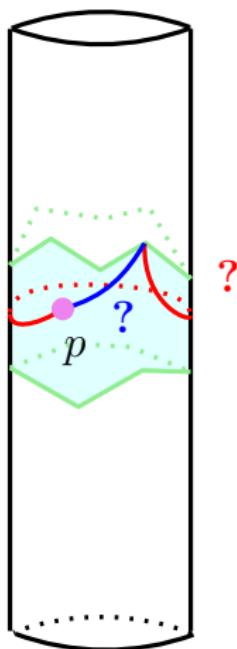
# What if the hole is **not** a topological disk?



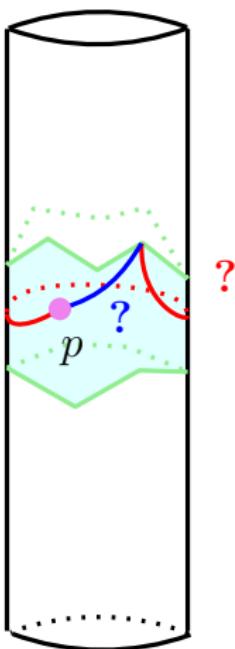
# What if the hole is **not** a topological disk?



# What if the hole is **not** a topological disk?



# What if the hole is not a topological disk?



Does the insertion by flips work better?

# Rounding 3D meshes

Proposed by Sylvain Lazard

PhD on the same topic

# Rounding 3D meshes

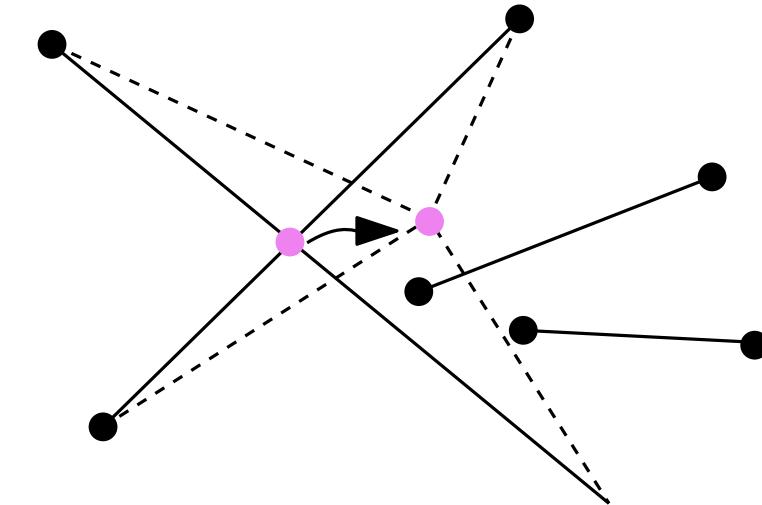
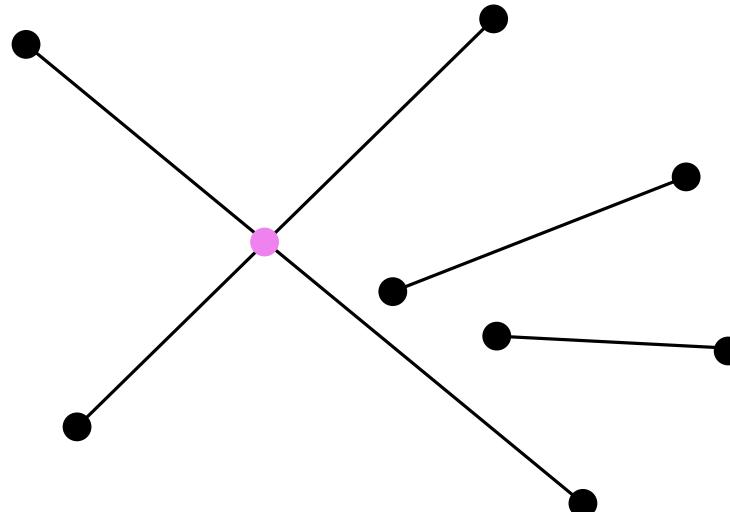
Intersecting 3D meshes output vertices whose coordinates cannot be *exactly* represented with doubles.

Problem: Given a set of interior-disjoint triangles in 3D

Vertices with arbitrary coordinates

Round the coordinates on a grid – Subdivision are allowed

Naively rounding the coordinates to the closest double may create intersections, recursively

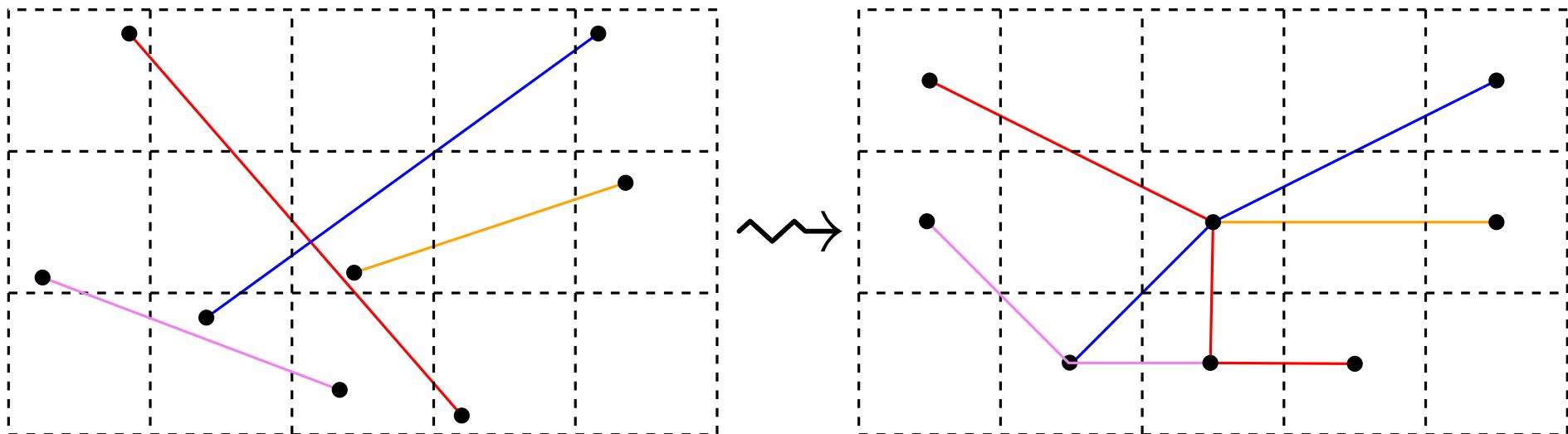


# Rounding 3D meshes

Problem well solved in 2D

For simplicity, the rounding is done on the integer grid

- Every pixel that contains a vertex or an intersection is tagged *hot*
- Every segment that intersects a hot pixel is subdivided in that pixel
- All vertices are rounded on  $\mathbb{Z}^2$



# Rounding 3D meshes

1999 : Complicated and not satisfactory algorithm [S. Fortune]

2018 : “simple” algorithm but very bad worst-case complexity  
[Us]

In short:

1. Apply some projections of the faces along  $X$ ,  $Y$  and  $Z$
2. Let  $c_1, \dots, c_k$  be the vertex  $X$ -coordinates rounded on  $\mathbb{Z}$   
Subdivide all faces by the planes  $X = c_i \pm 1/2$
3. Triangulate the faces and round all vertices on  $\mathbb{Z}^3$

Goal: **Implement a simplified version of the algorithm**  
**Study and develop improvements**

# Smoothed analysis of walking strategies in Delaunay triangulation

Proposed by Olivier Devillers

Straight walk analysis

$X$  a Poisson point process of density  $n$

(0,0)

(1,0)

Random case

Straight walk analysis

$X$  a Poisson point process of density  $n$

(0,0)

(1,0)

Random case

## Straight walk analysis

$X$  a Poisson point process of density  $n$

$$\mathbb{E} [\text{ crossed edges}] = 2.16\sqrt{n}$$

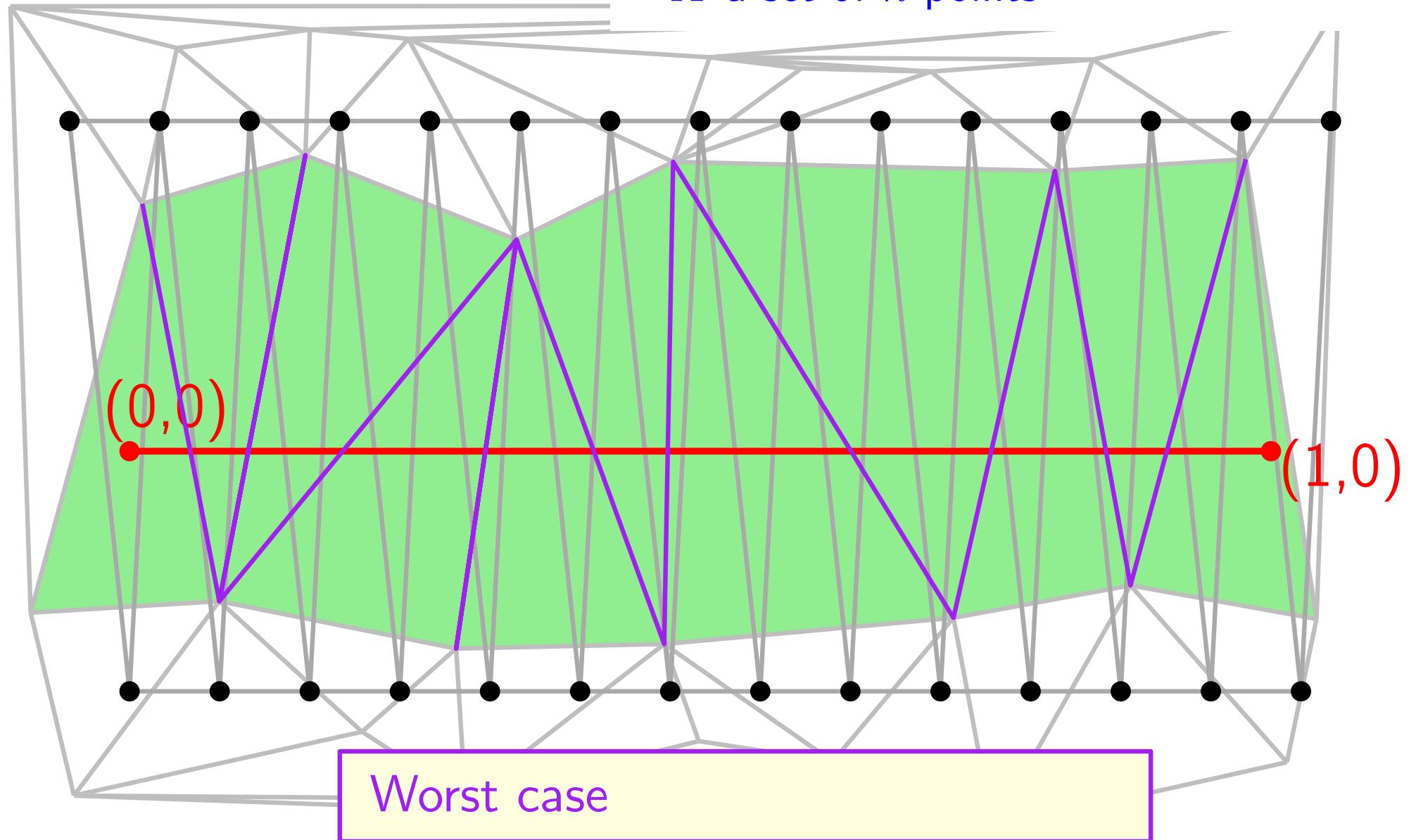
(0,0)

(1,0)

Random case

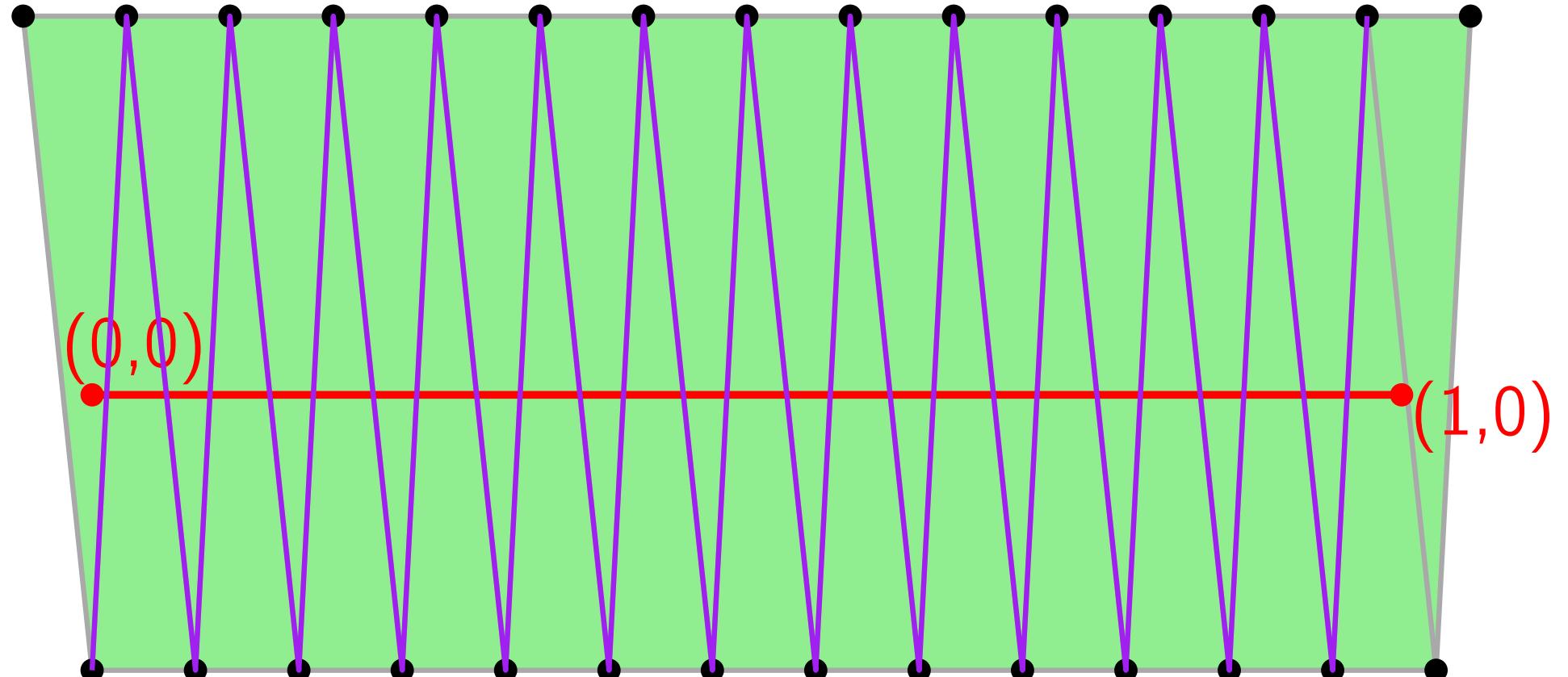
# Straight walk analysis

$X$  a set of  $n$  points



# Straight walk analysis

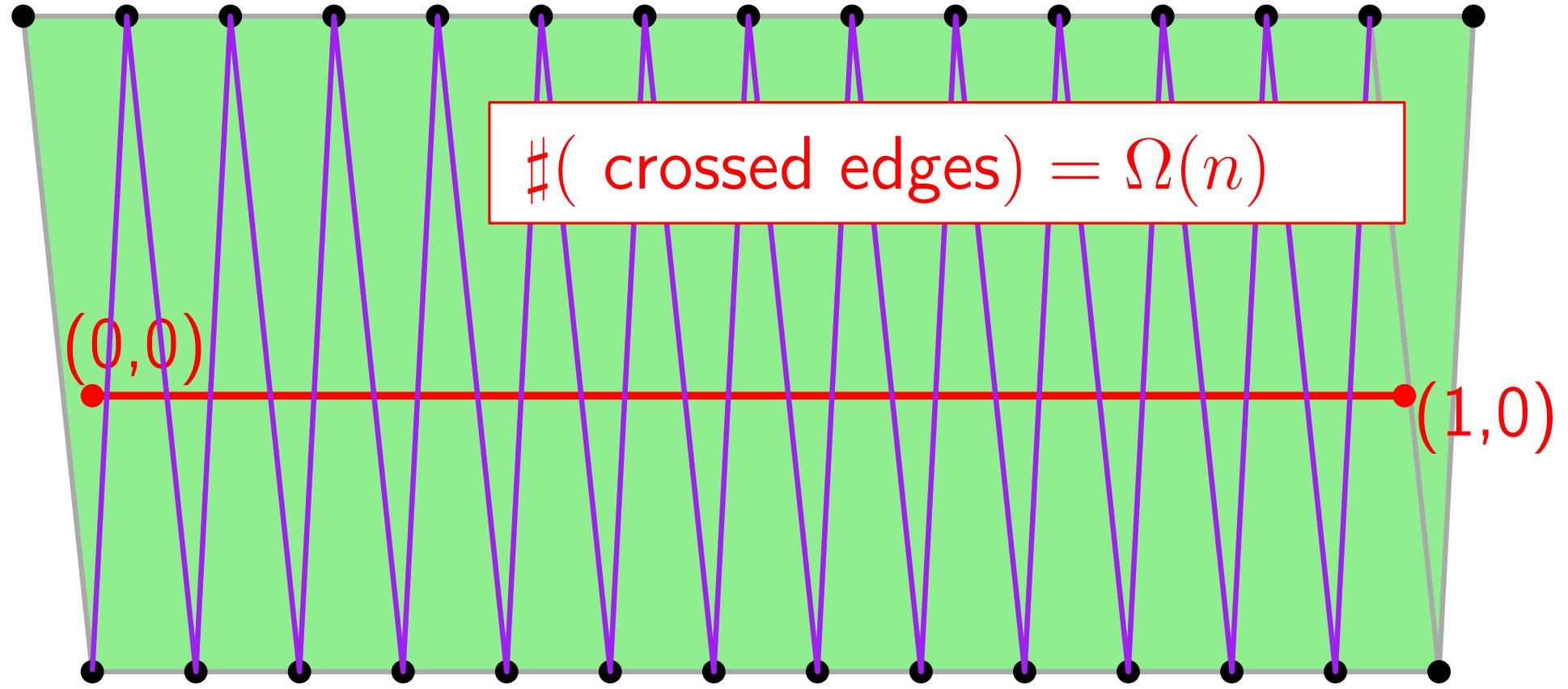
$X$  a set of  $n$  points



Worst case

# Straight walk analysis

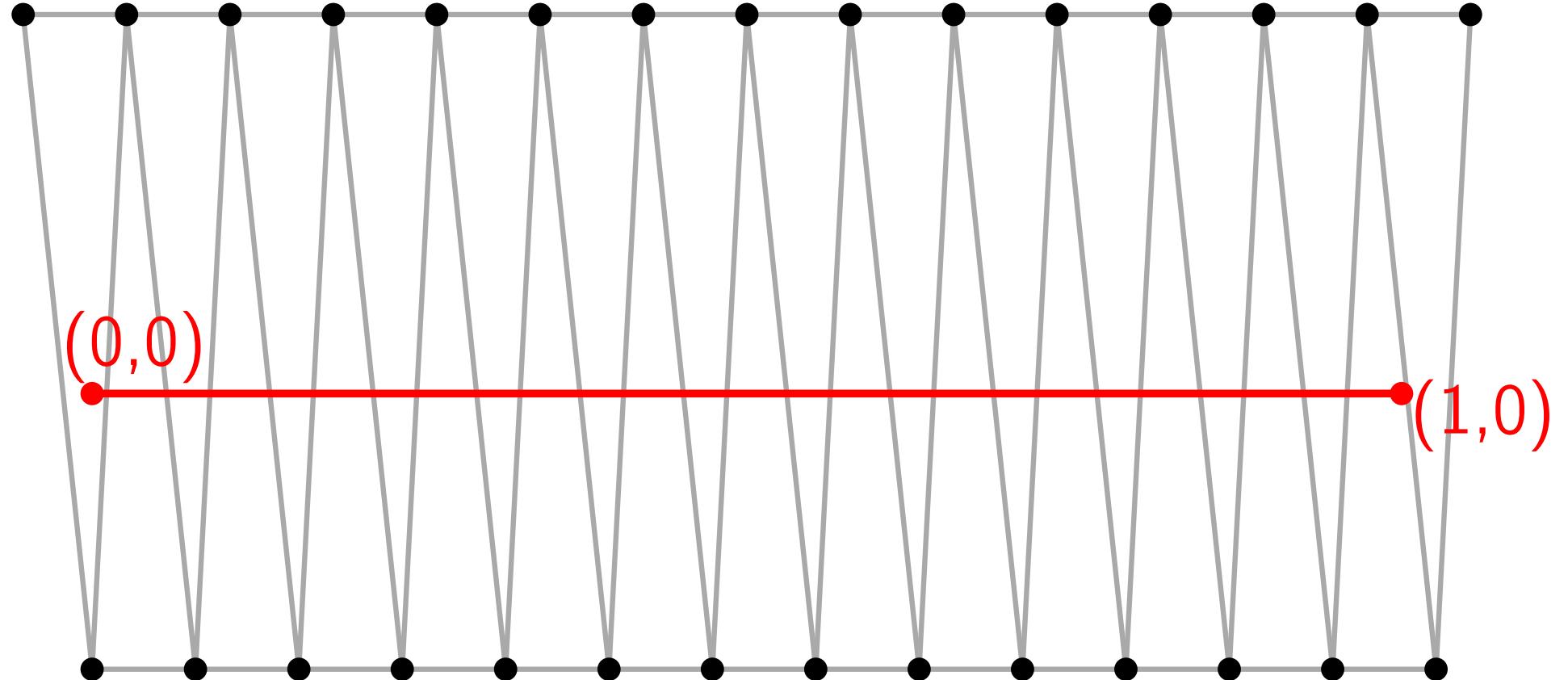
$X$  a set of  $n$  points



Worst case

# Straight walk analysis

$X$  a set of  $n$  points



Smoothed case

# Straight walk analysis

$X$  a set of  $n$  points + noise

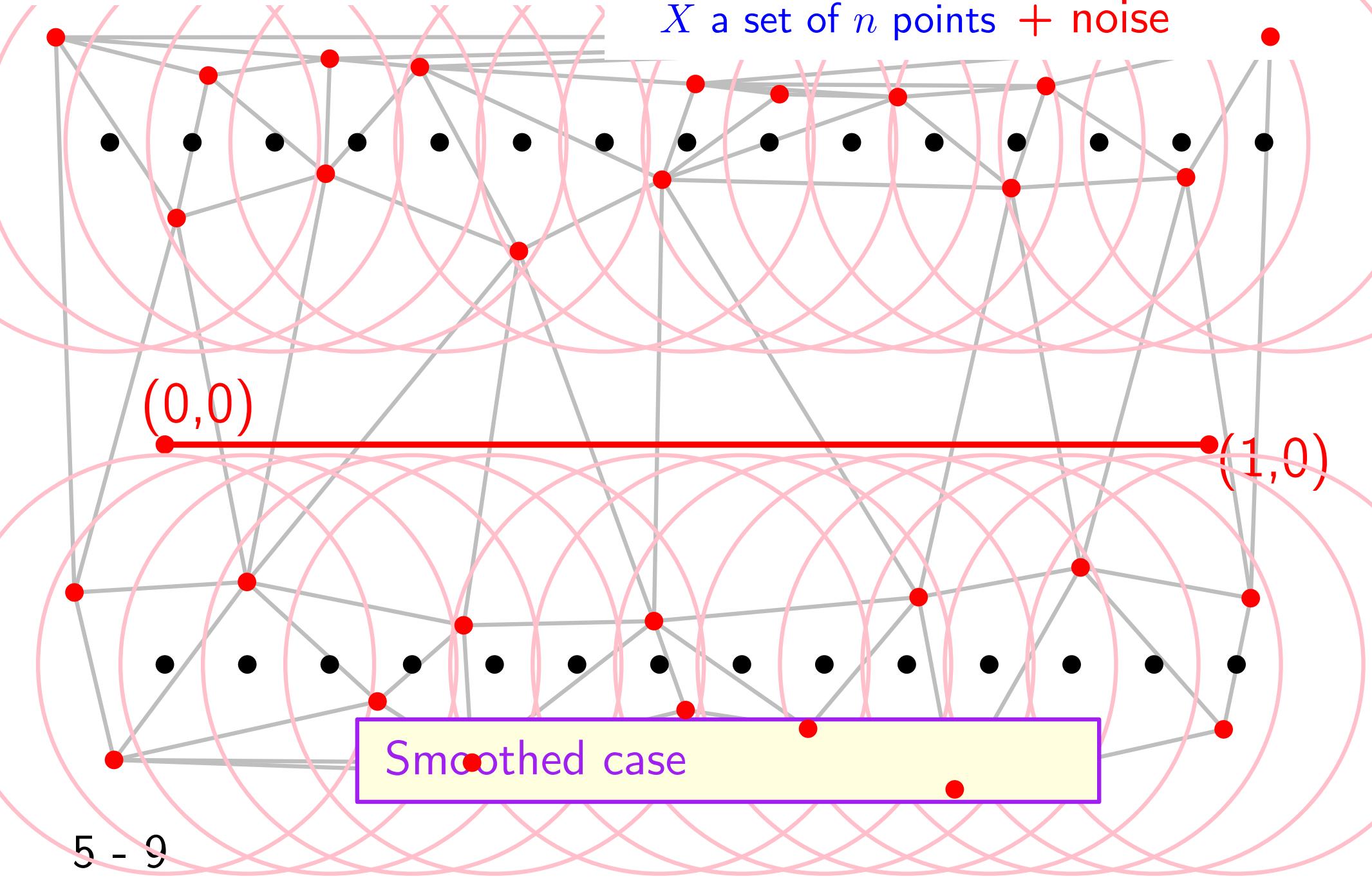
(0,0)

(1,0)

Smoothed case

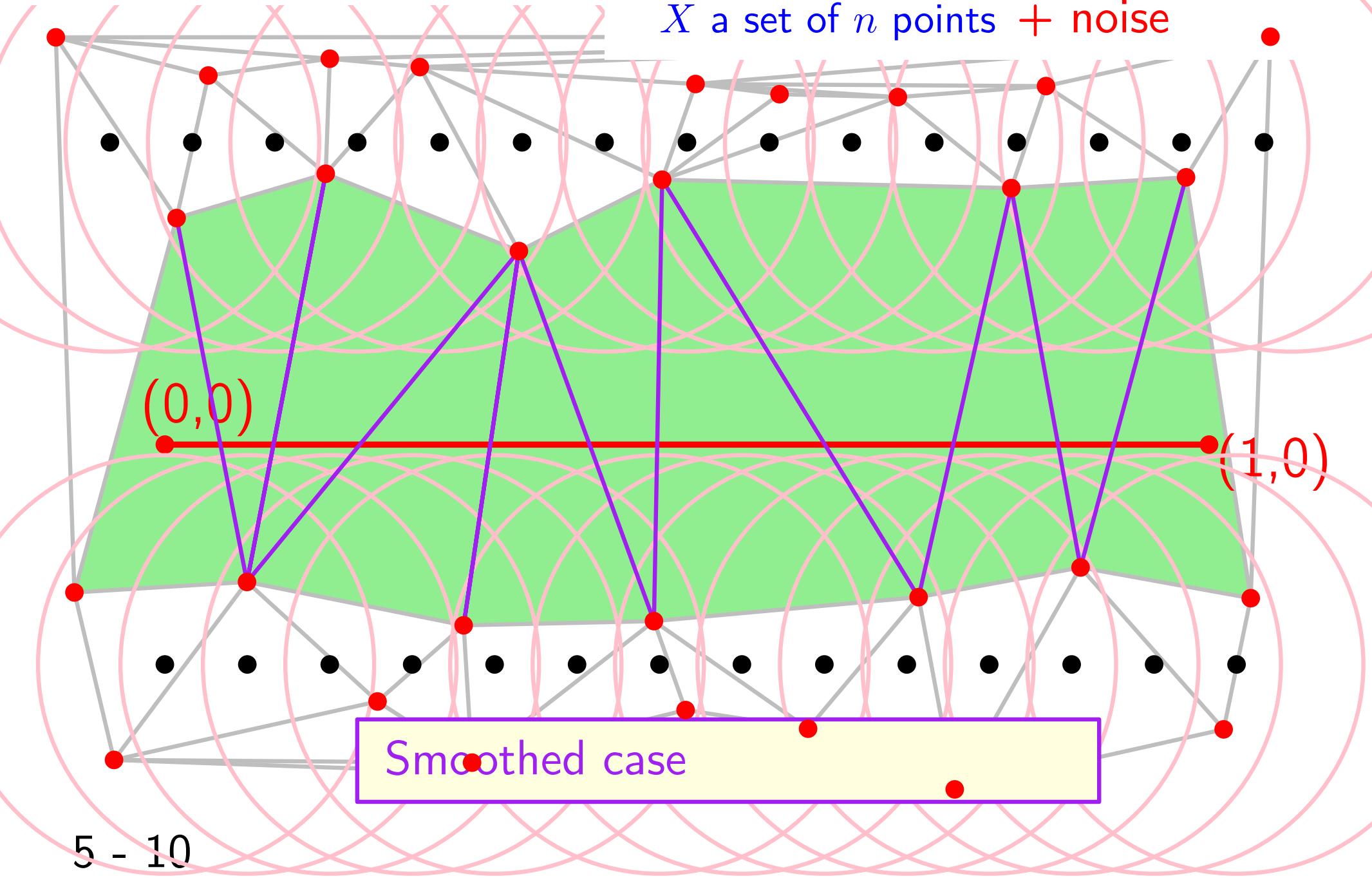
# Straight walk analysis

$X$  a set of  $n$  points + noise



# Straight walk analysis

$X$  a set of  $n$  points + noise



# Straight walk analysis

$X$  a set of  $n$  points + noise

$$\mathbb{E} [ \text{ crossed edges}] = ?$$

(0, 0)

(1, 0)

Smoothed case

# Solving generic polynomial equations

Proposed by Guillaume Moroz

# Solving equations with Chebyshev approximations

## Polynomial equation

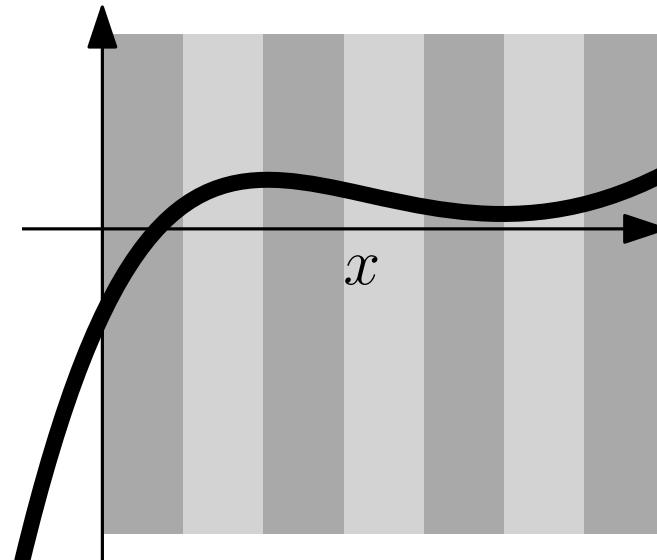
$$a_0 + a_1 x + \cdots + a_n x^n = 0$$

## Applications

- *Robotics*: find the parameters of a robotic arm to reach a given position.
- *Dynamic system*: find the position of an equilibrium
- ...

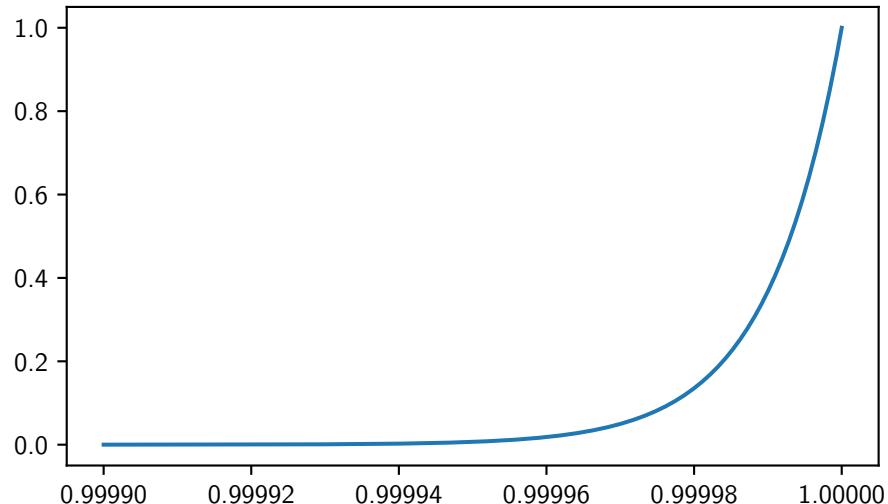
## A dichotomic approach

1. Do: subdivide the domain in smaller intervals
2. Until: the intervals contain either exactly 1 or 0 solutions.

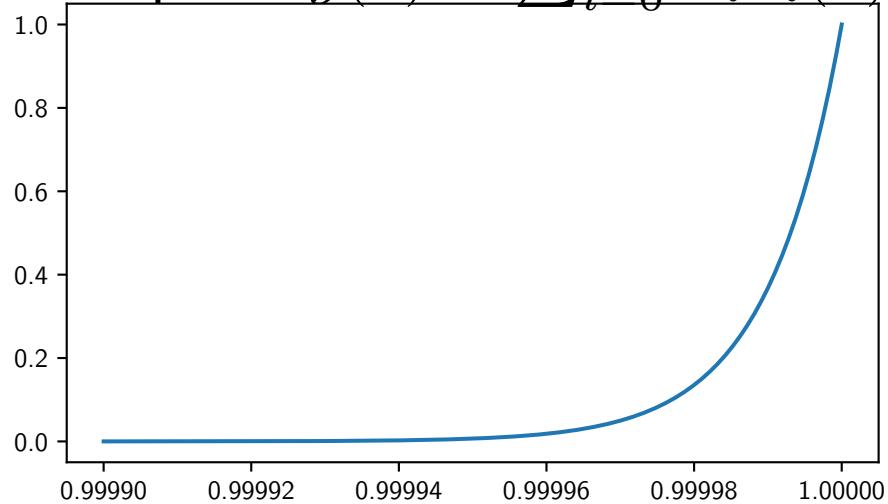


# Solving equations with Chebyshev approximations

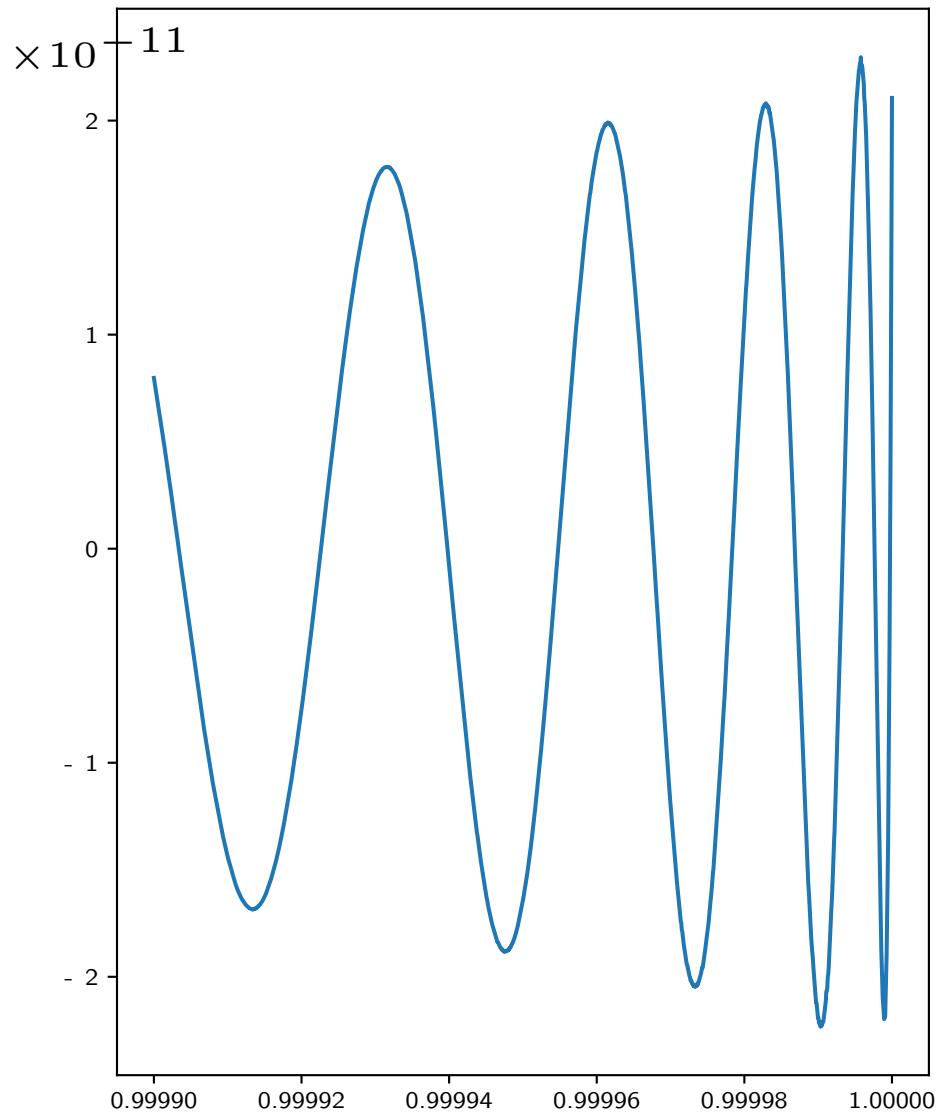
Graph of  $f(x) = x^{100000}$



Graph of  $g(x) = \sum_{i=0}^{1518} c_i T_i(x)$



Graph of error  $f(x) - g(x)$



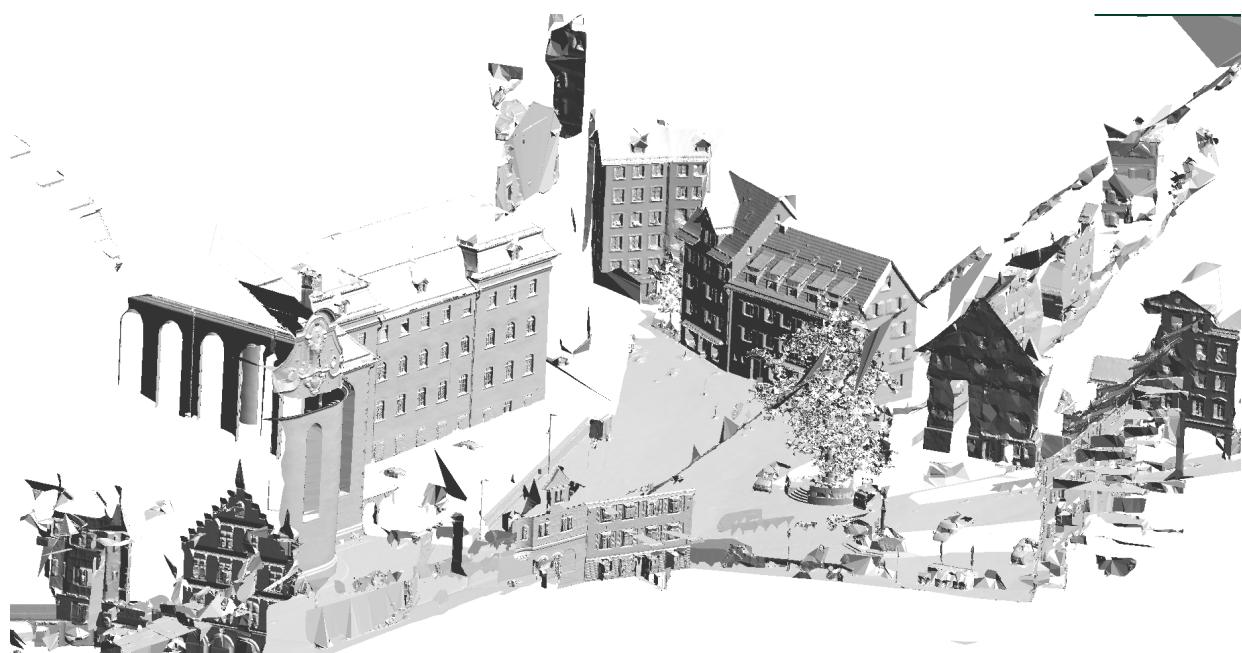
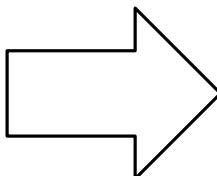
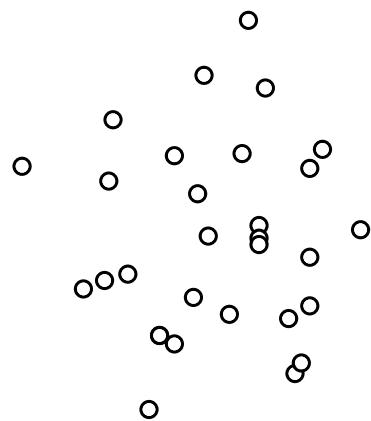
Approximation of  $x^n$  by a polynomial of degree  $d = \sqrt{n \ln(10^{-k})}$   
with  $n = 100000$ ,  $k = 10$  and  $d = 1518$

# Analyse d'un algorithme de reconstruction surfacique

Proposed by Xavier Goaoc & Dobrina Boltcheva

# Analyse d'un algorithme de reconstruction surfacique

Problématique : construire "un bon" maillage à partir d'un nuage de points.



## Objectif :

Étudier les garanties théoriques offertes par un algorithme efficace en pratique.

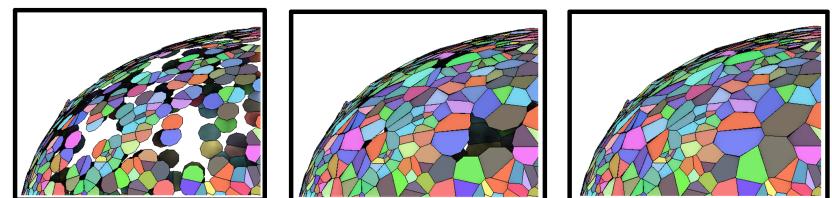
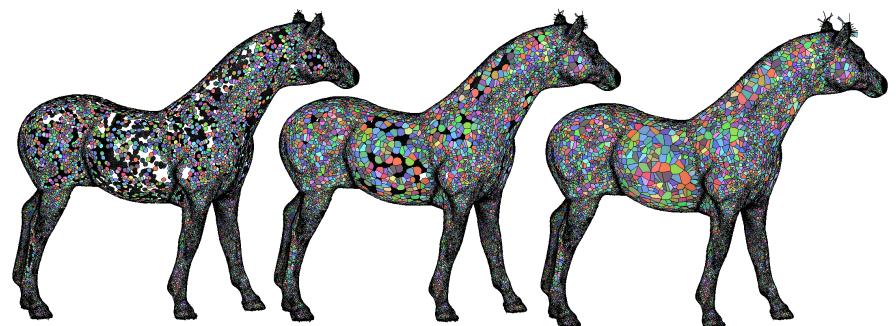
## Ingédients :

Delaunay/Voronoi, dualité

topologie, théorème du nerf

analyses locales, recollements...

analyse probabiliste



## Encadrants :

Dobrina Boltcheva, équipe ALICE (infographie, traitement de données géométriques)

Xavier Goaoc, équipe GAMBLE (géométrie algorithmique)

## Contacts :

[dobrina.boltcheva@univ-lorraine.fr](mailto:dobrina.boltcheva@univ-lorraine.fr), [xavier.goaoc@loria.fr](mailto:xavier.goaoc@loria.fr)

