

# Techniques for Mesh Movement and Curved Mesh Generation for Computational Fluid Dynamics

Aditya Kashi

Department of Mechanical and Aerospace Engineering,  
North Carolina State University

April 27, 2016

# Table of Contents

## 1 Section I : Introduction

- Need for mesh movement
- Need for curved meshes

## 2 Section II : Mesh movement

- Review of methods
- Large rotation in inviscid mesh
- Mesh quality
- Large rotation with mesh quality

## 3 Curved mesh generation

- Viscous 3-component airfoil
- 3D Bump Channel Turbulent Flow Case

## 4 Conclusion

# Why mesh movement?

Mesh movement is useful in several areas of CFD

- aeroelasticity, and fluid-structure interaction in general
- shape optimization
- generation of curved meshes for spatially high-order discretizations
- others

# Fluid-Structure Interaction

- For a body-fitted grid, robust mesh-movement is required to maintain validity and quality of the mesh after imposing motion of the structural domain<sup>1</sup>.
- Immersed boundary methods can also be used; both have advantages and disadvantages<sup>2</sup>.

---

<sup>1</sup>C. Farhat and M. Lesoinne. "Two efficient staggered algorithms for the serial and parallel solution of three-dimensional nonlinear transient aeroelastic problems". In: *Comput. Methods Appl. Mech. Engrg.* 182 (2000), pp. 499–515.

<sup>2</sup>G. Hou, J. Wang, and A. Layton. "Numerical methods for fluid-structure interaction - a review". In: *Commun. Comput. Phys.* 12.2 (2012), pp. 337–377.

# Shape optimization

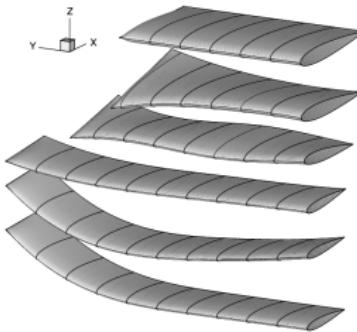


Figure : Convergence history of lift-constrained drag minimization

3

<sup>3</sup>J.E. Hicken and D.W. Zingg. "Aerodynamic optimization algorithm with integrated geometry parameterization and mesh movement". In: *AIAA Journal* 48.2 (2010), pp. 400–413.

# High-order methods

- According to Wang, Fidkowski *et. al.*<sup>4</sup>, spatially high-order methods perform better than prevailing second order methods for some kinds of simulations considering CPU time taken to achieve a given error level.
- One area of challenge they mention is generation of high-order meshes.

---

<sup>4</sup>Z.J. Wang et al. “High-order CFD methods: current status and perspective”. In: *Intl. J. Numer. Meth. Fluids* 72 (2013), pp. 811–845.



# The need for curved meshes

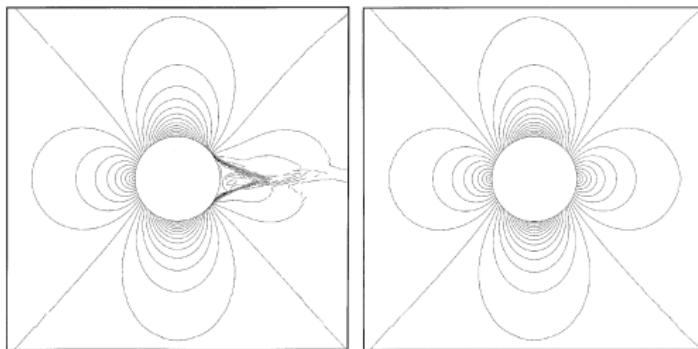


Figure : Inviscid subsonic flow over a cylinder; left: DGP1 solution with regular linear mesh, right: DGP1 solution with quadratic ('Q2') mesh<sup>5</sup>

---

<sup>5</sup>F. Bassi and S. Rebay. "High-order accurate discontinuous finite element solution of the 2D Euler equations". In: *J. Comput. Phys.* 128 (1997), pp. 251–285.

# The need for curved meshes

Even in less extreme cases, curved meshes are required to obtain design  $(p+1)$  order of accuracy for high-order methods such as discontinuous Galerkin methods<sup>6</sup>.

---

<sup>6</sup>X. Luo, M.S. Shephard, and J.-F. Remacle. "The influence of geometric approximation on the accuracy of high order methods". In: *Rensselaer SCOREC report 1* (2001).

# Table of Contents

## 1 Section I : Introduction

- Need for mesh movement
- Need for curved meshes

## 2 Section II : Mesh movement

- Review of methods
- Large rotation in inviscid mesh
- Mesh quality
- Large rotation with mesh quality

## 3 Curved mesh generation

- Viscous 3-component airfoil
- 3D Bump Channel Turbulent Flow Case

## 4 Conclusion

# Introduction

The problem is to move the interior nodes of a mesh when a given displacement is imposed on the boundary.

- At the very least: mesh elements should not get invalidated.
- Mesh elements should not suffer much deterioration in quality.
- The technique should be computationally inexpensive.

# Element validity for finite element methods

Since we cannot integrate over arbitrary sets, physical elements are mapped to a reference element using a geometric mapping.

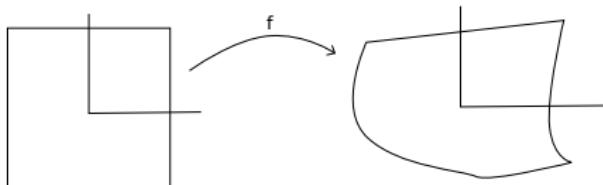


Figure : Geometric mapping between reference element  $S$  (left) and physical element  $\Omega$  (right)

Change of variables theorem: if the mapping  $f : S \rightarrow \Omega$  is bijective and differentiable, we can integrate over the reference element instead. This requires the Jacobian of  $f$ ,  $Df$  to be nonsingular throughout the element.

Many mesh-movement methods for unstructured meshes can be found in literature. They can be broadly classified into two types.

- Elasticity-based methods
- Interpolation methods

Combination of these with each other and with other techniques such as topological (connectivity) smoothing are also used<sup>7</sup>.

---

<sup>7</sup>F. Alauzet. "A changing-topology moving mesh technique for large displacements". In: *Engrg. Comput.* 30 (2014), pp. 175–200.



# Lineal spring analogy

Every mesh edge is treated as a linear spring in each coordinate direction<sup>8</sup>.

$$\sum_j k_{ij}(\Delta \mathbf{r}_i - \Delta \mathbf{r}_j) = \mathbf{0} \quad \forall i \quad (1)$$

where  $i$  ranges over all nodes,  $j$  ranges over points surrounding node  $i$  and  $\Delta \mathbf{r}_i$  is the displacement of node  $i$ .  $k_{ij}$  is the stiffness of the spring between nodes  $i$  and  $j$ , which can be taken as

$$k_{ij} = \frac{1}{\|\mathbf{r}_i - \mathbf{r}_j\|}. \quad (2)$$

---

<sup>8</sup>J.T. Batina. “Unsteady Euler Algorithm with Unstructured Dynamic Mesh for Complex-Aircraft Aerodynamic Analysis”. In: *AIAA Journal* 29.3 (1991), pp. 327–333.

# Lineal spring analogy

This scheme requires the solution of a (SPD) linear system of size  $N_n$  (the number of mesh nodes),  $n_{dim}$  (the spatial dimension of the problem) times.

# Torsional springs

The lineal spring analogy fails for large deformations or stretched elements. Farhat *et. al.* came up with a more robust scheme, which is also a spring analogy<sup>9</sup>. They introduce two improvements over Batina's model.

- The model is closer to a structural analogy in that the displacements in each coordinate direction are coupled.
- 'Torsional springs' are introduced at each node in each element. These are designed to prevent edges collapsing into each other due to rotational motion.

---

<sup>9</sup>C. Farhat et al. "Torsional springs for two-dimensional dynamic unstructured fluid meshes". In: *Comput. meth. appl. mech. engrg.* 163 (1998), pp. 231–245.

# Torsional springs

The stiffness of the torsional springs at a node in an element is inversely related to the node angle in that element.

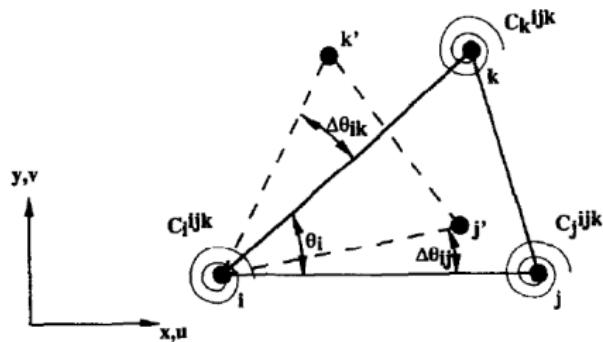


Figure : Movement of an element in torsion spring analogy

A coupled SPD system of  $n_{dim}N_n$  equations must be solved.

# Linear elasticity

The mesh is assumed to model a deformable solid body, which is then deformed according to the equations of solid mechanics, that is, linear or non-linear elasticity.

The simplest approach is linear elasticity.

$$\nabla \cdot \boldsymbol{\sigma} = \mathbf{0} \quad \text{in } \Omega \quad (3)$$

$$\boldsymbol{\sigma} = 2\mu\boldsymbol{\epsilon} + \lambda(\text{tr}\boldsymbol{\epsilon})\mathbf{I} \quad (4)$$

$$\boldsymbol{\epsilon} = \frac{1}{2}(\nabla \mathbf{u} + \nabla \mathbf{u}^T) \quad (5)$$

$$\mathbf{u} = \mathbf{u}_b \quad \text{on } \partial\Omega \quad (6)$$

# Stiffened Linear elasticity

- The linear elasticity scheme is often modified by ‘stiffening’ the mesh appropriately.
- We attain some control over the propagation of deformation into the interior of the mesh, as done, for instance, by Stein *et. al.*<sup>10</sup>.
- The material is stiffened based on the determinant of the local Jacobian matrix of the reference-to-physical mapping; i.e., smaller elements are stiffer than larger ones.

---

<sup>10</sup>K. Stein, T. Tezduyar, and R. Benney. “Mesh moving techniques for fluid-structure interactions with large displacements”. In: *J. Appl. Mech.* 70 (2003), pp. 58–63.

# Nonlinear elasticity

Claimed to be a highly robust method for mesh movement by Persson and Peraire<sup>11</sup>.

The constitutive equation (4) and strain-displacement relation (5) are replaced by the ‘neo-Hookean’ constitutive model

$$\boldsymbol{\sigma} = \mu((\boldsymbol{F}^T \boldsymbol{F}) \boldsymbol{F}^{-T} - \boldsymbol{F}^{-T}) + \lambda(\ln \det \boldsymbol{F}) \boldsymbol{F}^{-T} \quad (7)$$

Here,  $\boldsymbol{F} = \frac{\partial \mathbf{x}}{\partial \xi}$ ,  $\mathbf{x}$  is the physical position vector of a point with coordinate  $\xi$  in the reference configuration.

The system is solved using Newton-GMRES iterations.

---

<sup>11</sup>P.-O. Persson and J. Peraire. “Curved mesh generation and mesh refinement using lagrangian solid mechanics”. In: *47<sup>th</sup> AIAA Aerospace Sciences Meeting*. 2009.

# Elasticity-based methods

- Advantages: stiffened linear elasticity is found to be robust, and nonlinear elasticity is claimed to be very robust.
- Disadvantage: expensive!
- Also, implementation is dependent on element type and spatial dimension.

# Delaunay graph mapping (DGM)

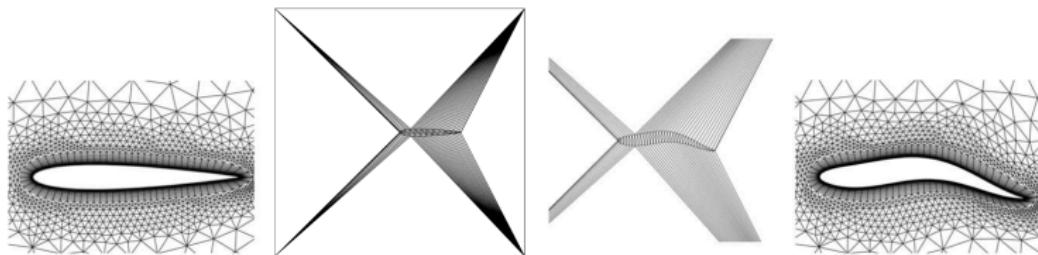


Figure : The DGM process (from left to right): original mesh, Delaunay graph, deformed Delaunay graph, deformed mesh (ref:<sup>12</sup>)

<sup>12</sup>X. Liu, N. Qin, and H. Xia. “Fast dynamic grid deformation based on Delaunay graph mapping”. In: *Jo. Comput. Phys.* 211 (2006), pp. 405–423.

# Radial basis function (RBF) interpolation

Considering  $n_b$  boundary points in the mesh, we express the displacement at any point in the mesh as<sup>13</sup>

$$\mathbf{s}(\mathbf{x}) = \sum_{j=1}^{n_b} \mathbf{a}_j \phi(\|\mathbf{x} - \mathbf{x}_{bj}\|) \quad (8)$$

$\phi$  is a radial basis function. As a function of  $\mathbf{x}$ , it is radially symmetric about the origin.

---

<sup>13</sup>A. de Boer, M.S. van der Schoot, and M. Bijl. "Mesh deformation based on radial basis function interpolation". In: *Computers & Structures* 85 (2007), pp. 784–795.

# RBF interpolation

Since we know the displacements of the boundary nodes, we can solve for the coefficients  $\mathbf{a}_j$  using

$$\mathbf{s}(\mathbf{x}_{ib}) = \sum_{j=1}^{n_b} \mathbf{a}_j \phi(\|\mathbf{x}_{bi} - \mathbf{x}_{bj}\|). \quad (9)$$

In each coordinate direction  $k$ , this leads to a system of  $n_b$  equations in  $n_b$  unknowns:

$$\mathbf{A}\mathbf{a}_k = \mathbf{s}_k \quad (10)$$

# RBF interpolation

The quality of the final mesh depends on which RBF is used.  
We use Wendland's  $C^2$  function<sup>14</sup>

$$\phi(x) = \begin{cases} \left(1 - \frac{x}{r_s}\right)^4 \left(4\frac{x}{r_s} + 1\right) & x < r_s \\ 0 & x \geq r_s \end{cases} \quad (11)$$

$r_s$  is a real number called the 'support radius'.

- Compact support - leads to sparse LHS
- Positive definite - systems can (usually) be solved quickly
- $C^2$  - smooth mesh movement

---

<sup>14</sup>H. Wendland. "Error estimates for interpolation by compactly supported radial basis functions of minimal degree". In: *J. Approx. Theory* 93 (1998), pp. 258–272.

# RBF interpolation

Catch: The condition number of the left hand side scales as

$$\begin{aligned}\text{cond}_2(\mathbf{A}) &= \mathcal{O}(q^{-5}) \quad \text{in 2D, and} \\ \text{cond}_2(\mathbf{A}) &= \mathcal{O}(q^{-6}) \quad \text{in 3D}\end{aligned}\tag{12}$$

where

$$q := \min_{i \neq j} \|\mathbf{x}_{bi} - \mathbf{x}_{bj}\|_2 \tag{13}$$

is the minimum distance between any two boundary nodes.  
Iterative solvers do not always work! Sparse direct solvers are found to be a robust option.

# RBF interpolation on a Delaunay Graph (DGRBF)

Wang *et. al.*<sup>15</sup> interpolate the displacements over each Delaunay graph element using RBFs instead of barycentric coordinates. The displacement of a node with initial position  $\mathbf{x}$  is given by

$$\mathbf{s}(\mathbf{x}) = \sum_{j=1}^{n_t} \mathbf{a}_j \phi(\|\mathbf{x} - \mathbf{x}_{tj}\|) \quad (14)$$

where  $\mathbf{x}_{tj}$  are positions of nodes of the Delaunay simplex that contains the node,  $n_t$  is the number of nodes in that simplex (3 in 2D and 4 in 3D).

<sup>15</sup>Y. Wang, N. Qin, and N. Zhao. “Delaunay graph and radial basis function for fast quality mesh deformation”. In: *J. Comput. Phys.* 294 (2015), pp. 149–172.

# DGRBF

We need to solve for the  $\mathbf{a}_j$ , the RBF coefficients, by solving a  $3 \times 3$  system in 2D or a  $4 \times 4$  system in 3D, in each Delaunay element.

$$\mathbf{s}(\mathbf{x}_{ti}) = \sum_{j=1}^{n_t} \mathbf{a}_j \phi(\|\mathbf{x}_{ti} - \mathbf{x}_{tj}\|). \quad (15)$$

This is more robust than DGM, but not as robust as RBF for large complex deformations.

# DGRBF with angle interpolation - 'DGRBF2'

For large rotational deformation, Wang *et. al.* interpolate *rotation angles* from the boundary to the interior nodes. If  $\mathbf{s}$  is the rotation angle instead of displacement, we get DGRBF2.

Can recover displacements using the usual rotation transformations.

In 2D

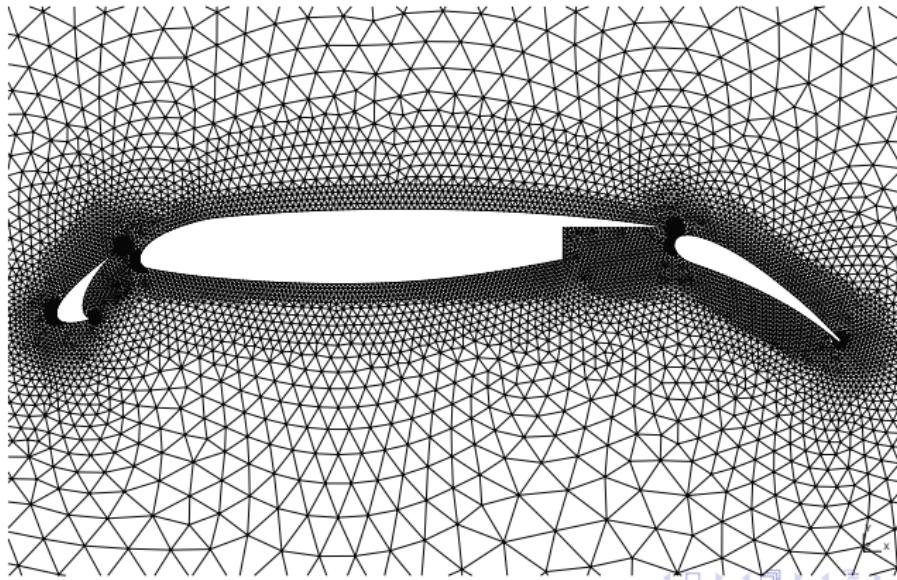
$$x_{new} = (x - x_0) \cos a_z - (y - y_0) \sin a_z + x_0 \quad (16)$$

$$y_{new} = (x - x_0) \sin a_z + (y - y_0) \cos a_z + y_0 \quad (17)$$

But this is difficult for general movement.

# Large rotation case 1

We first present visual results of large rotation for an inviscid flow mesh of a 3-component airfoil. The flap is rotated while keeping the rest of the boundary fixed.



# Torsion springs

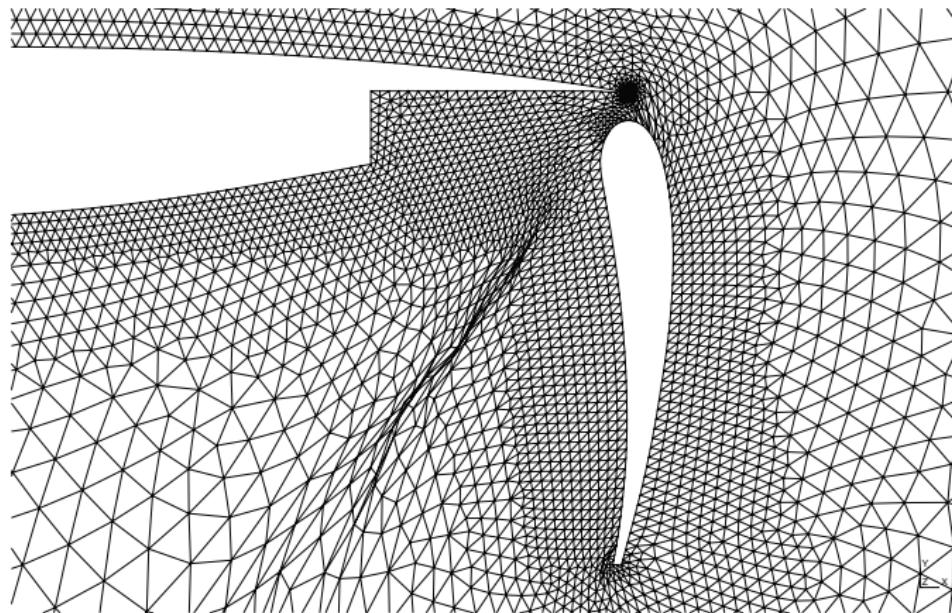


Figure : Inviscid 3-component airfoil mesh with  $60^\circ$  rotation of flap by torsion spring method

# Linear elasticity

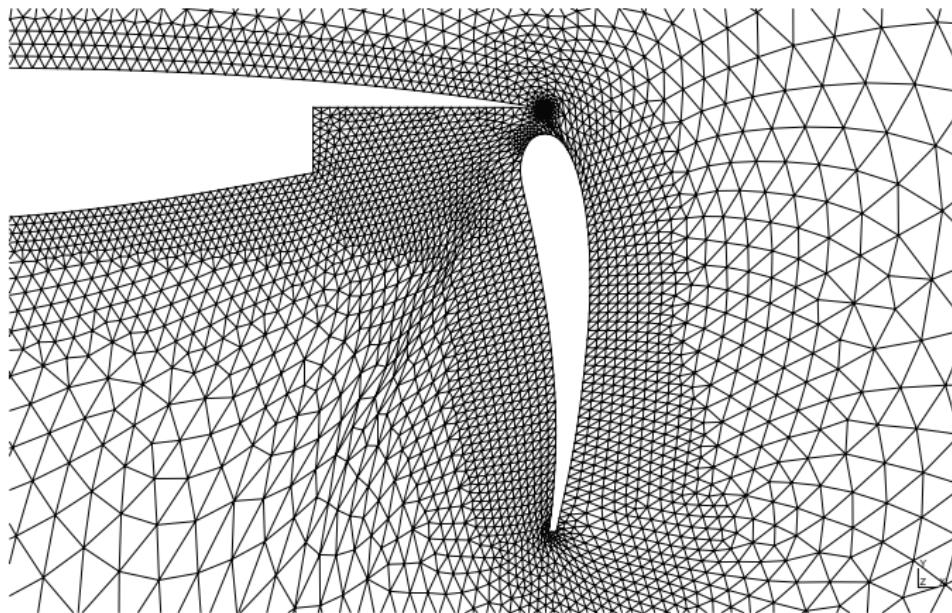


Figure : Inviscid 3-component airfoil mesh with 60° rotation of flap by linear elasticity method

# Linear elasticity

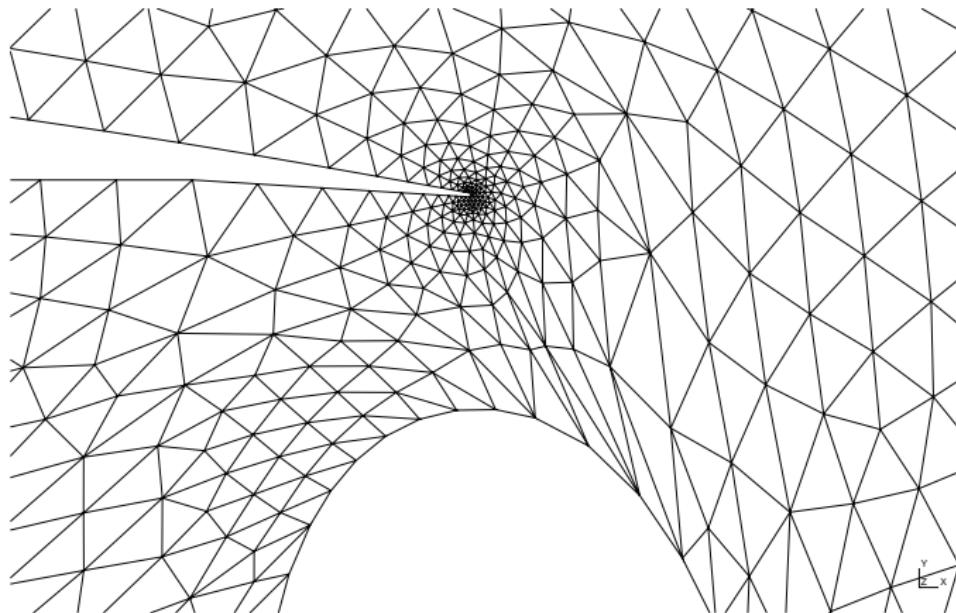


Figure : Inviscid 3-component airfoil mesh with  $60^\circ$  rotation of flap by linear elasticity method; zoomed to where the flap meets the wing

# Linear elasticity

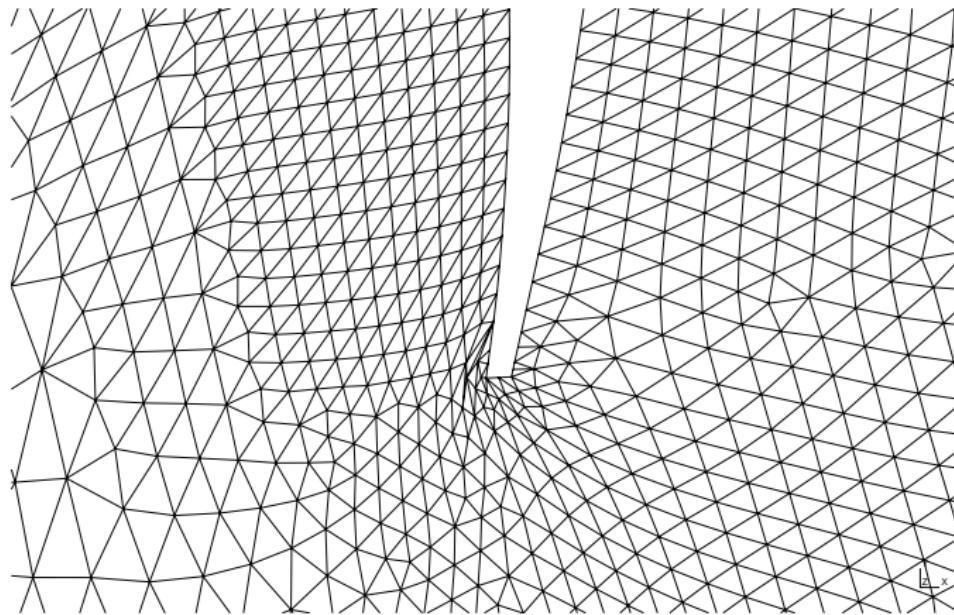


Figure : Inviscid 3-component airfoil mesh with  $60^\circ$  rotation of flap by linear elasticity method, zoomed to the trailing edge of the flap

# DGM

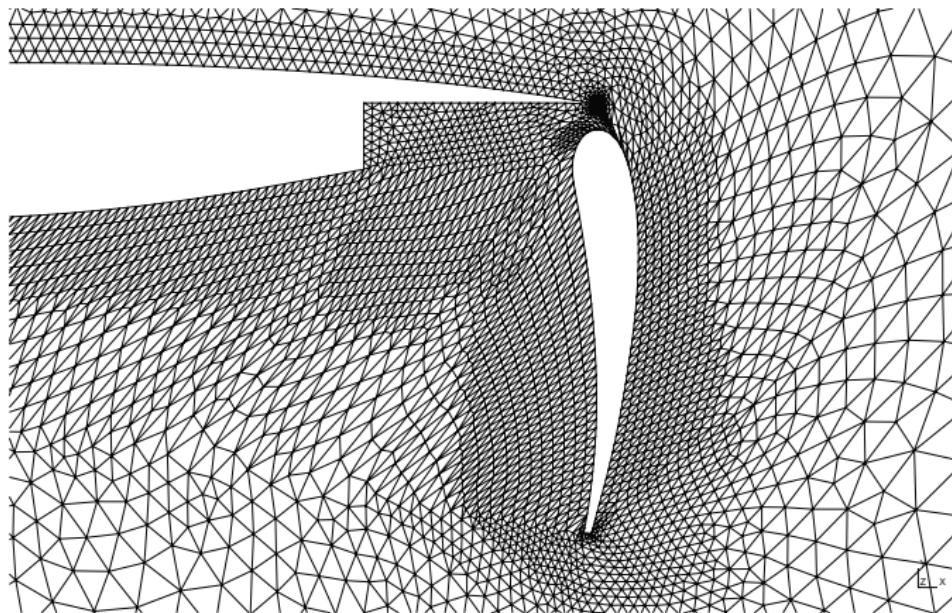


Figure : Inviscid 3-component airfoil mesh with  $60^\circ$  rotation of flap by DGM method

# DGM

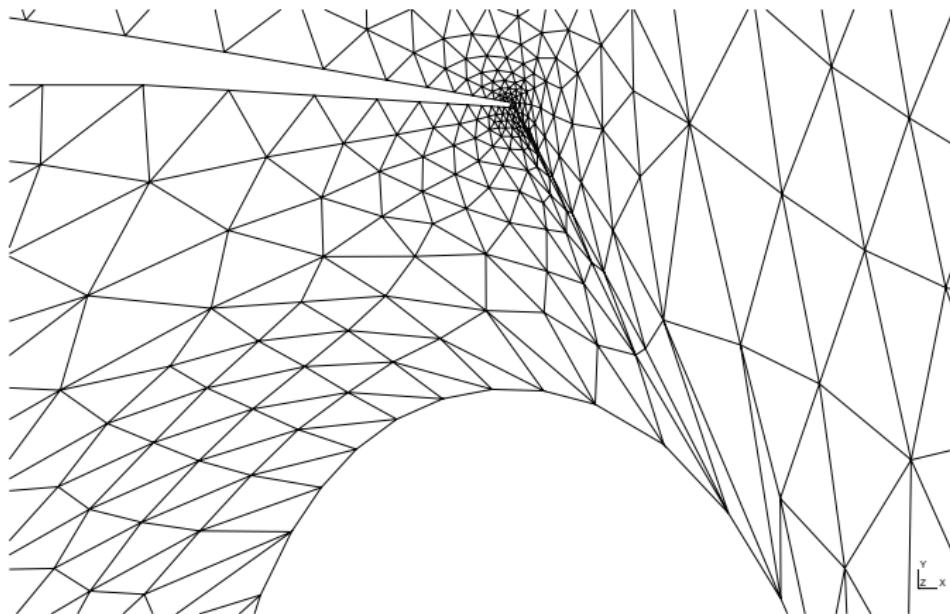


Figure : Inviscid 3-component airfoil mesh with  $60^\circ$  rotation of flap by DGM method; zoomed to where the flap meets the wing

# RBF

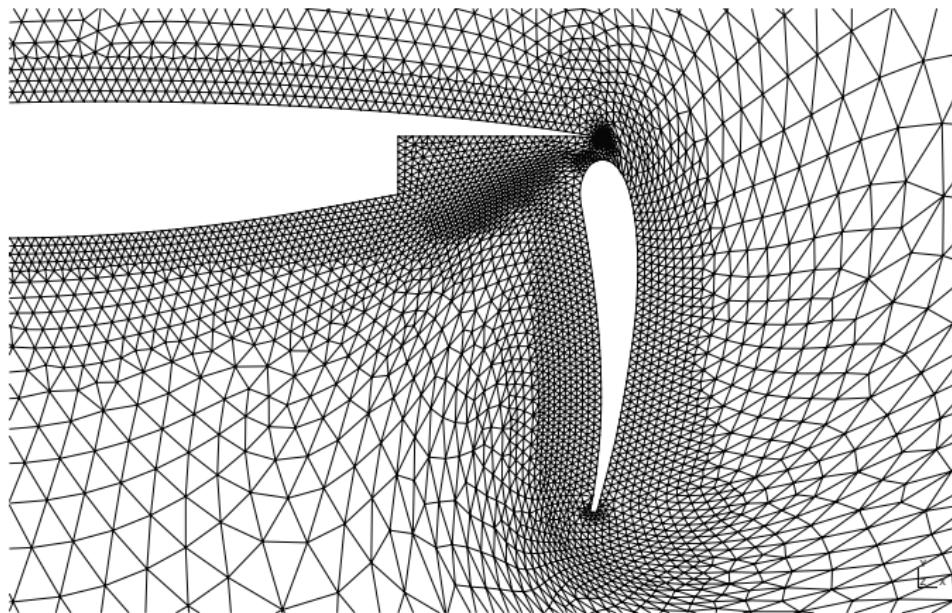


Figure : Inviscid 3-component airfoil mesh with  $60^\circ$  rotation of flap by RBF method

# RBF

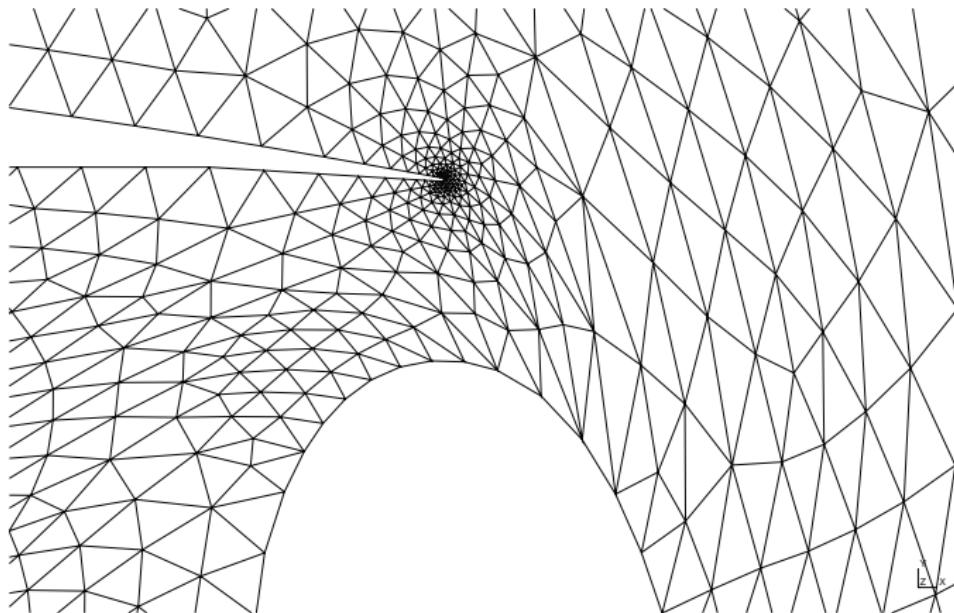


Figure : Inviscid 3-component airfoil mesh with  $60^\circ$  rotation of flap by RBF method; zoomed to where the flap meets the wing

# RBF

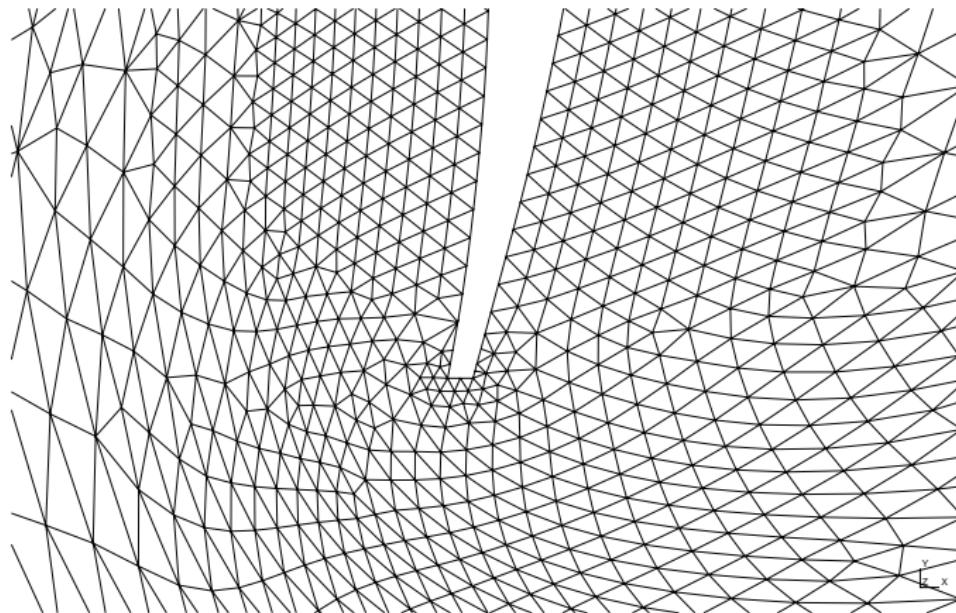


Figure : Inviscid 3-component airfoil mesh with  $60^\circ$  rotation of flap by RBF method, zoomed to the trailing edge of the flap

# Mesh quality metrics

In order to judge the effectiveness of mesh-movement methods, we need to measure the quality of the deformed mesh. Some mesh quality measures have been derived for linear 2D and 3D elements by Knupp<sup>16</sup>:

- size
- shape
- skew
- size-shape and size-skew

metrics for triangles, quadrangles, tetrahedra and hexahedra.

---

<sup>16</sup>P.M. Knupp. "Algebraic mesh quality measures for unstructured initial meshes". In: *Finite Elements in Analysis and Design* 39 (2003), pp. 217–241.

# Shape metric

Shape metric takes into account only the relative lengths of the edges of a cell.

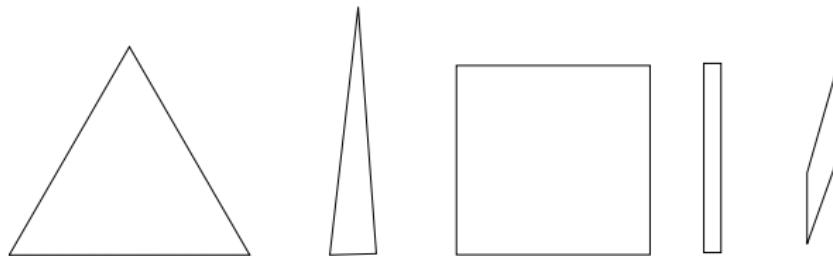
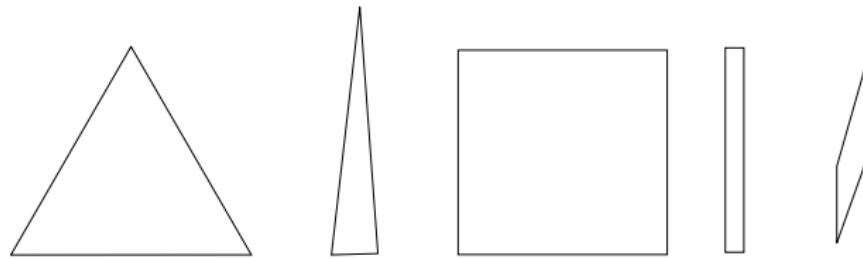


Figure : Cell shape quality. From left to right: triangle with good shape, triangle with bad shape, quadrilateral with good shape, quadrilateral with bad shape and another quadrilateral with bad shape

# Skew Metric

Skew metric takes into account only the relative angles between edges of a cell. For a triangle, it is equivalent to the shape metric.



**Figure :** Cell skew quality. From left to right: good triangle, bad triangle, good quadrilateral, *good* quadrilateral (but with bad shape) and a quadrilateral with both bad skew and bad shape qualities

# Mesh quality metrics

Shape quality:

$$f_{shape} = \frac{\sqrt{3}r \sin \theta}{1 - r \cos \theta + r^2} \text{ for triangles} \quad (18)$$

$$f_{shape} = \frac{8}{\sum_0^3 (1 + r_k^2) / (r_k \sin \theta_k)} \text{ for quads.} \quad (19)$$

Skew quality for quadrilaterals:

$$f_{skew} = \frac{4}{\sum_0^3 1 / \sin \theta_k}. \quad (20)$$

$r$  is the ratio of consecutive edge lengths.

We choose to use the shape metric for triangles and the skew metric for quadrilaterals.

# Comparison of interpolation methods

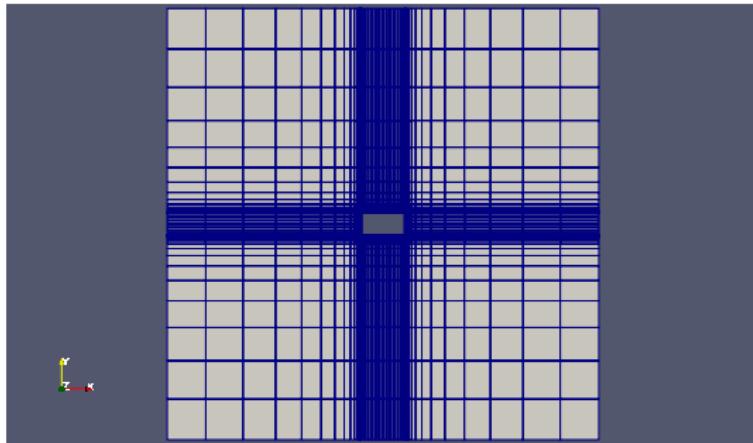


Figure : Original mesh

This test case is taken from Wang *et. al.*<sup>17</sup>.

---

<sup>17</sup>21.

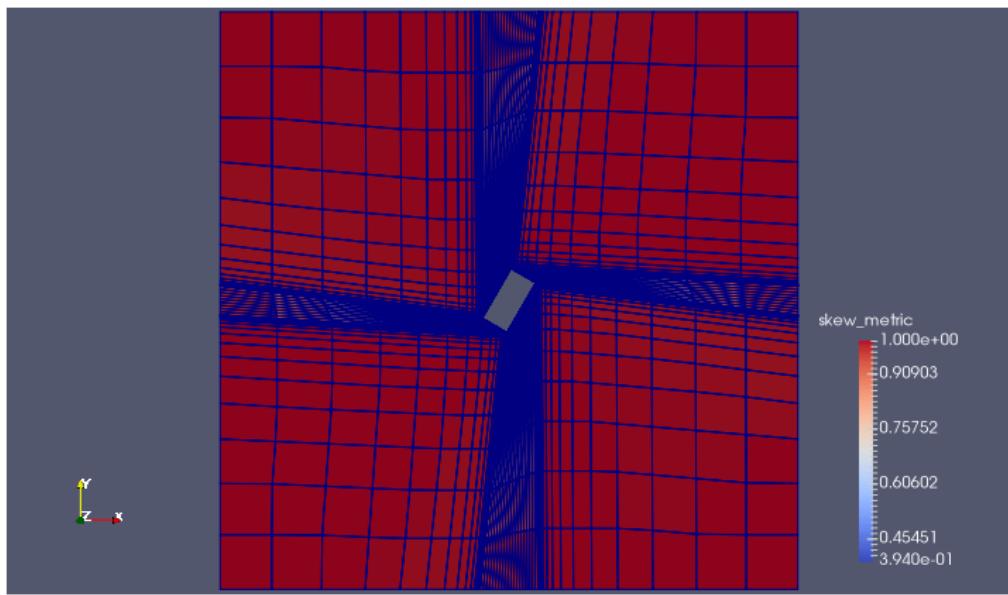


Figure : 60 degrees rotation by DGM

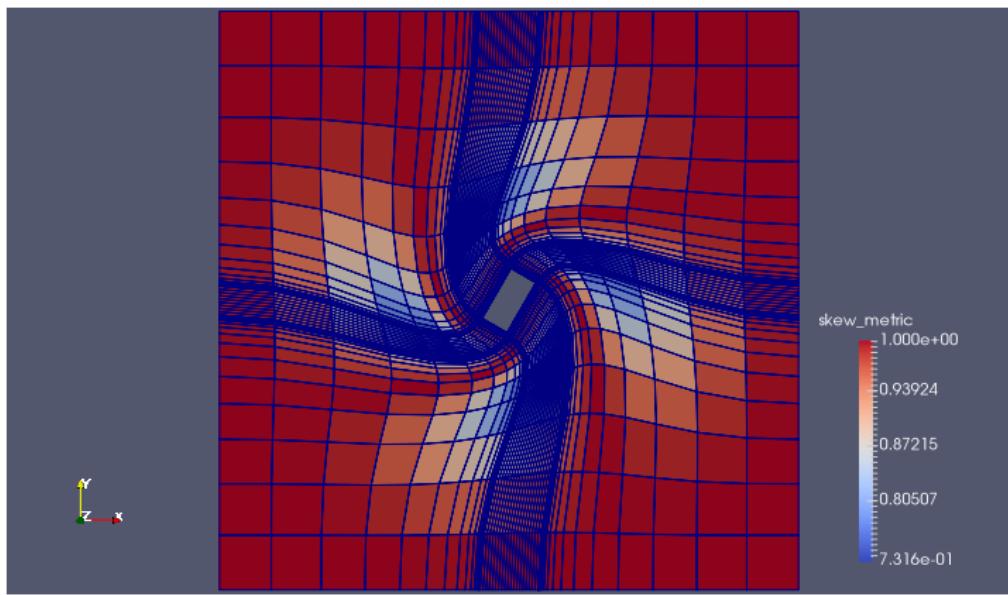


Figure : 60 degrees rotation by RBF

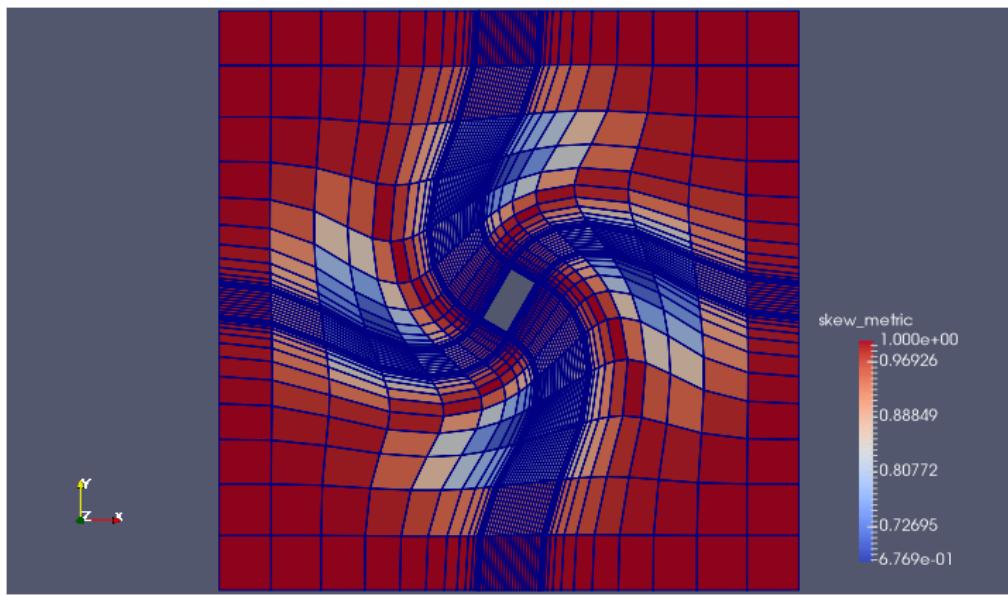


Figure : 60 degrees rotation by DGRBF2

# Performance

Method	Wall-clock time
RBF	1.926 s
DGRBF2	0.08 s

Table : Performance comparison between RBF and DGRBF2 methods

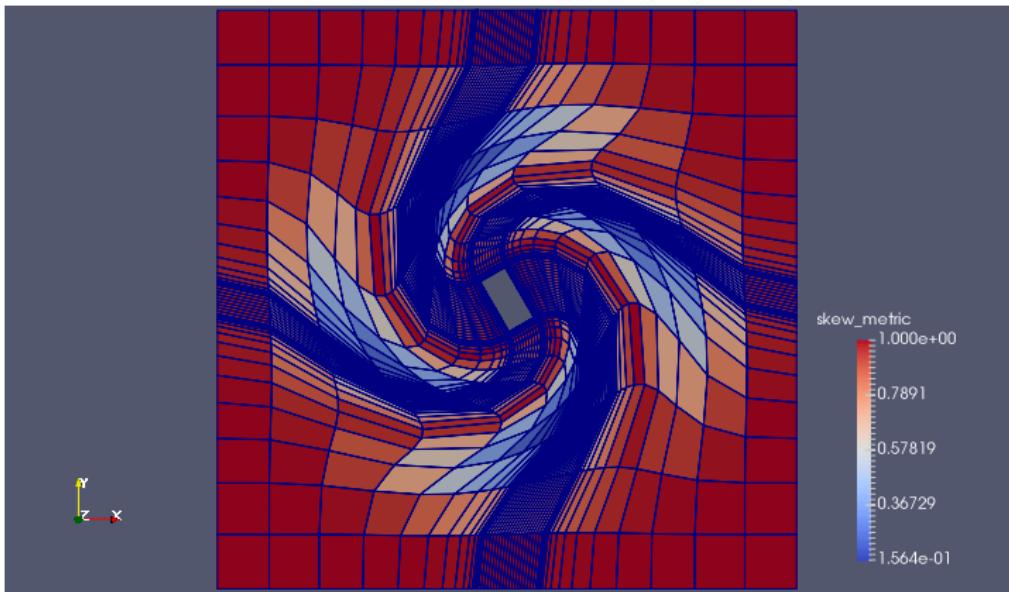


Figure : Large rotational motion ( $120^\circ$ ) carried out by DGRBF2

RBF can achieve about  $117^\circ$  rotation before giving invalid elements.

DGRBF2 is very robust for the case it is designed for - large

# Table of Contents

## 1 Section I : Introduction

- Need for mesh movement
- Need for curved meshes

## 2 Section II : Mesh movement

- Review of methods
- Large rotation in inviscid mesh
- Mesh quality
- Large rotation with mesh quality

## 3 Curved mesh generation

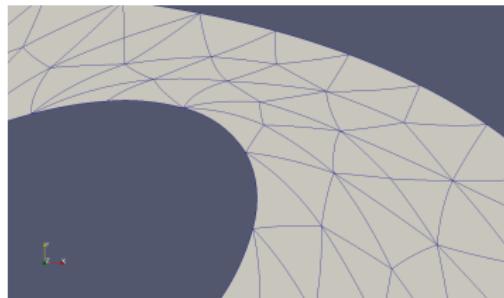
- Viscous 3-component airfoil
- 3D Bump Channel Turbulent Flow Case

## 4 Conclusion

# Curved Mesh Generation

Involves four steps:

- Add 'high-order' nodes to edges, faces and inside cells
- Obtain information about the true boundary, either using CAD data or reconstruction
- Determine displacements of boundary nodes and move them
- Regularize the interior mesh



# Boundary reconstruction

Based on the assumption of smoothness of the boundary, we can reconstruct a piecewise polynomial boundary from a piecewise linear boundary. 2D boundary reconstruction code:

Reconstructs an almost globally  $C^2$  curve using cubic splines. Corners are detected by comparing normals of consecutive boundary facets.

# Interior mesh movement

Why?

- Once boundary nodes are moved, the quality of elements near the boundary deteriorates even for inviscid flow cases. This could worsen the conditioning of the problem.
- In case of meshes for turbulent flows, high-aspect-ratio elements in the boundary layer will get invalidated upon boundary movement. This will cause the solver to fail.

# Interior mesh movement

Three main ways to achieve a valid and high-quality mesh:

- Elasticity-based methods (Peraire and Persson [17], Hartmann [7], many others)
- Interpolation methods (Z.J. Wang [10] and others)
- Optimization (Toulorge *et. al.* [20])

# Curved mesh quality

We need a way for measuring curved mesh quality independent of linear mesh quality. This is done as a post-processing step using the plugin ‘AnalyseCurvedMesh’ available in Gmsh<sup>18</sup>. The quantity computed by this plugin is given as

$$m_i = \frac{\inf_{\mathbf{x} \in \Omega_i} \det \mathbf{J}(\mathbf{x})}{\det \mathbf{J}_{I_i}}, \quad (21)$$

It measures the ‘distortion’ of the element from the corresponding linear element.

---

<sup>18</sup>A. Johnen, J.-F. Remacle, and C. Geuzaine. “Geometrical validity of curvilinear finite elements”. In: *J. Comput. Phys.* 233 (2013), pp. 359–372.

# Viscous 3-component airfoil

We reconstruct a high-order boundary using cubic splines.  
Interior mesh movement is compared using linear elasticity, RBF  
and stiffened linear elasticity (SLE).

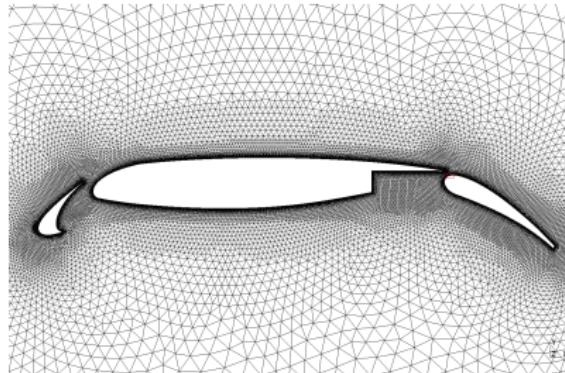
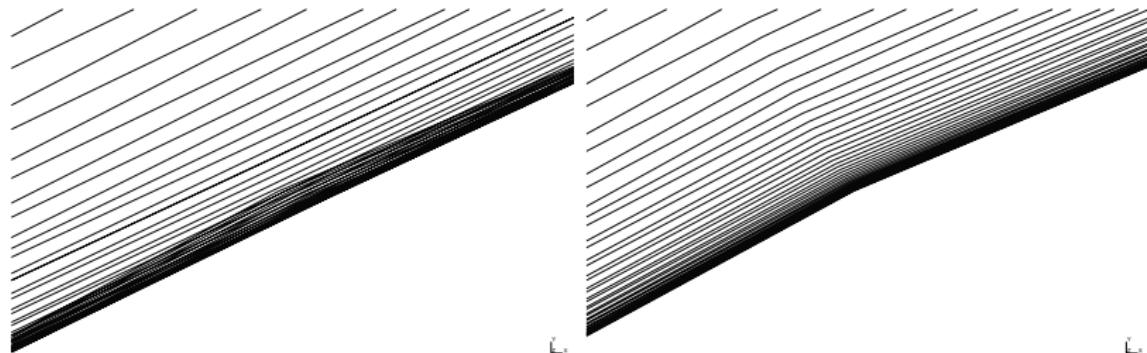


Figure : Boundary-layer mesh of multi-element airfoil



**Figure :** Portion of quadratic viscous mesh for multi-element airfoil, showing a boundary face in the flap, generated by linear elasticity (left) and RBF (right) methods

Regular linear elasticity does not work.

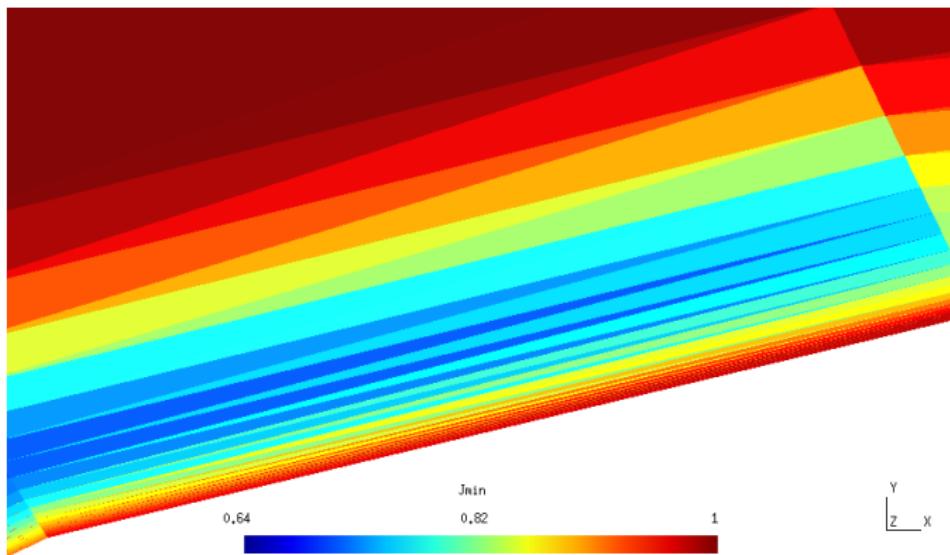


Figure : Minimum scaled Jacobian over each element for a portion mesh generated by RBF interpolation

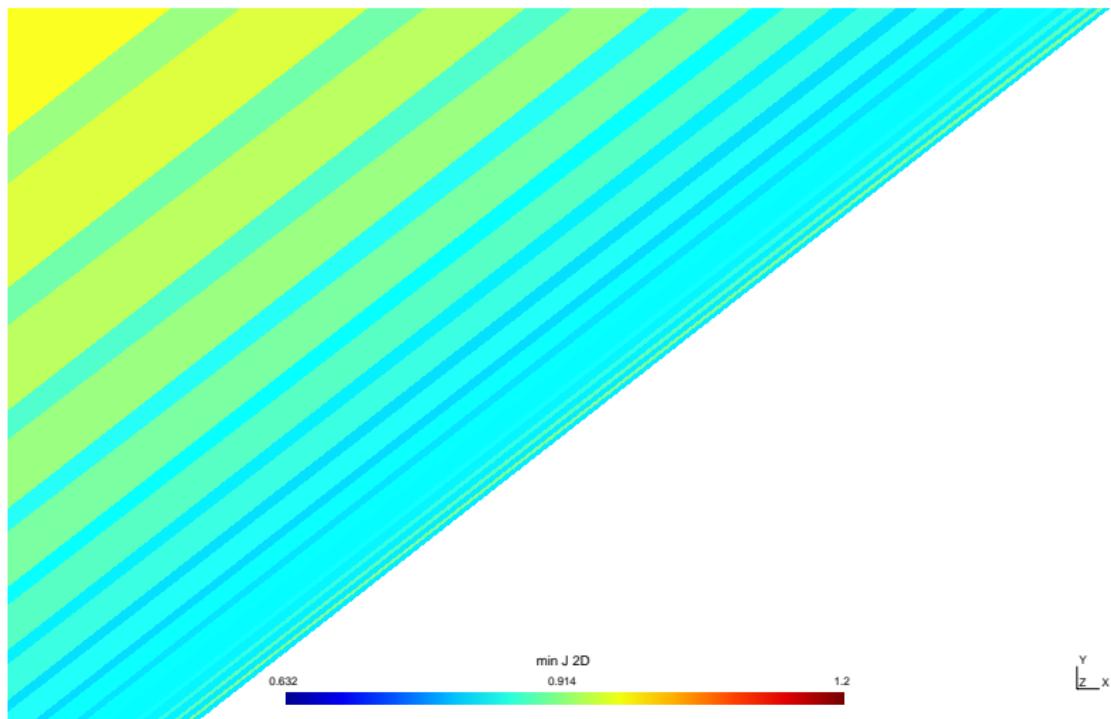
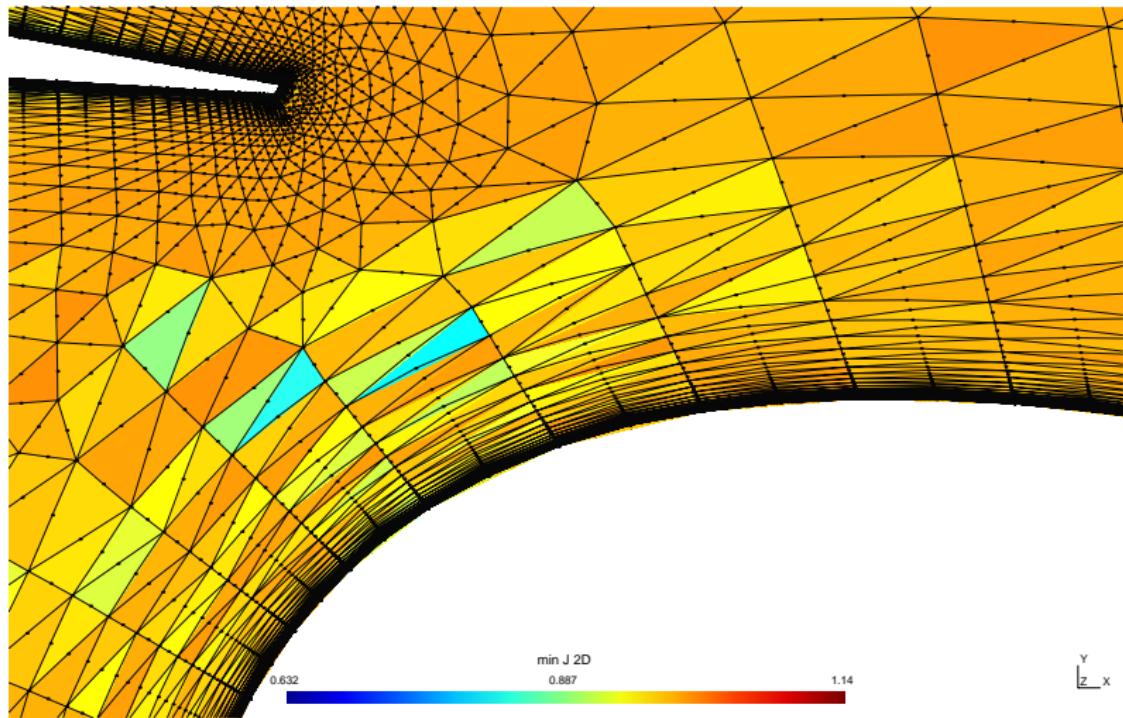


Figure : A portion curved mesh generated by stiffened elasticity method; minimum scaled Jacobian over each element is shown



**Figure :** An illustrative example of a curved mesh generated by stiffened elasticity method; minimum scaled Jacobian over each element is shown

# Performance

Solutions are computed by a point-Jacobi-preconditioned conjugate gradient solver.

Method	Parameters	Min. quality	Avg. quality	Time	Solver iterations
SLE	$\chi = 2.75$	0.632	0.994922	19.3s	324
	$\chi = 2.9$	0.632	0.994535	24.4s	423
RBF	1-step $r_s = 0.04$	0.640	0.991093	1.86s	365 x 2
	1-step $r_s = 0.08$	0.639	0.991562	1.88s	1150 x 2
	1-step $r_s = 0.12$	0.637	0.99159	1.89s	2295 x 2
	2-step $r_s = 0.04$	0.641	0.991099	2.58s	365 x 4

Table : Comparison of RBF (radial basis function) and SLE (stiffened linear elasticity) methods

# 3D Bump Channel Turbulent Flow Case

Reynolds-averaged Navier-Stokes (RANS) flow simulation test case from the NASA Turbulence models website [18].

The mesh is a hexahedral structured grid.

Boundary displacements were first obtained from the analytical expression for the bump given on the website. The solver used for RBF is a sparse LU decomposition.

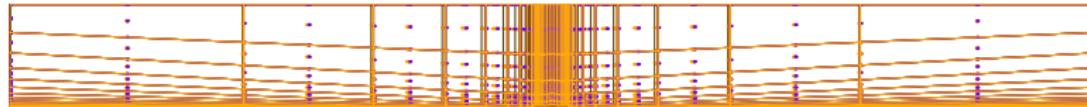


Figure : Coarsest curved mesh of 3D bump

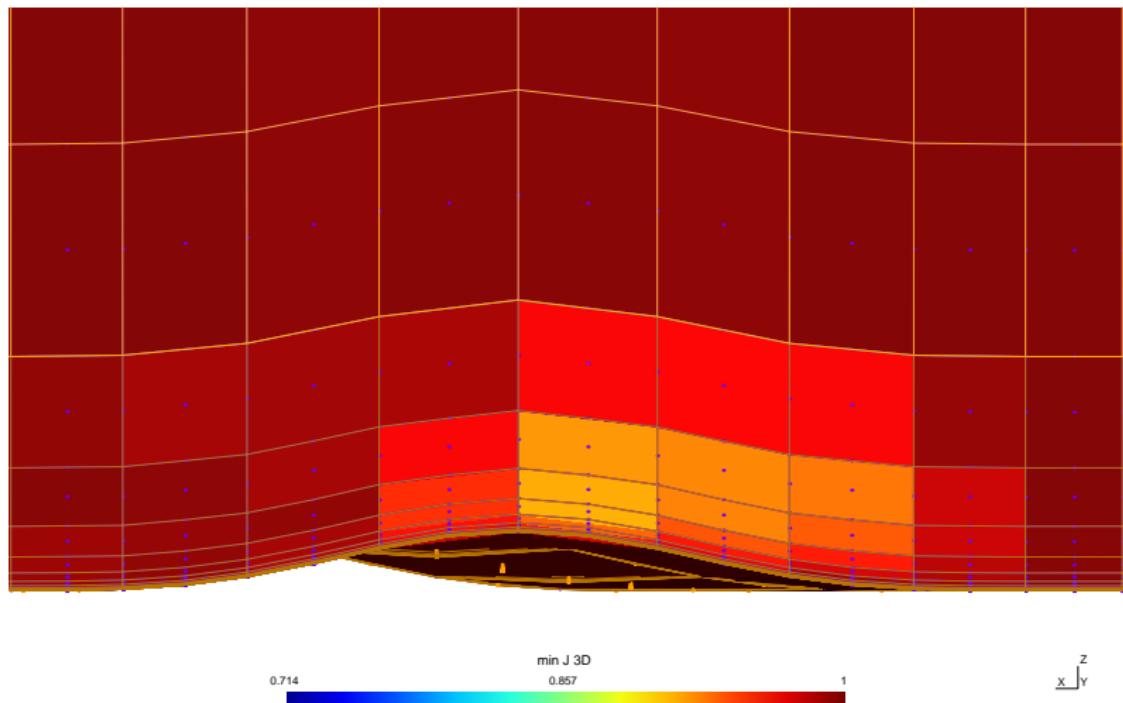


Figure : Minimum scaled Jacobian of coarse curved mesh of 3D bump

# Performance of curved mesh generator

	Coarse	Medium	Fine
No. of points	32,841	243,729	1,875,489
No. of boundary points	2,304	8,704	33,792
Support radius	0.06	0.06	0.04
Minimum quality	0.714	0.852	0.926
Wall-clock time	0.68s	12.6s	338s

Table : Summary of 3D bump curved mesh generation

# Results from RANS simulation run by rDGFL<sup>19</sup>

- Spalart-Allmaras eddy-viscosity model
- DG P1
- HLLC and Bassi Rebay 2 numerical fluxes
- First order in time, implicit time stepping
- LU SGS preconditioned GMRES solver

---

<sup>19</sup>Xiaodong Liu et al. "Development and Assessment of a Reconstructed Discontinuous Galerkin Method for the Compressible Turbulent Flows on Hybrid Grids". In: *54th AIAA Aerospace Sciences Meeting*. American Institute of Aeronautics and Astronautics. 2016.

# Residual convergence for mass flux

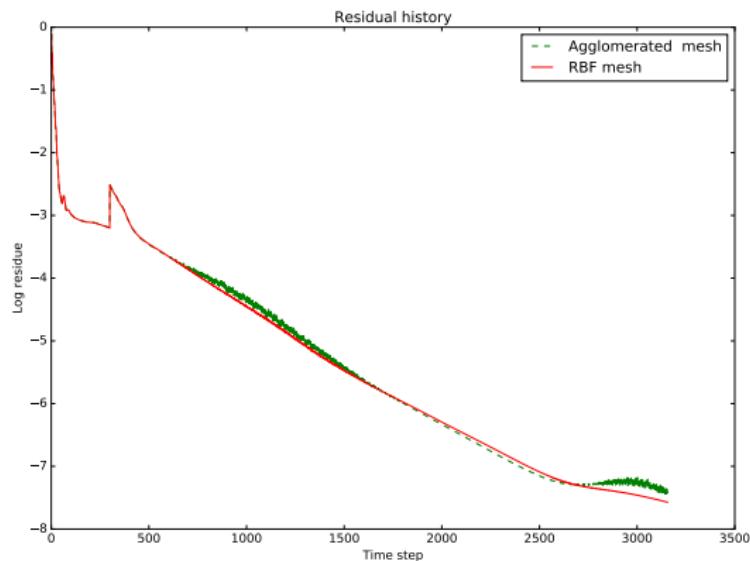


Figure : Comparison of mass-flux residual convergence history with time steps for implicit DG P1 solution

# Grid convergence for $C_d$

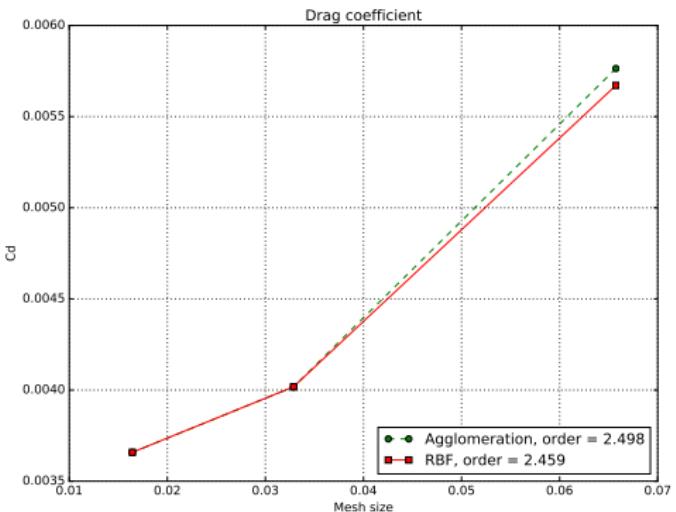


Figure : Accuracy (grid-convergence) of the solver using the agglomerated and RBF-curved mesh; the reference solution [18] is 0.0035897

# Table of Contents

## 1 Section I : Introduction

- Need for mesh movement
- Need for curved meshes

## 2 Section II : Mesh movement

- Review of methods
- Large rotation in inviscid mesh
- Mesh quality
- Large rotation with mesh quality

## 3 Curved mesh generation

- Viscous 3-component airfoil
- 3D Bump Channel Turbulent Flow Case

## 4 Conclusion

# Conclusions - general mesh movement I

- Interpolation methods are generally well-suited for unsteady simulations that require mesh movement, as they are very fast.
- When deformations are relatively small and not highly rotational, Delaunay graph mapping (DGM) is a good choice.
- When larger and more general deformations are needed, the pure radial basis function (RBF) and to some extent the DGM with RBF interpolation (DGRBF) methods are good.

# Conclusions - general mesh movement II

- The pure RBF method, while usually giving good robust results, is more expensive than the other interpolation methods considered here. The DGRBF methods, while being inexpensive, are only robust when implemented with angle interpolation (may be difficult to do for general mesh movements).
- While linear elasticity methods generally perform well for mesh movement, they are much more expensive than the other methods and probably too expensive to use in unsteady simulations.

# Conclusions - curved mesh generation

- We find that RBF method is much more cost-effective than Jacobian-stiffened linear elasticity method for comparable results
- Both methods provide ‘knobs’ to tune, such as the basis function and support radius in case of RBF and the stiffening criterion and stiffening exponent for the linear elasticity method
- Delaunay graph mapping methods, in their current state, are generally unusable for curved mesh generation.

# Future Directions

- Surface reconstruction in 3D - started, currently works only for smooth meshes. The technique used is ‘Weighted averaging of local fittings’<sup>20</sup>. Need detection of various kinds of singularities (‘ $C^1$  discontinuities’) in the surface mesh.
- Automatic estimation of support radius for curved mesh generation by RBF; could try this based on local curvature estimates and density of interior mesh points nearby.

---

<sup>20</sup>X. Jiao and D. Wang. “Reconstructing high-order surfaces for meshing”.

In: *Engineering with computers* 28 (2012), pp. 361–373.



# Table of Contents

## 1 Section I : Introduction

- Need for mesh movement
- Need for curved meshes

## 2 Section II : Mesh movement

- Review of methods
- Large rotation in inviscid mesh
- Mesh quality
- Large rotation with mesh quality

## 3 Curved mesh generation

- Viscous 3-component airfoil
- 3D Bump Channel Turbulent Flow Case

## 4 Conclusion

-  F. Alauzet. "A changing-topology moving mesh technique for large displacements". In: *Engrg. Comput.* 30 (2014), pp. 175–200.
-  F. Bassi and S. Rebay. "High-order accurate discontinuous finite element solution of the 2D Euler equations". In: *J. Comput. Phys.* 128 (1997), pp. 251–285.
-  J.T. Batina. "Unsteady Euler Algorithm with Unstructured Dynamic Mesh for Complex-Aircraft Aerodynamic Analysis". In: *AIAA Journal* 29.3 (1991), pp. 327–333.
-  A. de Boer, M.S. van der Schoot, and M. Bijl. "Mesh deformation based on radial basis function interpolation". In: *Computers & Structures* 85 (2007), pp. 784–795.
-  C. Farhat and M. Lesoinne. "Two efficient staggered algorithms for the serial and parallel solution of three-dimensional nonlinear transient aeroelastic

problems". In: *Comput. Methods Appl. Mech. Engrg.* 182 (2000), pp. 499–515.

-  C. Farhat et al. "Torsional springs for two-dimensional dynamic unstructured fluid meshes". In: *Comput. meth. appl. mech. engrg.* 163 (1998), pp. 231–245.
-  Ralf Hartmann and Tobias Leicht. *High-order unstructured grid generation and Discontinuous Galerkin discretization applied to a 3D high-lift configuration*. 53<sup>rd</sup> AIAA Aerospace Sciences Meeting (SciTech 2015), AIAA 2015-0819, Kissimmee, Florida. 2015.
-  J.E. Hicken and D.W. Zingg. "Aerodynamic optimization algorithm with integrated geometry parameterization and mesh movement". In: *AIAA Journal* 48.2 (2010), pp. 400–413.

-  G. Hou, J. Wang, and A. Layton. "Numerical methods for fluid-structure interaction - a review". In: *Commun. Comput. Phys.* 12.2 (2012), pp. 337–377.
-  J. Ims, Z. Duan, and Z.J. Wang. "meshcurve : An automated low-order to high-order mesh generator". In: *22nd AIAA Computational Fluid Dynamics conference*. 2015.
-  X. Jiao and D. Wang. "Reconstructing high-order surfaces for meshing". In: *Engineering with computers* 28 (2012), pp. 361–373.
-  A. Johnen, J.-F. Remacle, and C. Geuzaine. "Geometrical validity of curvilinear finite elements". In: *J. Comput. Phys.* 233 (2013), pp. 359–372.

-  P.M. Knupp. "Algebraic mesh quality measures for unstructured initial meshes". In: *Finite Elements in Analysis and Design* 39 (2003), pp. 217–241.
-  X. Liu, N. Qin, and H. Xia. "Fast dynamic grid deformation based on Delaunay graph mapping". In: *Jo. Comput. Phys.* 211 (2006), pp. 405–423.
-  Xiaodong Liu et al. "Development and Assessment of a Reconstructed Discontinuous Galerkin Method for the Compressible Turbulent Flows on Hybrid Grids". In: *54th AIAA Aerospace Sciences Meeting*. American Institute of Aeronautics and Astronautics. 2016.
-  X. Luo, M.S. Shephard, and J.-F. Remacle. "The influence of geometric approximation on the accuracy of high order methods". In: *Rensselaer SCOREC report* 1 (2001).

-  P.-O. Persson and J. Peraire. "Curved mesh generation and mesh refinement using lagrangian solid mechanics". In: *47<sup>th</sup> AIAA Aerospace Sciences Meeting*. 2009.
-  Christopher Rumsey. *VERIF/3DB: 3D Bump-in-channel Verification Case*. NASA Langley Research Center Turbulence Models website, <http://turbmodels.larc.nasa.gov/bump3d.html>.
-  K. Stein, T. Tezduyar, and R. Benney. "Mesh moving techniques for fluid-structure interactions with large displacements". In: *J. Appl. Mech.* 70 (2003), pp. 58–63.
-  T. Toulorge et al. "Robust untangling of curvilinear meshes". In: *Journal of Computational Physics* 254 (2013), pp. 8–26.



Y. Wang, N. Qin, and N. Zhao. "Delaunay graph and radial basis function for fast quality mesh deformation". In: *J. Comput. Phys.* 294 (2015), pp. 149–172.



Z.J. Wang et al. "High-order CFD methods: current status and perspective". In: *Intl. J. Numer. Meth. Fluids* 72 (2013), pp. 811–845.



H. Wendland. "Error estimates for interpolation by compactly supported radial basis functions of minimal degree". In: *J. Approx. Theory* 93 (1998), pp. 258–272.