1 Automatic Mesh Generation

1.1 Mesh Definition

Mesh M is a discrete representation of geometric model in terms of its geometry G, topology T, and associated attributes A.

$$M = \{G, T, A\}$$

- geometry - nodal coordinates
- topology - element types, adjacency relationships
- attributes - color, loading, boundary conditions, ...

Typeset in LaTeX by Daniel Rypl

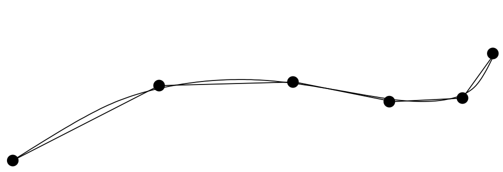
1.2 Requirements on Mesh Generation

- generality (broad range of geometries and topologies)
- automation (minimum user intervention)
- validity (valid mesh)
- accuracy (accurate resolution)
- convergence (guaranteed convergence)
- quality (guaranteed quality)
- invariance (to model rigid body motions)
- flexible mesh density control (uniform, graded meshes)
- robustness (reliability)
- compactness (storage requirements)
- efficiency (computational speed)
- linear computational complexity

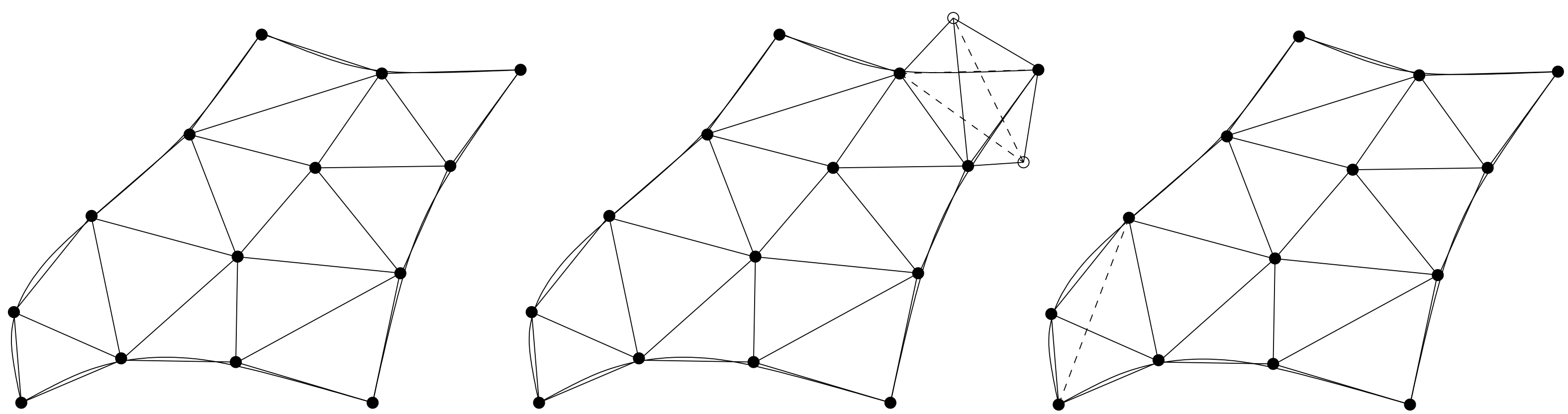
All these requirements can be hardly fulfilled at the same time and it is necessary to compromise.

1.3 Mesh Validity

- topological compatibility
  - mesh is topologically compatible with model entity E^d of dimension d
    - if each mesh entity M^{d-1} classified to E^d is shared exactly by two mesh entities M^d classified to E^d
    - if each mesh entity M^{d-1} classified to E_m^{d-1} forming m times boundary of model entity E^d is shared exactly by m mesh entities M^d classified to E^d
  - mesh is topologically compatible if it is compatible with all model entities
  - topological incompatibilities
    - topological redundancy
    - topological holes



Topological compatibility (left), topological hole (middle), and topological redundancy (right) on a curve mesh.



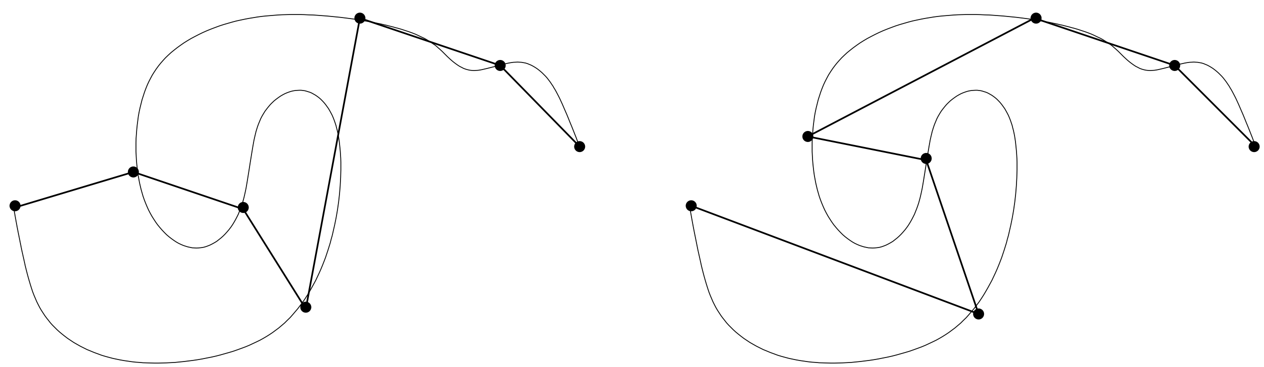
Topological compatibility (left), topological hole (middle), and topological redundancy (right) on a surface mesh.

Typeset in LaTeX by Daniel Rypl

- geometrical similarity
  - topologically compatible mesh is geometrically similar to model entity E^d $(1 \leq d \leq 3)$ if for any two different mesh entities M_i^d and M_j^d classified to E^d holds

$$M_i^d \cap M_j^d = \emptyset^d \qquad \text{if } d = 3$$

$$M_i^d \cap^* M_j^d = \emptyset^d \qquad \text{if } d = 1 \text{ or } d = 2$$



Mesh violating (left) and satisfying (right) geometrical similarity.

1.4 Computational Complexity

- number of operations needed to generate N elements expressed in terms of N and some appropriate constants
- corresponds to the time needed to generate N elements
- algorithm dependent
- average, worst

- ◦ $O(1)$ - constant computational complexity (not achievable)
- ◦ $O(N)$ - linear computational complexity (ideal)
- ◦ $O(Nlog(N))$ - logarithmic computational complexity (acceptable)
- ◦ $O(N^2)$ - quadratic computational complexity (unacceptable)
- ◦ $\cdots$

1.5 Mesh Quality

Elementary quality criteria

- element shape based criteria
  - inscribed circle/sphere radius to circumscribed circle/sphere radius ratio (simplices only)
  - area/volume^2 to perimeter^2/surface^3 ratio
  - min edge to max edge ratio (aspect ratio)
  - dihedral angle criterion (min angle to max angle ratio)
  - jacobian
- mesh topology based criteria
  - nodal valence criterion
- mesh density based criteria
  - deviation of real element spacing from desired one

Overall mesh quality

- arithmetic mean
- harmonic mean
- worst quality
- quality distribution function

1.6 Mesh Classification

- dimension - 1D mesh, 2D mesh, 3D mesh
- element type - triangular, tetrahedral, quadrilateral, hexahedral, quad-dominant, mixed, ... mesh
- element aspect ratio - isotropic, anisotropic mesh
- mesh density - uniform, graded mesh
- topology - structured, unstructured mesh

1.7 Mesh Generation Method Classification

- manual and semi-automatic methods
  - applicable to geometrically simple domains (usually 2D)
  - enumerative methods (user supplied mesh entities)
  - explicit methods (revolution, extrusion)
- mapping (parameterization) methods
  - mapping from parameter space to the physical space
  - explicit mapping - algebraic interpolation methods
  - implicit mapping - PDE solution methods
- domain decomposition methods
  - block decomposition methods (multiblock method)
  - spatial decomposition methods (quadtree/octree method)

- constructive methods
  - applicable to arbitrary geometry and topology
  - element creation - advancing front method
  - point insertion - Delaunay method

2 Structured Mesh Generation

- implicit mesh topology
- applicable to topologically simple domains
- limited mesh density control
- typical for CFD

Methods

- algebraic methods
- PDE based methods
- multiblock methods

3 Unstructured Mesh Generation

- explicit mesh topology
- applicable to domains of arbitrary geometrical and topological complexity
- flexible mesh density control
- typical for structural analysis

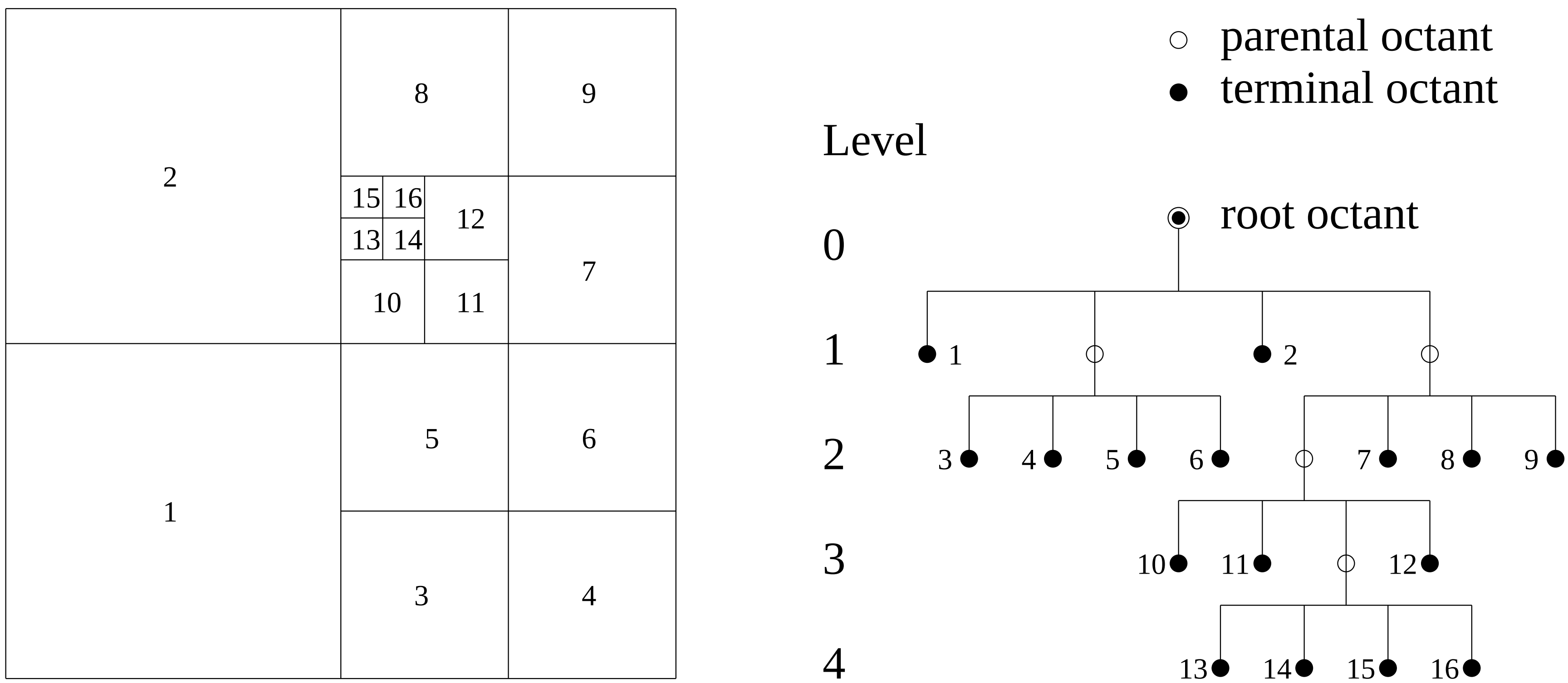
3.1 Triangular and Tetrahedral Mesh Generation

3.1.1 Quadtree/Octree Based Methods

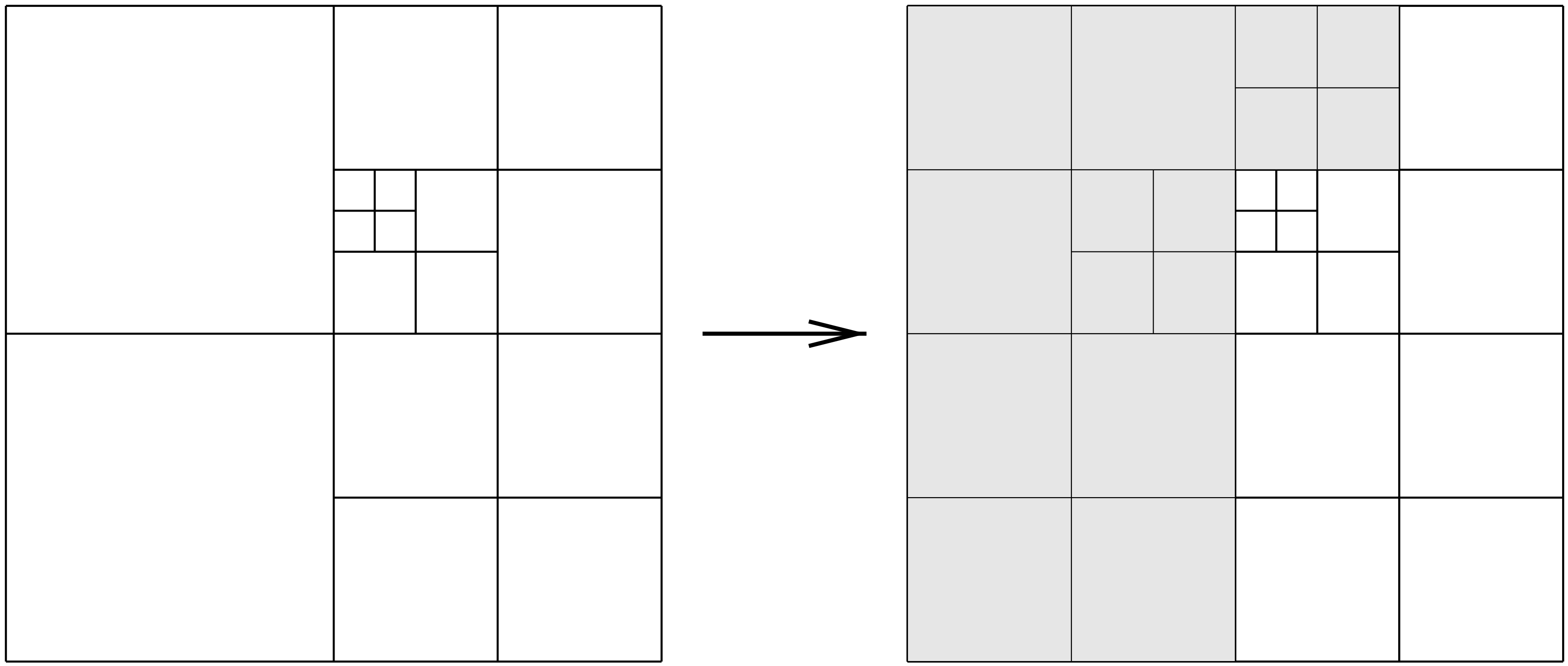
Algorithm

1. tree construction
2. mesh generation
3. mesh optimization

- quadtree/octree data structure
  - hierarchic data structure
    - vertical (top-down, bottom-up) traversal - $O(Nlog(N))$
    - horizontal (neighbour at the same level) traversal - $O(1)$
  - root cell (quadrant/octant) - bounding box (square/cube)
  - boundary refinement by recursive subdivision to equal cells (4/8 quadrants/octants) up to sufficient resolution
    - desired cell size, desired cell level
    - geometry representation (curvature, boundary features)
  - global refinement by recursive subdivision using mesh density control (background mesh, grid, sources)
  - one-level difference rule enforcement (tree balancing)
    - each two cells sharing at least an edge are at the same or subsequent levels of hierarchic tree data structure

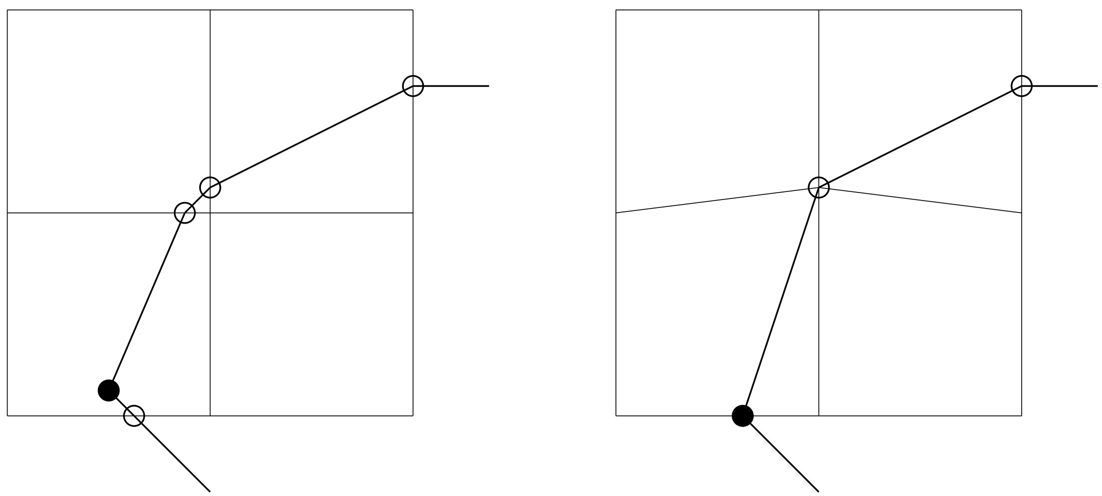


Octree hierarchy.



One-level difference rule.

- terminal cell classification
  - interior / boundary / exterior
  - in/out test (boundary orientation, ray intersection, modeller queries)
  - classification propagation
- interior refinement by recursive subdivision (to minimum level)
- evaluation of intersection of domain boundary with boundary cells

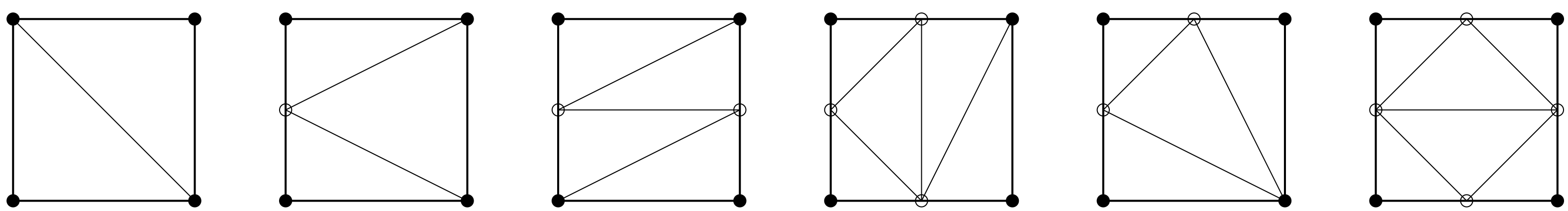


Filtering of intersection points.

- filtering of intersection points
  - violation of topology not allowed
  - modification of geometry permissible
    (original geometry restored after mesh completion)
  - reclassification of boundary cells

  - association of cell corner with closest (boundary or intersection) point (if close enough) - modification of cell shape
  - merging two intersection points
  - association of boundary point with closest intersection point (if close enough) - modification of boundary geometry
- smoothing of corner nodes of interior cells
  (repositioning of corners to barycenter of corners of all cells incident to smoothed corner)

- mesh generation
  - exterior cells - skipped
  - interior cells - templates (predefined patterns of elements topologically compatible with the cell)
    - cell corner points - mesh nodes
    - there is at maximum one midside point per octree edge (consequence of one-level difference rule)
    - 2D: $\sum_{i=0}^{4} \binom{4}{i} = 2^4 = 16$ templates (6 basic templates)
    - 3D: $\sum_{i=0}^{12} \binom{12}{i} = 2^{12} = 4096$ templates (78 basic templates)



Basic 2D templates.

Typeset in LaTeX by Daniel Rypl

- boundary cells - specific algorithm
  - only interior parts of boundary cells subjected to triangulation
  - discretization of relevant boundary of the cell
    (must ensure compatibility with neighbouring cells)
  - discretization of domain boundary within the cell
    (must comply with model topology)
  - discretization of interior of boundary cell (e.g. AFT)

- mesh optimization
  - Laplacian smoothing of interior mesh nodes
    - repositioning of nodes to barycenter of nodes of all elements incident to smoothed node
    - modifies mesh geometry
    - preserves mesh topology
    - iterative process (convergence after about 5 cycles)

Features (• - advantages, ∘ - disadvantages)

- very fast
- very robust
- reasonable quality meshes
  (slivers avoided, guaranteed quality for interior cells)
- guaranteed convergence (for sufficiently simple boundary cells)
- validity guaranteed by proper use of templates
- favourable computational complexity $O(Nlog(N)) \longrightarrow O(N)$
- boundary discretization is part of output
- good cache usage (for appropriate cell ordering)

∘ less flexible mesh density control (less suitable for adaptive analysis)
  - limited cell sizing (power of 2)
  - limited mesh density gradation
    (consequence of one-level difference rule)

- ∘ boundary layer of elements - worst quality
- ∘ not invariant with respect to rotations of the model
- ∘ cannot fully comply with given boundary triangulation
- ∘ hardly usable for anisotropic meshing

Alternatives and Extensions

- instead of templates using
  - ∘ Delaunay based cell corners insertion
  - ∘ element removal concept
- quadtree/octree data structure only used as control space (mesh density control, spatial localization)
- generation of mixed meshes (there are no all-hexahedral templates)
- applicable to curved surface meshing (direct approach)
- extensible to generalized tree approach in parametric space

3.1.2 Advancing Front Based Methods

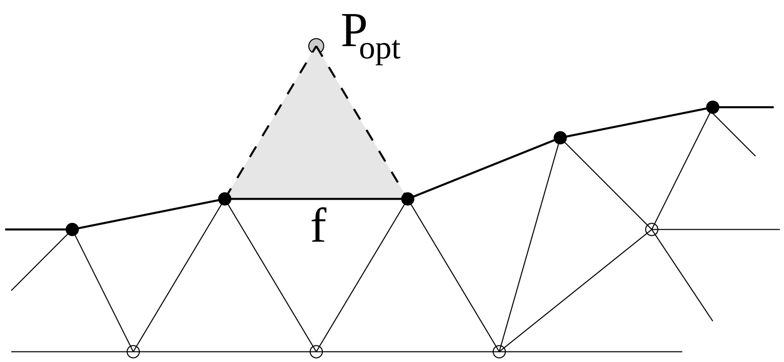
Algorithm

1. front setup
2. mesh generation
3. mesh optimization

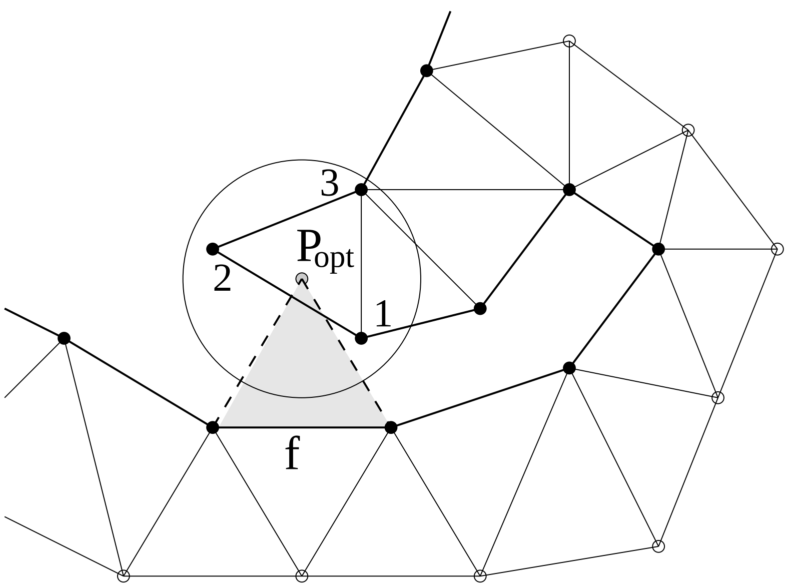
- front data structure
  - oriented interface between the meshed part and not yet meshed part of the domain
  - initially formed by all boundary facets (segments/faces)
  - evolves during mesh generation until becomes empty
  - may be multiple connected (is not allowed to overlap itself)
  - front management (facet selection, insertion, removal)

- mesh generation (until the front is not empty)
  1. facet selection: select facet f from the front
     - geometrical criteria (length, area, angles)
     - topological criteria (neighbour, minimize the front)
  2. optimal point placement: find position of the optimal point $\boldsymbol{P}_{opt}$ to form together with the selected facet f a new tentative element e
     - 2D uniform mesh: equilateral triangle
     - 3D uniform mesh: the most regular tetrahedron
     - graded mesh: mesh density variation should be taken into account
  3. potential candidate selection: search for a point $\boldsymbol{P}$ in the mesh to be used instead of $\boldsymbol{P}_{opt}$

- spatial search
- shape - circle (2D), sphere (3D) with center at $\boldsymbol{P}_{opt}$
- size - related to mesh density at $\boldsymbol{P}_{opt}$
- points in the neighbourhood ordered with respect to the increasing distance from $\boldsymbol{P}_{opt}$

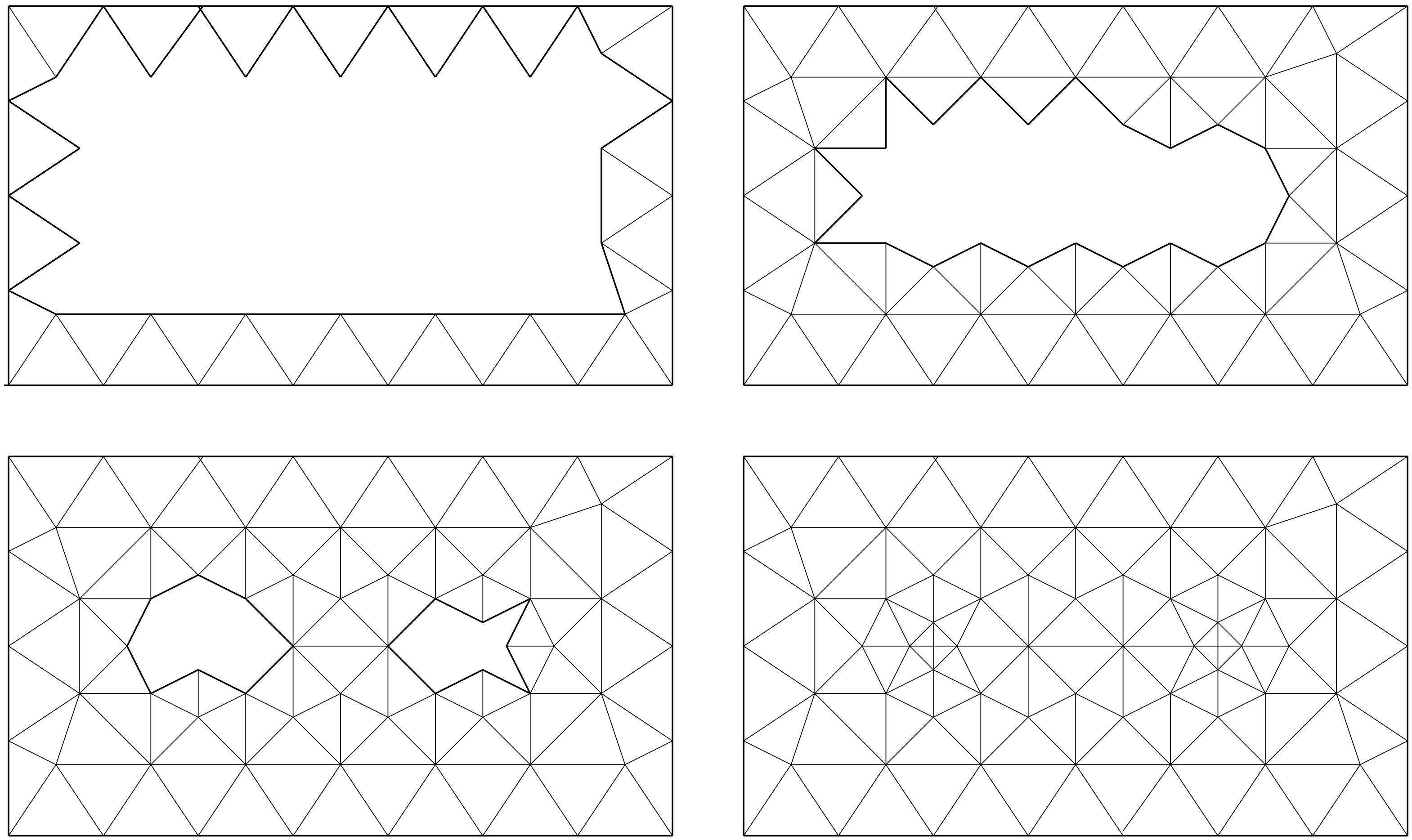


Optimal point placement.



Potential candidate selection.

4. intersection check: check if the new element e is topologically valid; if not, the point is rejected and the potential candidate selection is repeated; if there is no more candidates, optimal point placement is repeated with reduced element size
   - spatial search
   - shape - circle/rectangle (2D), sphere/parallelepiped (3D)
   - size - large enough to be reliable, small enough to be efficient
   - intersection check - tentative element against front
   - enclosure check - tentative element against nodes
5. element forming: form the new element
6. front update: update the front
   - remove existing facets used to form the new triangle from the front
   - insert new facets used to form the new triangle to the front



Front propagation.

- mesh optimization
  - Laplacian smoothing of interior mesh nodes
    - repositioning of nodes to barycenter of nodes of all elements incident to smoothed node
    - modifies mesh geometry
    - preserves mesh topology
    - iterative process (convergence after about 5 cycles)
  - topological transformations (to remove slivers)
    - diagonal edge swapping, generalized face swapping, node merging
    - modify mesh topology
    - preserve mesh geometry

Computational Aspects

- front management
  - hashing
- spatial search
  - background grid
  - background octree data structure
- intersection check
  - bounding box intersection
  - alternating digital tree

Features (• - advantages, ◦ - disadvantages)

- high quality graded meshes
- high quality boundary layer of elements
- flexible mesh density control
- validity guaranteed (if meshing completed)
- complies with given boundary discretization
- favourable computational complexity $O(Nlog(N)) \longrightarrow O(N)$
- theoretically invariant with respect to rigid body motions
- extensible to anisotropic meshing
- good cache usage (for appropriate front processing)
- appropriate for adaptive analysis (local remeshing)

◦ rather low speed

- convergence not guaranteed in 3D (requires node insertion to complete mesh even in very simple 3D case - Schönhardt polyhedron)
- creation of slivers in mesh generation phase is not avoided (slivers eliminated during mesh optimization)
- requires existence of boundary discretization

Alternatives and Extensions

- mesh density control by octree (lost of invariance) or background grid
- node acceptance driven by local Delaunay circle/sphere empty property
- extensible to curved surface meshing (direct approach)
- extensible to anisotropic meshing (using appropriate metric)
- extensible to boundary layer anisotropy (using offsetting)
- applicable to curve surface meshing (indirect approach)

3.1.3 Delaunay Based Methods

Delaunay Triangulation

Triangulation of convex hull $\mathcal{S}$ of points $\boldsymbol{P}_i,\ i = 1, 2, 3, \ldots, m$ in $\mathcal{R}^n,\ n \geq 2$ where for each simplex K holds Delaunay criterion

$$\forall i : \parallel \boldsymbol{SP}_i \parallel \geq \ \rho(K) \qquad \boldsymbol{S} = \{\boldsymbol{P} \in \mathcal{R}^n : \parallel \boldsymbol{SP}_j \parallel = \rho(K) \text{ for } \boldsymbol{P}_j \in K\}$$

is called Delaunay triangulation $\mathcal{T}_m$.

- if the equality in Delaunay criterion is fulfilled also for $\boldsymbol{P}$ not incident to K then the Delaunay triangulation is degenerated
- empty circle property – Delaunay criterion in 2D
- empty sphere property – Delaunay criterion in 3D

Voronoi Diagram

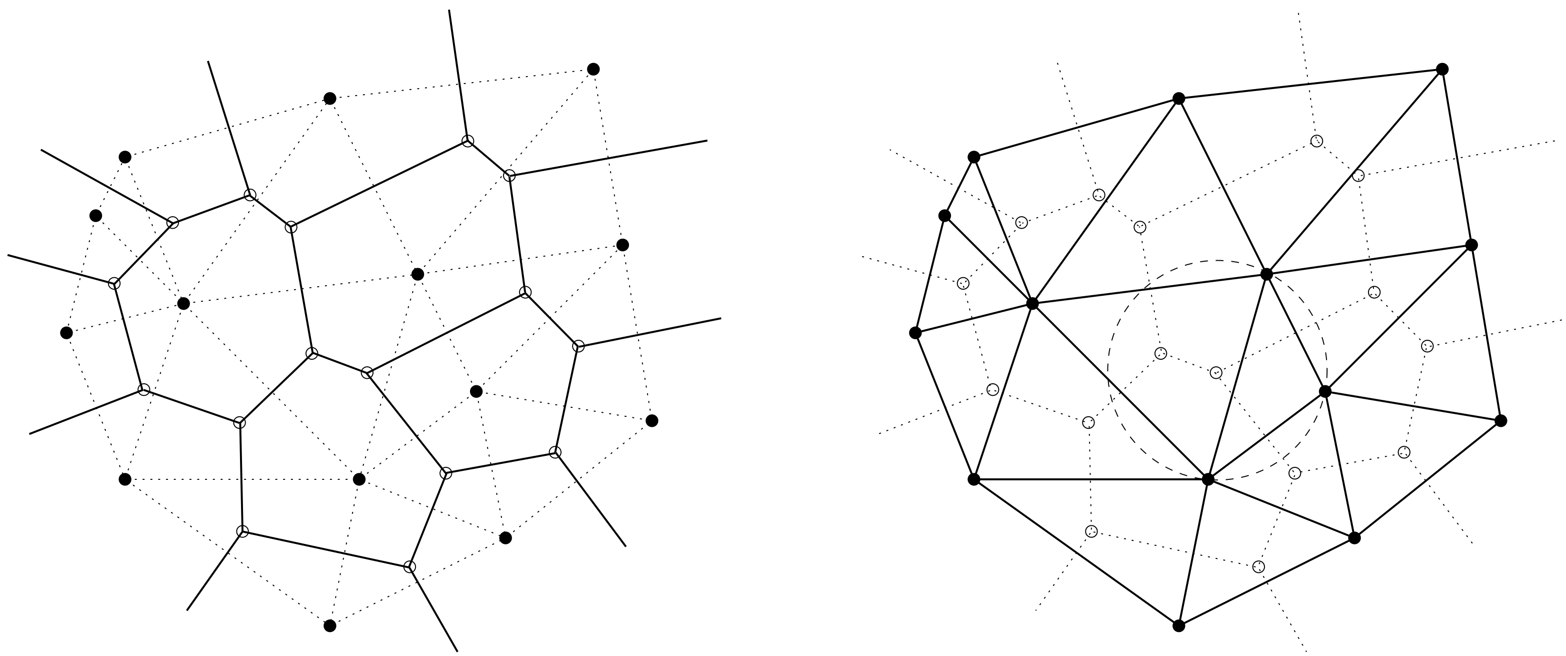
Set of cells V_i around m points $\boldsymbol{P}_i,\ i = 1, 2, 3, \ldots, m$ in $\mathcal{R}^n,\ n \geq 2$ defined as

$$V_i = \{\boldsymbol{P} \in \mathcal{R}^n \mid \forall i \neq j : \parallel \boldsymbol{PP}_i \parallel \leq \parallel \boldsymbol{PP}_j \parallel\}$$

is called Voronoi Diagram.

Duality between Delaunay Triangulation and Voronoi Diagram

- corners of Voronoi cells are centers of discs (circles, spheres) circumscribed to simplices in Delaunay triangulation
- faces of Voronoi cells correspond to faces of simplices in Delaunay triangulation

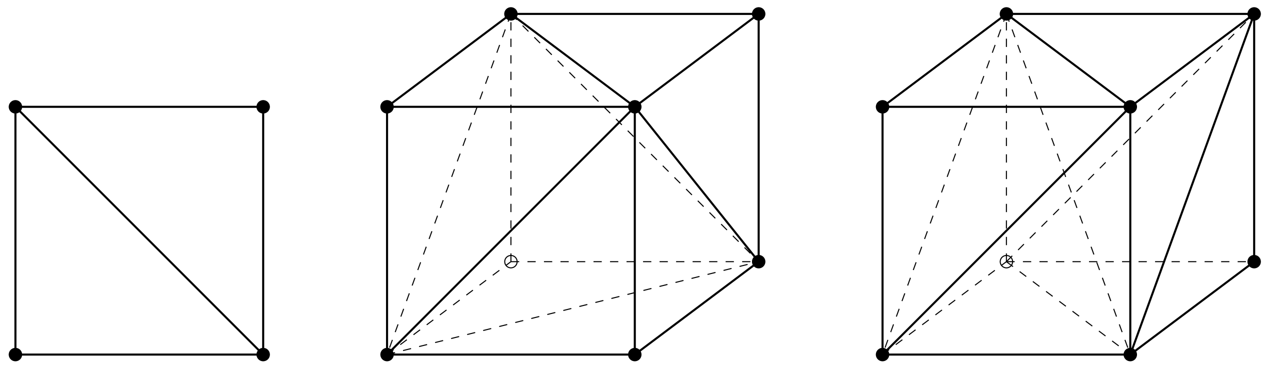


Duality between Voronoi diagram (left) and Delaunay triangulation (right) in 2D.

Algorithm

1. initial Delaunay triangulation setup
2. mesh generation
   - boundary node insertion
   - recovery of domain boundary
   - interior classification
   - interior node insertion
3. mesh optimization

- initial Delaunay triangulation setup
  - one or few simplices (completely surrounding the domain) with *a priori* fulfilled Delaunay criterion
  - can be degenerated
  - 2D: typically 2 triangles of square bounding box
  - 3D: typically 5 or 6 tetrahedrons of cubic bounding box



Initial Delaunay triangulation.

- mesh generation
  - incremental point insertion algorithm - Delaunay kernel
    - Bowyer (Watson) algorithm

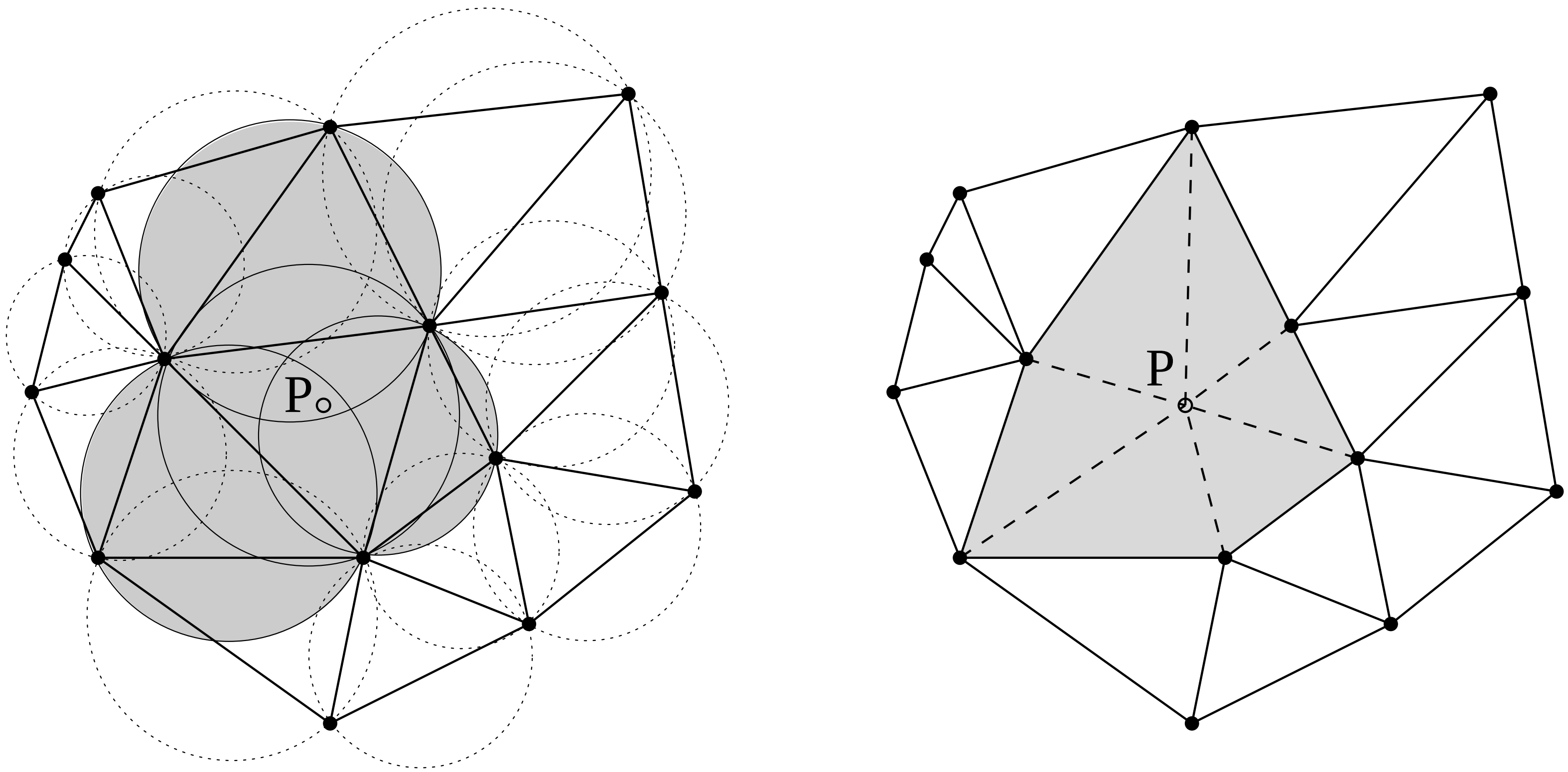
$$\mathcal{T}_{i+1} = \mathcal{T}_i - \mathcal{C}_{\boldsymbol{P}} + \mathcal{B}_{\boldsymbol{P}}$$

$\boldsymbol{P}$ - $(i+1)^{\text{th}}$ point from a convex hull $\mathcal{S}$

$\mathcal{T}_j$ - Delaunay triangulation of first j points from a convex hull $\mathcal{S}$

$\mathcal{C}_{\boldsymbol{P}}$ - cavity, set of elements K from $\mathcal{T}_i$ whose circumball contains $\boldsymbol{P}$

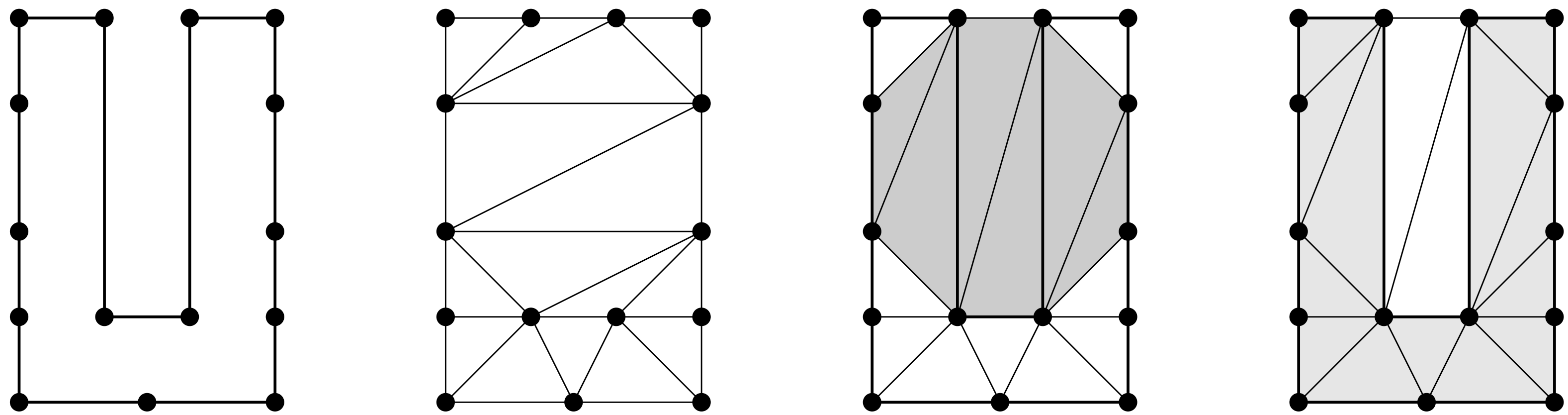
$\mathcal{B}_{\boldsymbol{P}}$ - ball, set of new elements formed by boundary facets of $\mathcal{C}_{\boldsymbol{P}}$ and $\boldsymbol{P}$



Bowyer algorithm.

- cavity $\mathcal{C}_{\boldsymbol{P}}$ is star-shaped
- boundary facets of $\mathcal{C}_{\boldsymbol{P}}$ are visible from $\boldsymbol{P}$

- insertion of boundary points
  - all points of boundary discretization are incrementally insert in the Delaunay triangulation using Bowyer algorithm
- recovery of domain boundary
  - boundary Delaunay triangulation may be not boundary conforming (domain boundary facets are not present in the triangulation)
  - topological transformations (edge and face swapping)
    not sufficient in 3D (Schönhardt polyhedron)
    violate Delaunay property (unless applied to degenerated scheme)
    $\longrightarrow$ **constrained Delaunay triangulation**
  - insertion of additional points on boundary (Steiner points)
- interior classification
  - simplices of the constrained Delaunay triangulation are classified either as interior or exterior with respect to the model topology



Recovery of boundary (left 3) and interior classification (right).

- insertion of interior points
  - points along edges of interior simplices not complying with desired mesh density or of poor shape
  - barycenters of interior simplices not complying with desired mesh density or of poor shape
  - circumcenters of interior simplices not complying with desired mesh density or of poor shape

- points created on basis of AFT
  (front is the interface between interior simplices complying and not complying with desired mesh density and quality)
- points are inserted using modified Bowyer algorithms
  (cavity $\mathcal{C}_{\boldsymbol{P}}$ cannot propagate over boundary facets)

- mesh optimization
  - Laplacian smoothing of interior mesh nodes
    - repositioning of nodes to barycenter of nodes of all elements incident to smoothed node
    - modifies mesh geometry
    - preserves mesh topology
    - iterative process (convergence after about 5 cycles)

- topological transformations (to remove slivers)
  - diagonal edge swapping, generalized face swapping, node merging
  - modify mesh topology
  - preserve mesh geometry

Computational Aspects

- Delaunay kernel - cavity construction
  - spatial search
    (adjacency search from simplex containing point being inserted)
  - robustness of in-circle and in-sphere test
    (rounding errors, ill-conditioned simplices, perturbation)
- boundary recovery
  - topological issue (localization of missing facet)
  - geometrical issue (localization of entities intersecting missing facet)

Features (• - advantages, ◦ - disadvantages)

- strong mathematical background
- high quality graded meshes
- rather high speed
- flexible mesh density control
- convergence guaranteed
- validity of raw mesh through the meshing process
- favourable computational complexity $O(N)$
- extensible to anisotropic meshing

◦ validity not guaranteed (boundary recovery necessary)

◦ creation of slivers in mesh generation phase is not avoided (slivers eliminated during mesh optimization)

- not invariant with respect to rotations
- requires existence of boundary discretization
- does not comply with given boundary discretization

Alternatives and Extensions

- mesh density control by octree or background grid
- use of preplaced points (octree - high probability of degeneracy)
- extensible to anisotropic meshing (using appropriate metric)
- applicable to curved surface meshing (indirect approach)
- extensible to curved surface meshing (direct approach) with applying Delaunay property in tangent plane at a given location

3.2 Quadrilateral and Hexahedral Mesh Generation

3.2.1 Grid Based Methods

- quadtree-like (2D)
  - 9-tree over domain interior
  - all-quad templates
- octree-like (3D)
  - 27-tree over domain interior
  - all-hexa templates with one exception
- isomorphism technique (used to mesh boundary region between the domain boundary and interior tree)
- poor quality elements along boundary
- limited mesh density flexibility
- cannot handle domains with internal faces
  (multiple region and multiple material domains)

3.2.2 Advancing Front Based Methods

Paving and Plastering

- whole layer of elements is constructed along the part of the front at a time
- paving (2D)
  - all-quadrilateral meshes of high quality
  - even number of boundary segments must be maintained in each closed part of the front
- plastering (3D)
  - mixed meshes of good quality
  - seams and wedges used to resolve a conforming mesh closure in areas where layers would overlap or coincide with the boundary

Triangle Merging

- extension of 2D advancing front technique
  - merging two consequently generated triangles to form a quad
  - modification of the optimal point placement
  - preserving even number of segments in each closed part of the front
- high quality mesh

3.2.3 Topology Based Methods

Whisker Weaving

- spatial twisted continuum
- constructs a dual (topology based) representation of mesh (from a boundary discretization)
- identifies chords - chains of elements neighbouring by a facet
- mesh geometry is derived from topology

3.2.4 Postprocessing Based Methods

Simplex Splitting

- initial grid of half density
- triangles split into 3 quadrilaterals
- tetrahedrons split into 4 hexahedrons
- poor connectivity and quality (especially in 3D)

Triangle Merging

- initial triangular grid of half density
- merging neighbouring triangles forming well-shaped quadrilaterals
- one-level refinement
  - triangles split into 3 quadrilaterals
  - quadrilaterals split into 4 quadrilaterals
- high quality mesh
- unable to form single row mesh

References

[1] T.J. Baker, *Automatic mesh generation for complex three-dimensional regions using a constrained Delaunay triangulation*, Engineering with Computers **5** (1989), 161–175.

[2] T.D. Blacker, *Paving: A new approach to automated quadrilateral mesh generation*, International Journal for Numerical Methods in Engineering **32** (1991), 811–847.

[3] J. Bonet and J. Peraire, *An alternating digital tree (ADT) algorithm for 3D geometric searching and intersection problem*, International Journal for Numerical Methods in Engineering **31** (1991), 1–17.

[4] A. Bowyer, *Computing Dirichlet tessellations*, Computer Journal **24** (1981), 162–167.

[5] P.J. Frey, H. Borouchaki, and P.L. George, *Delaunay tetrahedrization using an advancing-front approach*, Proceedings of 5th International Meshing Roundtable, Sandia National Laboratories, 1996.

[6] H. Jin and R.I. Tanner, *Generation of unstructured tetrahedral meshes by*

advancing front technique, International Journal for Numerical Methods in Engineering **36** (1993), 1805–1823.

[7] B.P. Johnston, J.M. Sullivan, and A. Kwasnik, *Automatic conversion of triangular finite element meshes to quadrilateral elements*, International Journal for Numerical Methods in Engineering **31** (1991), 67–84.

[8] C.K. Lee and S.H. Lo, *A new scheme for the generation of graded quadrilateral meshes*, Computers and Structures **52** (1994), 847–857.

[9] S.H. Lo, *A new mesh generation scheme for arbitrary planar domains*, International Journal for Numerical Methods in Engineering **21** (1985), 1403–1426.

[10] R. Löhner and P. Parikh, *Three dimensional grid generation by the advancing front technique*, International Journal for Numerical Methods in Fluids **8** (1988), 1135–1149.

[11] P. Möller and P. Hansbo, *On advancing front mesh generation in three dimensions*, International Journal for Numerical Methods in Engineering **38** (1995), 3551–3569.

[12] M.A. Price, C.G. Armstrong, and M.A. Sabin, *Hexahedral mesh generation*

by medial axis subdivisions: I. solids with convex edges, 1994, Submitted to International Journal for Numerical Methods in Engineering.

[13] R. Schneiders, *A grid-based algorithm for the generation of hexahedral element meshes*, Engineering with Computers **12** (1996), 168–177.

[14] R. Schneiders, R. Schindler, and F. Weiler, *Octree-based generation of hexahedral element meshes*, Proceedings of 5th International Meshing Roundtable, Sandia National Laboratories, 1996.

[15] W.J. Schroeder and M.S. Shephard, *A combined octree/Delaunay method for fully automatic 3-D mesh generation*, International Journal for Numerical Methods in Engineering **29** (1990), 37–55.

[16] M.S. Shephard and M.K. Georges, *Automatic three-dimensional mesh generation by the finite octree technique*, International Journal for Numerical Methods in Engineering **26** (1991), 709–749.

[17] ———, *Reliability of automatic 3-D mesh generation*, Computer Methods in Applied Mechanics and Engineering **101** (1992), 443–462.

[18] M.B. Stephenson, S. Canann, and T.D. Blacker, *Plastering: A new approach to automated 3D hexahedral mesh generation*, Tech. Report SAND 89-2192,

Sandia National Laboratories, 1990.

[19] T.J. Tautges, T.D. Blacker, and S. Mitchell, *The whisker weaving algorithm: A connectivity-based method for constructing all-hexahedral finite element meshes*, International Journal for Numerical Methods in Engineering **39** (1996), 3327–3349.

[20] D.F. Watson, *Computing the n-dimensional Delaunay tessellation with application to Voronoi polytopes*, The Computer Journal **24** (1981), 167–172.

[21] N.P. Weatherill and O. Hassan, *Efficient three-dimensional Delaunay triangulation with automatic point creation and imposed boundary constraints*, International Journal for Numerical Methods in Engineering **37** (1994), 2005–2039.

[22] J.Z. Zhu, O.C. Zienkiewicz, E. Hinton, and J. Wu, *A new approach to the development of automatic quadrilateral mesh generation*, International Journal for Numerical Methods in Engineering **32** (1991), 849–866.