A brief introduction to approximate solution of boundary value problems (BVP)

Yuri Vassilevski

Institute of Numerical Mathematics
Russian Academy of Science

The Rome-Moscow school of Matrix Methods and Applied Linear Algebra

September 2011, Moscow

1. Unstructured mesh generation technologies (9.09.11)
2. Solution of linear systems with square sparse matrices (12.09.11)
3. Discretization of BVP on unstructured meshes (13.09.11)

1. Unstructured mesh generation technologies (9.09.11)
2. Solution of linear systems with square sparse matrices (12.09.11)
3. Discretization of BVP on unstructured meshes (13.09.11)

Lecture + Practical work

Unstructured mesh generation technologies

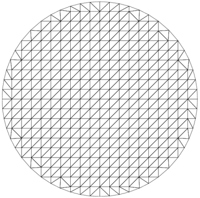
Basic types of computational meshes

Basic types of computational meshes

structured (ij)
1950'

Basic types of computational meshes

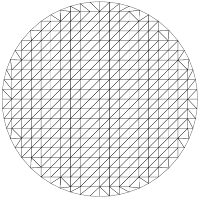
structured (ij)
1950'



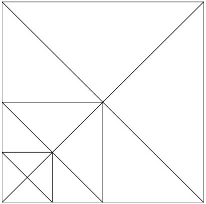
Locally fitted
1970'

Basic types of computational meshes

structured (ij)
1950'



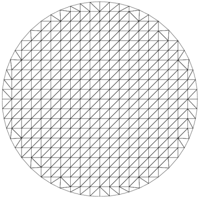
Locally fitted
1970'



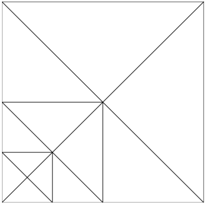
hierarchical
1980'

Basic types of computational meshes

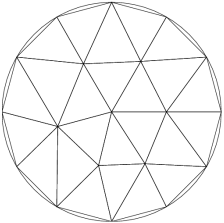
structured (ij)
1950'



Locally fitted
1970'



hierarchical
1980'



unstructured
1980'

Unstructured conformal triangular/tetrahedral meshes

Unstructured conformal triangular/tetrahedral meshes

Advantages

Unstructured conformal triangular/tetrahedral meshes

Advantages

+ General 2D/3D domains

Unstructured conformal triangular/tetrahedral meshes

Advantages

+ General 2D/3D domains
+ Meshing technologies

Unstructured conformal triangular/tetrahedral meshes

Advantages

+ General 2D/3D domains
+ Meshing technologies
+ Adaptation flexibility

Unstructured conformal triangular/tetrahedral meshes

Advantages

+ General 2D/3D domains
+ Meshing technologies
+ Adaptation flexibility

Drawbacks

Unstructured conformal triangular/tetrahedral meshes

Advantages

+ General 2D/3D domains
+ Meshing technologies
+ Adaptation flexibility

Drawbacks

− Many cells per node

Unstructured conformal triangular/tetrahedral meshes

Advantages

+ General 2D/3D domains
+ Meshing technologies
+ Adaptation flexibility

Drawbacks

− Many cells per node
− Non-smoothness

Unstructured conformal triangular/tetrahedral meshes

Advantages

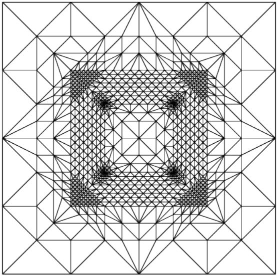
+ General 2D/3D domains
+ Meshing technologies
+ Adaptation flexibility

Drawbacks

− Many cells per node
− Non-smoothness
− Non-orthogonality

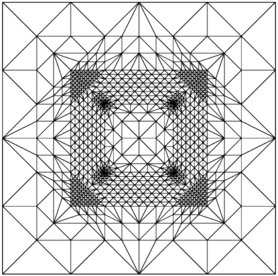
Types of mesh adaptation

Types of mesh adaptation

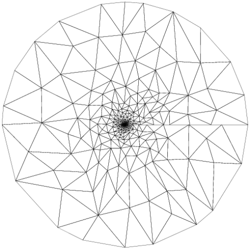


hierarchical

Types of mesh adaptation

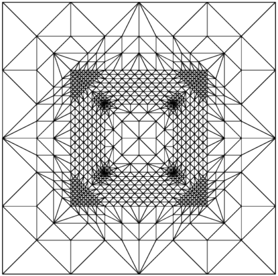


hierarchical

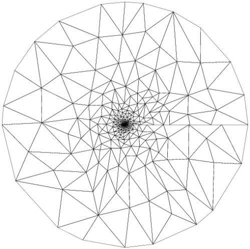


regular

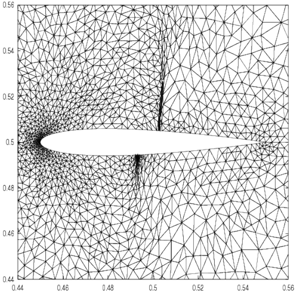
Types of mesh adaptation



hierarchical



regular



anisotropic

Ani2D	Ani3D
`www.sf.net/projects/ani2d`	`www.sf.net/projects/ani3d`
4500 downloads	2600 downloads

K.Lipnikov, Y.Vassilevski, A.Danilov, V.Chugunov, ...

Alternatives: FreeFEM (F.Hecht), ALBERTA (K.Siebert et al.), PLTMG (R.Bank) ...

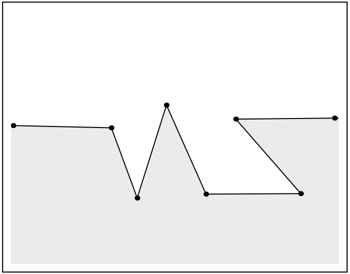
Statement I:

Unstructured meshes
can be generated
in complex domains

Generation of unstructured triangulations

Two basic methods

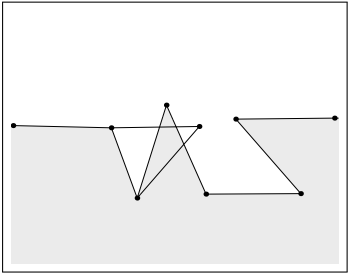
1. Advancing front technique
2. Delaunay triangulation



1. Select element from front
2. Construct triangle
   and check intersections
3. If intersection occurs
   consider neighbouring vertices
4. Add triangle into mesh, update front

R. Löhner, P. Parikh, *Generation of three-dimensional unstructured grids by the advancing front method.* Int.J.Numer.Meth.Fluids, 1988, V.8, p.1135–1149.

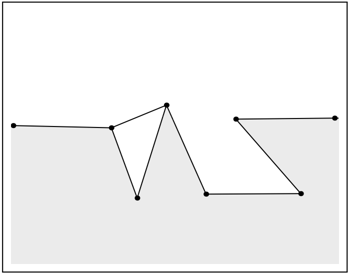
A. Danilov, *Unstructured mesh generation technology.* Comp.Math. Math.Phys., 2010, V.50, p.139–156.



1. Select element from front
2. Construct triangle and check intersections
3. If intersection occurs consider neighbouring vertices
4. Add triangle into mesh, update front

R. Löhner, P. Parikh, *Generation of three-dimensional unstructured grids by the advancing front method.* Int.J.Numer.Meth.Fluids, 1988, V.8, p.1135–1149.

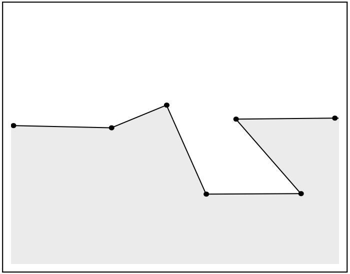
A. Danilov, *Unstructured mesh generation technology.* Comp.Math. Math.Phys., 2010, V.50, p.139–156.



1. Select element from front
2. Construct triangle
   and check intersections
3. If intersection occurs
   consider neighbouring vertices
4. Add triangle into mesh, update front

R. Löhner, P. Parikh, *Generation of three-dimensional unstructured grids by the advancing front method.* Int.J.Numer.Meth.Fluids, 1988, V.8, p.1135–1149.

A. Danilov, *Unstructured mesh generation technology.* Comp.Math. Math.Phys., 2010, V.50, p.139–156.



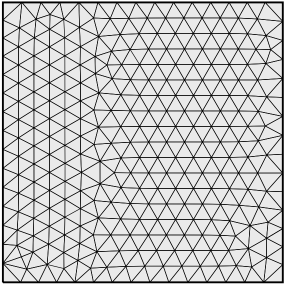
1. Select element from front
2. Construct triangle
   and check intersections
3. If intersection occurs
   consider neighbouring vertices
4. Add triangle into mesh, update front

R. Löhner, P. Parikh, *Generation of three-dimensional unstructured grids by the advancing front method.* Int.J.Numer.Meth.Fluids, 1988, V.8, p.1135–1149.

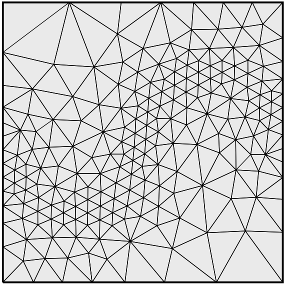
A. Danilov, *Unstructured mesh generation technology.* Comp.Math. Math.Phys., 2010, V.50, p.139–156.

Generation of unstructured triangulations

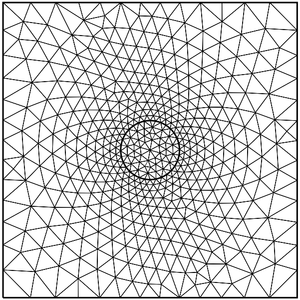
Choice of the mesh step in AFT



- Quasiuniform mesh
- User defined mesh size
- Automatic coarsening



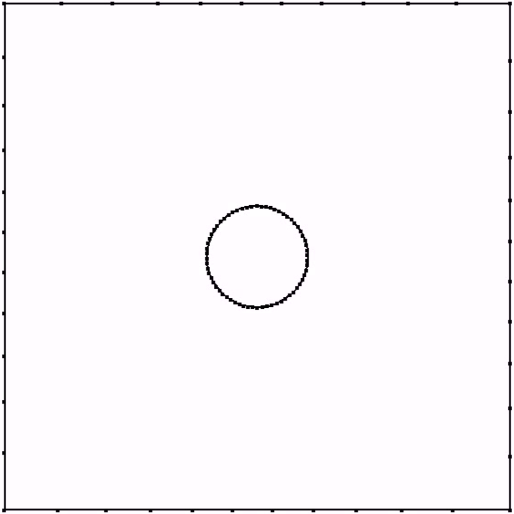
- Quasiuniform mesh
- User defined mesh size
- Automatic coarsening



- Quasiuniform mesh
- User defined mesh size
- Automatic coarsening

Generation of unstructured triangulations

Advancing front meshing



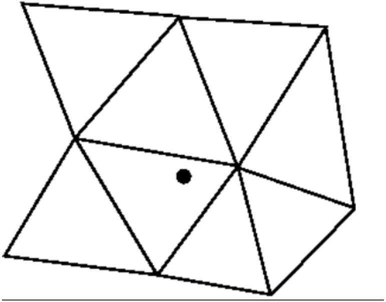
- In Delaunay triangulation (DT) of nodes $V_n = \{v_1, \dots, v_n\}$ for any triangle the circumscribed circle does not contain vertices of other triangles
- Any conformal triangulation can be modified to be DT using edge flips
- The simplest generation of DT is iterative: given DT for V_k, add v_{k+1} and modify mesh to be DT for V_{k+1}

- In Delaunay triangulation (DT) of nodes $V_n = \{v_1, \ldots, v_n\}$ for any triangle the circumscribed circle does not contain vertices of other triangles
- Any conformal triangulation can be modified to be DT using edge flips
- The simplest generation of DT is iterative: given DT for V_k, add v_{k+1} and modify mesh to be DT for V_{k+1}

- In Delaunay triangulation (DT) of nodes $V_n = \{v_1, \dots, v_n\}$ for any triangle the circumscribed circle does not contain vertices of other triangles
- Any conformal triangulation can be modified to be DT using edge flips
- The simplest generation of DT is iterative: given DT for V_k, add v_{k+1} and modify mesh to be DT for V_{k+1}

Generation of unstructured triangulations

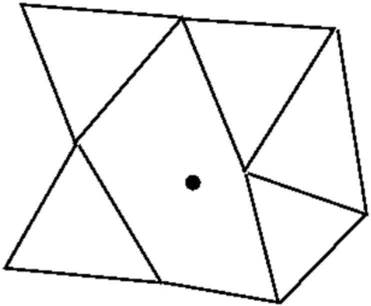
Single step in DT generation (convex set)



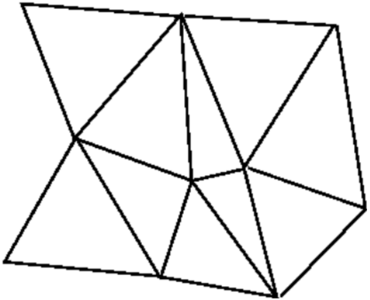
- Addition of new node
- Deletion of triangles violating DT condition
- Generation of new triangles

Generation of unstructured triangulations

Single step in DT generation (convex set)



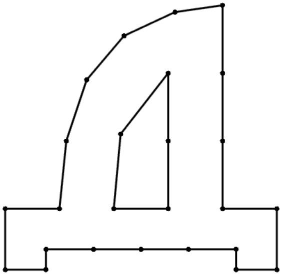
- Addition of new node
- Deletion of triangles violating DT condition
- Generation of new triangles



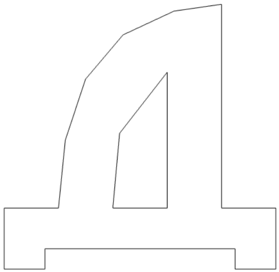
- Addition of new node
- Deletion of triangles violating DT condition
- Generation of new triangles

Advancing front technique vs. Delaunay triangulation

Advancing front technique



Delaunay triangulation



Generation of tetrahedral meshes (overview)

1. Geometry representation
2. Surface meshing
3. Volume meshing

Generation of tetrahedral meshes (volume meshing)

AFT $\to$ Delaunay $\to$ Cosmetics

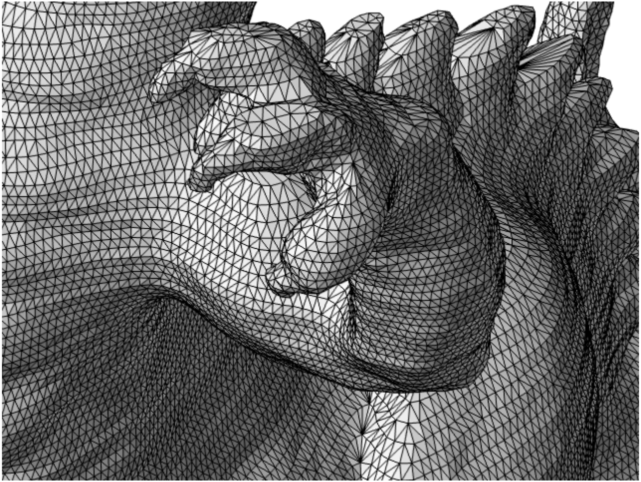
Three stages:

1. Advancing front technique (99-100%)
2. Delaunay tetrahedrization of remaining lacunas (0.1%)
3. Mesh cosmetics

K.Morton, B.Soni, P.George, ...

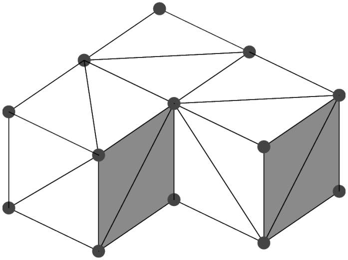
Generation of tetrahedral meshes (volume meshing)

Front dynamics in AFT



Generation of tetrahedral meshes (volume meshing)

Delaunay-based technique meshes unmeshed isolated lacunas



initial front

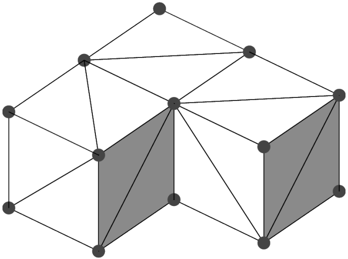
P.-L. George, H. Borouchaki, E. Saltel, *'Ultimate' robustness in meshing an arbitrary polyhedron.* Int. J. Numer. Meth. Eng., 2003, V.58, p.1061–1089.

A. Danilov, *Unstructured mesh generation technology.* Comp.Math. Math.Phys., 2010, V.50, p.139–156.

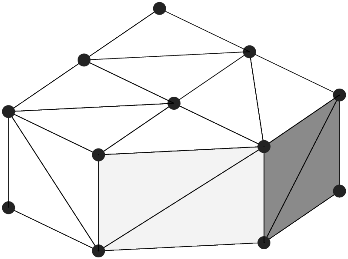
AFT+DT build a topologically correct mesh but may leave a few slivers behind

Generation of tetrahedral meshes (volume meshing)

Delaunay-based technique meshes unmeshed isolated lacunas



initial front



Delaunay triangulation

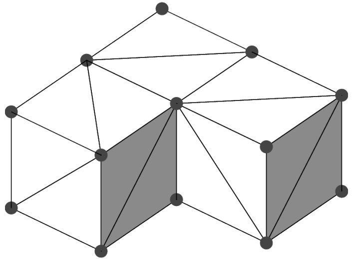
P.-L. George, H. Borouchaki, E. Saltel, *'Ultimate' robustness in meshing an arbitrary polyhedron.* Int. J. Numer. Meth. Eng., 2003, V.58, p.1061–1089.

A. Danilov, *Unstructured mesh generation technology.* Comp.Math. Math.Phys., 2010, V.50, p.139–156.

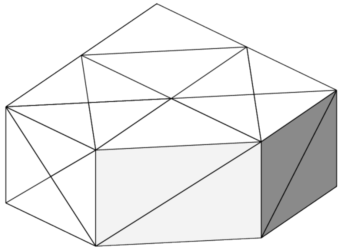
AFT+DT build a topologically correct mesh but may leave a few slivers behind

Generation of tetrahedral meshes (volume meshing)

Delaunay-based technique meshes unmeshed isolated lacunas



initial front



refined mesh

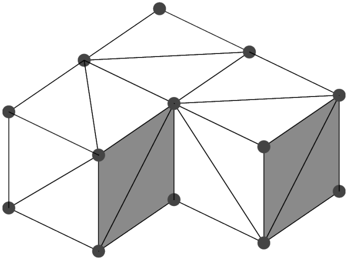
P.-L. George, H. Borouchaki, E. Saltel, *'Ultimate' robustness in meshing an arbitrary polyhedron.* Int. J. Numer. Meth. Eng., 2003, V.58, p.1061–1089.

A. Danilov, *Unstructured mesh generation technology.* Comp.Math. Math.Phys., 2010, V.50, p.139–156.

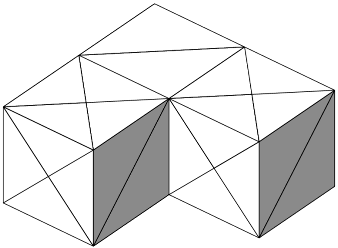
AFT+DT build a topologically correct mesh but may leave a few slivers behind

Generation of tetrahedral meshes (volume meshing)

Delaunay-based technique meshes unmeshed isolated lacunas



initial front



geometry recovery

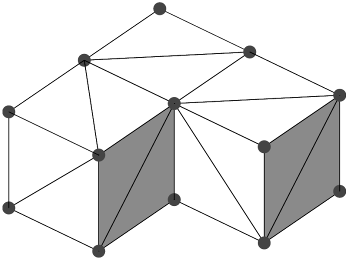
P.-L. George, H. Borouchaki, E. Saltel, *'Ultimate' robustness in meshing an arbitrary polyhedron.* Int. J. Numer. Meth. Eng., 2003, V.58, p.1061–1089.

A. Danilov, *Unstructured mesh generation technology.* Comp.Math. Math.Phys., 2010, V.50, p.139–156.

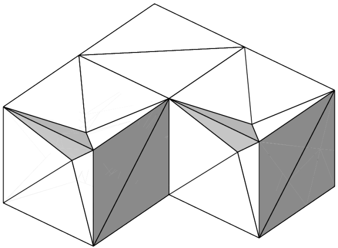
AFT+DT build a topologically correct mesh but may leave a few slivers behind

Generation of tetrahedral meshes (volume meshing)

Delaunay-based technique meshes unmeshed isolated lacunas



initial front



boundary faces recovery

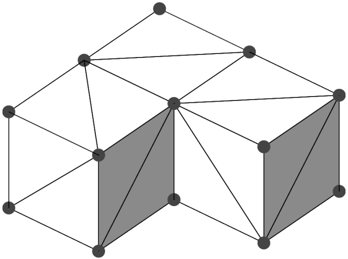
P.-L. George, H. Borouchaki, E. Saltel, *'Ultimate' robustness in meshing an arbitrary polyhedron.* Int. J. Numer. Meth. Eng., 2003, V.58, p.1061–1089.

A. Danilov, *Unstructured mesh generation technology.* Comp.Math. Math.Phys., 2010, V.50, p.139–156.

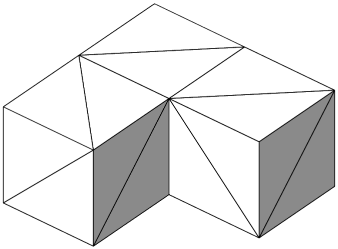
AFT+DT build a topologically correct mesh but may leave a few slivers behind

Generation of tetrahedral meshes (volume meshing)

Delaunay-based technique meshes unmeshed isolated lacunas



initial front



final mesh

P.-L. George, H. Borouchaki, E. Saltel, *'Ultimate' robustness in meshing an arbitrary polyhedron.* Int. J. Numer. Meth. Eng., 2003, V.58, p.1061–1089.

A. Danilov, *Unstructured mesh generation technology.* Comp.Math. Math.Phys., 2010, V.50, p.139–156.

AFT+DT build a topologically correct mesh but may leave a few slivers behind

Mesh cosmetics removes slivers by changing mesh topology via a sequence of local mesh modifications that increase quality of the worst mesh element

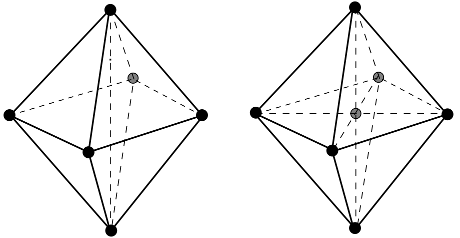
The tetrahedron shape quality $Q_s(\Delta)$ consists of a scaling factor and a shape controlling factor:

$$Q_s(\Delta) = 6^4\sqrt{2}\,\frac{|\Delta|}{p(\Delta)^3}$$

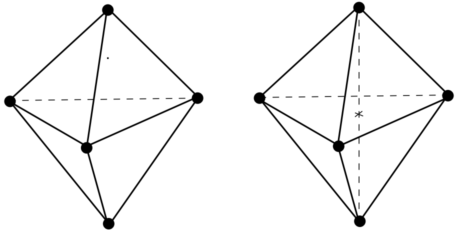
- $|\Delta|$ is the volume of Δ
- $p(\Delta)$ is the "perimeter" of Δ, i.e the sum of length of its edges:

$$p(\Delta) = \sum_{k=1}^{6} |\mathbf{e}_k|$$

Local topological operations



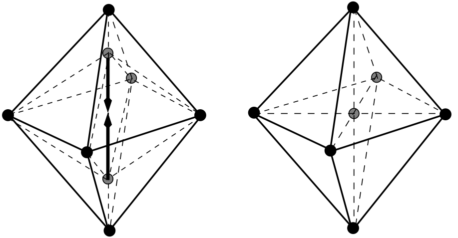
Op 1: point insertion



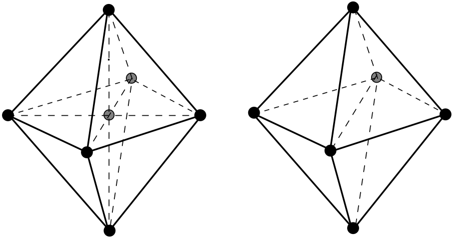
Ops 2 and 3: edge-face swaps

Coupez, Buscaglia, Dari, Freitag, Ollivier-Gooch, Joe, Misztal, Shewchuk, ...

Local topological operations



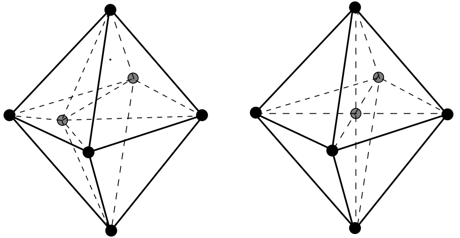
Op 4: edge collapsing



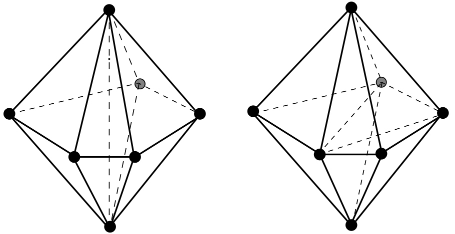
Op 5: point deletion

Locality is the key to robustness, rich set of operations provides faster convergence

Local topological operations



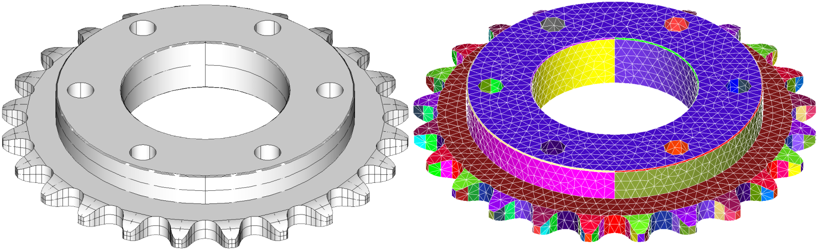
Op 6: point relocation



Op 7: generalized edge-face swapping

Locality is the key to robustness, rich set of operations provides faster convergence

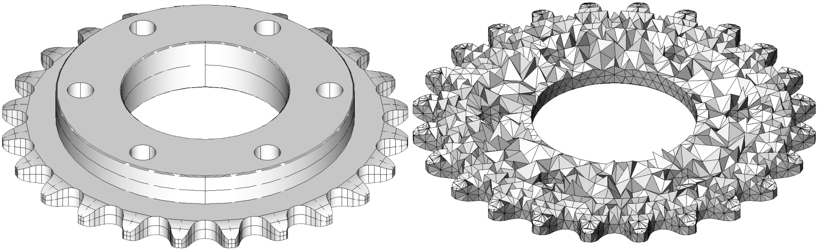
Gear model



CAD model of the gear has 422 vertices, 636 curvilinear edges, and 217 curvilinear faces

AFT re-meshed the surface mesh into a quasiuniform mesh with 19405 vertices and 38834 triangles

Gear model



AFT failed to mesh 0.04% of the domain and left 352 triangles in the front, DT meshed successfully the lacunas

Mesh cosmetics removed slivers from the mesh

Gear model

Tetrahedra distribution by quality shows reduction of the mesh quality after DT step and significant quality improvement after mesh cosmetics

Method	$Q(\Omega_h)$	N_T	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}
AFT	$1.01 \cdot 10^{-3}$	140442	139684	750	8	—	—
AFT+DT	$1.60 \cdot 10^{-5}$	140723	139817	854	37	5	10
AFT+DT+MC	$2.00 \cdot 10^{-1}$	156538	156538	—	—	—	—

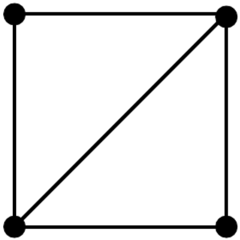
Total time of mesh construction is 9 min and 33 sec
Surface meshing of CAD model is 8 minutes and 40 seconds
Volume meshing with the AFT and DT is 41 seconds
Mesh cosmetics is 12 seconds

Statement II:

Conformal meshes can be hierarchically locally refined and derefined

Refinement by marked edge bisection

support of conformity



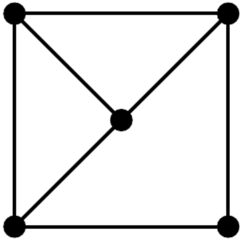
- Initial mesh
- Bisection of triangle
- Recovery of conformity

  (bisection of triangles where it is lost)

E. Bänsh, *Local mesh refinement in 2 and 3 dimensions.* IMPACT of Computing in Science and Engrg., 1991, V.3, p.181–191.

Refinement by marked edge bisection

support of conformity



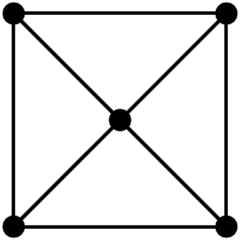
- Initial mesh
- Bisection of triangle
- Recovery of conformity

  (bisection of triangles where it is lost)

E. Bänsh, *Local mesh refinement in 2 and 3 dimensions.* IMPACT of Computing in Science and Engrg., 1991, V.3, p.181–191.

Refinement by marked edge bisection

support of conformity



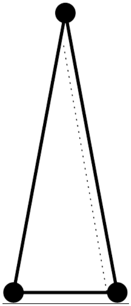
- Initial mesh
- Bisection of triangle
- Recovery of conformity

  (bisection of triangles where it is lost)

E. Bänsh, *Local mesh refinement in 2 and 3 dimensions.* IMPACT of Computing in Science and Engrg., 1991, V.3, p.181–191.

Refinement by marked edge bisection

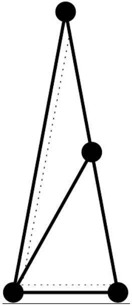
Bänsch method



- Triangle and marked edge
- 1st bisection and new marked edges
- 2nd bisection and new marked edges

Refinement by marked edge bisection

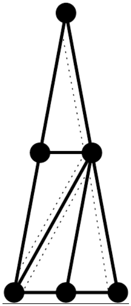
Bänsch method



- Triangle and marked edge
- 1st bisection and new marked edges
- 2nd bisection and new marked edges

Refinement by marked edge bisection

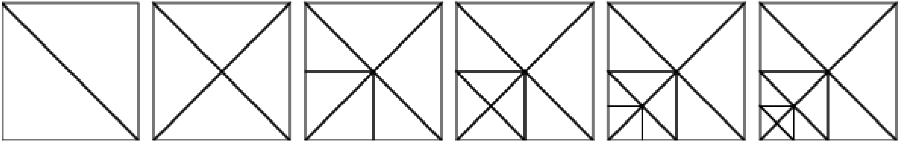
Bänsch method



- Triangle and marked edge
- 1st bisection and new marked edges
- 2nd bisection and new marked edges

Refinement by marked edge bisection

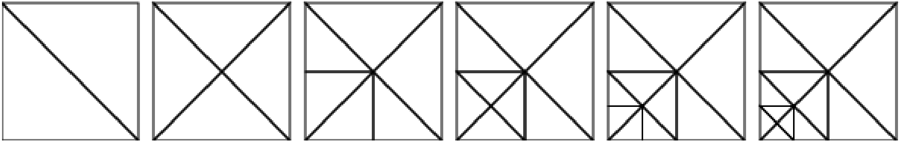
Sequence of refined triangulations



- Shape regularity is preserved: minimal angle is reduced at most by factor 2 independently of levels
- Multilevel bisection of conformal tetrahedral meshes (ani3D)

Refinement by marked edge bisection

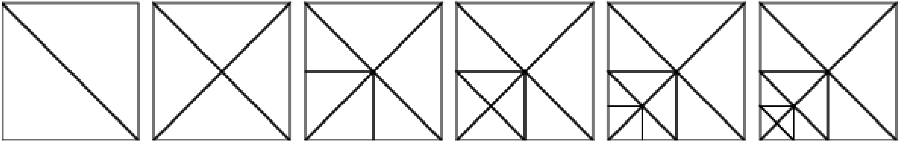
Sequence of refined triangulations



- Shape regularity is preserved: minimal angle is reduced at most by factor 2 independently of levels
- Multilevel bisection of conformal tetrahedral meshes (ani3D)

Refinement by marked edge bisection

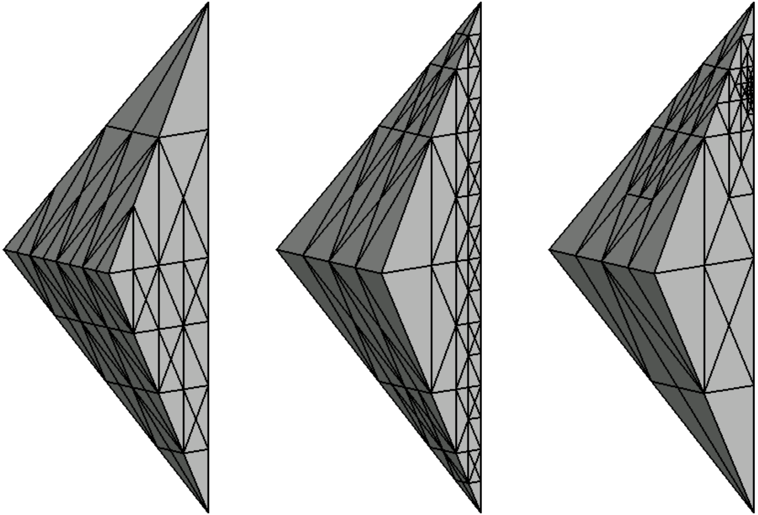
Sequence of refined triangulations



- Shape regularity is preserved: minimal angle is reduced at most by factor 2 independently of levels
- Multilevel bisection of conformal tetrahedral meshes (ani3D)

Refinement by marked edge bisection

Examples of refined tetrahedrizations



Coarsening by marked edge bisection

- Only refined cells can be coarsened
- Only cells with the same refinement level can be merged
- After merging conformity must be recovered
- History of refinement must be kept

Coarsening by marked edge bisection

- Only refined cells can be coarsened
- Only cells with the same refinement level can be merged
- After merging conformity must be recovered
- History of refinement must be kept

Coarsening by marked edge bisection

- Only refined cells can be coarsened
- Only cells with the same refinement level can be merged
- After merging conformity must be recovered
- History of refinement must be kept

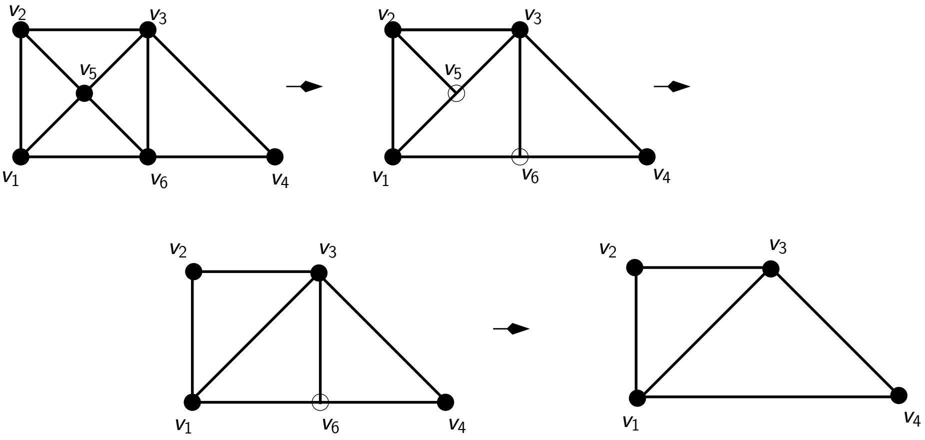
Coarsening by marked edge bisection

- Only refined cells can be coarsened
- Only cells with the same refinement level can be merged
- After merging conformity must be recovered
- History of refinement must be kept

Coarsening by marked edge bisection

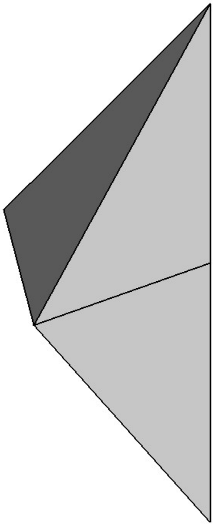
- Only refined cells can be coarsened
- Only cells with the same refinement level can be merged
- After merging conformity must be recovered
- History of refinement must be kept

Coarsening by marked edge bisection



Generation of dynamic meshes

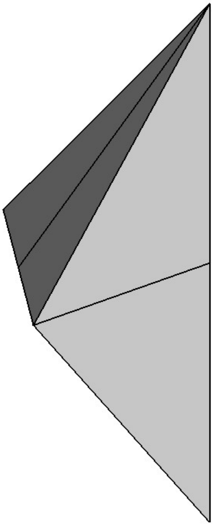
Marked edge bisection algorithm



AniRCB package from Ani3D library

Generation of dynamic meshes

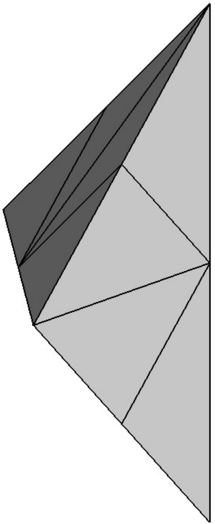
Marked edge bisection algorithm



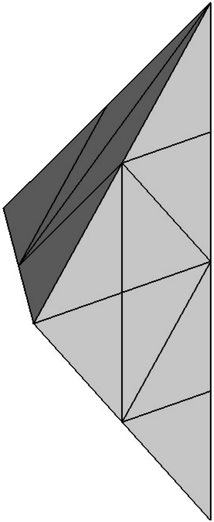
AniRCB package from Ani3D library

Generation of dynamic meshes

Marked edge bisection algorithm



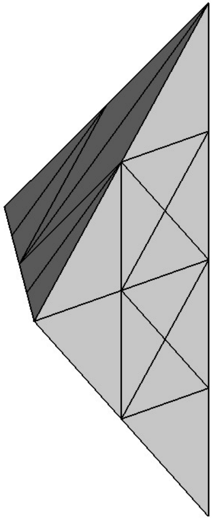
AniRCB package from Ani3D library



AniRCB package from Ani3D library

Generation of dynamic meshes

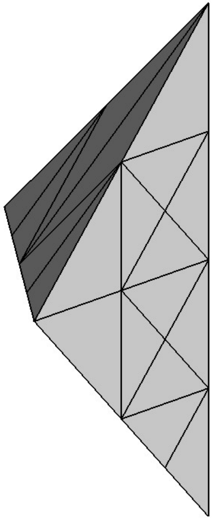
Marked edge bisection algorithm



AniRCB package from Ani3D library

Generation of dynamic meshes

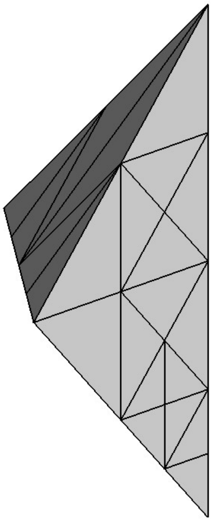
Marked edge bisection algorithm



AniRCB package from Ani3D library

Generation of dynamic meshes

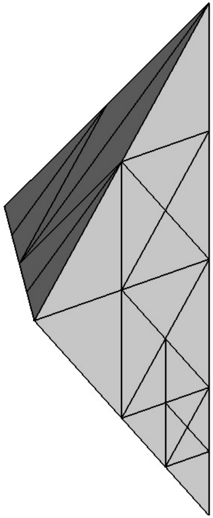
Marked edge bisection algorithm



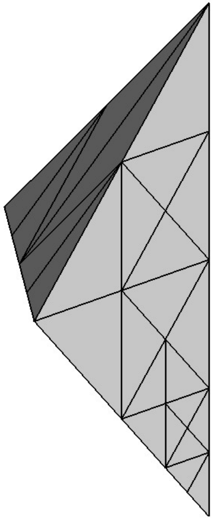
AniRCB package from Ani3D library

Generation of dynamic meshes

Marked edge bisection algorithm



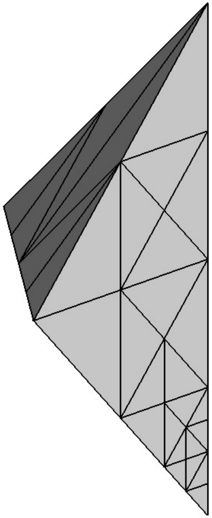
AniRCB package from Ani3D library



AniRCB package from Ani3D library

Generation of dynamic meshes

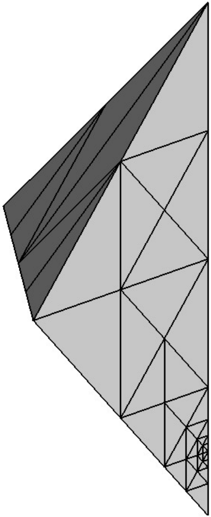
Marked edge bisection algorithm



AniRCB package from Ani3D library

Generation of dynamic meshes

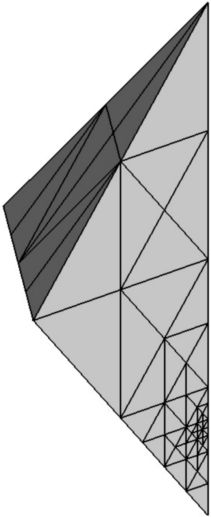
Marked edge bisection algorithm



AniRCB package from Ani3D library

Generation of dynamic meshes

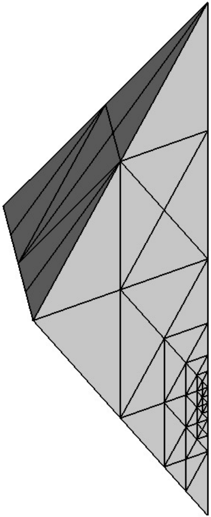
Marked edge bisection algorithm



AniRCB package from Ani3D library

Generation of dynamic meshes

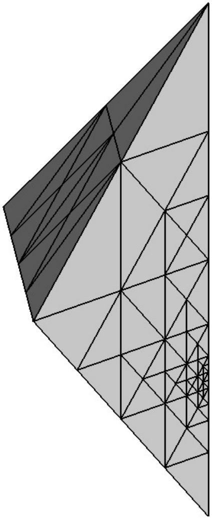
Marked edge bisection algorithm



AniRCB package from Ani3D library

Generation of dynamic meshes

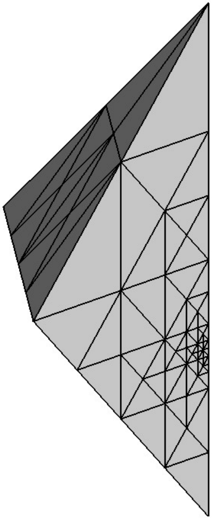
Marked edge bisection algorithm



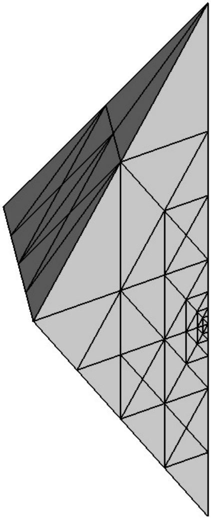
AniRCB package from Ani3D library

Generation of dynamic meshes

Marked edge bisection algorithm



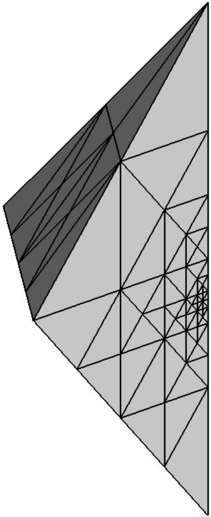
AniRCB package from Ani3D library



AniRCB package from Ani3D library

Generation of dynamic meshes

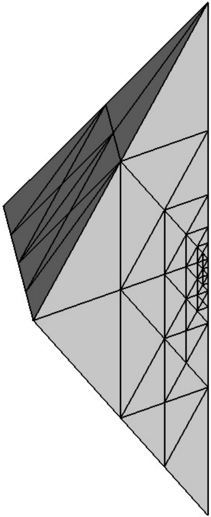
Marked edge bisection algorithm



AniRCB package from Ani3D library

Generation of dynamic meshes

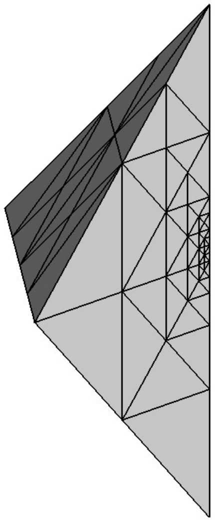
Marked edge bisection algorithm



AniRCB package from Ani3D library

Generation of dynamic meshes

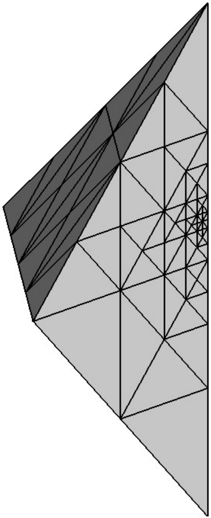
Marked edge bisection algorithm



AniRCB package from Ani3D library

Generation of dynamic meshes

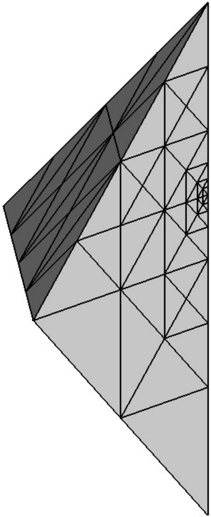
Marked edge bisection algorithm



AniRCB package from Ani3D library

Generation of dynamic meshes

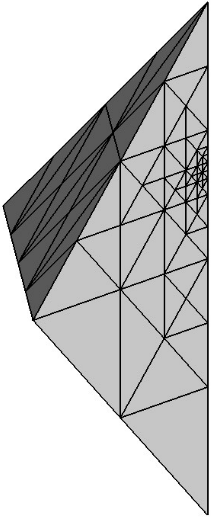
Marked edge bisection algorithm



AniRCB package from Ani3D library

Generation of dynamic meshes

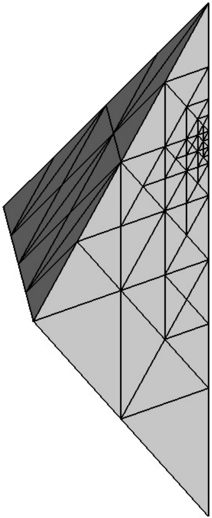
Marked edge bisection algorithm



AniRCB package from Ani3D library

Generation of dynamic meshes

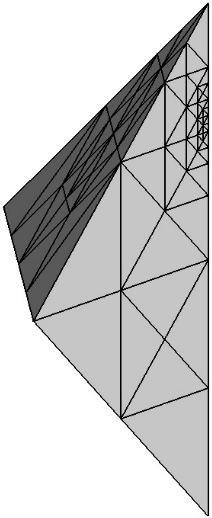
Marked edge bisection algorithm



AniRCB package from Ani3D library

Generation of dynamic meshes

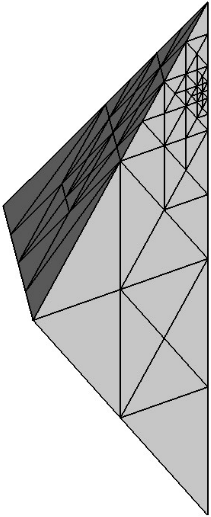
Marked edge bisection algorithm



AniRCB package from Ani3D library

Generation of dynamic meshes

Marked edge bisection algorithm



AniRCB package from Ani3D library

Statement III:

Conformal meshes can be controlled by a tensor metric field

Tensor metric $\mathfrak{M} = \mathfrak{M}^T > 0$

$$\mathfrak{M} = \begin{bmatrix} M_{xx} & M_{xy} \\ M_{yx} & M_{yy} \end{bmatrix}, \qquad \mathfrak{M} = \begin{bmatrix} M_{xx} & M_{xy} & M_{xz} \\ M_{yx} & M_{yy} & M_{yz} \\ M_{zx} & M_{zy} & M_{zz} \end{bmatrix}$$

- Area/volume of domain D:

$$|D|_{\mathfrak{M}} = \int_D \sqrt{\det(\mathfrak{M}(\mathbf{x}))}\,\mathrm{d}V \approx |D|\sqrt{\det(\mathfrak{M}(\mathbf{x}_*))}$$

- Length of parameterized curve ℓ:

$$|\ell|_{\mathfrak{M}} = \int_0^1 \sqrt{\gamma'(t)^T\mathfrak{M}(\gamma(t))\gamma'(t)}\,\mathrm{d}t$$

  Length of parameterized edge $\mathbf{x} = \mathbf{x}_1 + t(\mathbf{x}_2 - \mathbf{x}_1)$:

$$|\mathbf{e}|_{\mathfrak{M}} = \int_0^1 \sqrt{(\mathbf{x}_2 - \mathbf{x}_1)^T\mathfrak{M}(\gamma(t))(\mathbf{x}_2 - \mathbf{x}_1)}\,\mathrm{d}t \approx \sqrt{(\mathbf{x}_1 - \mathbf{x}_2)^T\mathfrak{M}(\mathbf{x}_{12})(\mathbf{x}_1 - \mathbf{x}_2)}$$

- "Perimeter" of triangle/tetrahedron: $p_{\mathfrak{M}}(\Delta) = \sum_{k=1}^{n_{edges}} |\mathbf{e}_k|_{\mathfrak{M}}$

Tensor metric $\mathfrak{M} = \mathfrak{M}^T > 0$

$$\mathfrak{M} = \begin{bmatrix} M_{xx} & M_{xy} \\ M_{yx} & M_{yy} \end{bmatrix}, \qquad \mathfrak{M} = \begin{bmatrix} M_{xx} & M_{xy} & M_{xz} \\ M_{yx} & M_{yy} & M_{yz} \\ M_{zx} & M_{zy} & M_{zz} \end{bmatrix}$$

- Area/volume of domain D:

$$|D|_{\mathfrak{M}} = \int_D \sqrt{\det(\mathfrak{M}(\mathbf{x}))}\,\mathrm{d}V \approx |D|\sqrt{\det(\mathfrak{M}(\mathbf{x}_*))}$$

- Length of parameterized curve ℓ:

$$|\ell|_{\mathfrak{M}} = \int_0^1 \sqrt{\gamma'(t)^T\mathfrak{M}(\gamma(t))\gamma'(t)}\,\mathrm{d}t$$

  Length of parameterized edge $\mathbf{x} = \mathbf{x}_1 + t(\mathbf{x}_2 - \mathbf{x}_1)$:

$$|\mathbf{e}|_{\mathfrak{M}} = \int_0^1 \sqrt{(\mathbf{x}_2 - \mathbf{x}_1)^T\mathfrak{M}(\gamma(t))(\mathbf{x}_2 - \mathbf{x}_1)}\,\mathrm{d}t \approx \sqrt{(\mathbf{x}_1 - \mathbf{x}_2)^T\mathfrak{M}(\mathbf{x}_{12})(\mathbf{x}_1 - \mathbf{x}_2)}$$

- "Perimeter" of triangle/tetrahedron: $p_{\mathfrak{M}}(\Delta) = \sum_{k=1}^{n_{edges}} |\mathbf{e}_k|_{\mathfrak{M}}$

Tensor metric $\mathfrak{M} = \mathfrak{M}^T > 0$

$$\mathfrak{M} = \begin{bmatrix} M_{xx} & M_{xy} \\ M_{yx} & M_{yy} \end{bmatrix}, \qquad \mathfrak{M} = \begin{bmatrix} M_{xx} & M_{xy} & M_{xz} \\ M_{yx} & M_{yy} & M_{yz} \\ M_{zx} & M_{zy} & M_{zz} \end{bmatrix}$$

- Area/volume of domain D:

$$|D|_{\mathfrak{M}} = \int_D \sqrt{\det(\mathfrak{M}(\mathbf{x}))}\,\mathrm{d}V \approx |D|\sqrt{\det(\mathfrak{M}(\mathbf{x}_*))}$$

- Length of parameterized curve ℓ:

$$|\ell|_{\mathfrak{M}} = \int_0^1 \sqrt{\gamma'(t)^T \mathfrak{M}(\gamma(t))\gamma'(t)}\,\mathrm{d}t$$

Length of parameterized edge $\mathbf{x} = \mathbf{x}_1 + t(\mathbf{x}_2 - \mathbf{x}_1)$:

$$|\mathbf{e}|_{\mathfrak{M}} = \int_0^1 \sqrt{(\mathbf{x}_2 - \mathbf{x}_1)^T \mathfrak{M}(\gamma(t))(\mathbf{x}_2 - \mathbf{x}_1)}\,\mathrm{d}t \approx \sqrt{(\mathbf{x}_1 - \mathbf{x}_2)^T \mathfrak{M}(\mathbf{x}_{12})(\mathbf{x}_1 - \mathbf{x}_2)}$$

- "Perimeter" of triangle/tetrahedron: $p_{\mathfrak{M}}(\Delta) = \sum_{k=1}^{n_{edges}} |\mathbf{e}_k|_{\mathfrak{M}}$

Tensor metric $\mathfrak{M} = \mathfrak{M}^T > 0$

$$\mathfrak{M} = \begin{bmatrix} M_{xx} & M_{xy} \\ M_{yx} & M_{yy} \end{bmatrix}, \qquad \mathfrak{M} = \begin{bmatrix} M_{xx} & M_{xy} & M_{xz} \\ M_{yx} & M_{yy} & M_{yz} \\ M_{zx} & M_{zy} & M_{zz} \end{bmatrix}$$

- Area/volume of domain D:

$$|D|_{\mathfrak{M}} = \int_D \sqrt{\det(\mathfrak{M}(\mathbf{x}))}\,\mathrm{d}V \approx |D|\sqrt{\det(\mathfrak{M}(\mathbf{x}_*))}$$

- Length of parameterized curve ℓ:

$$|\ell|_{\mathfrak{M}} = \int_0^1 \sqrt{\gamma'(t)^T \mathfrak{M}(\gamma(t))\gamma'(t)}\,\mathrm{d}t$$

  Length of parameterized edge $\mathbf{x} = \mathbf{x}_1 + t(\mathbf{x}_2 - \mathbf{x}_1)$:

$$|\mathbf{e}|_{\mathfrak{M}} = \int_0^1 \sqrt{(\mathbf{x}_2 - \mathbf{x}_1)^T \mathfrak{M}(\gamma(t))(\mathbf{x}_2 - \mathbf{x}_1)}\,\mathrm{d}t \approx \sqrt{(\mathbf{x}_1 - \mathbf{x}_2)^T \mathfrak{M}(\mathbf{x}_{12})(\mathbf{x}_1 - \mathbf{x}_2)}$$

- "Perimeter" of triangle/tetrahedron: $p_{\mathfrak{M}}(\Delta) = \sum_{k=1}^{n_{edges}} |\mathbf{e}_k|_{\mathfrak{M}}$

Mesh shape quality

Quality of triangle Δ in metric $\mathfrak{M}$:

$$Q_{\mathfrak{M}}(\Delta) = 12\sqrt{3}\frac{|\Delta|_{\mathfrak{M}}}{p_{\mathfrak{M}}(\Delta)^2}$$

Mesh shape quality:

$$Q_{\mathfrak{M}}(\Omega_h) = \min_{\Delta\in\Omega_h} Q_{\mathfrak{M}}(\Delta)$$

Mesh shape quality

Quality of tetrahedron Δ in metric $\mathfrak{M}$:

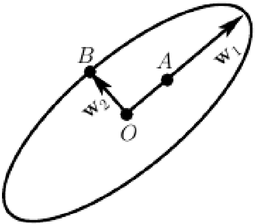
$$Q_{\mathfrak{M}}(\Delta) = 6^4\sqrt{2}\frac{|\Delta|_{\mathfrak{M}}}{p_{\mathfrak{M}}(\Delta)^3}$$

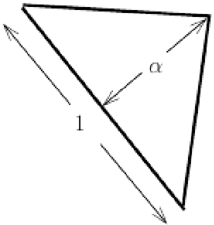
Mesh shape quality:

$$Q_{\mathfrak{M}}(\Omega_h) = \min_{\Delta\in\Omega_h} Q_{\mathfrak{M}}(\Delta)$$

Example

$$\mathfrak{M} = \begin{bmatrix} 5.89 & 2.51 \\ 2.51 & 5.01 \end{bmatrix}, \quad \lambda_1 = 8.0, \quad \lambda_2 = 2.9$$





$\mathfrak{M}$-equilateral triangle
height $\alpha = \sqrt{3\lambda_2/4\lambda_1}$

Control of mesh properties

Given tensor metric field $\mathfrak{M}(\mathbf{x})$ and desirable number of cells $N_\star$, we generate by a sequence of local modifications a $\mathfrak{M}$-quasiuniform mesh with $N_\star$ cells.

- $h_\star$ is a mesh size of $\mathfrak{M}$-quasiuniform mesh with $N_\star$ cells:

$$h_\star = \left(\frac{1}{N_\star\, V_d} \int_\Omega \sqrt{\det(\mathfrak{M}(\mathbf{x}))}\, dV \right)^{1/d}.$$

- $F(\cdot)$ is a smooth positive function with the only maximum $F(1) = 1$

Mesh quality:

$$Q(\Omega_h) = \min_{\Delta \in \Omega_h} Q(\Delta)$$

Monotone increase of $Q(\Omega_h)$ by a set of local modifications

Control of mesh properties

Quality of triangle Δ in metric $\mathfrak{M}$:

$$Q_{\mathfrak{M},N_\star}(\Delta) = 12\sqrt{3}\,\overbrace{\frac{|\Delta|_{\mathfrak{M}}}{p_{\mathfrak{M}}(\Delta)^2}}^{\textbf{shape}} F\overbrace{\left(\frac{p_{\mathfrak{M}}(\Delta)}{3h_\star}\right)}^{\textbf{size}}$$

- $h_\star$ is a mesh size of $\mathfrak{M}$-quasiuniform mesh with $N_\star$ cells:

$$h_\star = \left(\frac{1}{N_\star\, V_d}\int_\Omega \sqrt{\det(\mathfrak{M}(\mathbf{x}))}dV\right)^{1/d}.$$

- $F(\cdot)$ is a smooth positive function with the only maximum $F(1) = 1$

Mesh quality:

$$Q(\Omega_h) = \min_{\Delta\in\Omega_h} Q(\Delta)$$

Monotone increase of $Q(\Omega_h)$ by a set of local modifications

Control of mesh properties

Quality of tetrahedron Δ in metric $\mathfrak{M}$:

$$Q_{\mathfrak{M},N_\star}(\Delta) = 6^4\sqrt{2}\,\overbrace{\frac{|\Delta|_{\mathfrak{M}}}{p_{\mathfrak{M}}(\Delta)^3}}^{\textbf{shape}} \overbrace{F\left(\frac{p_{\mathfrak{M}}(\Delta)}{6h_\star}\right)}^{\textbf{size}}$$

- $h_\star$ is a mesh size of $\mathfrak{M}$-quasiuniform mesh with $N_\star$ cells:

$$h_\star = \left(\frac{1}{N_\star\, V_d}\int_\Omega \sqrt{\det(\mathfrak{M}(\mathbf{x}))}dV\right)^{1/d}.$$

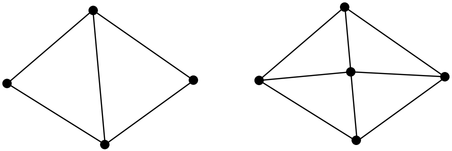
- $F(\cdot)$ is a smooth positive function with the only maximum $F(1) = 1$

Mesh quality:

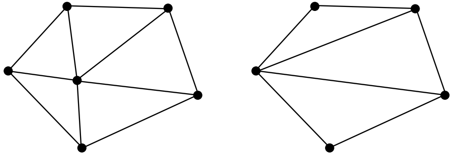
$$Q(\Omega_h) = \min_{\Delta\in\Omega_h} Q(\Delta)$$

Monotone increase of $Q(\Omega_h)$ by a set of local modifications

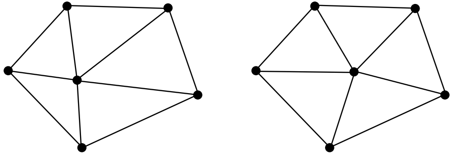
Local topological operations in 2D



Op 1: point insertion

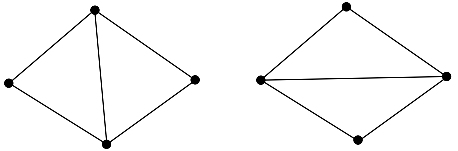


Op 2: point deletion

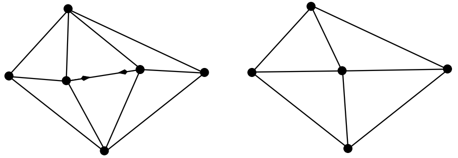


Op 3: point relocation

Local topological operations in 2D



Op 4: edge swap

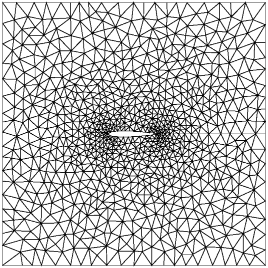


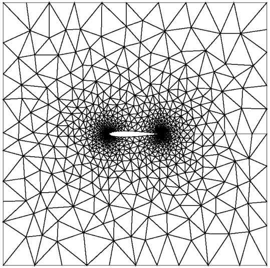
Op 5: edge collapsing

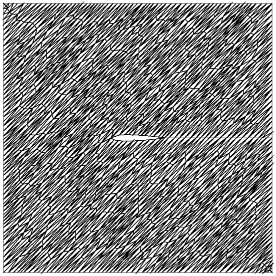
Locality is the key to robustness,
rich set of operations provides faster convergence

Examples of mesh control

Choice of $\mathfrak{M}$







$$h(\mathbf{x})^{-2}\mathbb{I}_2$$

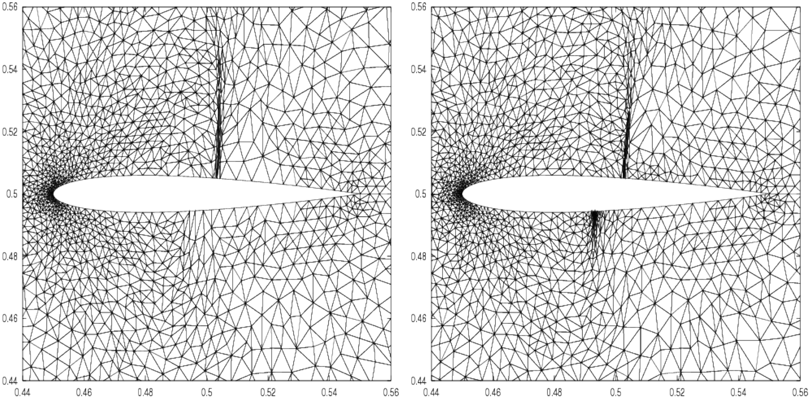
$$\max\{|(x-0.4)^2+(y-0.5)^2+10^{-4}|^{a/2},\\ |(x-0.6)^2+(y-0.5)^2+10^{-4}|^{a/2}\}$$

$$\mathbb{R}_{\pi/4}\begin{bmatrix} 100 & 0 \\ 0 & 1 \end{bmatrix}\mathbb{R}^T_{\pi/4}$$

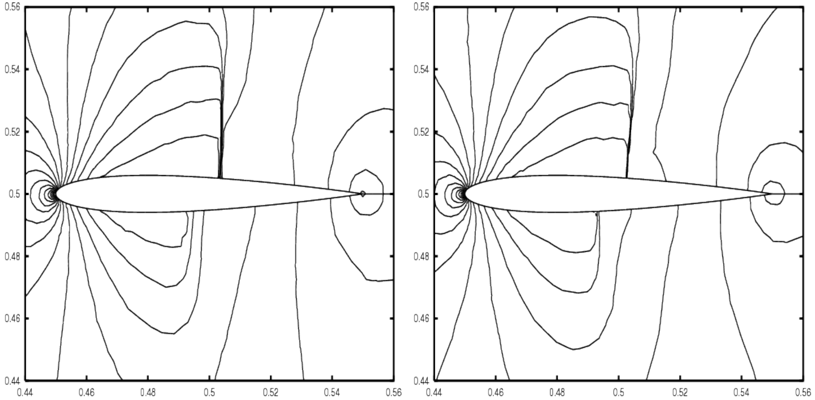
Statement IV:

Conformal meshes can be adapted isotropically and anisotropically

Transonic potential flow



Transonic potential flow



Example of adaptive loop

Initialization Step. Generate an initial triangulation Ω^h. Choose the final mesh quality Q_0, $Q_0 < 1$, and the final number $N_\star$ of mesh elements.

Iterative Step.

1. Compute the discrete solution $\mathcal{P}_{\Omega^h} u$ for triangulation Ω^h.
2. Recover the tensor metric field $\mathfrak{M}$ from $\mathcal{P}_{\Omega^h} u$.
   Stop iterations if $Q_{\mathfrak{M},N_\star}(\Omega^h) \geq Q_0$.
3. Generate the next mesh $\widetilde{\Omega}^h$ such that $Q_{\mathfrak{M},N_\star}(\widetilde{\Omega}^h) \geq Q_0$.
4. Set $\Omega^h = \widetilde{\Omega}^h$ and go to 1.

Approaches to recovery of $\mathfrak{M}$

- Recover discrete Hessian
  - + black-box
  - — lack of analysis, error control
- Use a posteriori error estimates
  - — problem dependent
  - + theory and error estimates exist

Approaches to recovery of $\mathfrak{M}$

- Recover discrete Hessian
  - $+$ black-box
  - $-$ lack of analysis, error control

- Use a posteriori error estimates
  - $-$ problem dependent
  - $+$ theory and error estimates exist

Approaches to recovery of $\mathfrak{M}$

- Recover discrete Hessian
  - \+ black-box
  - — lack of analysis, error control

- Use a posteriori error estimates
  - — problem dependent
  - \+ theory and error estimates exist

Metric and Hessian

If the adaptation goal is to minimize P_1-interpolation error

$$\|u - \mathcal{P}_{\Omega^h} u\|_{L_p(\Omega)}, \qquad 0 < p \leq \infty,$$

take

$$\mathfrak{M}(x) = (det|H(x)|)^{-1/(2p+2)}|H(x)|$$

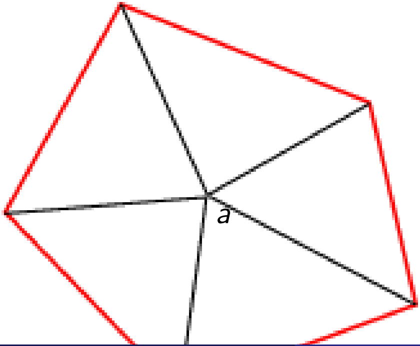
where H is the Hessian of u.

- $|H| = W^T|\Lambda|W$ from local spectral decomposition $H = W^T\Lambda W$.
- u is unknown, hence H is replaced with H^h. Then $\mathfrak{M} \leftarrow |H^h|$.

Variational recovery of Hessian

Weak definition of the Hessian

$$\int_\sigma H_{ij}^h(a)\,\phi_a^h\,\mathrm{d}x = -\int_\sigma \frac{\partial u^h}{\partial x_i}\frac{\partial \phi_a^h}{\partial x_j}\,\mathrm{d}x$$



a interior point

$\phi_a^h(a)$ is P_1-basis function for a.

$$\mathfrak{M} = \begin{bmatrix} \mathfrak{m}_{11} & \mathfrak{m}_{12} \\ \mathfrak{m}_{12} & \mathfrak{m}_{22} \end{bmatrix}$$

Theorem. Let α_k be the errors prescribed to edges of a triangle Δ such that

$$\alpha_k \geq 0 \qquad \text{and} \qquad \sum_{k=1}^{\#edges} \alpha_k > 0.$$

Then, there exists a constant tensor metric $\mathfrak{M}$ such that

$$0.4\,|\Delta|_{\mathfrak{M}} \leq \sum_{k=1}^{\#edges} \alpha_k \leq p_{\mathfrak{M}}(\Delta)^2.$$

A. Agouzal, Yu. Vassilevski *Minimization of gradient errors of piecewise linear interpolation on simplicial meshes.* Comp.Meth. Appl.Mech.Engnr., 2010, V.199, p.2195–2203.

Conclusions

- Unstructured simplicial meshes can be generated in complex domains
- Simplicial meshes can be hierarchically locally refined and derefined
- Simplicial meshes can be controlled by a tensor metric field
- Simplicial meshes can be adapted isotropically and anisotropically
- Next step: practical work with Ani2D

Conclusions

- Unstructured simplicial meshes can be generated in complex domains
- Simplicial meshes can be hierarchically locally refined and derefined
- Simplicial meshes can be controlled by a tensor metric field
- Simplicial meshes can be adapted isotropically and anisotropically
- Next step: practical work with Ani2D

Conclusions

- Unstructured simplicial meshes can be generated in complex domains
- Simplicial meshes can be hierarchically locally refined and derefined
- Simplicial meshes can be controlled by a tensor metric field
- Simplicial meshes can be adapted isotropically and anisotropically
- Next step: practical work with Ani2D

Conclusions

- Unstructured simplicial meshes can be generated in complex domains
- Simplicial meshes can be hierarchically locally refined and derefined
- Simplicial meshes can be controlled by a tensor metric field
- Simplicial meshes can be adapted isotropically and anisotropically
- Next step: practical work with Ani2D

Conclusions

- Unstructured simplicial meshes can be generated in complex domains
- Simplicial meshes can be hierarchically locally refined and derefined
- Simplicial meshes can be controlled by a tensor metric field
- Simplicial meshes can be adapted isotropically and anisotropically
- Next step: practical work with Ani2D