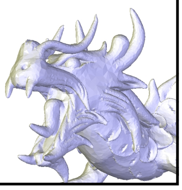
Lyon 1

Mesh and Computational Geometry

Raphaëlle Chaine
Université Claude Bernard Lyon 1

M2 ID3D
Image, Développement
et Technologie 3D
et 3A Centrale



1

Problem of a triangulating a surface passing through points

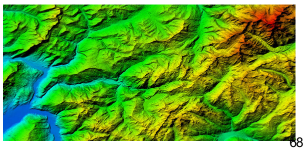
- Case of points belonging to a plane
- Ideas for constructing a mesh from these data?
  - Link the projected points together avoiding crossings to produce triangular pieces of surface.

67

67

Problem of a triangulating a surface passing through points

- Case of points belonging to a plane
- Case of a digital terrain model
  - Data can be parameterized as a height function with respect to a reference plane
  - 2D ½ dimension
- Ideas for reconstructing from these data?



68

68

Triangulation of a terrain

- Amounts to a 2D Problem
  - Work on the projection of the points on the reference plane
  - Then move the vertices up to their initial altitude

69

69

2D Triangulation

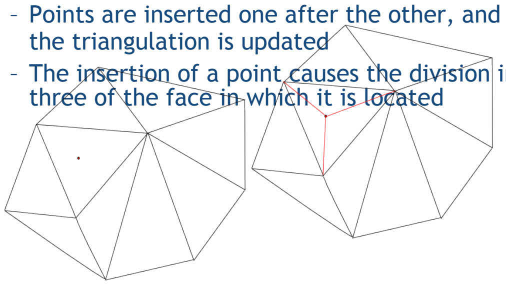
- Naive incremental construction
  - Add the points one after the other

70

70

2D Triangulation

- Naive incremental construction
  - Triangulation of 3 points: triangle oriented in the trigonometric direction
  - Points are inserted one after the other, and the triangulation is updated
  - The insertion of a point causes the division in three of the face in which it is located



71

71

2D Triangulation

- If the point does not fall into any face:
  - The insertion of a point outside the convex hull creates new triangles: One for each boundary edge that is "visible"
- The triangulation remains convex after each insertion

72

72

2D Triangulation

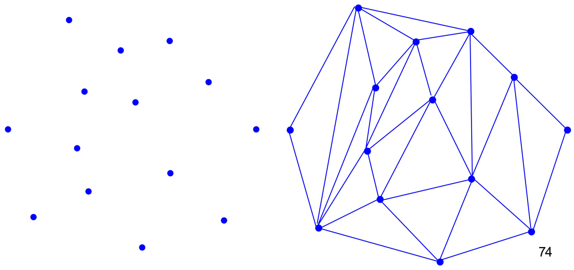
- Visibility test
  - Boundary edges are oriented (clockwise)
  - The oriented edge AB is visible by P if the triangle ABP is oriented counter-clockwise
    - Consider the sign of (**ABxAP**).**k**
    - A(x_A,y_A,0), B(x_B,y_B,0), P(x,y,0), k(0,0,1)
- Reminder :
  - edge of the convex envelope = pair (index of an infinite face + local index of the infinite vertex in that face)

73

73

2D Triangulation

- Naive incremental construction
  - Result clearly depending on the order in which the points are inserted



74

74

2D Triangulation

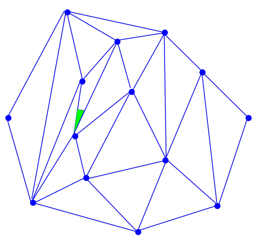
- Quality of a triangle
  - In most applications a"good triangle" is a balanced triangle (ie. as equilateral as possible)
  - Aspect ratio of a triangle
    - Inscribed circle radius / circumscribed circle radius
    - Minimum edge length / circumscribed circle radius
    - sin(smallest angle)
  - We would like triangles with aspect ratio as large as possible

75

75

Quality of a 2D triangulation

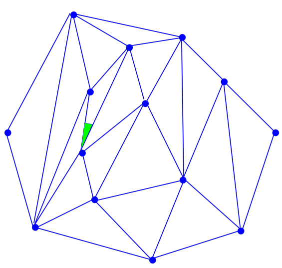
- Each triangulation is characterized by a smallest angle
- From all possible triangulations, choose one that maximizes the smallest angle

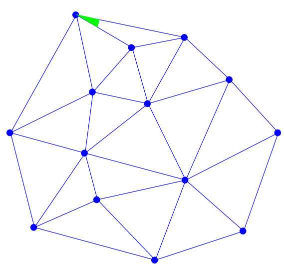


76

76

Quality of a 2D triangulation



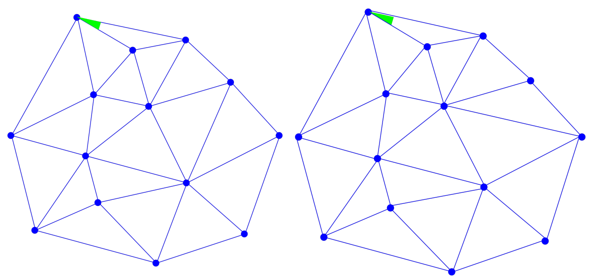


Optimal triangulation

77

77

Quality of a 2D triangulation

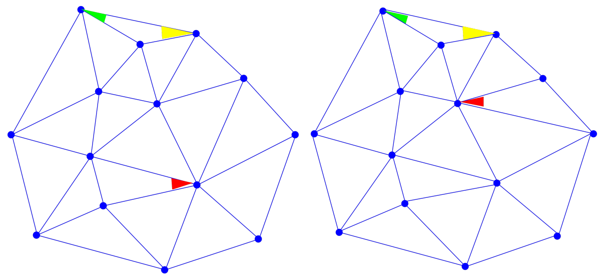


Optimal triangulation

What about this triangulation?

78

Quality of a 2D triangulation



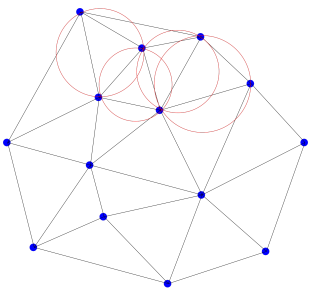
Optimal triangulation

Triangulation not optimal in the lexicographic order of the smallest angles

79

Delaunay Triangulation

- Triangulation with triangles having an empty circumscribed circle



80

Theorem

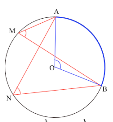
- The triangulation that maximizes the smallest angles is the Delaunay triangulation

81

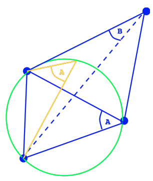
Proof of the equivalence

- Delaunay ⇔ Maximizes the smallest angles
  - **True for 4 points :**
  - Delaunay => Maximizes the smallest angles :
    - Proof : Regardless of where the smallest angle is in the Delaunay triangulation, the alternative triangulation always contains smaller angles. Therefore, the smallest angle of the Delaunay triangulation is larger than the smallest angle of the alternative triangulation.

Theorem of the inscribed angle



$$(\overrightarrow{OA}, \overrightarrow{OB}) \equiv 2(\overrightarrow{MA}, \overrightarrow{MB}) \mod 2\pi$$



82

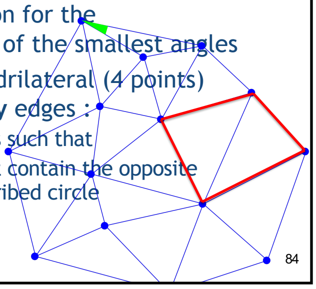
Proof of the equivalence

- Delaunay ⇔ Maximizes the smallest angles
  - **True for 4 points :**
  - Delaunay <= Maximizes the smallest angles :
    - Proof : Suppose that the optimum triangulation is not Delaunay and prove that this leads to a contradiction.

83

Proof of the equivalence

- Delaunay ⇔ Maximizes the smallest angles
  - **Proof for more than 4 points?**
  - Maximum triangulation for the lexicographical order of the smallest angles
    - --> max in each quadrilateral (4 points)
    - **--> Locally Delaunay** edges :
      - Two incident triangles such that
      - Each triangle does not contain the opposite vertex in its circumscribed circle

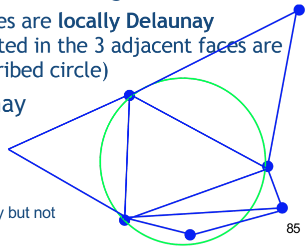


84

84

Theorem

- Delaunay ⇔ Maximizes the smallest angles
  - **Proof for more than 4 points?**
  - Maximum triangulation for angle order
    - -> All edges / triangles are **locally Delaunay** (ie. the vertices located in the 3 adjacent faces are outside the circumscribed circle)
  - Does locally Delaunay everywhere Implies globally Delanay?



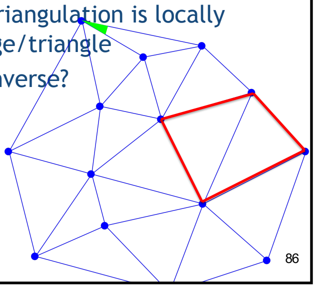
Example of a triangle locally Delaunay but not Delaunay

85

85

Theorem

- Delaunay ⇔ Maximizes the smallest angles
  - **Proof for more than 4 points?**
  - A globally Delaunay triangulation is locally Delaunay on each edge/triangle
  - But what about the inverse?



86

86

Other theorem

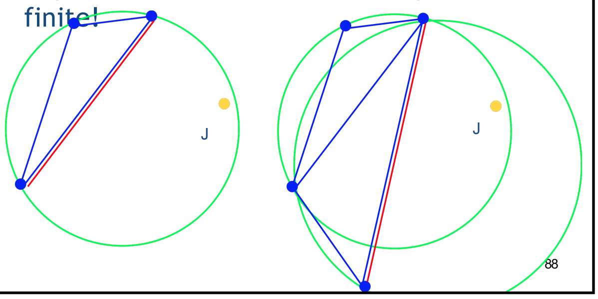
- Locally Delaunay on all triangles ⇔ Delaunay triangulation
- Demonstration :
  - Suppose there is a triangulation whose triangles are all locally Delaunay, but which is not Delaunay
  - This means that the circumscribed circle of one triangle T has a vertex J in its interior
  - But the edge of T visible from J is incident to a triangle T' whose third vertex is outside the circle circumscribed to T
  - J is in the circle circumscribed to T'....

87

87

Demonstration

- One thing leading to another, we build an infinity of triangles.
- Impossible since the number of points is finite!



88

88

Transformation of a 2D triangulation

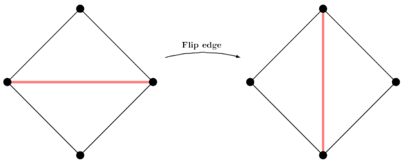
- How to improve a badly shaped triangulation?

89

89

Transformation of a 2D triangulation

- Notion of non-locally Delaunay edge
- Lawson algorithm
  - Flip non-locally Delaunay edges

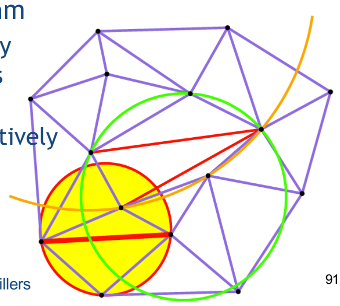


90

90

Transformation of a 2D triangulation into a Delaunay triangulation

- Notion of non-locally Delaunay edge
- Lawson algorithm
  - Push non-locally Delaunay edges into a queue
  - Flip them iteratively
  - Note : Locally Delaunay faster to check than Delaunay!!!!!



O. Devillers

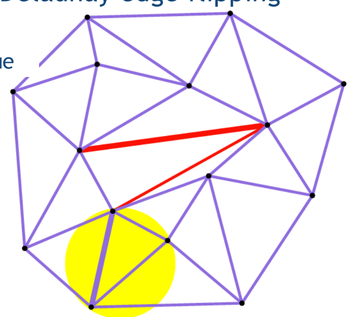
91

91

Transformation of a 2D triangulation into a Delaunay triangulation

- Non-locally Delaunay edge flipping

... Using a queue

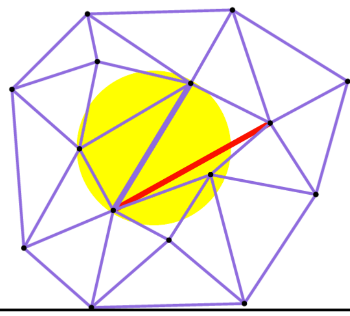


92

92

Transformation of a 2D triangulation into a Delaunay triangulation

- By non-locally Delaunay edge flipping

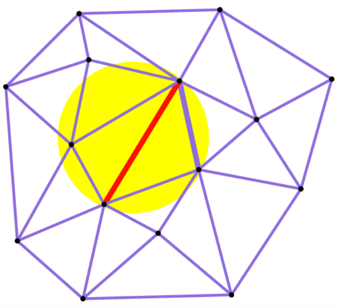


93

93

Transformation of a 2D triangulation into a Delaunay triangulation

- By non-locally Delaunay edge flipping

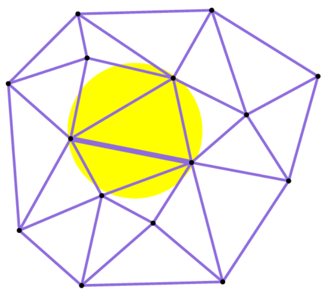


94

94

Transformation of a 2D triangulation into a Delaunay triangulation

- By non-locally Delaunay edge flipping



95

95

Transformation of a 2D triangulation into a Delaunay triangulation

- Do you think that such an algorithm should always terminate?
- Think of an energy of the triangulation based on the sum of the smallest angles of all the triangles.

96

96

Note

- It is always possible to flip a non-locally Delaunay edge, because the 4 vertices of the two incident triangles are always in convex position

97

97

Properties

- The Delaunay triangulation of four cocyclic points is not unique.
- Delaunay triangulation of points in « general » position is unique
  - No four cocyclic points
- Possible concept of disturbance
  - The points are given with a speed vector, as if they were moving...

98

98