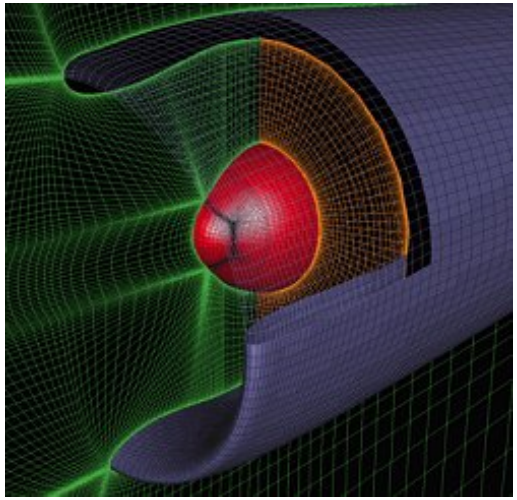
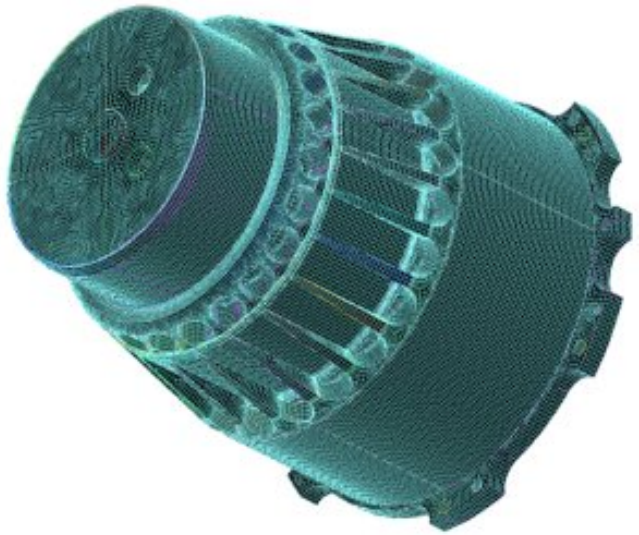


Introduction au maillage pour le calcul scientifique



Franck Ledoux
CEA DAM Île-de-France, Bruyères-le-Châtel
franck.ledoux@cea.fr

Présentation adaptée du tutorial de ***Steve Owen***,
Sandia National Laboratories, Albuquerque, NM, USA

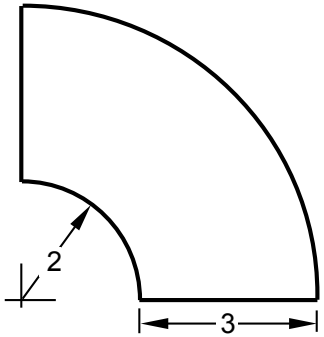
Introduction

- La simulation numérique consiste à reproduire informatiquement un phénomène physique, chimique, biologique ou autre, au moyen de
  - Équations physiques — dans un monde continu
  - Schémas numériques — dans un monde discret
  - Algorithmes informatiques
  - Maillages — une partie du monde discret

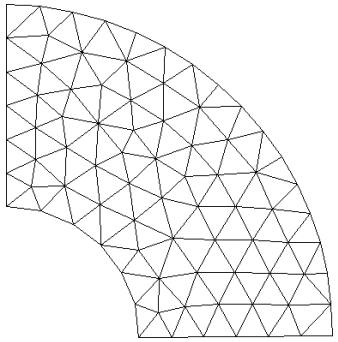
- Procédé très gourmand en temps de calcul et en espace mémoire
  - Parfait candidat pour le HPC

Préparation de la simulation numérique

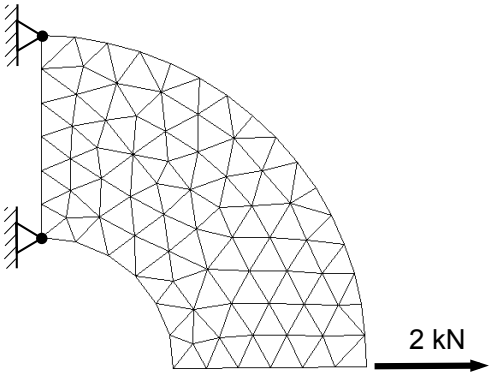
Processus simplifiée



1. Création du modèle géométrique

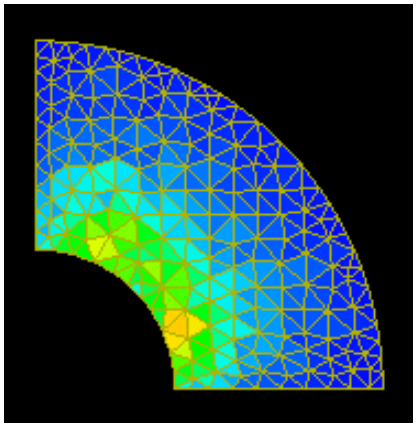


2. Maillage



3. Définition des conditions physiques (conditions limites, densités, matériaux,…)

4. Simulation numérique (CALCUL, potentiellement plusieurs heures)



5. Visualisation

Dassault Aviation-CEA/DIF : « Falcon 7x »

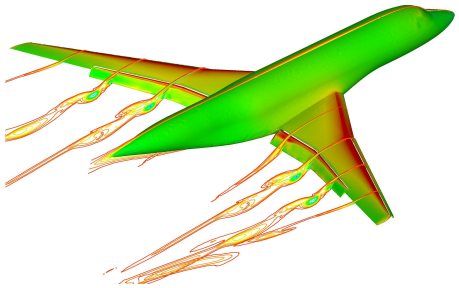
Ecoulements 3D autour d'un avion complet - 800 processeurs de TERA 10 pendant 250 heures

Démonstration de compatibilité des outils de conception avec les architectures informatiques nécessaires dans le futur

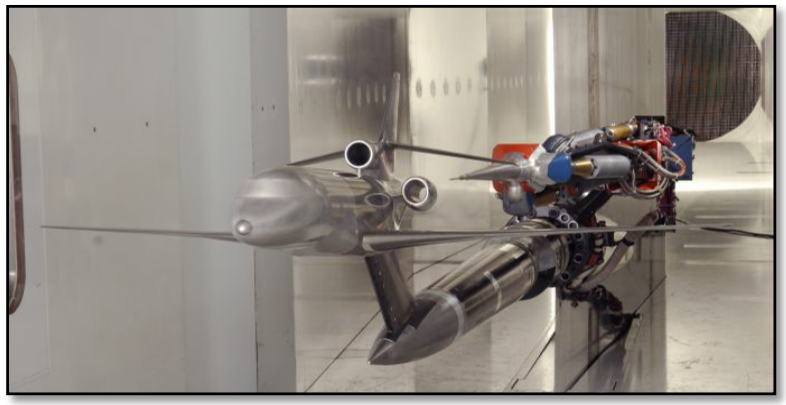
Amélioration du processus de conception actuel : maîtrise des incertitudes induites par les maillages de calcul de traînée

Application possible à des problèmes plus complexes

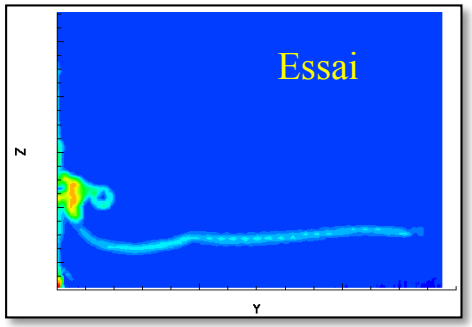
Maillage de 115 millions d'éléments, calcul de 7 valeurs physiques (pression, vitesses, température, énergie cinétique, ...) en chaque maille pour chaque pas de temps (140 millions d'équation à résoudre pour l'ensemble du calcul)

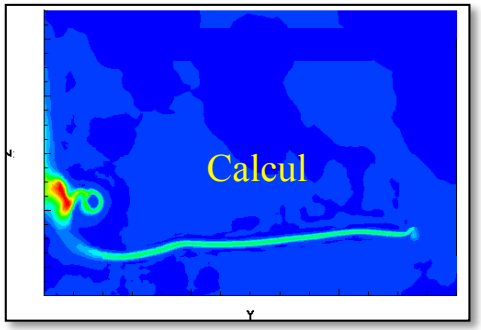


Démonstration technologique
Amélioration de la prédiction de la traînée



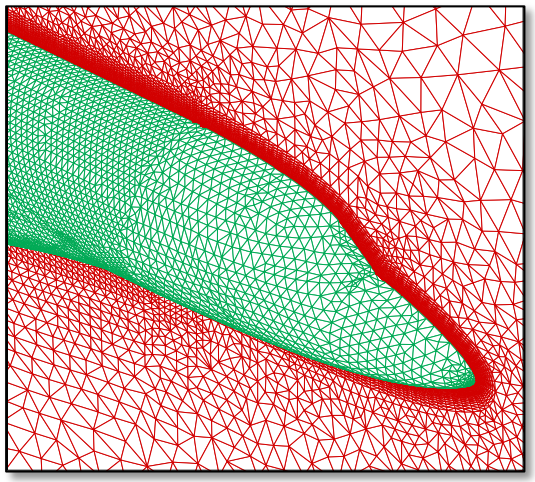
Soufflerie ONERA S2MA





La prédiction du sillage est un effet direct du raffinement en maillage
⇒ amélioration de la prédiction de la trainée

Projection dans l'avenir
à 4 ans par rapport à l'évolution du processus outillé



Maillage industriel :
6.10^6 points
&
30.10^6 éléments

Autres exemples

- Deux exemples de mécaniques des solides

Plan très simple !

Présentation des algorithmes traditionnels en maillage

- Maillage triangulaire et tétraédrique
- Maillage quadrangulaire et hexaédrique

On n'abordera pas

- Les maillages mixtes
- Les algorithmes de recherche
- Les détails qui vont qu'un algorithme est optimisé
- Les structures de données informatiques nécessaires

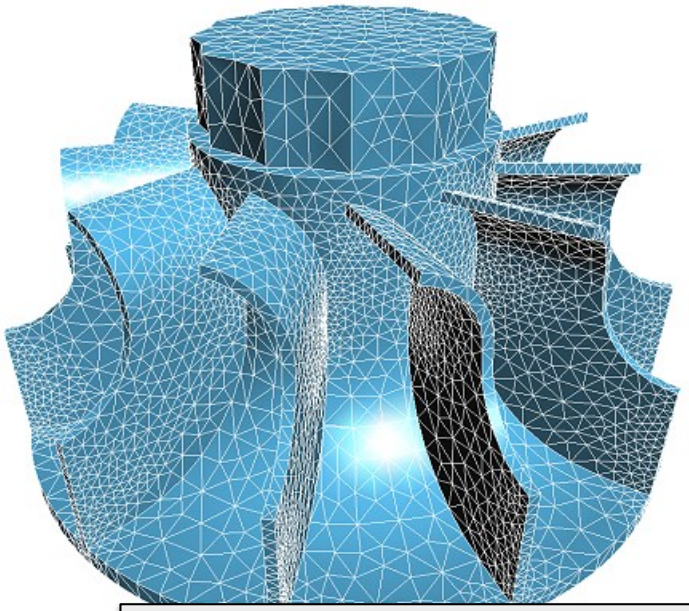
Partie 1

Maillage triangulaire et tétraédrique

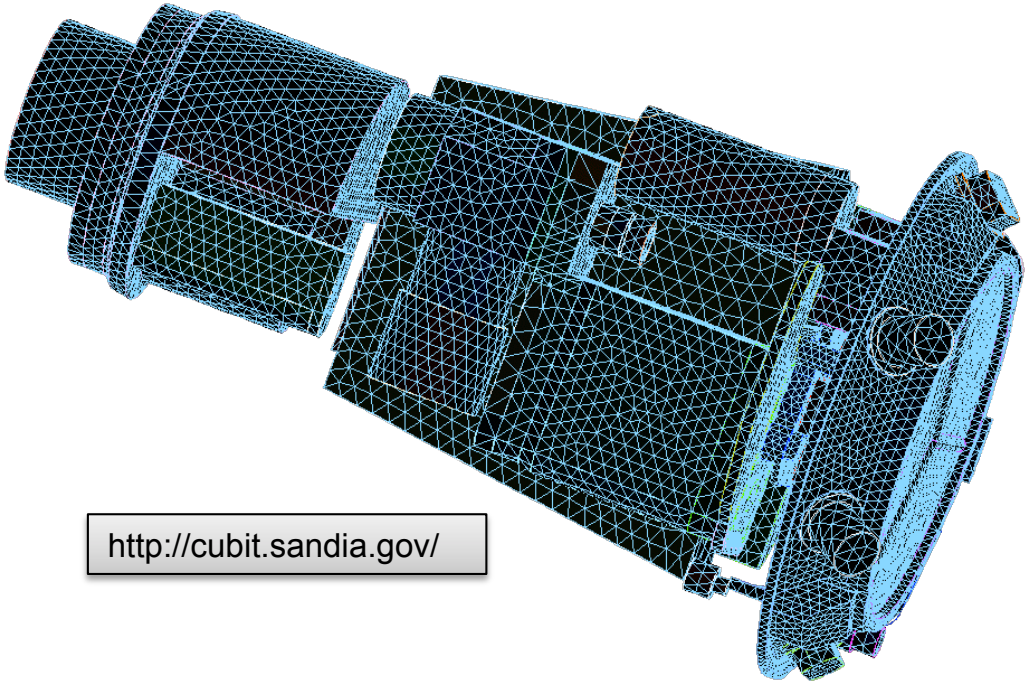
Maillage triangulaire et tétraédrique

3 grandes familles d'algorithmes

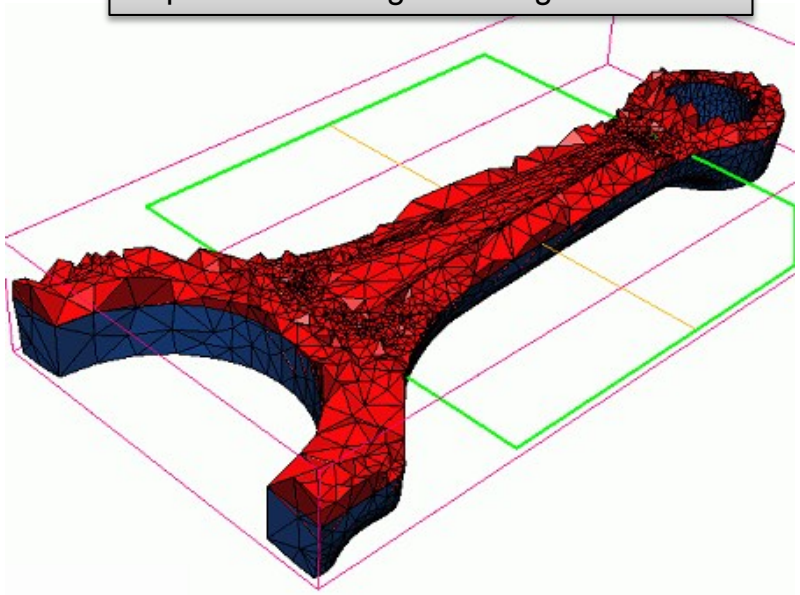
- Quadtree / Octree
- Avancée de front
- Delaunay



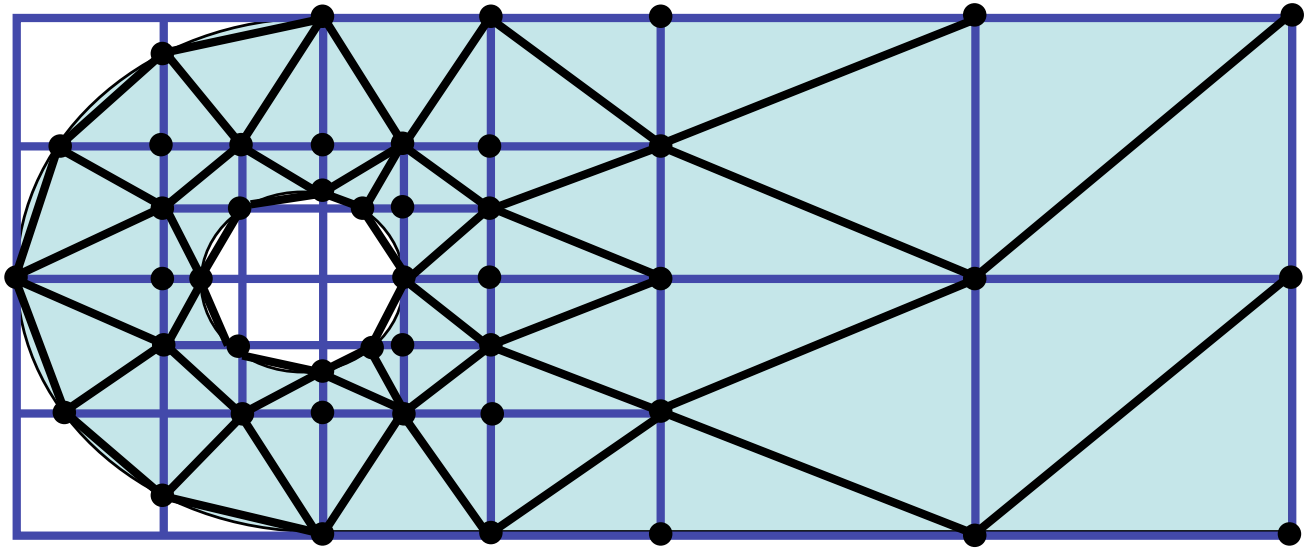
http://www.simulog.fr/mesh/gener2.htm



http://cubit.sandia.gov/



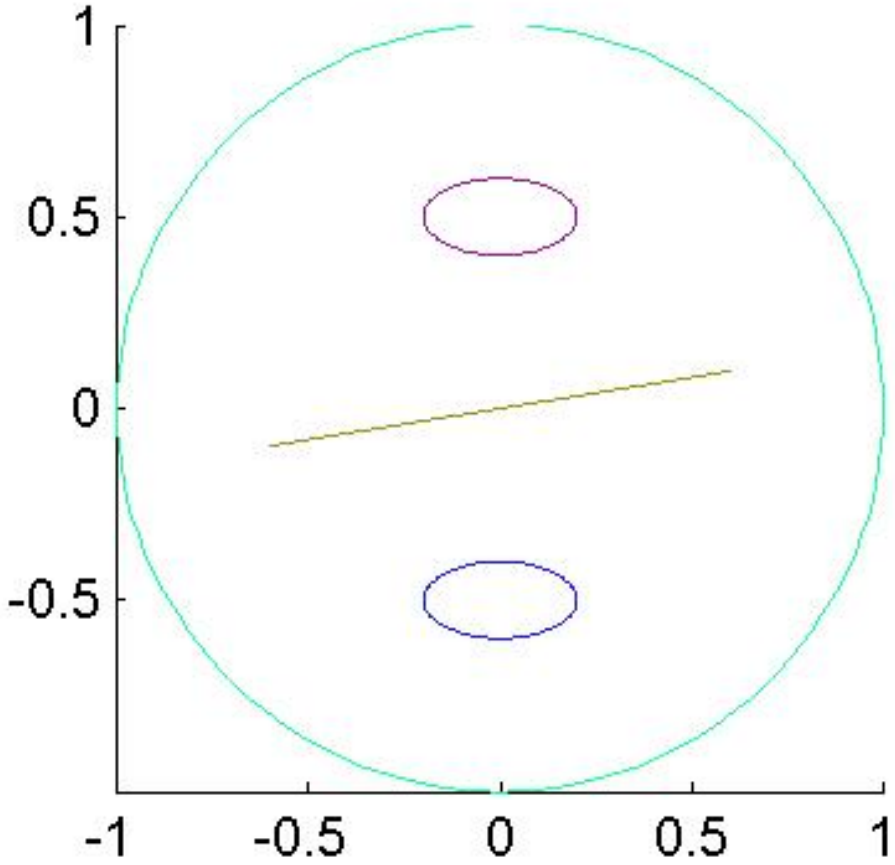
Méthodes QuadTree (2D) – Octree (3D)



- Définition d'une boîte englobante initiale (racine du quadtree)
- Découpage récursif de chaque quadrangle en 4 feuilles dans les zones où la géométrie est mal capturée
- Récupération des intersection entre les arêtes des feuilles et la géométrie
- Maillage de chaque feuille en utilisant les sommets (coins) de la feuille et les intersections avec la gométrie
- Suppression des triangles extérieurs à la géométrie

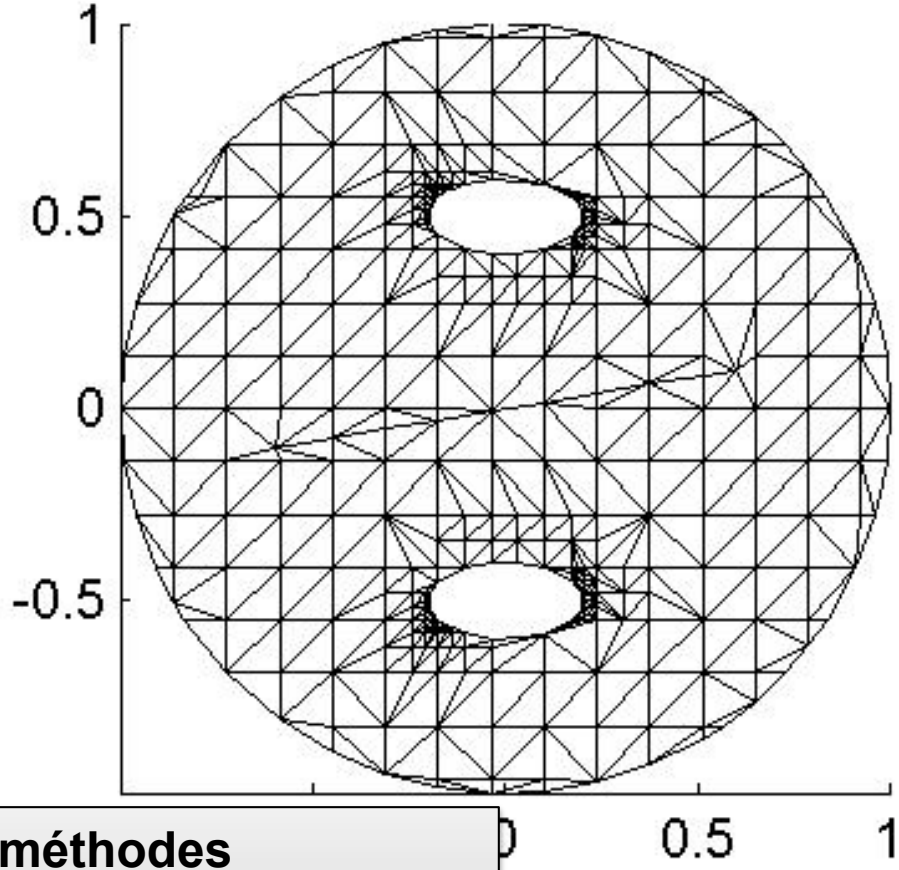
[Yerry and Shephard, 84] [Shepherd and Georges, 91]

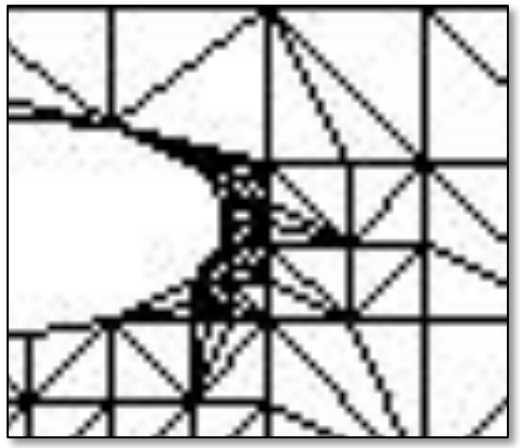
Méthodes QuadTree (2D) – Octree (3D)



QMG,
Cornell University

Méthodes QuadTree (2D) – Octree (3D)

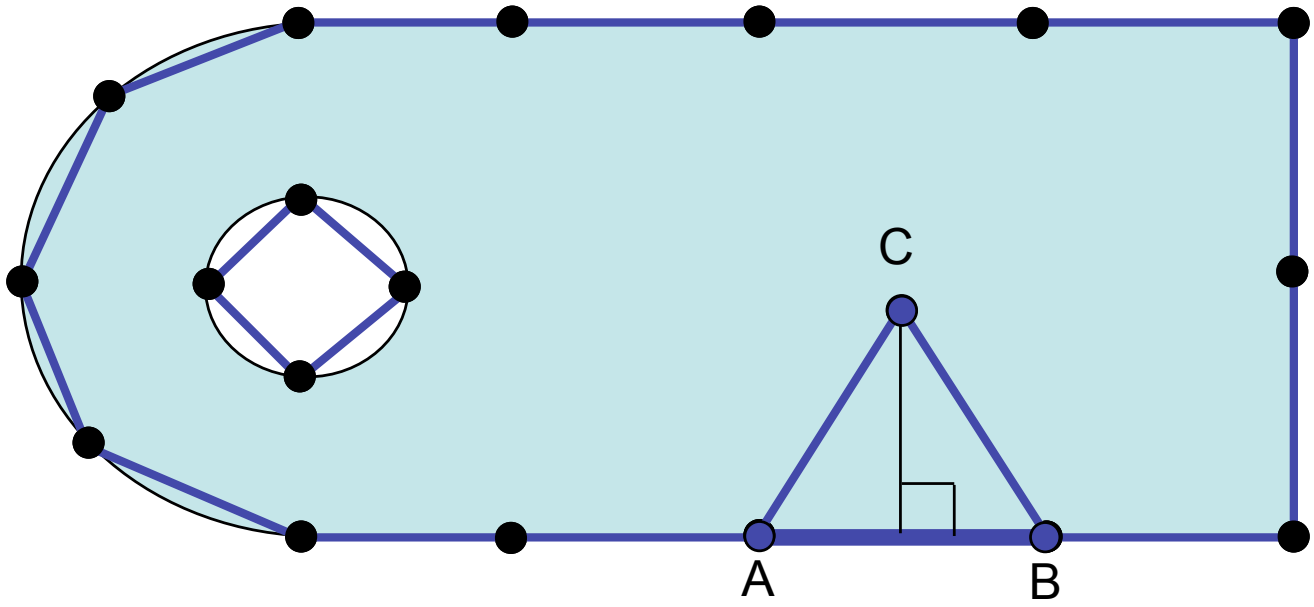




QMG,
Cornell University

Problèmes de ces méthodes
-Rapport de tailles entre éléments voisins
-Angles des éléments
→ Maillages assez laids

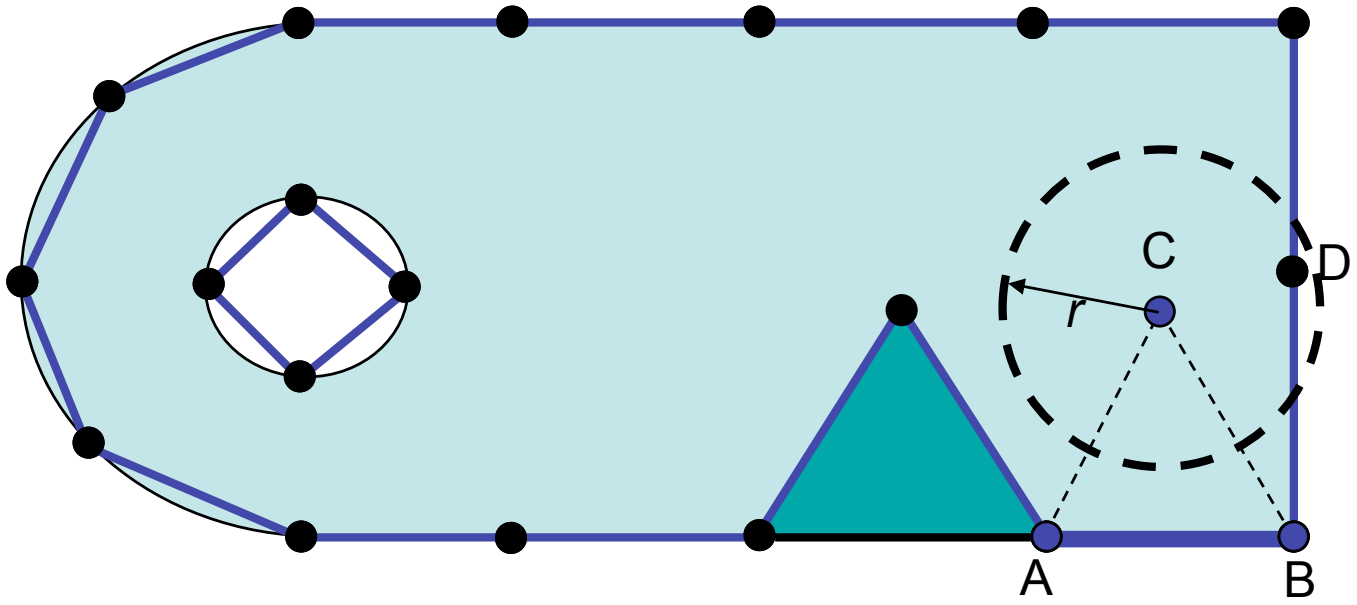
- Front initial – maillage du bord (nœuds + arêtes)
- Pour chaque arête [A,B] du front, on cherche la position idéal du point C
  - ⇨ ABC triangle équilatéral



Avancée de front 2D

Algorithme par avancée de front

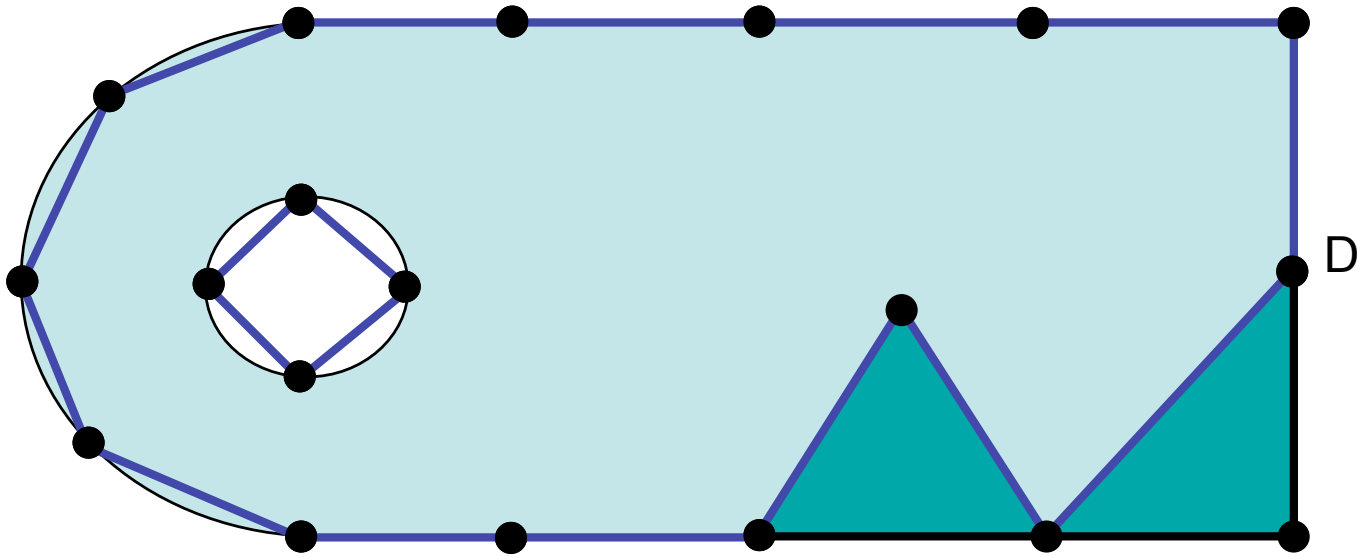
- On détermine si des nœuds du front courant ne seraient pas de candidats pour être le point C du triangle
  - Utilisation d'un paramètre distance r



  - On cherche un point du front existant dans le cercle de centre C idéal (i.e. ABC équilatéral)
    - Si oui, on prend ce point

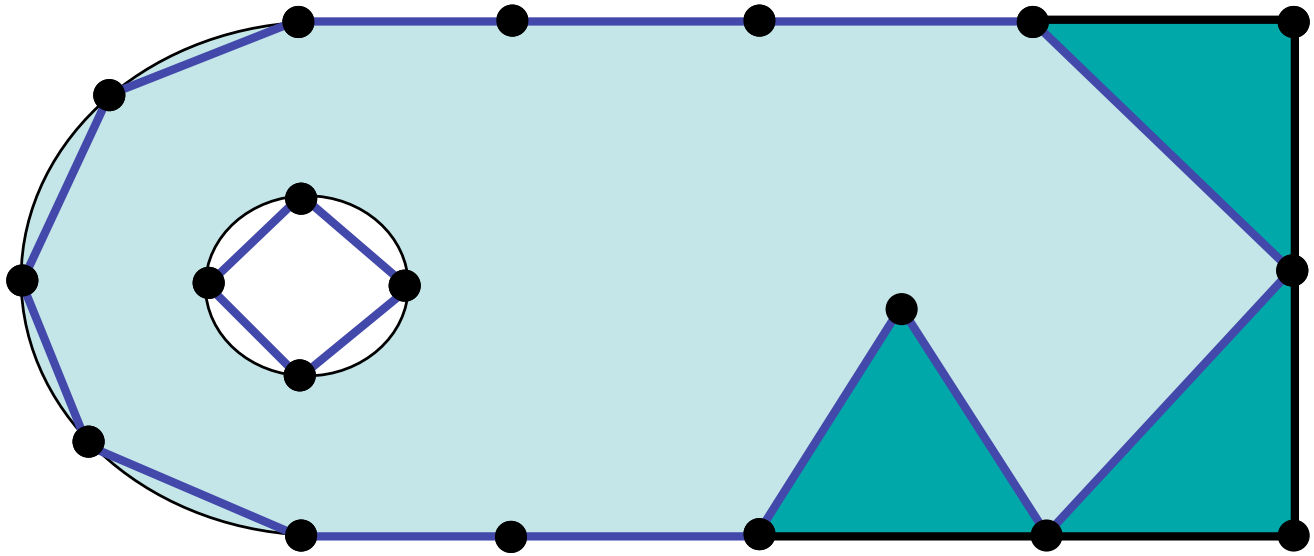
Algorithme par avancée de front

- Mise à jour permanente du front (qui est composée d'une liste d'arêtes)
- On continue tant qu'il reste des arêtes dans le front



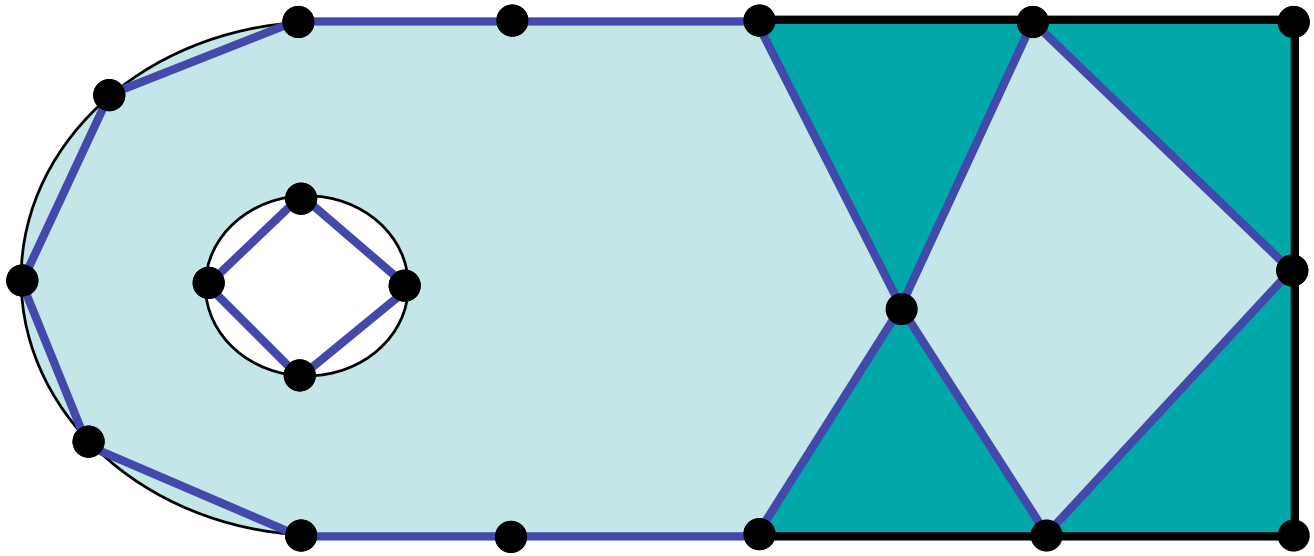
Algorithme par avancée de front

- Mise à jour permanente du front (qui est composée d'une liste d'arêtes)
- On continue tant qu'il reste des arêtes dans le front

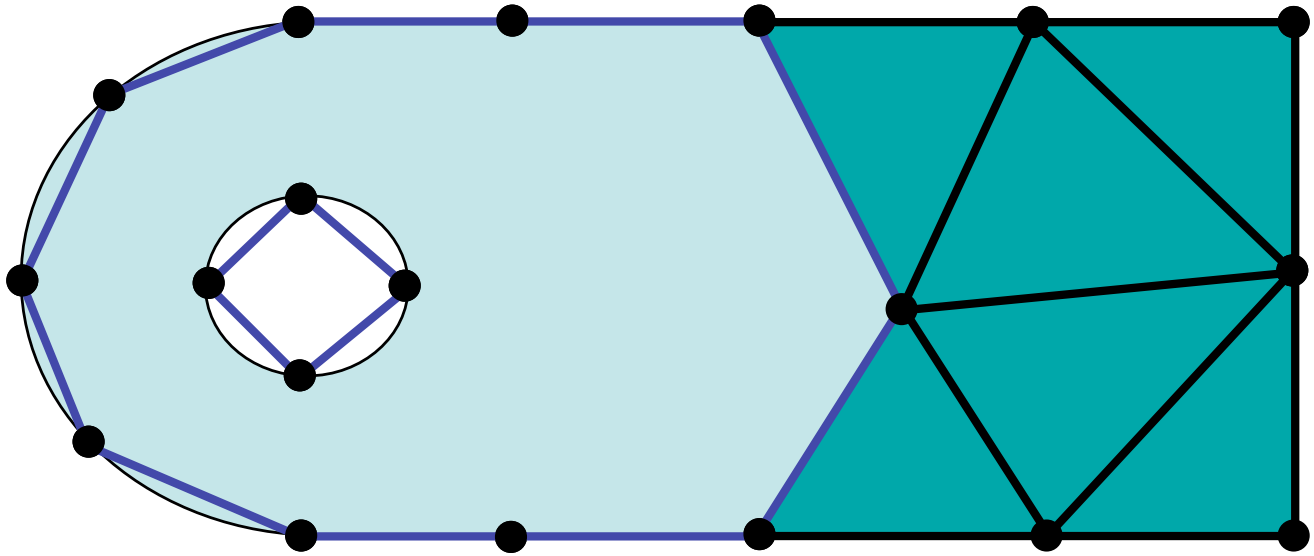


Algorithme par avancée de front

- Mise à jour permanente du front (qui est composée d'une liste d'arêtes)
- On continue tant qu'il reste des arêtes dans le front
- Attention, plusieurs fronts peuvent cohabiter



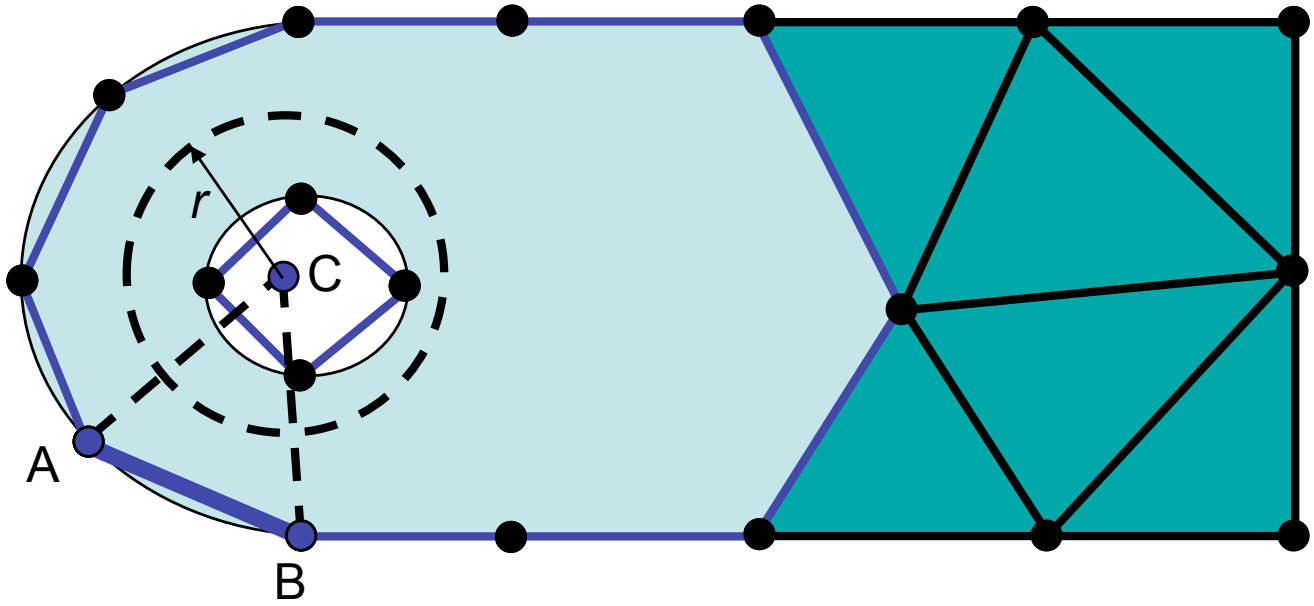
Algorithme par avancée de front



- Mise à jour permanente du front (qui est composée d'une liste d'arêtes)
- On continue tant qu'il reste des arêtes dans le front

Algorithme par avancée de front

- Si plusieurs choix pour le point C ?
  - On prend celui qui nous fournit le triangle le plus proche d'un triangle équilatéral



  - On supprime toutes les possibilités qui intersecteraient un front existant
  - On rejette les triangles inversées (I AB x AC I > 0)

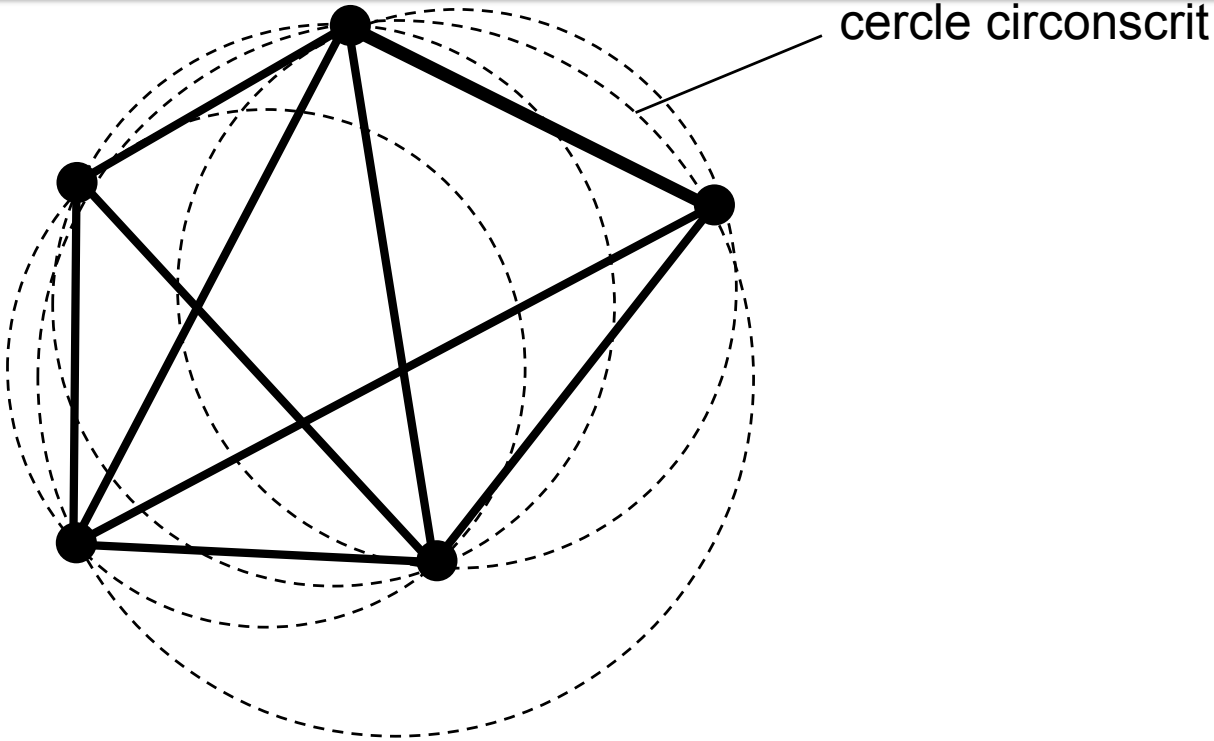
[Lohner,88;96] [Lo,91]

Avantages de ces méthodes
-Permet de contrôler l'aspect des triangles au bord

Algorithme de “type” Delaunay

- Algorithmes basés sur la propriété de cercle vide

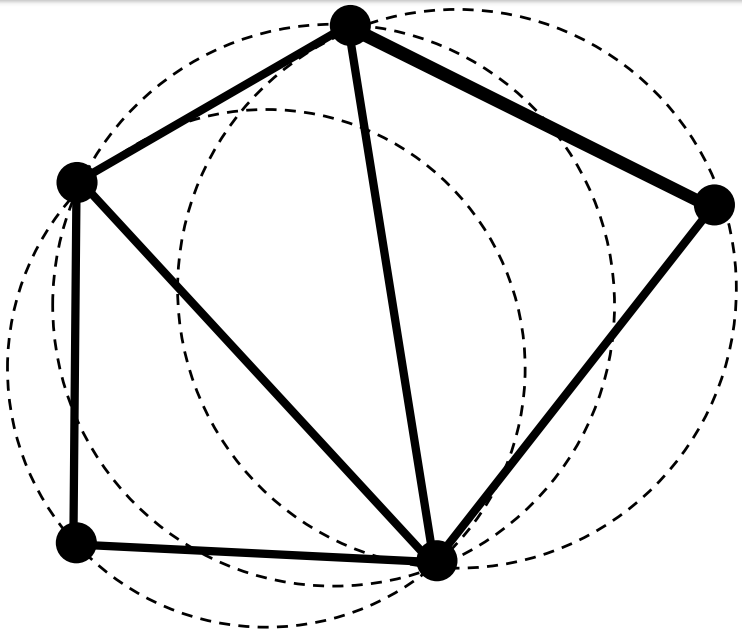
Pour tout triangle ABC, aucun sommet différent de A, B et C appartient au cercle circonscrit à ABC



Algorithme de “type” Delaunay

- Algorithmes basés sur la propriété de cercle vide

> Pour tout triangle ABC, aucun sommet différent de A, B et C appartient au cercle circonscrit à ABC



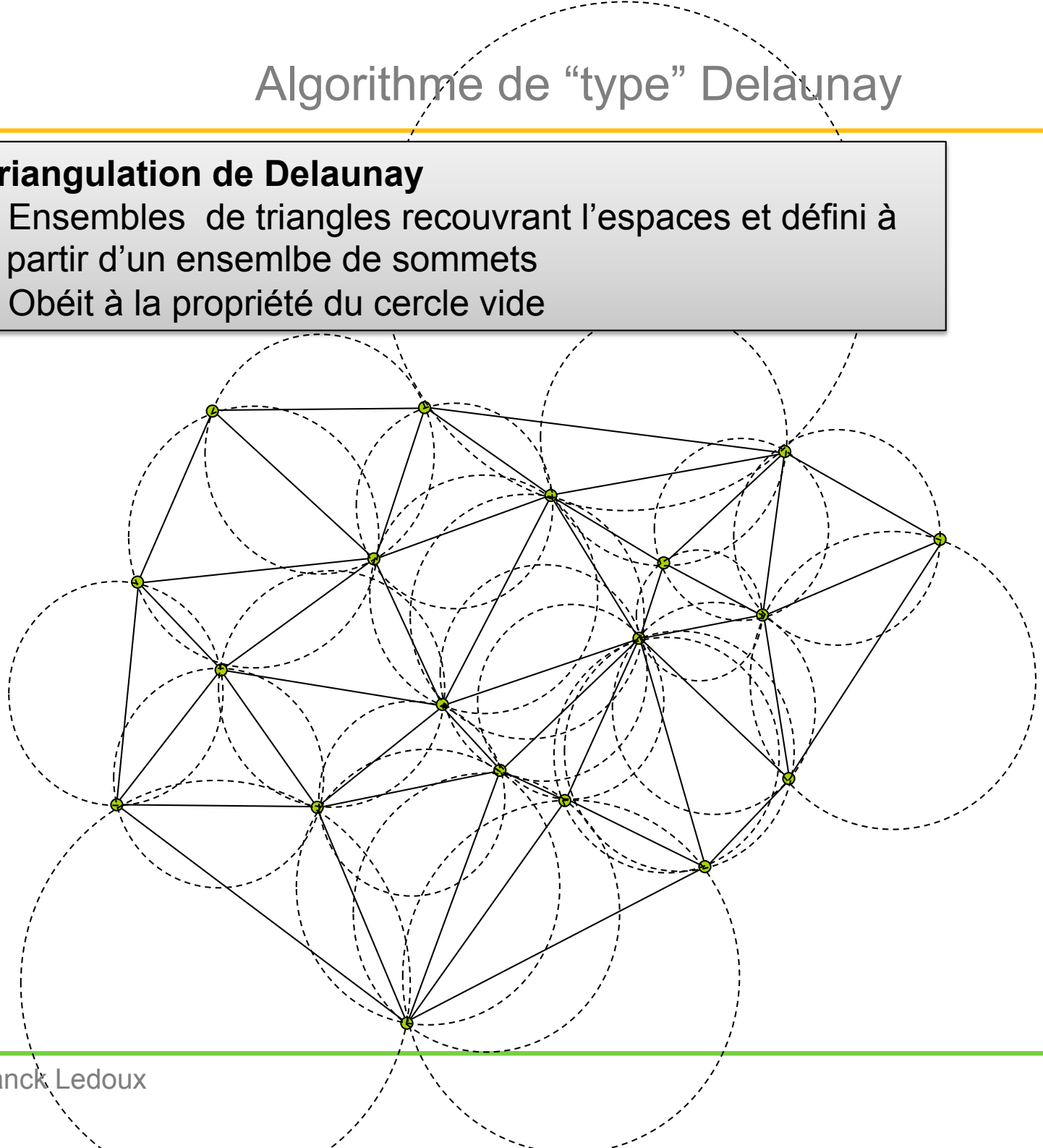
- Assure des mailles de bonne qualité
- Extensible en 3D

> Pour tout tétraèdre ABCD, aucun sommet différent de A, B, C et D appartient à la sphère circonscrite à ABCD

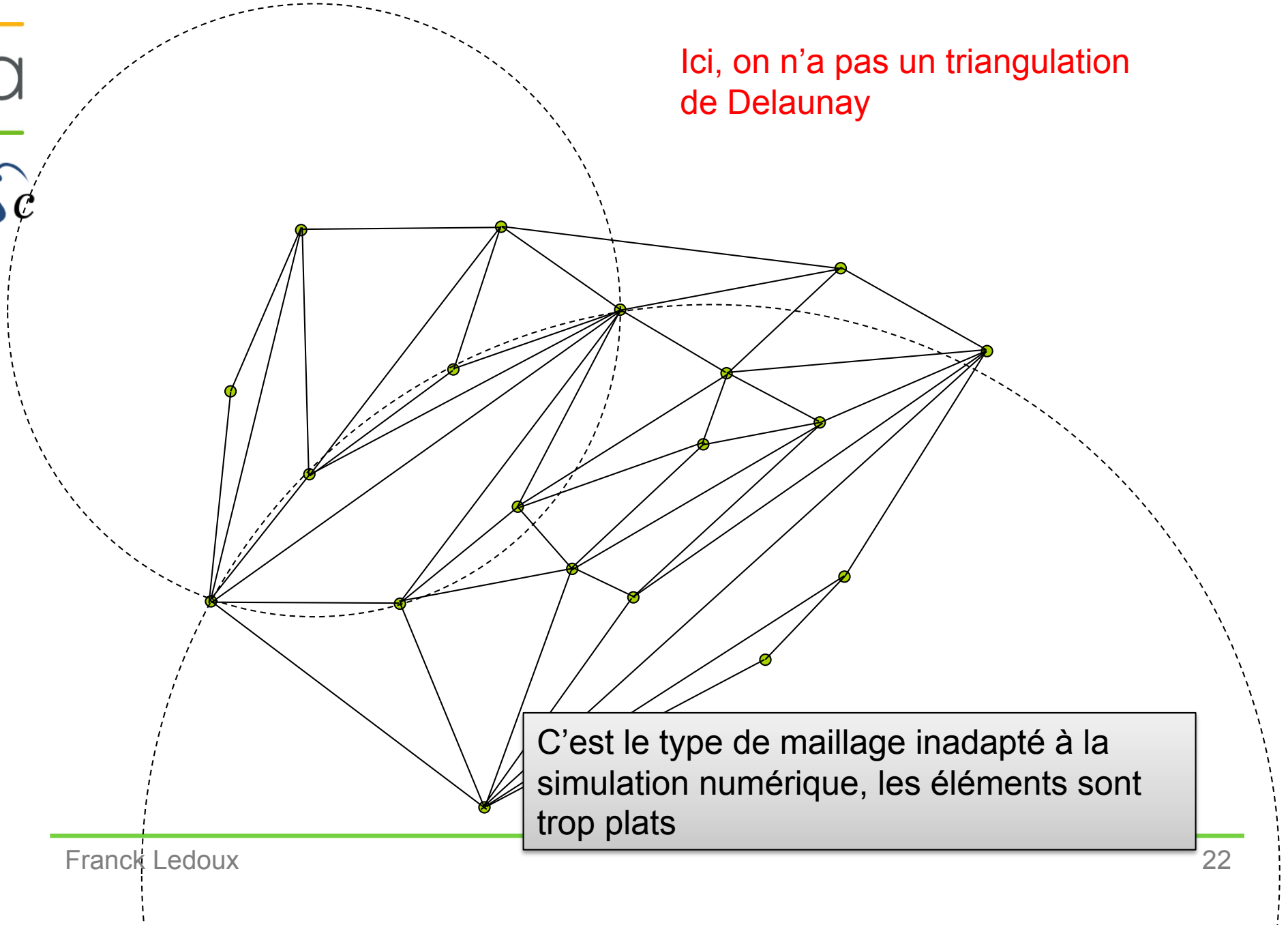
Algorithme de “type” Delaunay

Triangulation de Delaunay

- Ensembles de triangles recouvrant l’espaces et défini à partir d’un ensemlbe de sommets
- Obéit à la propriété du cercle vide



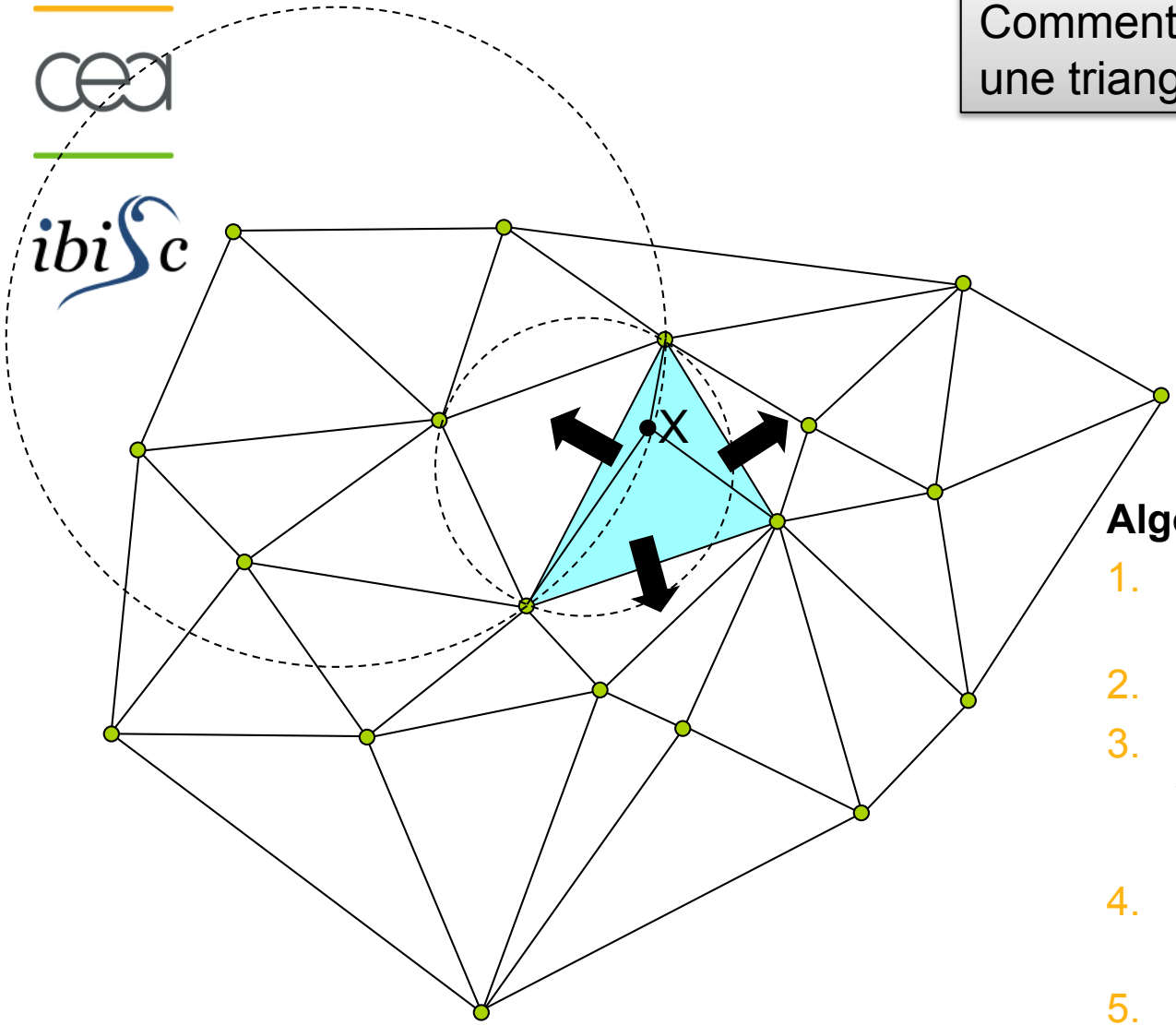
Algorithme de “type” Delaunay



Ici, on n’a pas un triangulation de Delaunay

C’est le type de maillage inadapté à la simulation numérique, les éléments sont trop plats

Algorithme de “type” Delaunay



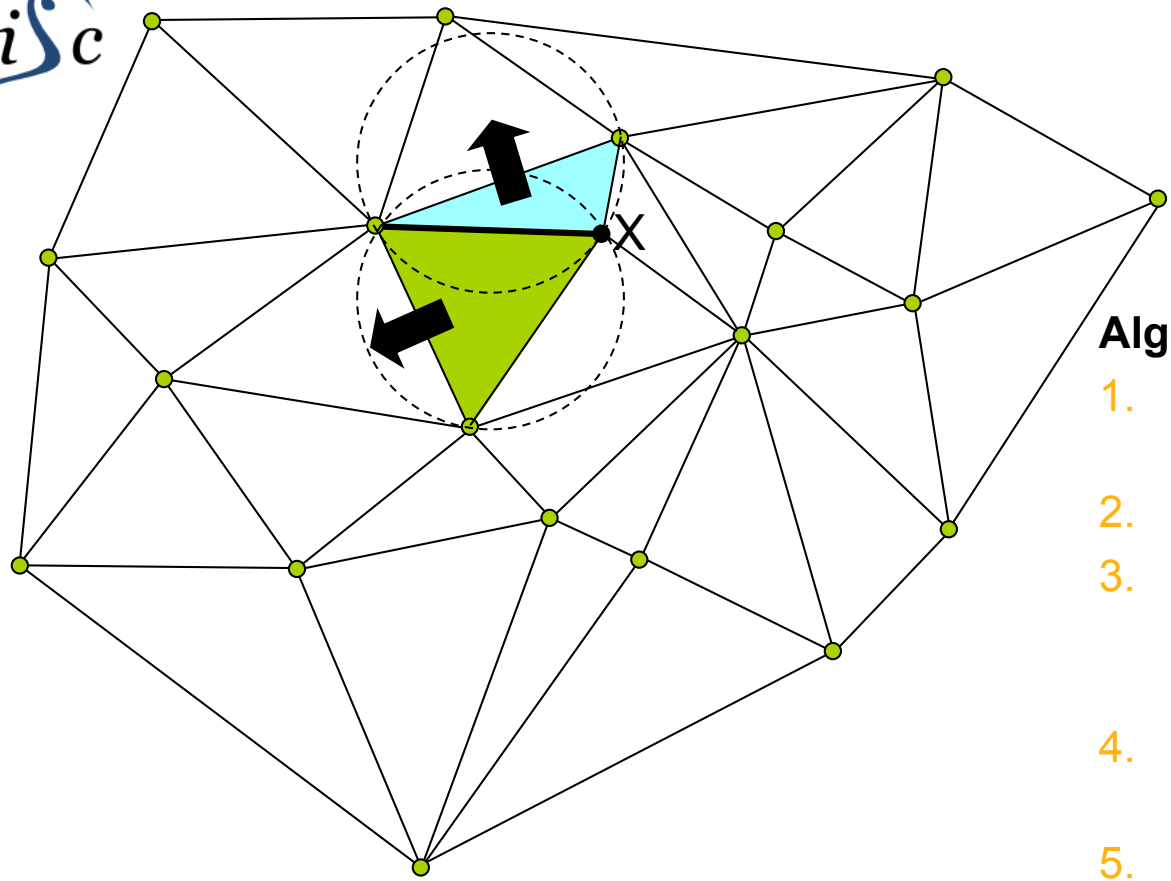
Comment ajouter un sommet dans une triangulation de Delaunay ?

Algorithme de Lawson (1977)

1. On détermine le triangle contenant X
2. On subdivise ce triangle en 3
3. On teste la propriété de sphère vide sur ces nouveaux triangles et les triangles adjacents
4. On effectue un swap d’arête si nécessaire
5. Les étapes 3 et 4 sont répétées tant qu’un swap est effectué

Algorithme de “type” Delaunay

Comment ajouter un sommet dans une triangulation de Delaunay ?

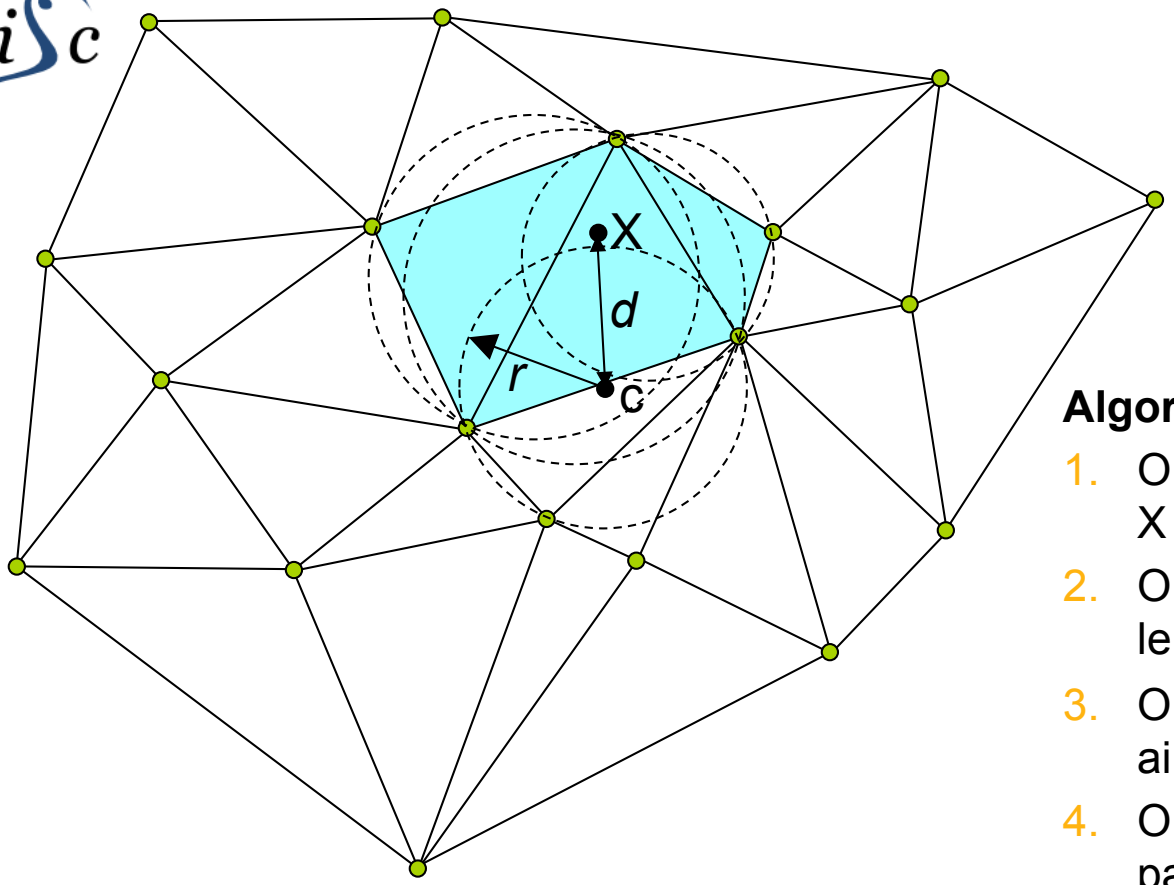


Algorithme de Lawson (1977)

1. On détermine le triangle contenant X
2. On subdivise ce triangle en 3
3. On teste la propriété de sphère vide sur ces nouveaux triangles et les triangles adjacents
4. On effectue un swap d’arête si nécessaire
5. Les étapes 3 et 4 sont répétées tant qu’un swap est effectué

Algorithme de “type” Delaunay

Comment ajouter un sommet dans une triangulation de Delaunay ?

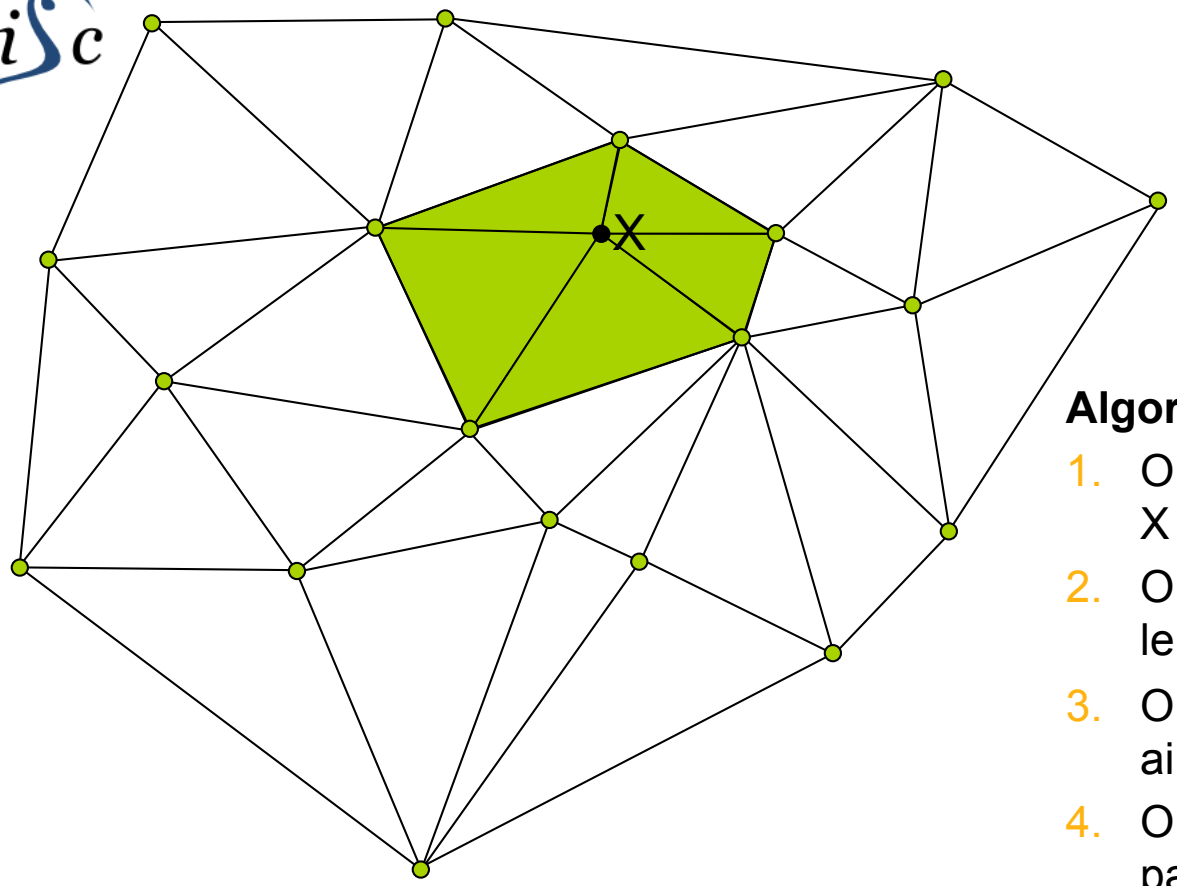


Algorithme de Bowyer-Watson (1981)

1. On détermine le triangle contenant X
2. On cherche tous les triangles dont le cercle circonscrit contient X
3. On supprime ces triangles formant ainsi une **cavité**
4. On crée de nouveaux triangles à partir des arêtes bordant cette cavité et du sommet X

Algorithme de “type” Delaunay

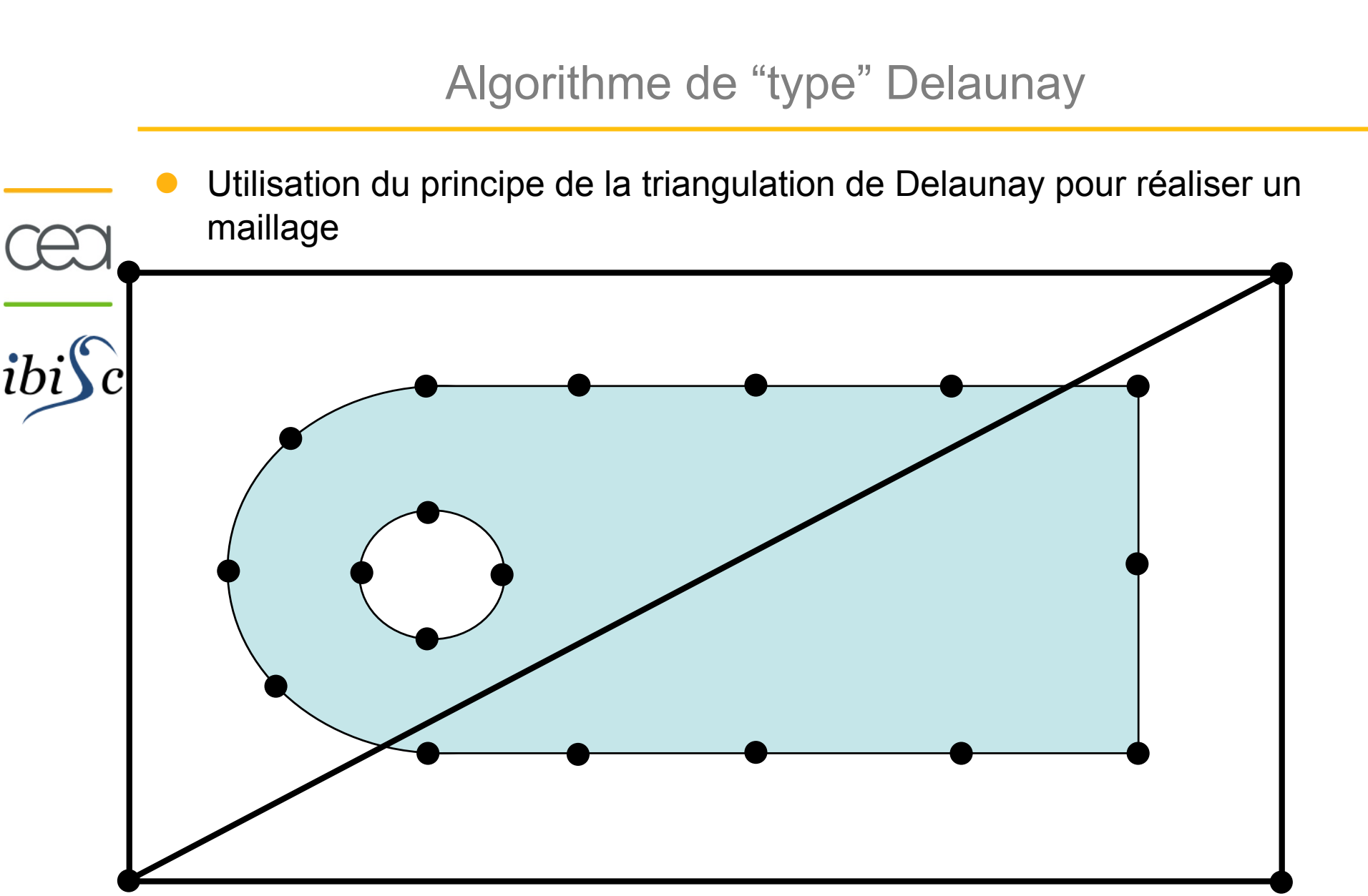
Comment ajouter un sommet dans une triangulation de Delaunay ?



Algorithme de Bowyer-Watson (1981)

1. On détermine le triangle contenant X
2. On cherche tous les triangles dont le cercle circonscrit contient X
3. On supprime ces triangles formant ainsi une **cavité**
4. On crée de nouveaux triangles à partir des arêtes bordant cette cavité et du sommet X

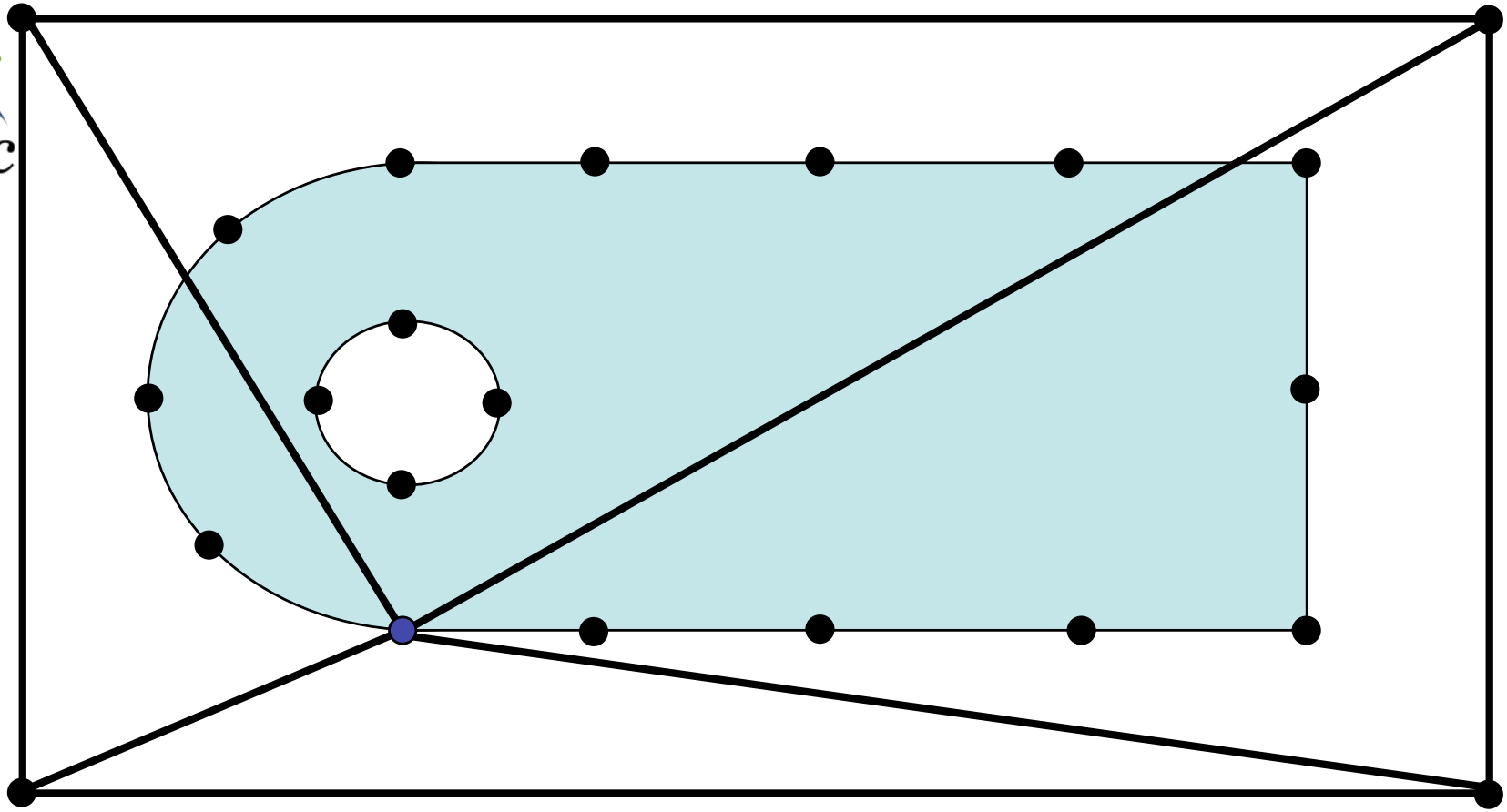
Algorithme de “type” Delaunay



- Utilisation du principe de la triangulation de Delaunay pour réaliser un maillage
- On commence avec une triangulation d’une boîte englobante élargie

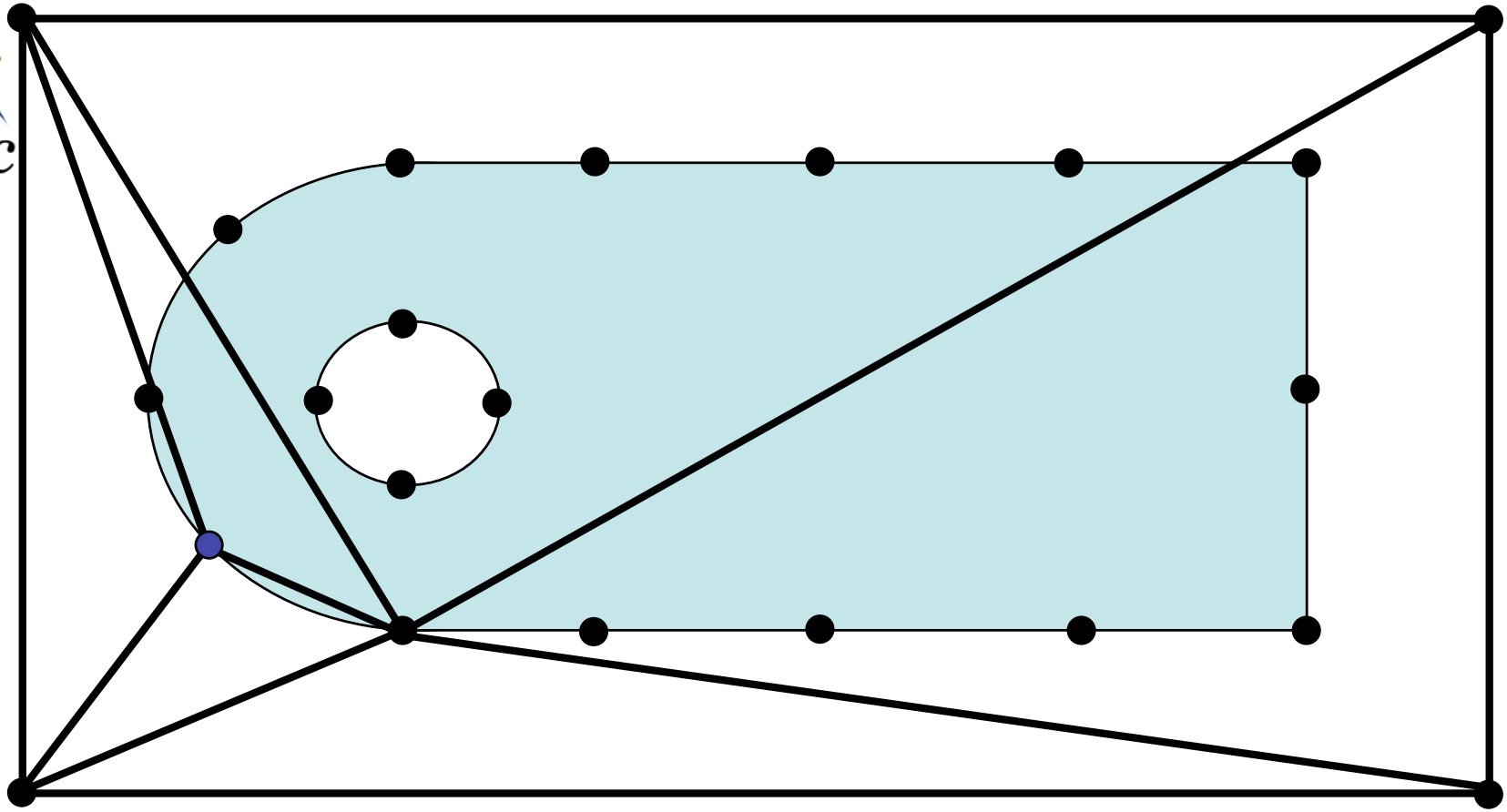
Algorithme de “type” Delaunay

- On insère les nœuds au bord du domaine à mailler avec un algorithme d’insertion tel que celui de Lawson ou de Bowyer-Watson



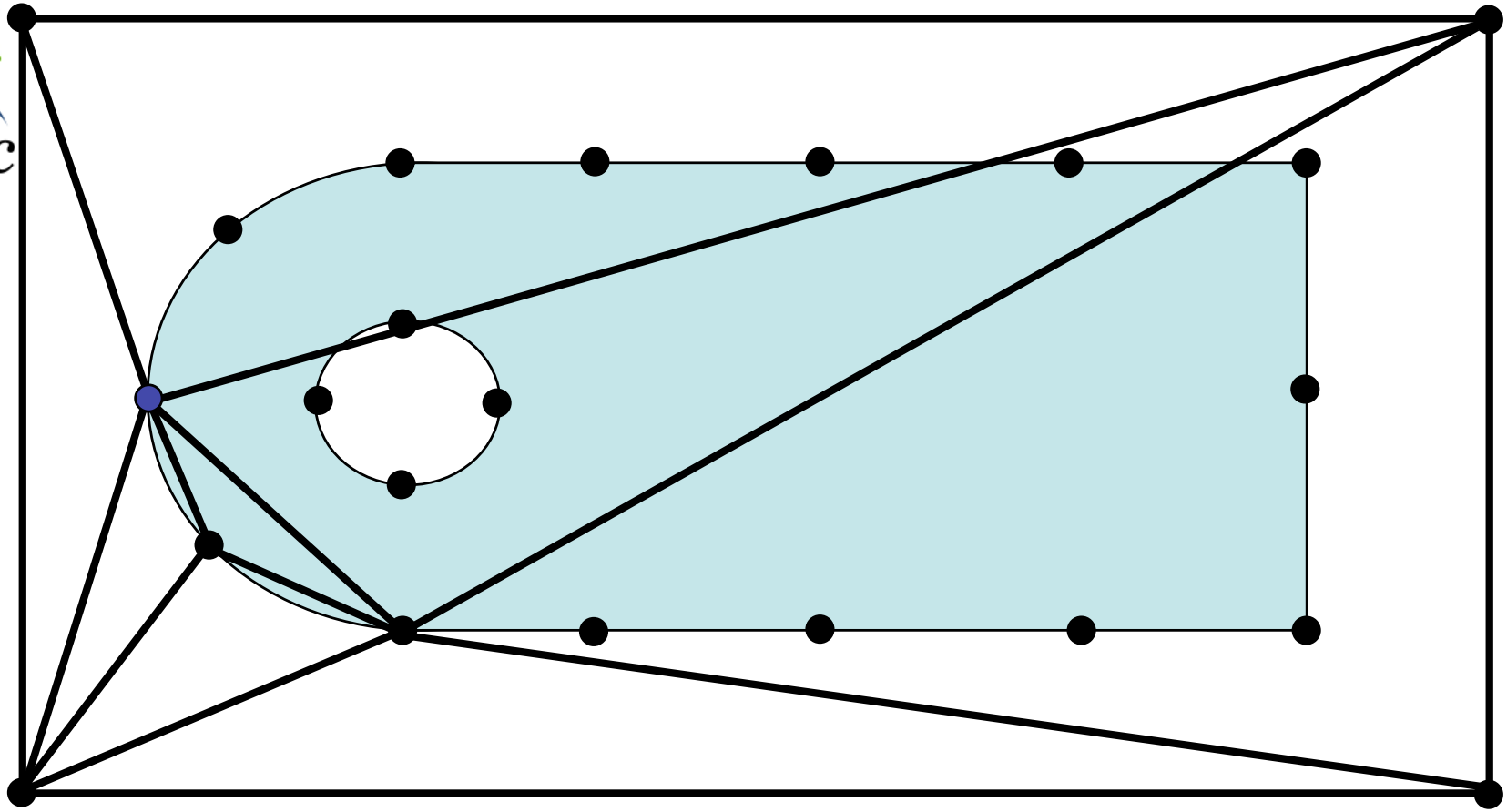
Algorithme de “type” Delaunay

- On insère les nœuds au bord du domaine à mailler avec un algorithme d’insertion tel que celui de Lawson ou de Bowyer-Watson



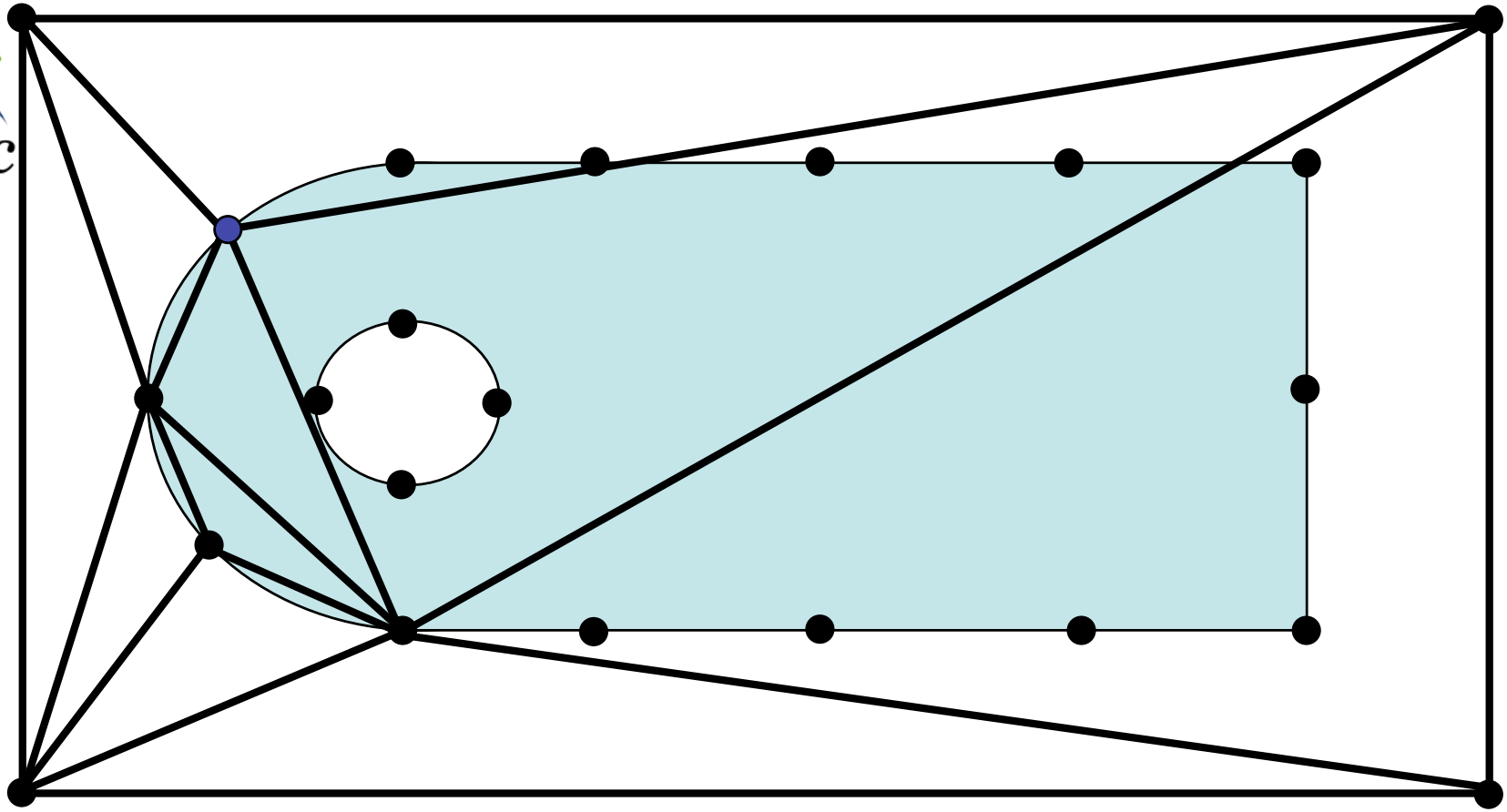
Algorithme de “type” Delaunay

- On insère les nœuds au bord du domaine à mailler avec un algorithme d’insertion tel que celui de Lawson ou de Bowyer-Watson



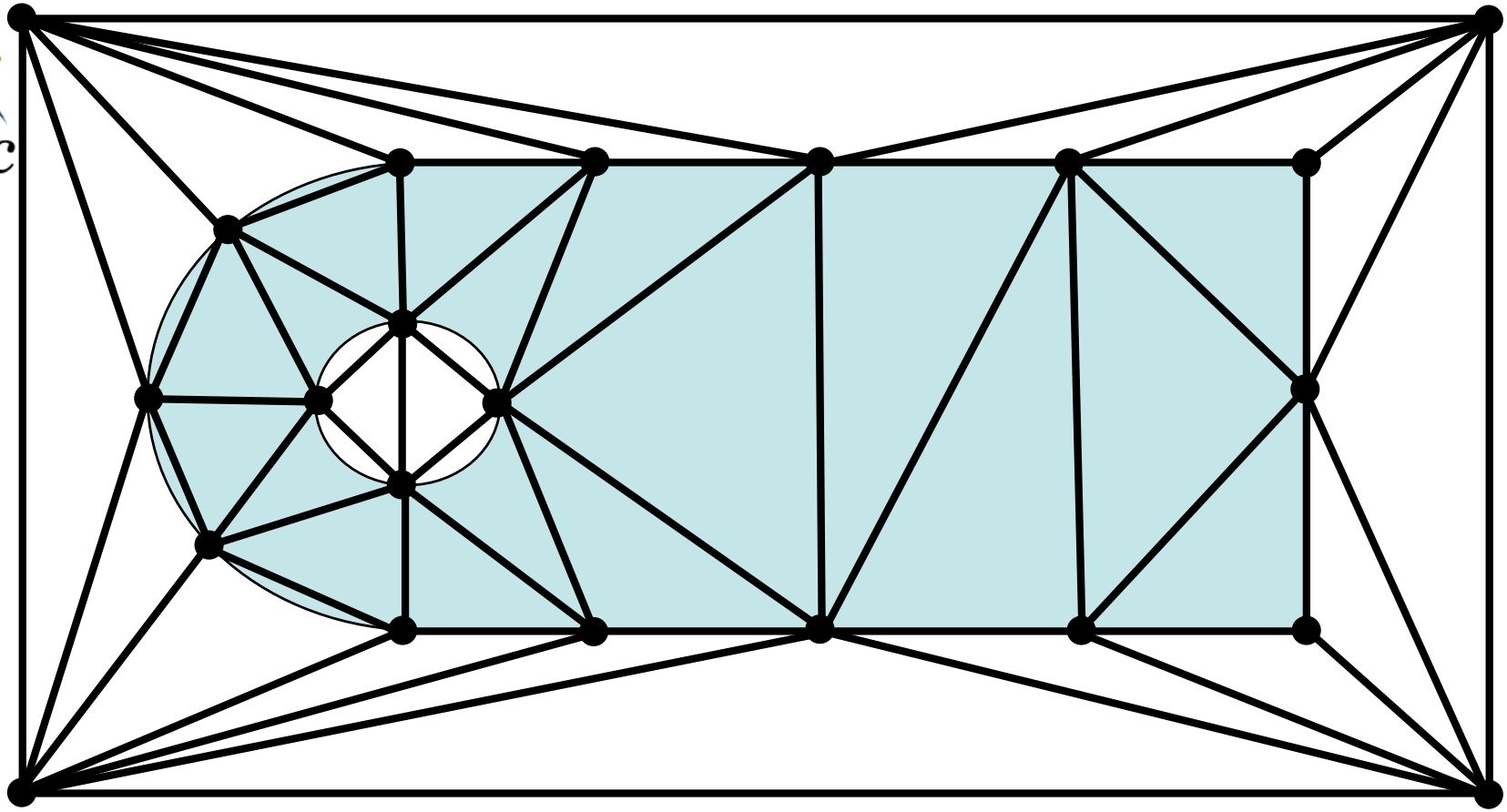
Algorithme de “type” Delaunay

- On insère les nœuds au bord du domaine à mailler avec un algorithme d’insertion tel que celui de Lawson ou de Bowyer-Watson



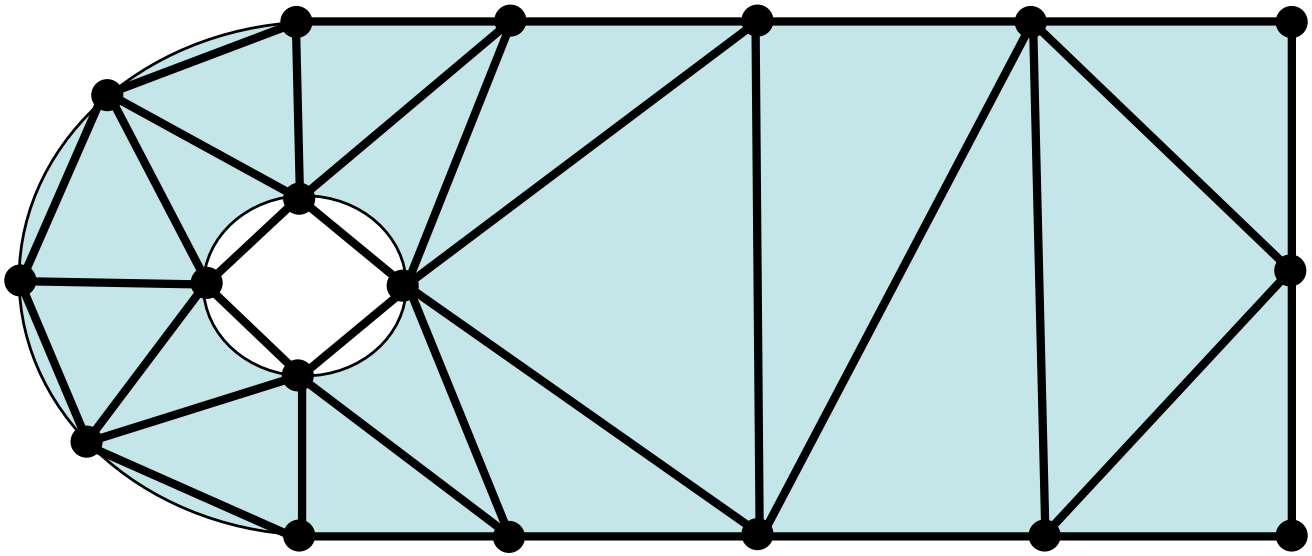
Algorithme de “type” Delaunay

- On insère les nœuds au bord du domaine à mailler avec un algorithme d’insertion tel que celui de Lawson ou de Bowyer-Watson



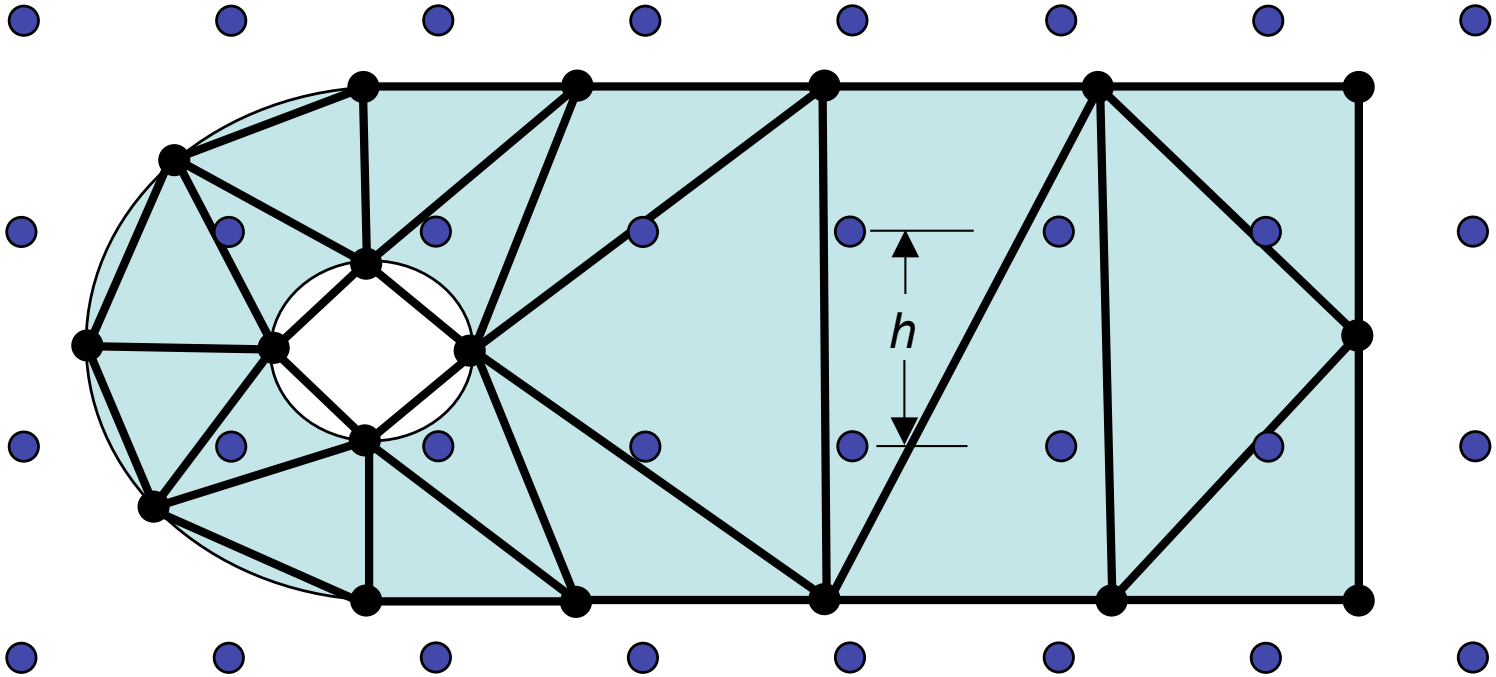
Algorithme de “type” Delaunay

- On récupère le bord du domaine si on ne l’a pas
- On supprime les triangles extérieurs
- On insert des nœuds internes selon les paramètres du maillage souhaité (densité de mailles dans un zone par exemple)



Algorithme de “type” Delaunay

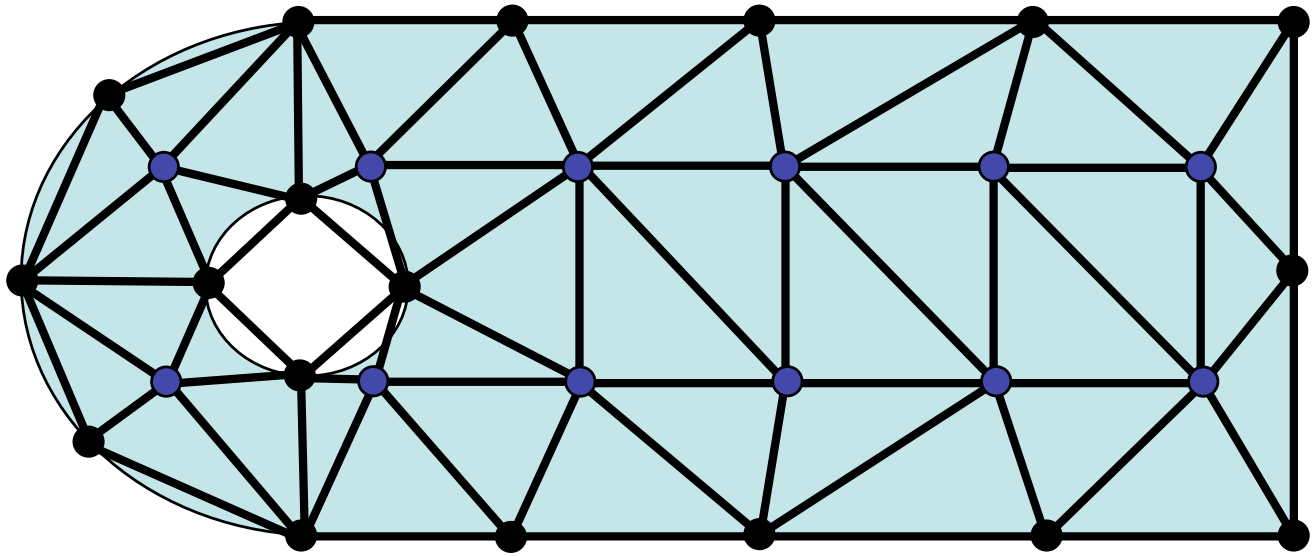
- Insertion de nœuds internes **à partir de la donnée d’une « grille »**



  - La grille peut être quelconques (rectangles de tailles différentes, triangles, quadtree)
  - On ignore les nœuds externes

Algorithme de “type” Delaunay

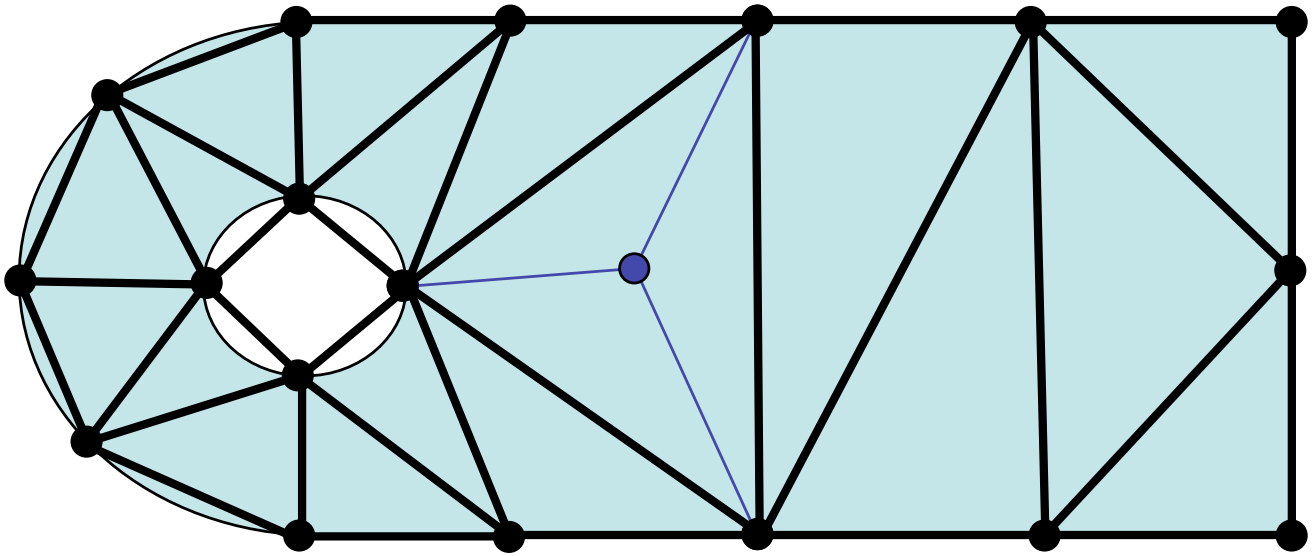
- Insertion de nœuds internes **à partir de la donnée d’une « grille »**



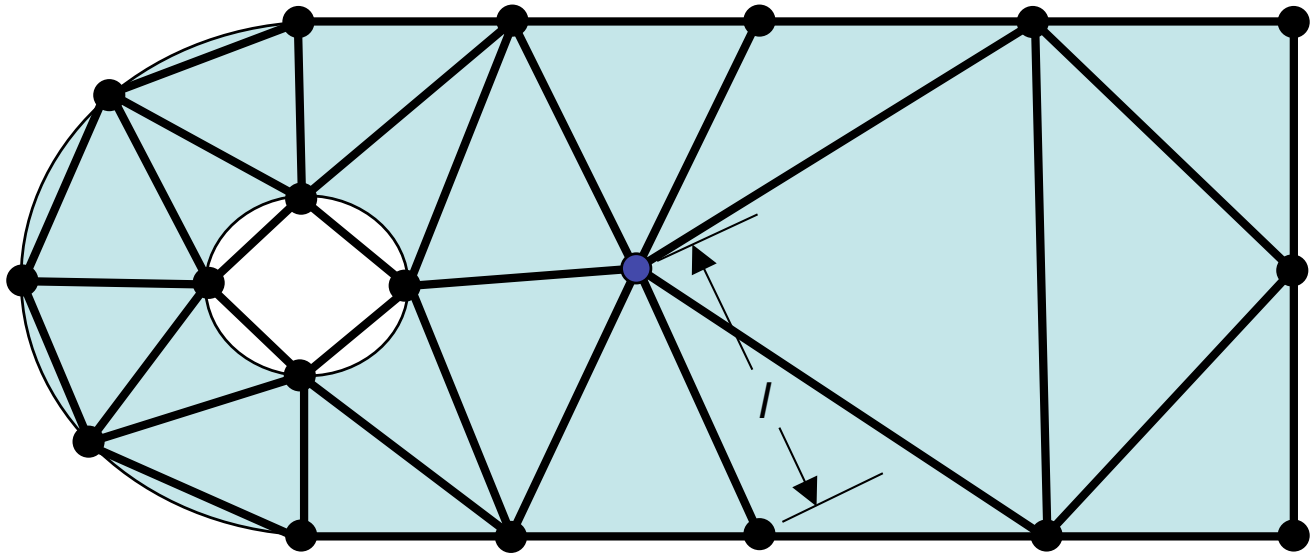
  - La grille peut être quelconques (rectangles de tailles différentes, triangles, quadtree)
  - On ignore les nœuds externes

Algorithme de “type” Delaunay

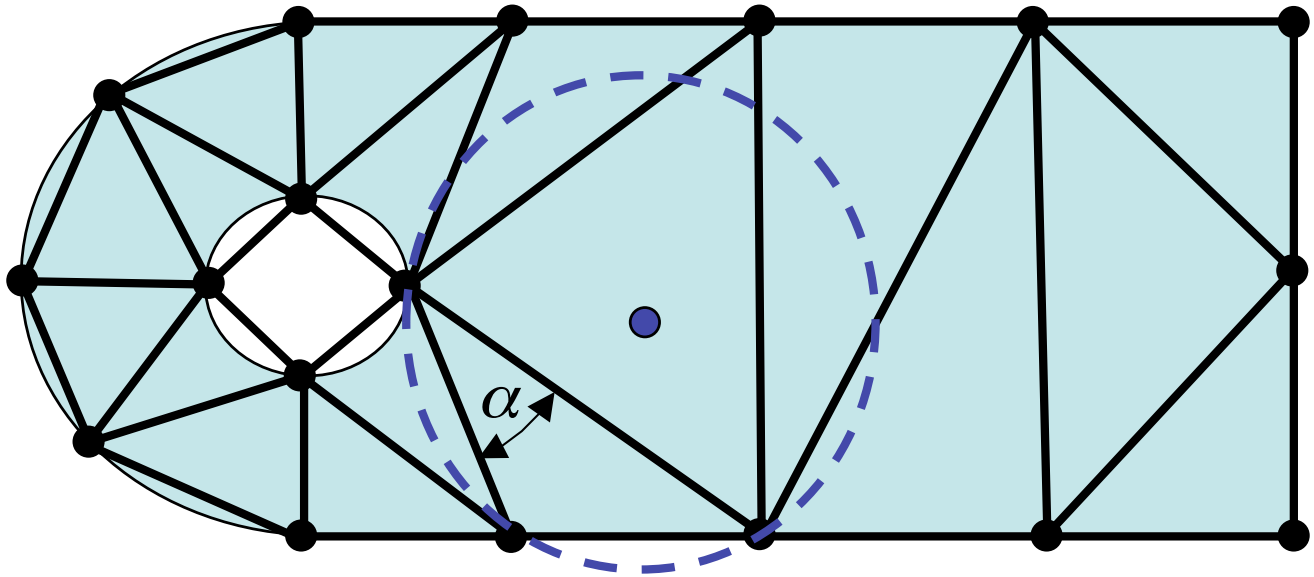
- Insertion de nœuds internes **comme centroïdes**
  - Les nouveaux nœuds sont définis comme centroïdes des triangles existants
  - On continue tant que les arêtes n’ont pas une longueur indiquée en paramètre



- Insertion de nœuds internes **comme centroïdes**
  - Les nouveaux nœuds sont définis comme centroïdes des triangles existants
  - On continue tant que les arêtes n’ont pas une longueur indiquée en paramètre



- Insertion de nœuds internes **à l’aide de cercles circonscrits**
  - Les nouveaux nœuds sont les centres des cercles circonscrits aux triangles existants

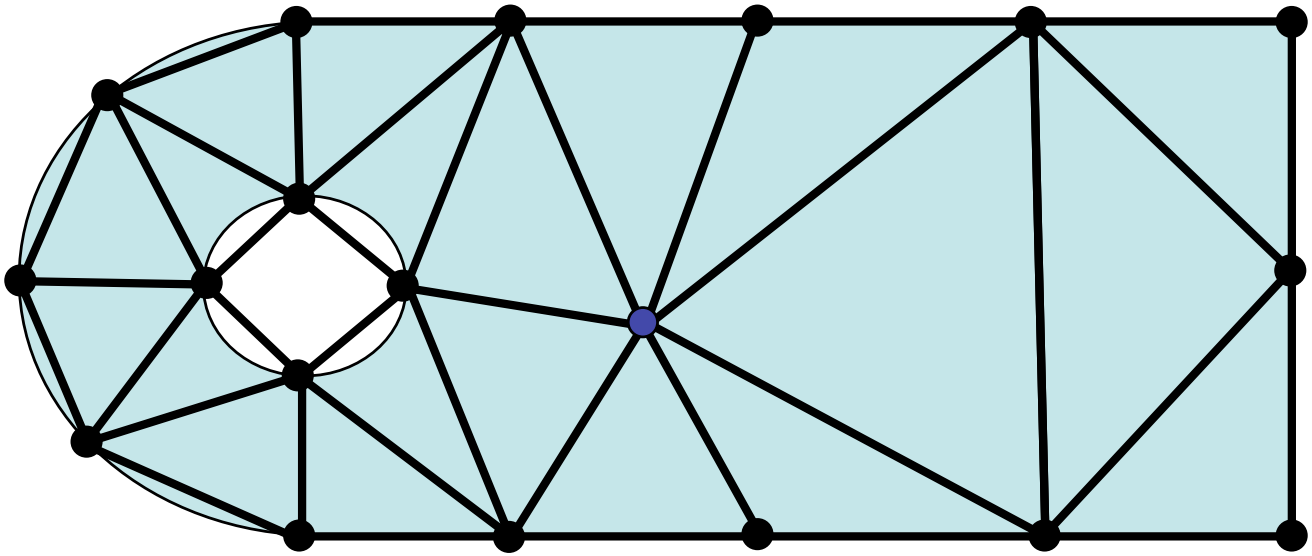


  - L’ordre d’insertion se base sur le plus petit angle au coin des triangles
  - Objectif (environ 30°)

(Chew,Ruppert,Shewchuk)

Algorithme de “type” Delaunay

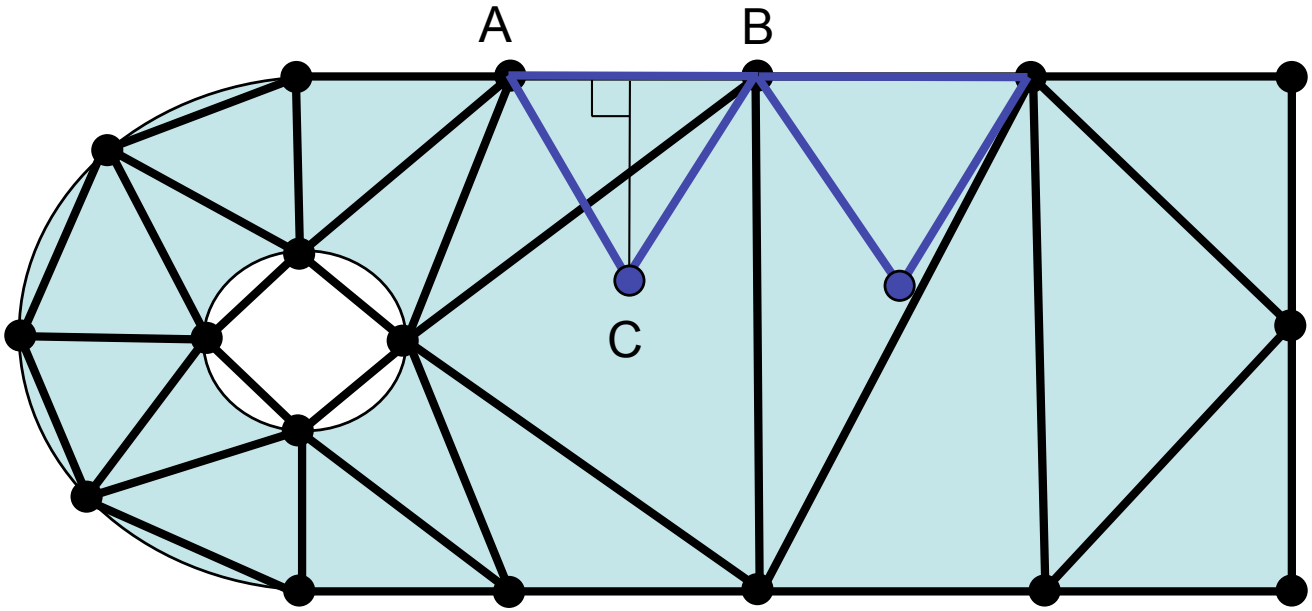
- Insertion de nœuds internes **à l’aide de cercles circonscrits**
  - Les nouveaux nœuds sont les centres des cercles circonscrits aux triangles existants



  - L’ordre d’insertion se base sur le plus petit angle au coin des triangles
  - Objectif (environ 30°)

(Chew,Ruppert,Shewchuk)

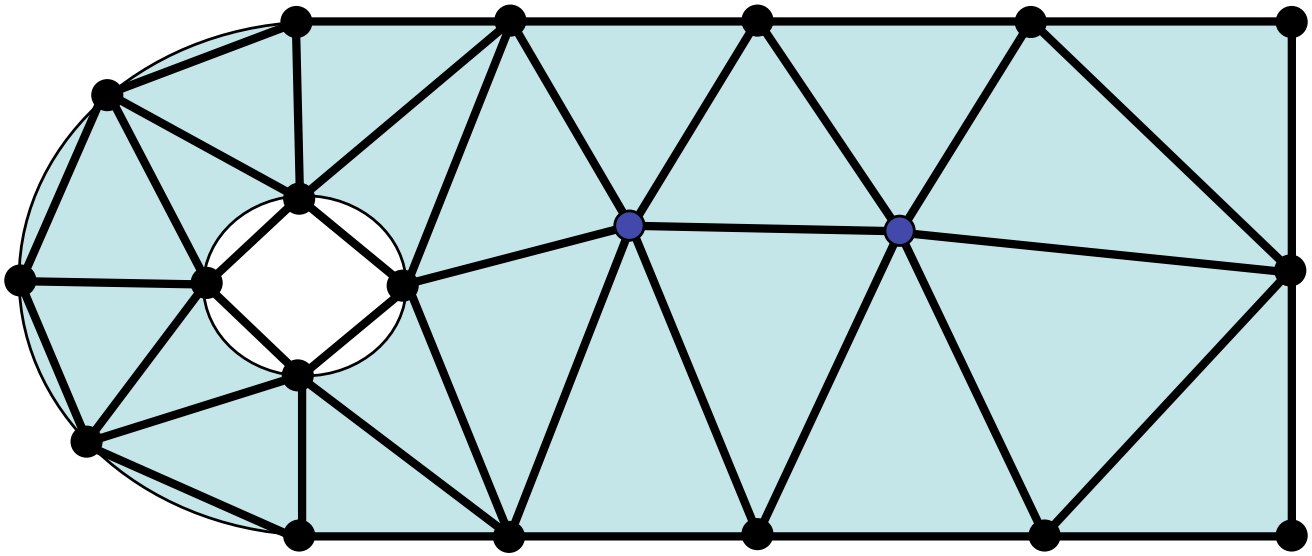
- Insertion de nœuds internes **par avancée de front**
  - Les nouveaux nœuds sont insérés à la position idéale à partir des arêtes du front



  - La structure du front doit être maintenue en permanence

[Marcum, 95]

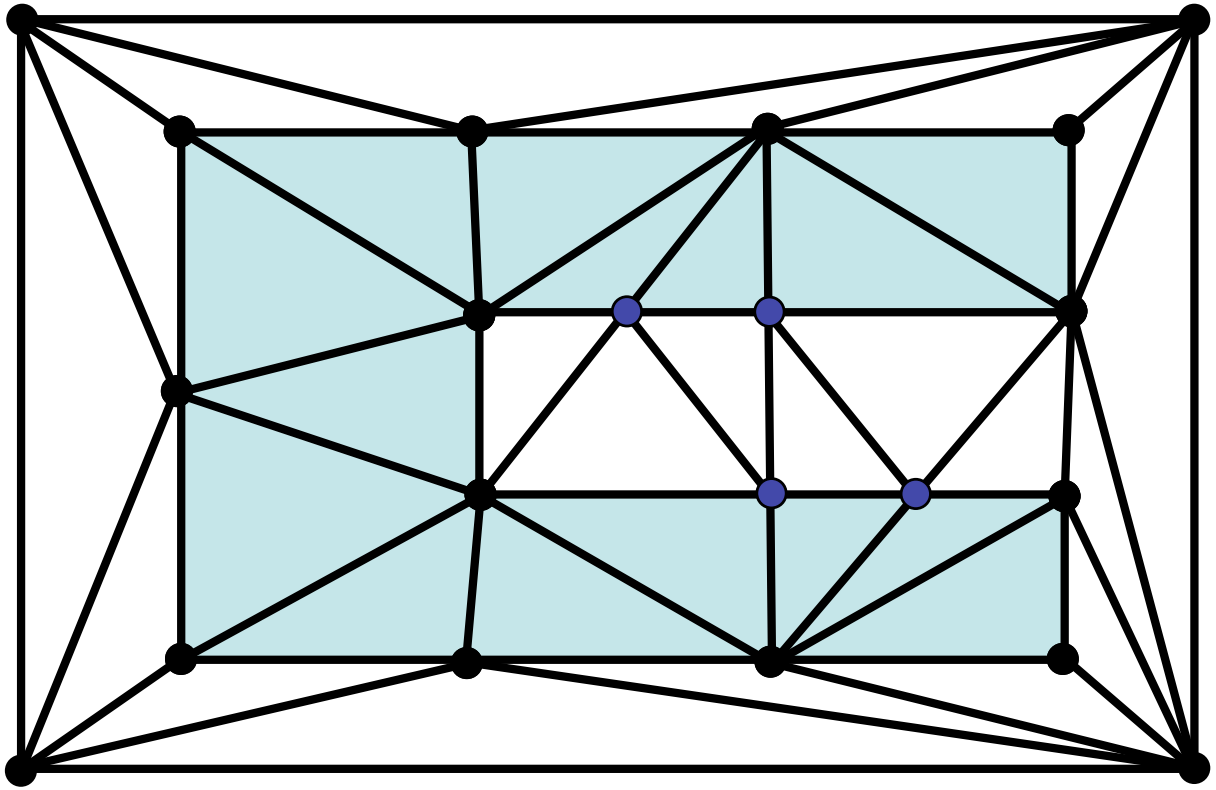
- Insertion de nœuds internes **par avancée de front**
  - Les nouveaux nœuds sont insérés à la position idéale à partir des arêtes du front



  - La structure du front doit être maintenue en permanence

[Marcum, 95]

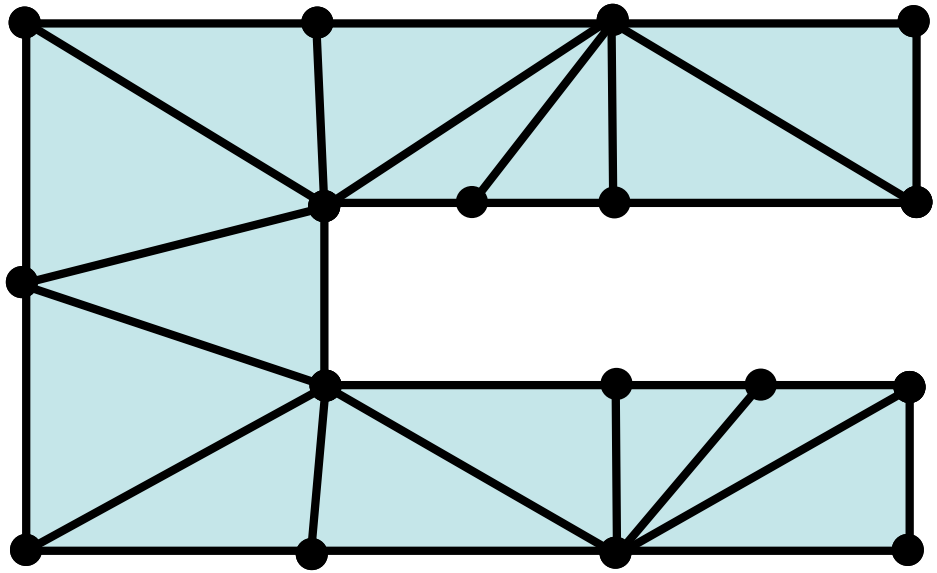
- On récupère l**e bord du domaine** si on ne l’a pas



  - Il faut insérer des nœuds et des arêtes là où les arêtes, introduites lors de la triangulation initiale, coupent le bord du domaine géométrique
  - **Ou faire du swapping d’arêtes pour récupérer les arêtes du bord géométrique**

Algorithme de “type” Delaunay

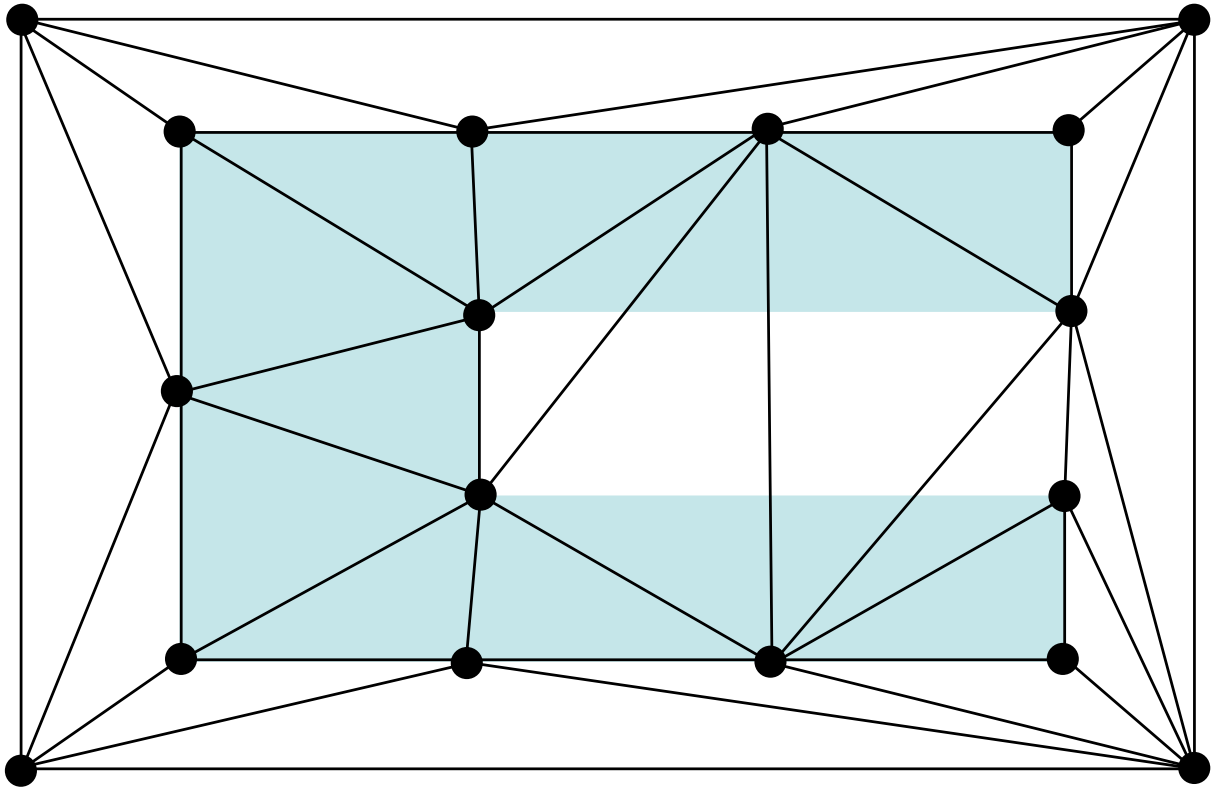
- On récupère l**e bord du domaine** si on ne l’a pas



  - Il faut insérer des nœuds et des arêtes là où les arêtes, introduites lors de la triangulation initiale, coupent le bord du domaine géométrique
  - **Ou faire du swapping d’arêtes pour récupérer les arêtes du bord géométrique**

Algorithme de “type” Delaunay

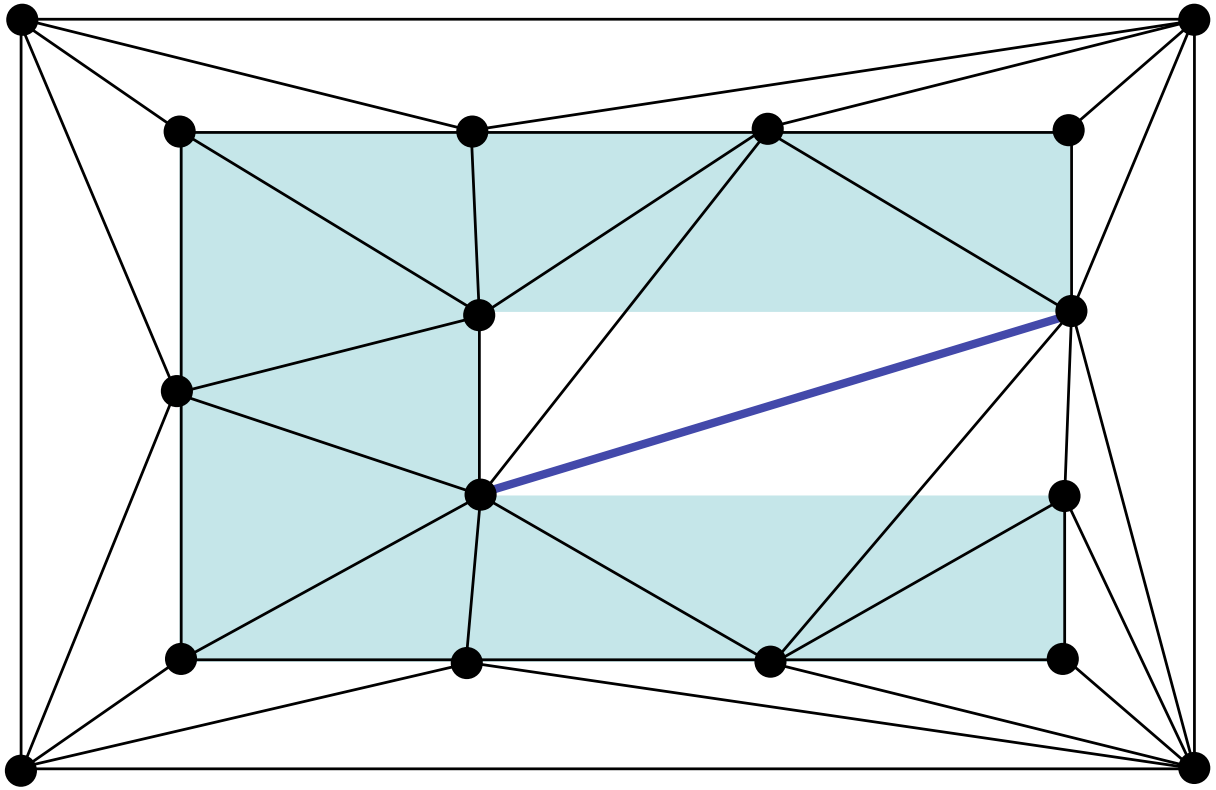
- On récupère l**e bord du domaine** si on ne l’a pas



  - Il faut insérer des nœuds et des arêtes là où les arêtes, introduites lors de la triangulation initiale, coupent le bord du domaine géométrique
  - **Ou faire du swapping d’arêtes pour récupérer les arêtes du bord géométrique**

Algorithme de “type” Delaunay

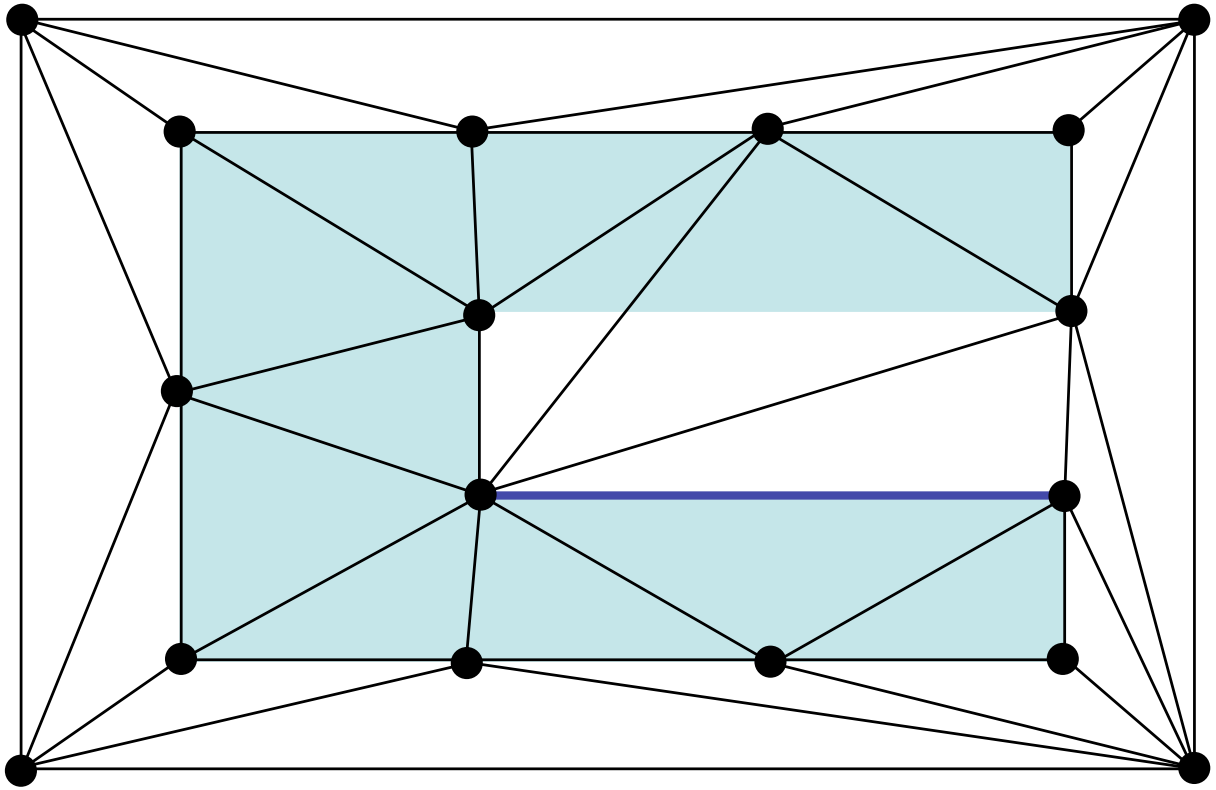
- On récupère l**e bord du domaine** si on ne l’a pas



  - Il faut insérer des nœuds et des arêtes là où les arêtes, introduites lors de la triangulation initiale, coupent le bord du domaine géométrique
  - **Ou faire du swapping d’arêtes pour récupérer les arêtes du bord géométrique**

Algorithme de “type” Delaunay

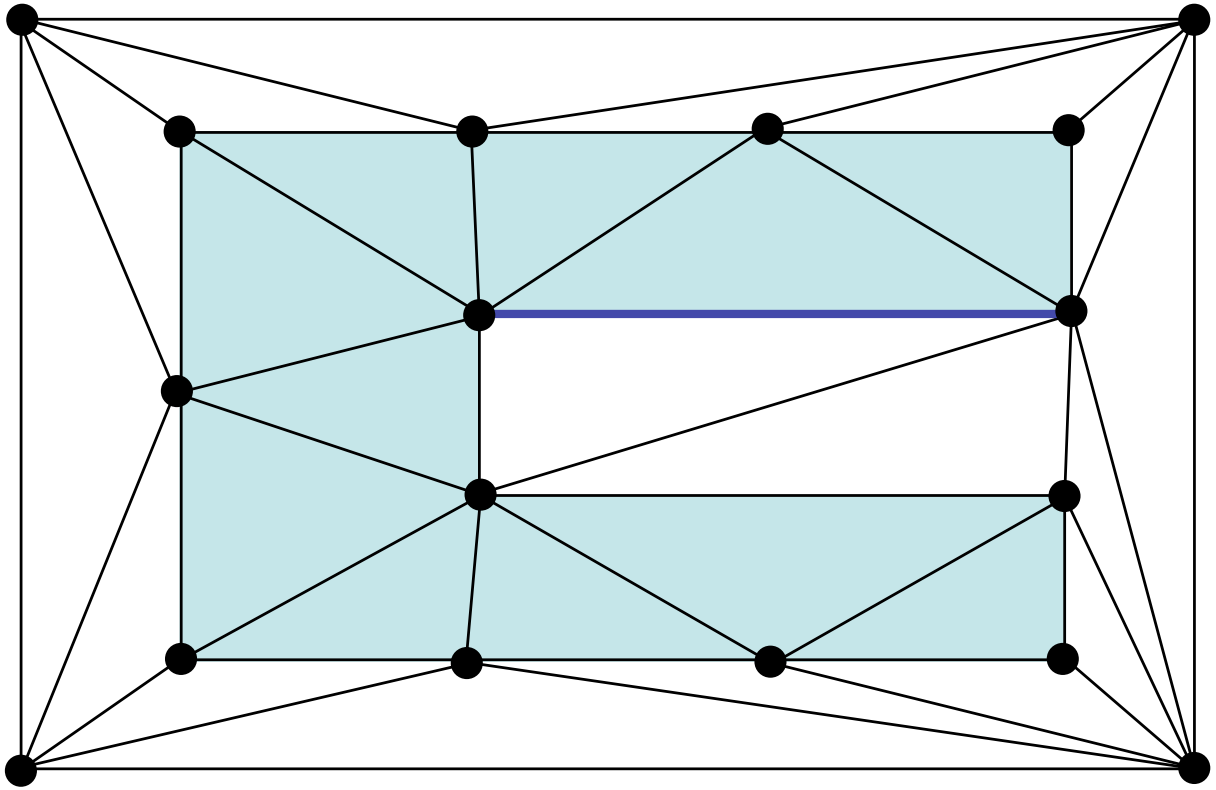
- On récupère l**e bord du domaine** si on ne l’a pas



  - Il faut insérer des nœuds et des arêtes là où les arêtes, introduites lors de la triangulation initiale, coupent le bord du domaine géométrique
  - **Ou faire du swapping d’arêtes pour récupérer les arêtes du bord géométrique**

Algorithme de “type” Delaunay

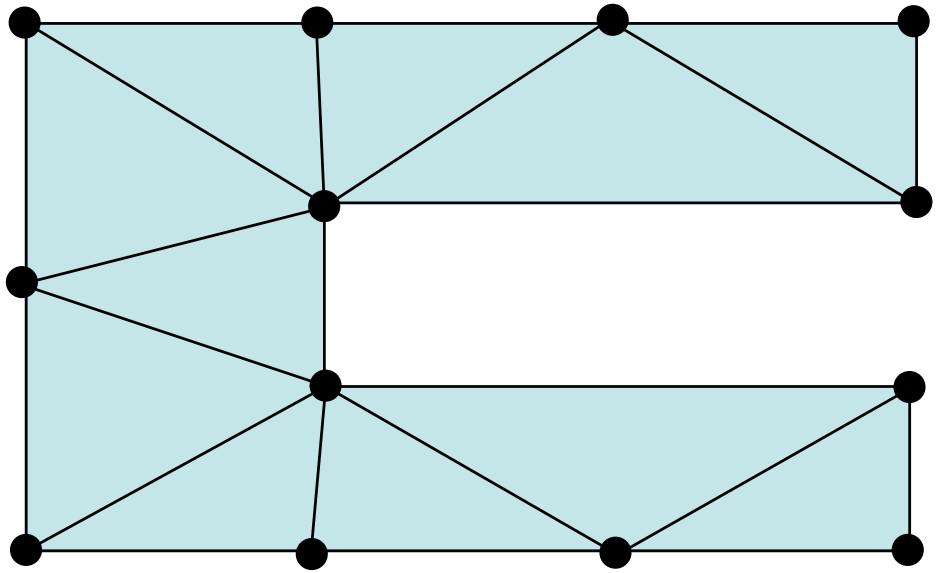
- On récupère l**e bord du domaine** si on ne l’a pas



  - Il faut insérer des nœuds et des arêtes là où les arêtes, introduites lors de la triangulation initiale, coupent le bord du domaine géométrique
  - **Ou faire du swapping d’arêtes pour récupérer les arêtes du bord géométrique**

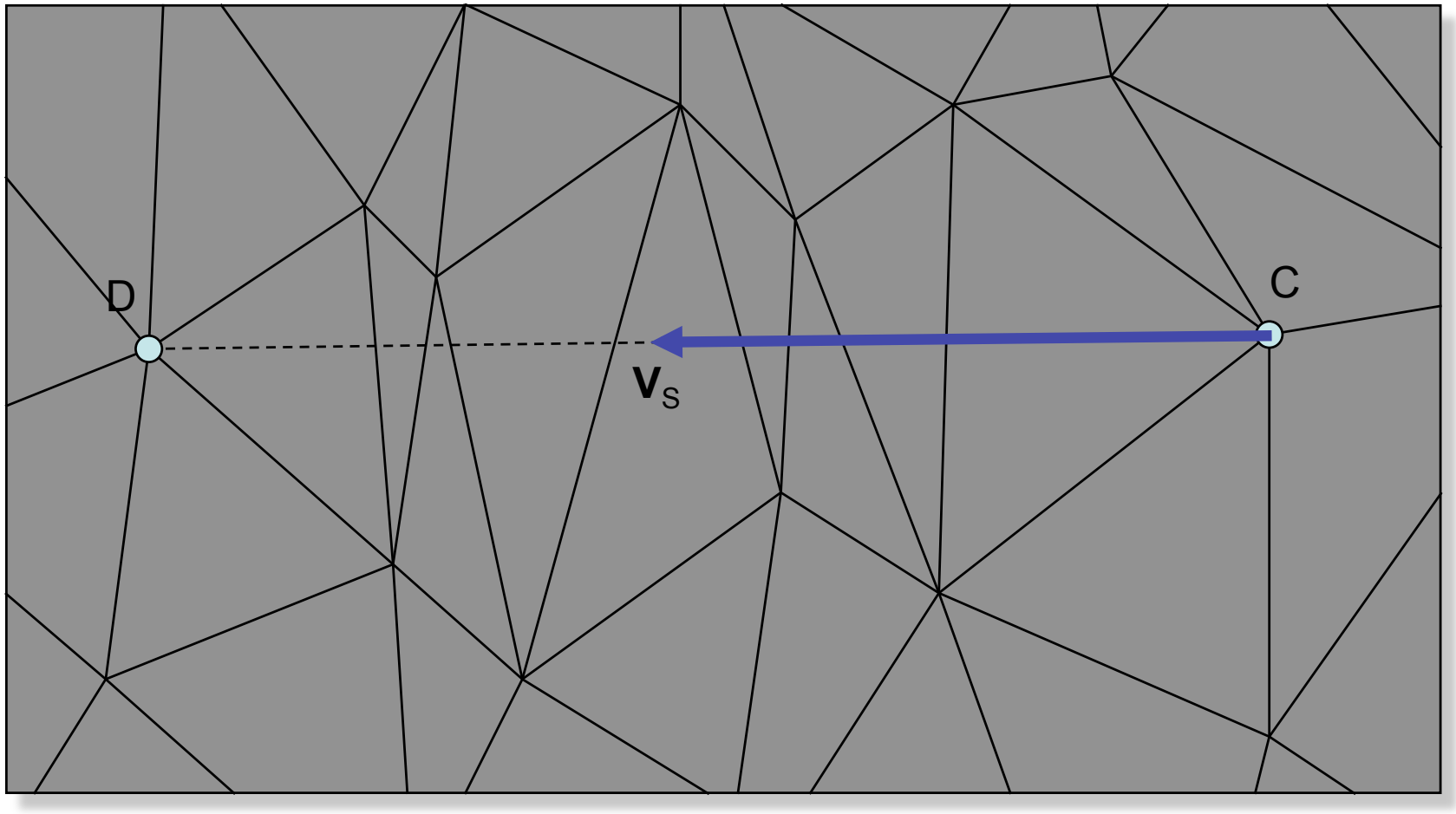
Algorithme de “type” Delaunay

- On récupère l**e bord du domaine** si on ne l’a pas



  - Il faut insérer des nœuds et des arêtes là où les arêtes, introduites lors de la triangulation initiale, coupent le bord du domaine géométrique
  - **Ou faire du swapping d’arêtes pour récupérer les arêtes du bord géométrique**

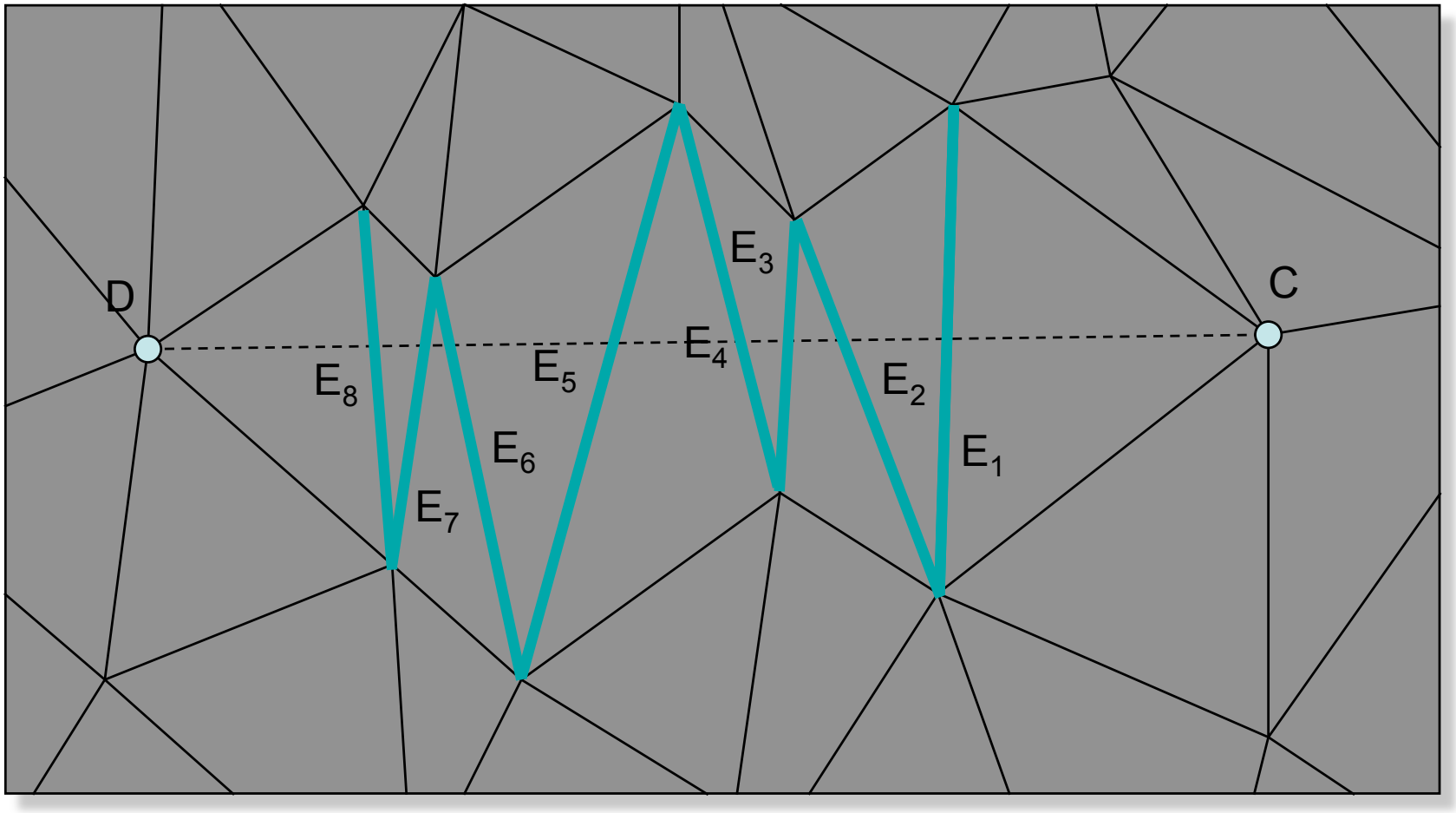
[George,91] [Owen,99]



Exemple de swapping local

- Récupération de l’arète CD le long du vecteur $\mathbf{V}_s$

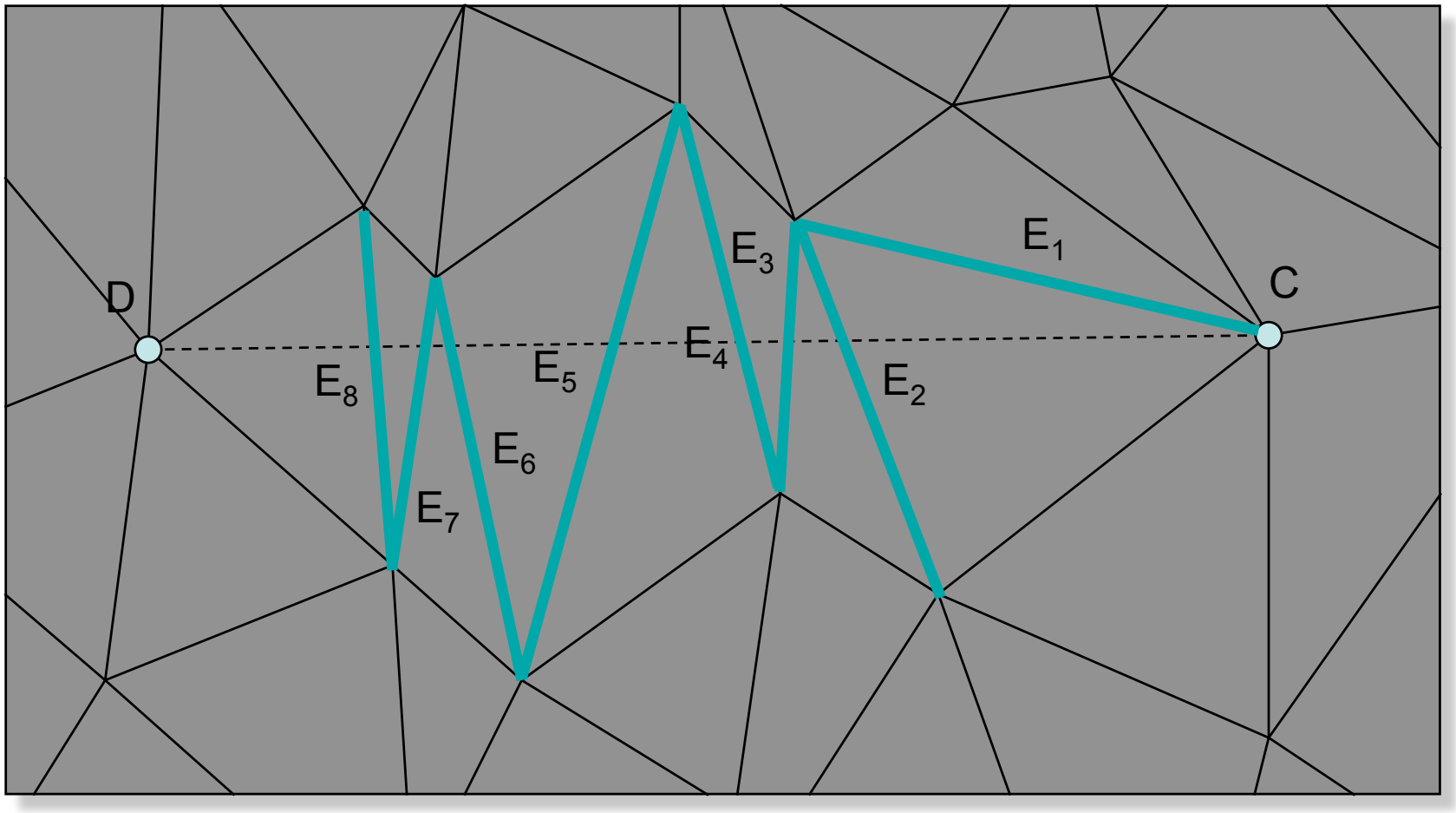
Algorithme de “type” Delaunay



Exemple de swapping local

- on liste l’ensemble des arêtes intersectées par $\mathbf{V}_s$

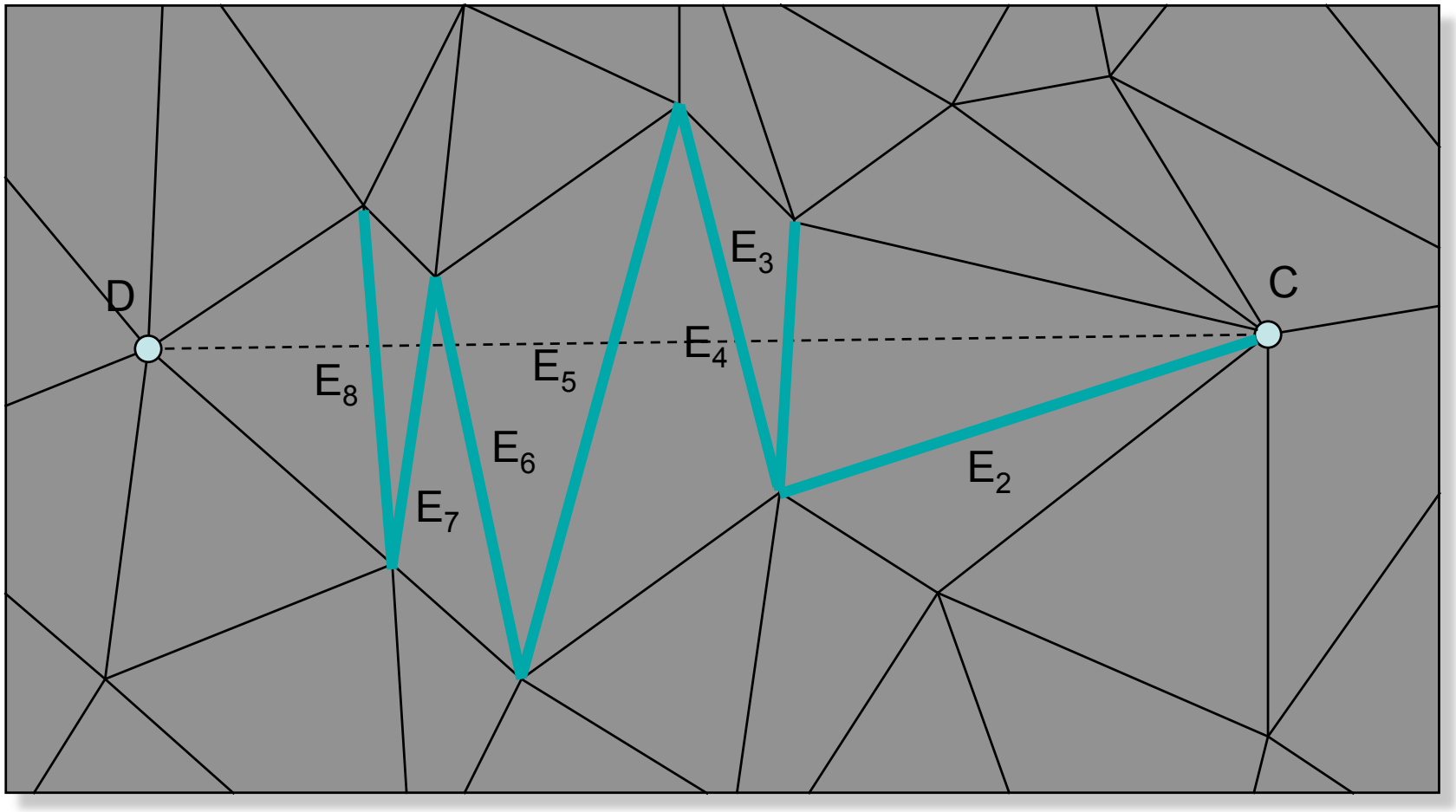
Algorithme de “type” Delaunay



Exemple de swapping local

- On fait du swapping d’arêtes pour les couples de triangles adjacent aux arêtes de la liste

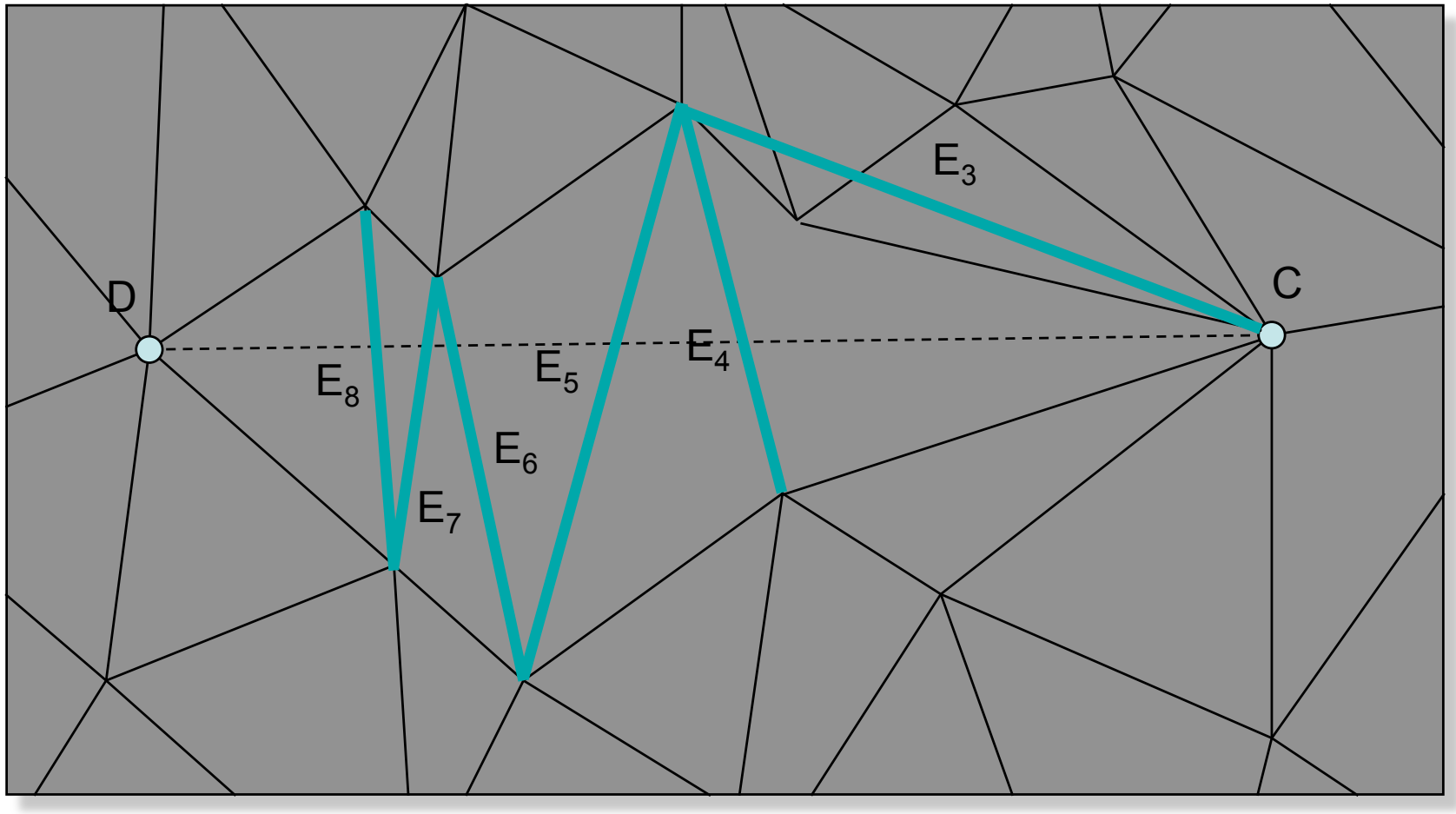
Algorithme de “type” Delaunay



Exemple de swapping local

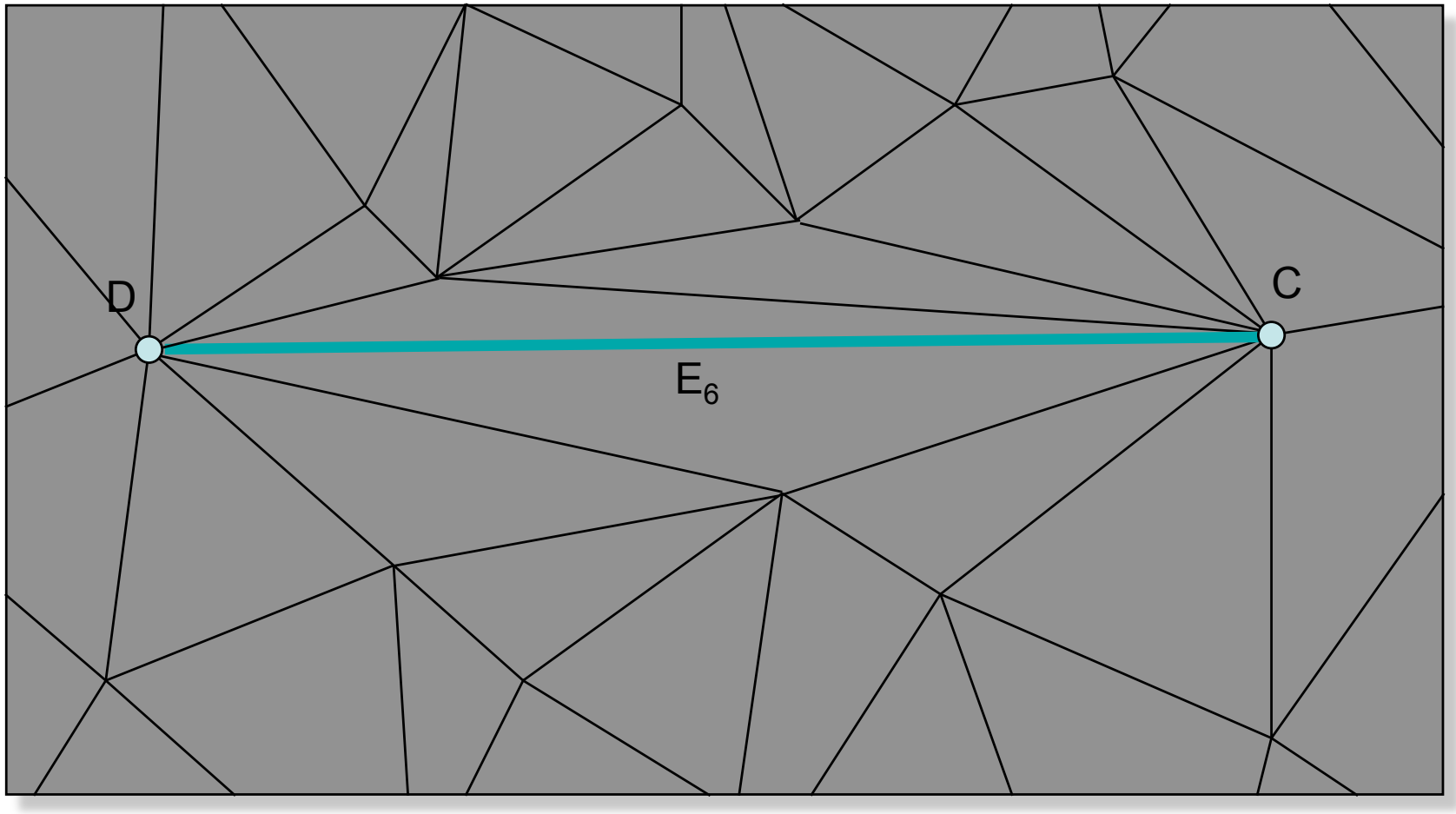
- On teste si le swapping n’induit pas de recouvrement (croisement) entre triangles
- Si oui, on reporte le swapping à plus tard

Algorithme de “type” Delaunay



Exemple de swapping local

- On teste si le swapping n’induit pas de recouvrement (croisement) entre triangles
- Si oui, on reporte le swapping à plus tard



Exemple de swapping local

- Au final, l’arête CD est récupérée mais la qualité des triangles peut être mauvaise
- Le critère de Delaunay nest plus respecté

Bilan sur les méthodes de triangles/tétraèdres

- Pas de problème de connectivité / topologie dans les maillages triangulaires / tétraédriques
  - Triangulation de Delaunay
    - Étant donné un ensemble de points, on peut les connecter entre eux pour former des triangles (2D), tétraèdres (3D), n-simplexes (nD)
    - La propriété de la sphère vide est locale et favorise les opérations locales
    - Cavités
      - Difficile d'avoir des maillages de très bonne qualité en 3D
      - Difficile d'assurer les contraintes au bord en 3D
  - Techniques à base d'octree
    - On divise des polyèdres jusqu'à avoir des simplexes
      - Inutile d'ajouter des sommets sur le bord
  - Méthodes par avancée de front
    - Se base sur le fait que la cavité restante est toujours subdivisable en tétraèdres