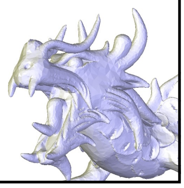
Lyon 1

Mesh and Computational Geometry

Raphaëlle Chaine
Université Claude Bernard Lyon 1

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

2024-25



1

Problem of a triangulating a surface passing through points

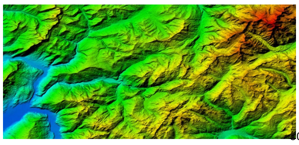
- 2D Case : points belonging to a plane
- Ideas for constructing a mesh from these data?
  - Connect the points together avoiding crossings to produce non overlapping triangles

79

79

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?



80

80

Triangulation of a terrain

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

81

81

2D Triangulation

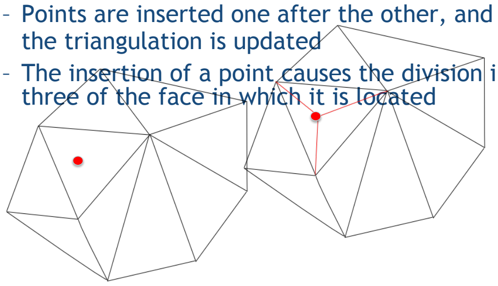
- Naive incremental construction
  - Add the points one after the other

82

82

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



83

83

2D Triangulation

- If the point to be inserted is outside every face:
  - The insertion of the point outside the convex hull creates new triangles:
    One for each boundary edge that is "visible by the point"
- The triangulation remains convex after each insertion

84

84

2D Triangulation

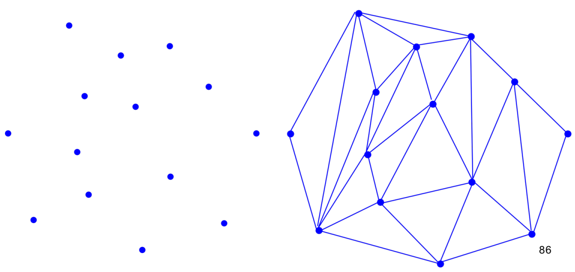
- Visibility test
  - Boundary edges are oriented (clockwise)
  - The oriented edge AB is visible by P if the triangle ABP is oriented counter-clockwise
    - Consider the sign of (**ABxAP**).**k**
    - $A(x_A,y_A,0)$, $B(x_B,y_B,0)$, $P(x,y,0)$, $k(0,0,1)$
- Reminder :
  - edge of the convex envelope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

85

85

2D Triangulation

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



86

86

2D Triangulation

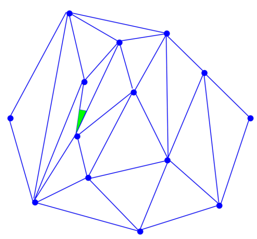
- Quality of a triangle
  - In most applications a"good triangle" is as equilateral as possible
  - Aspect ratio of a triangle
    - Inscribed circle radius / circumscribed circle radius
    - Minimum edge length / circumscribed circle radius
    - sin(smallest angle)
  - We would like triangles with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

87

87

Quality of a 2D triangulation

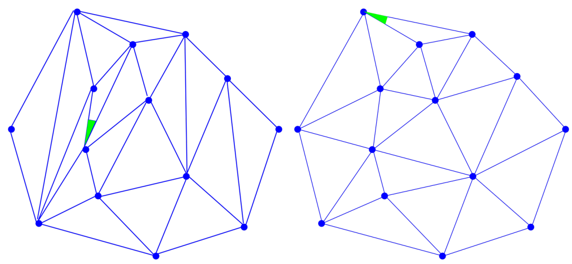
- Each triangulation is characterized by a smallest angle
- From all possible triangulations, choose one that maximizes the smallest angle



88

88

Quality of a 2D triangulation

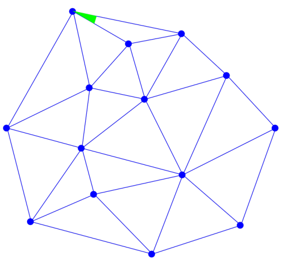


Optimal triangulation

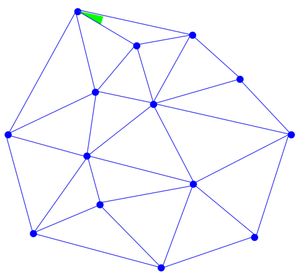
89

89

Quality of a 2D triangulation



Optimal triangulation

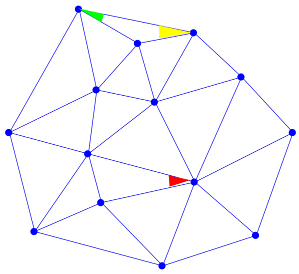


What about this triangulation?

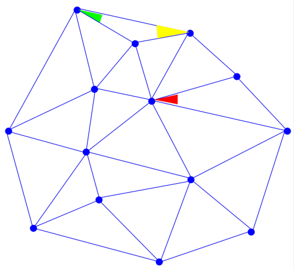
90

90

Quality of a 2D triangulation



Optimal triangulation



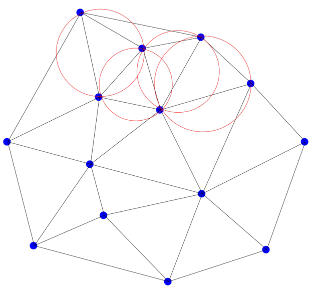
Triangulation not optimal in the lexicographic order of the smallest angles

91

91

Delaunay Triangulation

- Triangulation with triangles having an empty circumscribed circle



92

92

Theorem

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

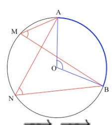
93

93

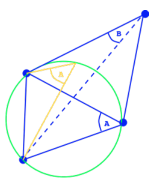
Proof of the equivalence

- 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



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



94

94

Proof of the equivalence

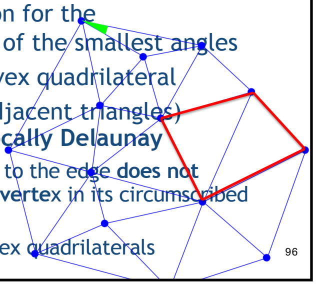
- Delaunay ⇔ Maximizes the smallest angles
  - **True for 4 points :**
  - Delaunay <- Maximizes the smallest angles :
    - Proof : Suppose that the optimum triangulation is not Delaunay and prove that this leads to a contradiction.

95

95

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 convex quadrilateral (composed of 2 adjacent triangles)
    - **--> Every edge is Locally Delaunay**
      - Each triangle incident to the edge **does not contain the opposite vertex** in its circumscribed circle
      - Also true for non convex quadrilaterals

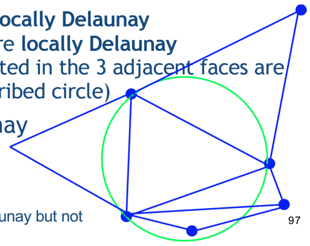


96

96

Theorem

- Delaunay ⇔ Maximizes the smallest angles (<-)
  - **Proof for more than 4 points?**
  - Maximum triangulation for angle order
    - -> All the edges are **locally Delaunay**
    - -> All the 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

97

97

Other theorem

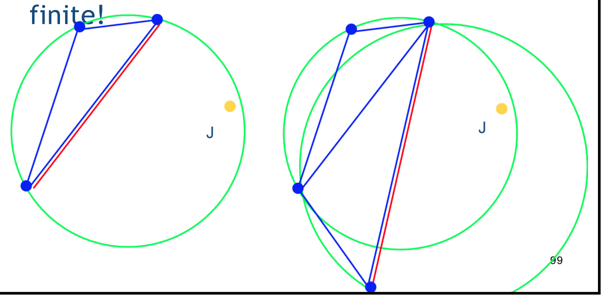
- Locally Delaunay on all triangles ⇔ Delaunay triangulation (<-)
- Demonstration :
  - Suppose there is a triangulation with locally Delaunay triangles, but which is not Delaunay
  - This means that the circumscribed circle of ONE triangle T has a vertex J in its interior
  - But the edge of T visible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

98

98

Demonstration

- Progressively, we build an infinity of triangles.
- Impossible since the number of points is finite!



99

99

Transformation of a 2D triangulation

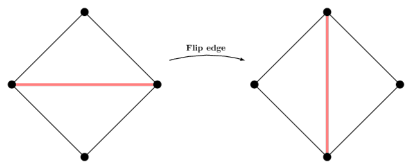
- How to improve a badly shaped triangulation?

100

100

Transformation of a 2D triangulation

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

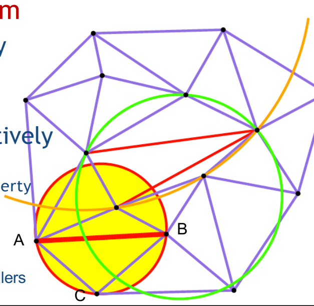


101

101

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 property faster to check than Delaunay!!!!!



O. Devillers

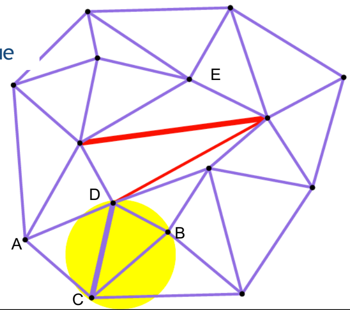
102

102

Transformation of a 2D triangulation into a Delaunay triangulation

- Flip non-locally Delaunay edges

... Using a queue

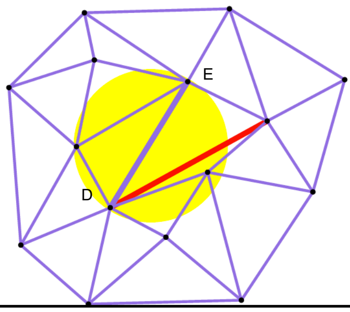


103

103

Transformation of a 2D triangulation into a Delaunay triangulation

- By non-locally Delaunay edge flipping

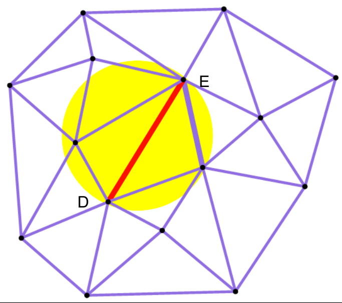


)4

104

Transformation of a 2D triangulation into a Delaunay triangulation

- By non-locally Delaunay edge flipping

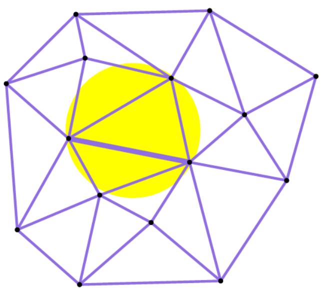


105

105

Transformation of a 2D triangulation into a Delaunay triangulation

- By non-locally Delaunay edge flipping



106

106

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 adjacent triangle pairs.

107

107

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

108

108

Properties

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

109

109

Properties

- There are interesting interpretations of the Delaunay triangulation, lifting the points into a space of higher dimension
  - What is Delaunay in 1D?
    - Delaunay is like sorting and linking each point with the next one

    OR
    - Projecting the 1D points on the lower side of a 2D convex and computing the convex envelop of the 2D lifted points
  - Delaunay = lower envelope of points

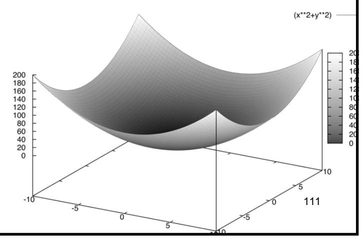
110

110

Properties

- Nice interpretation of Delaunay in the space (« space of spheres »)
  - Correspondence between the Delaunay triangulation and the lower convex envelope of the points lifted on the paraboloid

$$z = x^2 + y^2$$

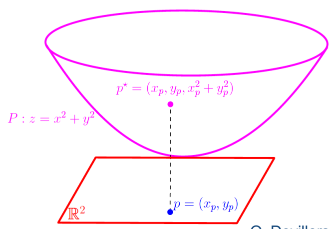


111

111

- Each 2D point x_P, y_P is lifted to $x_P, y_P, x_P^2+y_P^2$ on the paraboloid

$$z = x^2 + y^2$$



112

112

Demonstration

- Let us consider a circle of the plan

$$(M - C)^2 = R^2$$

$$(x - x_C)^2 + (y - y_C)^2 = R^2$$

$$x^2 + y^2 - 2xx_C - 2yy_C + x_C^2 + y_C^2 - R^2 = 0$$

$$x^2 + y^2 - 2ax - 2by + c = 0$$

  - Where are the points of the circle lifted on the paraboloid?

113

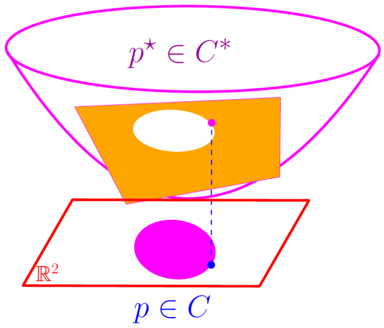
113

Demonstration

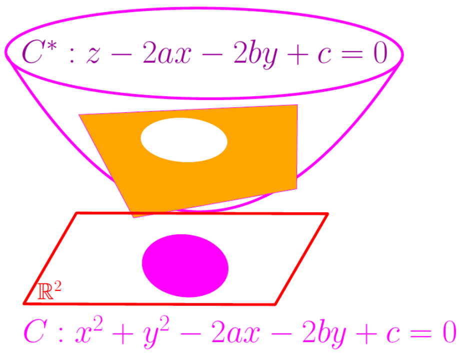
- Circle $x^2 + y^2 - 2ax - 2by + c = 0$
  - all the points belonging to this circle are lifted to points belonging simultaneously to the paraboloid and the plane

$$z - 2ax - 2by + c = 0$$

Demonstration

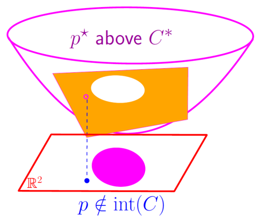


Demonstration



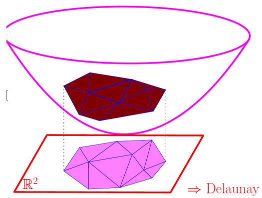
Demonstration

- What about the points outside the circle C when they are lifted?
  - They are lifted above the plane C*



Demonstration

- If 3 points correspond to a Delaunay triangle, none of the other points can be lifted under the plane corresponding to the 3 points :
  - Correspondance between the Delaunay triangulation and the lower envelopppe of the points lifted to the paraboloid.



Back to Lawson's algorithm

- Transformation of a 2D triangulation into a Delaunay triangulation
  - As long as there is a non-locally Delaunay edge (ie. a non Delaunay subtriangulation of 4 points)
    - Replace the 4 points' subtriangulation by the alternative subtriangulation (flip)
- What complexity?

Back to Lawson's algorithm

- By using Delaunay interpretation in the spheres space, let’s show that **an edge cannot appear more than once in the queue** (even with other incident triangles)

120

120

Back to Lawson's algorithm

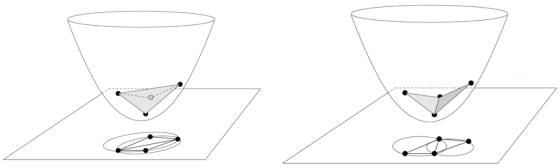
- Interpretation of the *flip* in the “space of spheres”
  - Before the *flip*, each of the two triangles incident to the edge has the 4^{th} point in its circumscribed circle
  - The lifting of the plane of each triangle on the paraboloid is located above the 4^{th} point.

121

121

Back to Lawson's algorithm

- Interpretation of the *flip* in the “space of spheres”



- Each *flip*, allows the lifted surface to descend locally

122

122

Back to Lawson's algorithm

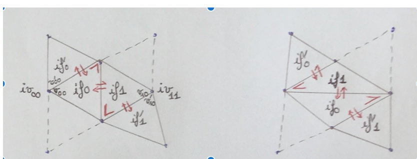
- This means that a flipped edge cannot reappear a second time in the flow of the algorithm
- The number of edges that can be formed with n points is n(n-1)/2
- Lawson's algorithm is therefore in $O(n^2)$
- Much more efficient in practice!

123

123

Operations for triangulating and flipping edges

- Updating the Mesh data structure
  - Division of a triangle into 3 triangles (FaceSplit)
  - Division of an edge into two edges (EdgeSplit)
  - Flip of an edge (Flip)



124

124