

Advanced Topics from Scientific Computing

TU Berlin Winter 2023/24

Notebook 10



Jürgen Fuhrmann

```
1 begin
2     using CairoMakie
3     using Triangulate
4     CairoMakie.activate!(type="svg")
5 end
```

Grid generation

- Regard boundary value problems for PDEs in a finite domain $\Omega \subset \mathbb{R}^d$
- Solution spaces for PDEs are infinite dimensional \Rightarrow develop finite dimensional approximations using finite element, finite volume and other methods
- One way of defining such approximations is based on the subdivision of Ω into a finite number of elementary closed subsets $T_1 \dots T_M$, called *mesh* or *grid*
- Elementary shapes:
 - triangles or quadrilaterals ($d = 2$)
 - tetrahedra or cuboids ($d = 3$)
 - more general cases possible, e.g. virtual element methods (VEM)
- During this course:
 - Assume the domain is polygonal, its boundary $\partial\Omega$ is the union of a finite number of subsets of hyperplanes in \mathbb{R}^d (line segments for $d = 2$, planar polygons for $d = 3$)
 - Focus on simplexes (triangles, tetrahedra)
 - Geometrically most flexible
 - Starting point for more general methods
 - Focus on $d = 2$

Admissible grids

Definiton: A grid $\{T_1 \dots T_M\}$ of Ω is admissible if

- Ω is the union of the elementary cells: $\bar{\Omega} = \cup_{m=1}^M T_m$
- If $T_m \cap T_n$ consists of exactly one point, then this point is a common vertex of T_m and T_n .
- If for $m \neq n$, $T_m \cap T_n$ consists of more than one point, then $T_m \cap T_n$ is a common edge (or a common facet for $d = 3$) of T_m and T_n .

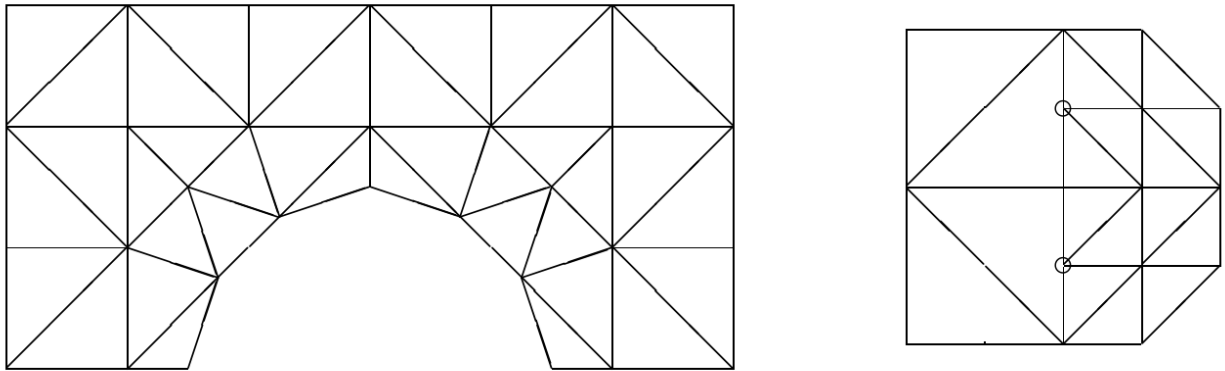


Image (Source: Braess, Finite Elements): Left - admissible mesh, right - mesh with "hanging" nodes

- There are generalizations of finite element and finite volume methods which can work with hanging nodes, however we will not go into these details in the course.

The Voronoi Diagram

After G.F. Voronoi

Definition: Let $\mathbf{p}, \mathbf{q} \in \mathbb{R}^d$. The set of points $H_{\mathbf{p}\mathbf{q}} = \{\mathbf{x} \in \mathbb{R}^d : \|\mathbf{x} - \mathbf{p}\| \leq \|\mathbf{x} - \mathbf{q}\|\}$ is the *halfspace* of points \mathbf{x} closer to \mathbf{p} than to \mathbf{q} .

Definition: Given a finite set of points $\mathcal{S} \subset \mathbb{R}^d$, the *Voronoi region* (*Voronoi cell*) of a point $\mathbf{p} \in \mathcal{S}$ is the set of points \mathbf{x} closer to \mathbf{p} than to any other point $\mathbf{q} \in \mathcal{S}$:

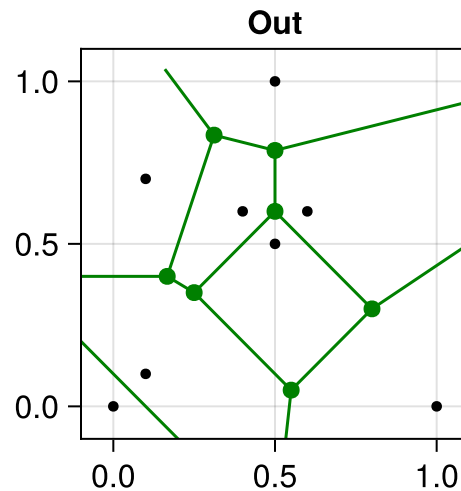
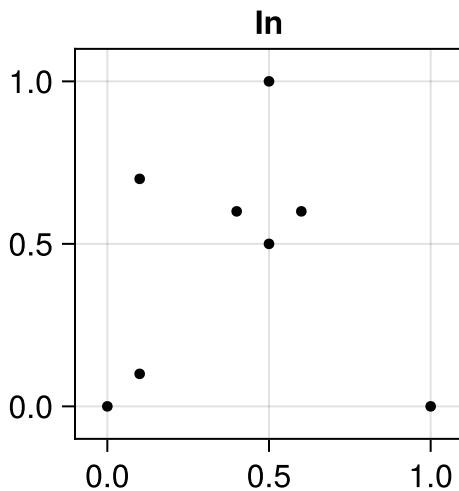
$$V_{\mathbf{p}} = \{\mathbf{x} \in \mathbb{R}^d : \|\mathbf{x} - \mathbf{p}\| \leq \|\mathbf{x} - \mathbf{q}\| \forall \mathbf{q} \in \mathcal{S}\}$$

The *Voronoi diagram* of \mathcal{S} is the collection of the Voronoi regions of the points of \mathcal{S} .

- The Voronoi diagram subdivides the whole space into "nearest neighbor" regions
- Being intersections of half planes, the Voronoi regions are convex sets

```
pts = 2×8 adjoint(::Matrix{Float64}) with eltype Float64:
 0.0  0.5  0.6  0.5  0.4  0.1  0.1  1.0
 0.0  1.0  0.6  0.5  0.6  0.1  0.7  0.0
```

```
1 pts=[0 0;
2     0.5 1;
3     0.6 0.6;
4     0.5 0.5;
5     0.4 0.6;
6     0.1 0.1;
7     0.1 0.7;
8     1 0]'
```

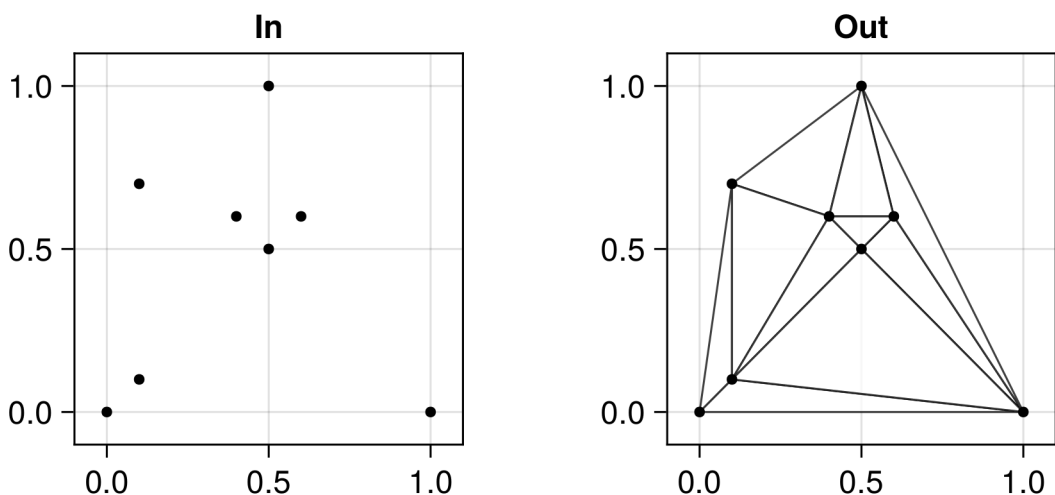


```
1 example_convex_hull_voronoi(;points=pts,show_tria=false)
```

Delaunay triangulation

After B.N. Delaunay (Delone), 1890-1980

- Given a finite set of points $S \subset \mathbb{R}^d$
- Assume that the points of S are in *general position*, i.e. no $d + 2$ points of S are on one sphere (in 2D: no 4 points on one circle)
 - Connect each pair of points whose Voronoi regions share a common edge with a line \Rightarrow *Delaunay triangulation* of the convex hull of S



```
1 example_convex_hull_voronoi(;points=pts,show_tria=true,show_voronoi=false)
```

- The *circumsphere* (*circumcircle* in 2D) of a d -dimensional simplex is the unique sphere containing all vertices of the simplex
- The *circumball* (*circumdisc* in 2D) of a simplex is the unique (open) ball which has the circumsphere of the simplex as boundary

Definition: A triangulation of the convex hull of a point set \mathcal{S} has the *Delaunay property* if each simplex (triangle) of the triangulation is Delaunay, i.e. its circumsphere (circumcircle) is empty wrt. \mathcal{S} , i.e. it does not contain any points of \mathcal{S} .

- The Delaunay triangulation of a point set \mathcal{S} , where all points are in general position is unique and has the Delaunay property
- Otherwise there is an ambiguity - if e.g. 4 points are one circle, there are two ways to connect them resulting in Delaunay triangles

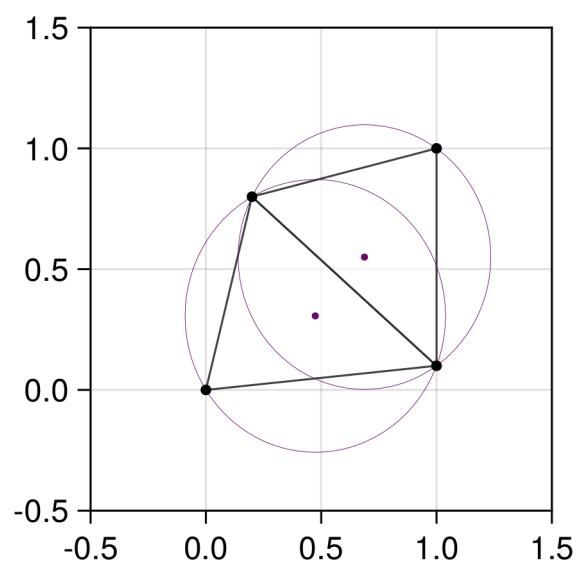
Edge flips and locally Delaunay edges (2D only)

- For any two triangles **abc** and **adb** sharing a common edge **ab**, there is the *edge flip* operation which reconnects the points in such a way that two new triangles emerge: **adc** and **cdb**.

An edge of a triangulation is locally Delaunay if it either belongs to exactly one triangle, or if it belongs to two triangles, and their respective circumdisks do not contain the points opposite wrt. the edge

- If an edge is locally Delaunay and belongs to two triangles, the sum of the angles opposite to this edge is less or equal to π .
- If all edges of a triangulation of the convex hull of \mathcal{S} are locally Delaunay, then the triangulation is the Delaunay triangulation
- If an edge is not locally Delaunay and belongs to two triangles, the edge emerging from the corresponding edge flip will be locally Delaunay

Flip edge to make the triangles Delaunay:



Lawson's Edge flip algorithm for constructing a Delaunay triangulation

This is one of the most elementary mesh generation algorithms

- Input: A stack L of edges of a given triangulation of a set S of n points
- While $L \neq \emptyset$
 - pop an edge \mathbf{ab} from L
 - If \mathbf{ab} is not locally Delaunay
 - Flip \mathbf{ab} to \mathbf{cd}
 - Push edges $\mathbf{ac}, \mathbf{cb}, \mathbf{db}, \mathbf{da}$ onto L

This algorithm is known to terminate after $O(n^2)$ operations. After termination, all edges will be locally Delaunay, so the output is the Delaunay triangulation of S .

- Among all triangulations of a finite point set S , the Delaunay triangulation maximises the minimum angle of all triangles
- For $d = 2$, the set of all possible triangulations of S is connected via the *flip graph*. Each edge of this graph corresponds to one particular flip operation.
- For $d = 3$, there are several kinds of flips, and it is not known if there is a connected flip graph as in 2D

Randomized incremental flip algorithm (2D only)

- Create Delaunay triangulation of point set S by inserting points one after another, and creating the Delaunay triangulation of the emerging subset of S using the flip algorithm
- Estimated complexity: $O(n \log n)$
- In 3D, there is no simple flip algorithm, generalizations are active research subject

Open source codes implementing mesh generation

- [CGAL](#): The Computational Geometry Algorithms Library
- ND: [qhull](#) mostly for pointsets
- 2D: [Triangle](#) by J.R.Shewchuk (UC Berkeley) with Julia wrapper [Triangulate.jl](#)
- 2D: [DelaunayTriangulation.jl](#) by Daniel VandenHeuvel
- 3D: [TetGen](#) by H. Si (WIAS Berlin) with Julia wrapper [TetGen.jl](#)
- 3D: [NetGen](#) by J. Schöberl and coworkers
- 2D/3D: [gmsh](#) By Ch. Geuzaine and J.F. Remacle with Julia wrapper [Gmsh.jl](#)

In the next lectures we will show how to interface with Triangle,TetGen and Gmsh such that the resulting grids can be used in simulation codes for partial differential equations.

`example_convex_hull_voronoi` (generic function with 1 method)

Table of Contents

Grid generation

Admissible grids

The Voronoi Diagram

Delaunay triangulation

Edge flips and locally Delaunay edges (2D only)

Lawson's Edge flip algorithm for constructing a Delaunay triangulation

Randomized incremental flip algorithm (2D only)

Open source codes implementing mesh generation