Methods for Mesh Movement and Curved Mesh Generation

Aditya Kashi
North Carolina State University

August 2015

1 Introduction

Various methods have been used to generate high-order curved meshes from linear meshes - including methods to optimize surface (boundary) meshing and methods to get valid internal mesh. Here, I focus on the latter.

There are mainly two classes of methods to achieve mesh movement:

- Elasticity-based methods
- Interpolation methods

2 Elasticity-based Methods

2.1 Lineal Spring Analogy

This is the simplest method that can be used to move internal mesh points in response to movement of the boundary, invented by J.T. Batina (NASA, 1991).

Each edge of the mesh is assumed to represent a lineal spring connecting the two nodes which make up the edge.

The idea is that after imposing boundary movement, the mesh is allowed to "go to equilibrium". So at every mesh point, we have

$$\sum_j k_{ij}(\Delta\vec{r_i} - \Delta\vec{r_j}) = \vec{0} \quad \forall i \tag{1}$$

where i ranges over all nodes, j ranges over points surrounding node i and $\Delta\vec{r_i}$ is the displacement of node i. k_{ij} is the stiffness of the spring between nodes i and j, calculated as

$$k_{ij} = \frac{1}{||\vec{r_i} - \vec{r_j}||} \tag{2}$$

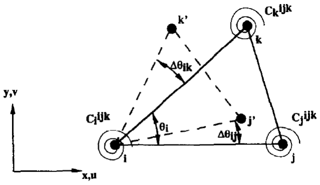


Fig. 1. Motion and deformation of a triangle.

2.2 Lineal Spring Analogy

It is not reliable as it generates invalid elements (containing zero and negative values of Jacobian determinant).

This method is used, for example, by Mavriplis et. al. [7]. In this case, however, the internal elements are linear - only boundary faces are high-order, since the problem considered is inviscid flow.

2.3 Torsional Spring Analogy

This is an extension to the previous method, which adds torsional springs at each node in each element.

These torsional springs resist the relative angular motion between edges of an element. This leads to a substantially more robust movement, according to Farhat et. al. [2] However, I have not found any literature (yet) where this method is used for curved mesh generation, though it is widely used for mesh deformation otherwise.

2.4 Linear Elasticity Analogy

In this method, the (partial differential) equations of linear elasticity are solved by high-order continuous FEM to move the internal nodes of a mesh, as done by Xie [9].

Even if a constant elasticity is used throughout, it gives decent results.

But the main power lies in the fact that the elasticity can be varied in a way that leads to a better mesh, such as making it inversely proportional to distance from the deforming boundary, or to the cell volume (as in Hartmann and Leicht [3]) (which I have not done yet).

Idea: A radial basis function could be used to determine the elasticity variation!

2.5 Non-linear Elasticity Analogy

The general idea of this method is same as that of linear elasticity. The difference is that a non-linear constitutive model is used (relating stress and strain) instead of a linear constitutive model.

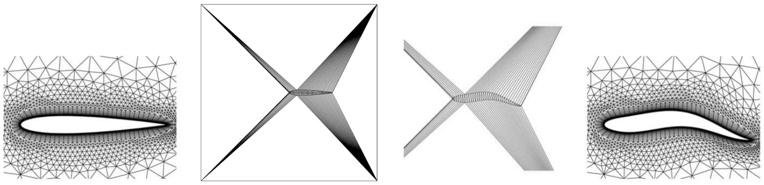


Figure 1: The DGM process

Persson and Peraire claim [6] “When the mesh is sufficiently fine relative to solid deformation, this method guarantees non-intersecting elements even for highly distorted or anisotropic initial meshes”.

3 Interpolation Methods

These methods are not based on any physical relation between node locations.

Instead, the boundary displacement is interpolated to the interior using any of several methods.

3.1 Delaunay Graph Mapping(DGM)

Developed by Liu, Qin and Xia [4], this method is a fast method of mesh movement. It consists of the following steps.

- Triangulate the boundary points of the mesh. This triangulation is referred to as the Delaunay graph.
- Each internal mesh point is located in the Delaunay graph, and its area coordinates are calculated w.r.t. the Delaunay element it lies in.
- Using the prescribed boundary displacements, the Delaunay graph is moved.
- Using the calculated area coordinates, the interior mesh points are mapped back to the deformed Delaunay elements, thus moving the mesh.

3.2 Radial Basis Function (RBF) Interpolation

In this method, global basis functions (which are radially symmetric) are used to represent the mesh deformation [1].

$$s(\mathbf{x}) = \sum_{i=1}^{N_b} \alpha_i \phi(||\mathbf{x} - \mathbf{x}_{\mathbf{b}i}||) \tag{3}$$

Here, $\mathbf{x}_{\mathbf{b}i}$ is the position of the ith boundary node, α_i is the coefficient of the basis function $\phi_i = \phi(||\mathbf{x} - \mathbf{x}_{\mathbf{b}i}||)$, $s(\mathbf{x})$ is the displacement at point $\mathbf{x}$ in the domain.

We first need to calculate the coefficients of the RBFs by using the boundary displacement information. This involves solving a symmetric positive-definite system of equations for the coefficients α_i.

$$s_{b_j}(\mathbf{x}) = \sum_{i=1}^{N_b} \alpha_i \phi(||\mathbf{x_b}_j - \mathbf{x_b}_i||) \quad \forall j \in \{1, 2, ...N_b\} \tag{4}$$

Thus, a linear system needs to be solved for RBF interpolation, as opposed to DG-based methods.

3.3 Delaunay graph mapping with radial basis functions

Wang, Qin and Zhao combined the Delaunay graph mapping method with RBF interpolation to produce what they call the DG-RBF method [8].

This method is similar to the original DGM, except that the interpolation is done using RBFs instead of area coordinates. This gives much more control over the nature of interpolation.

For problems with severe rotational motion, rotation angles can also be interpolated along with displacements.

4 Comparison in case of large mesh deformation

All of the above methods were tested on a case of flap-rotation in a 3-component airfoil inviscid mesh. It was found that the methods of linear elasticity, DGM and RBF were effective enough to be considered. Though DG-RBF is claimed to be better than DGM, I have been unable to make it work even as well as DGM for this case.

- DGM was observed to be faster than the RBF method. The RBF method is, in turn, observed to be significantly faster than linear elasticity.

  This is expected: DGM has to solve no linear system - it only needs to construct a triangulation of the boundary points; RBF needs to solve a linear system of size $N_b \times N_b$, where N_b is the number of boundary points; linear elasticity needs to solve a system of size $2N \times 2N$, where N is the total number of nodes in the mesh.

- At about 60° of rotation, mesh quality of DGM is worst, followed by linear elasticity, and RBF gives the best mesh quality (by visual inspection of the mesh). Linear elasticity does give somewhat better mesh quality away from the moving boundary, though.

- Linear elasticity failed beyond an angle of 60° CW, whereas RBF and DGM can go much beyond that. However, DGM fails for CCW angles of rotation, even as small as 10 to 15 degrees. Linear elasticity can go up to 30 degrees or so, while RBF did not produce invalid elements up to 40 degrees (possibly more).

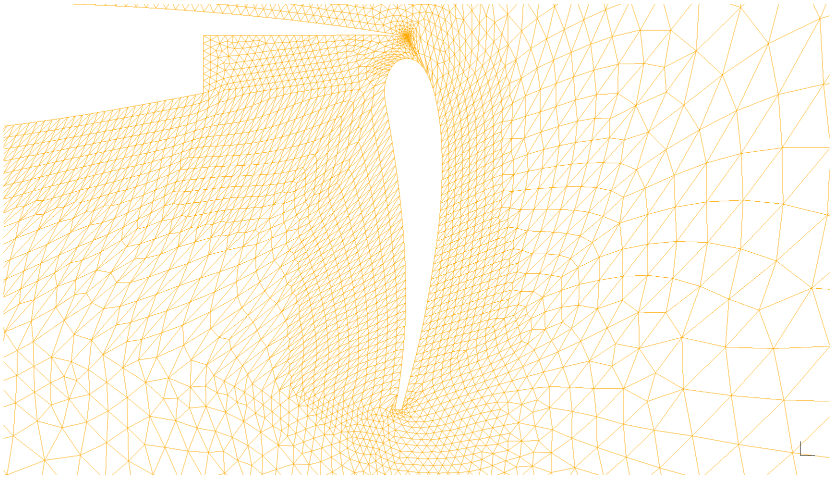


Figure 2: DGM

4.1 Deformed meshes

5 Curved Mesh Generation

If we only have boundary elements as high order, and move the boundary nodes to better conform to the true boundary, we may get invalid boundary elements.

Thus we need internal elements to be curved as well, and mesh movement methods can be used for this.

5.1 Curved Mesh Generation by Interpolation Methods

I tried to generate a curved mesh for 2D bump using DGM and RBF methods.

The RBF method works very well (it would seem) with a minimum scaled Jacobian range of 0.85 to 1.0, though I have not computed any skew (distortion) metric yet.

The DGM method, unfortunately, does not work due to the nature of this interpolation method - it is local in each Delaunay element, rather than global as in the case of RBF method.

6 Conclusion

The next steps are to

- Write pre-processing code so that 3D curved meshes can be generated.
- Write code to compute mesh-quality metrics.

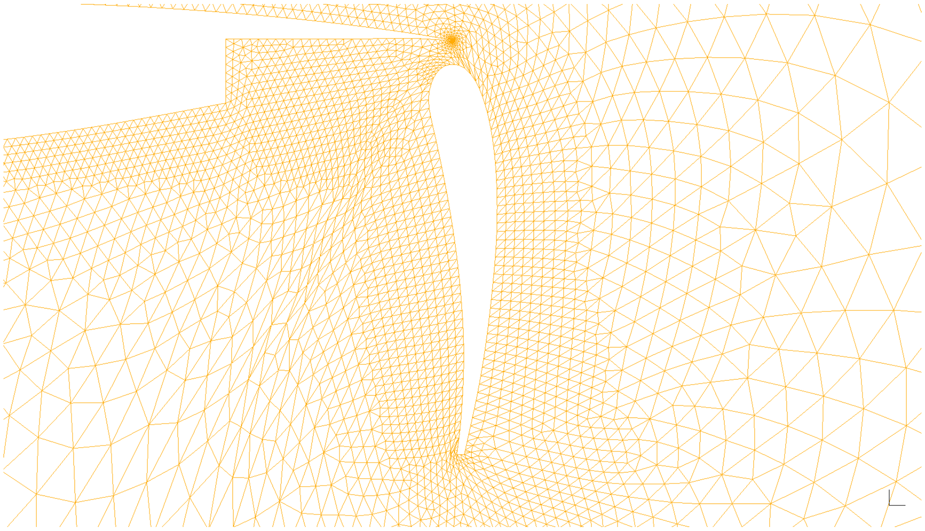


Figure 3: Linear elasticity

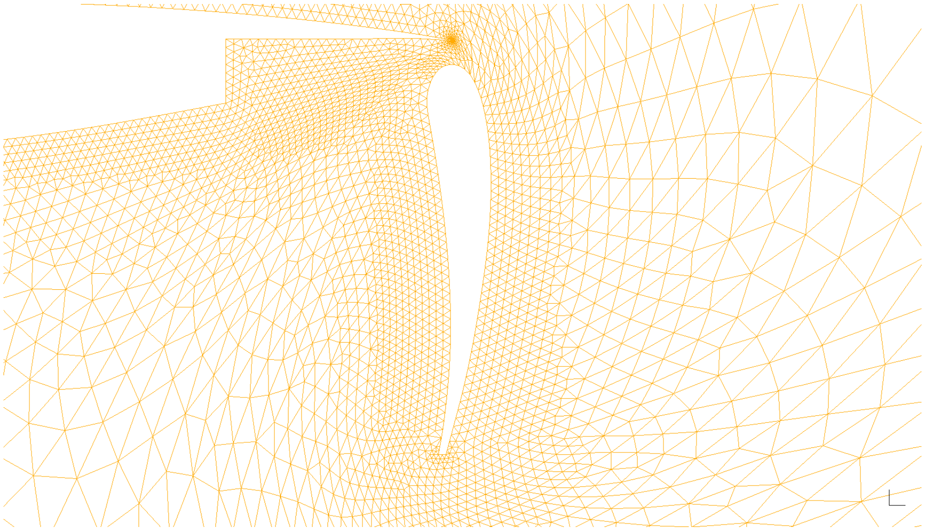


Figure 4: RBF

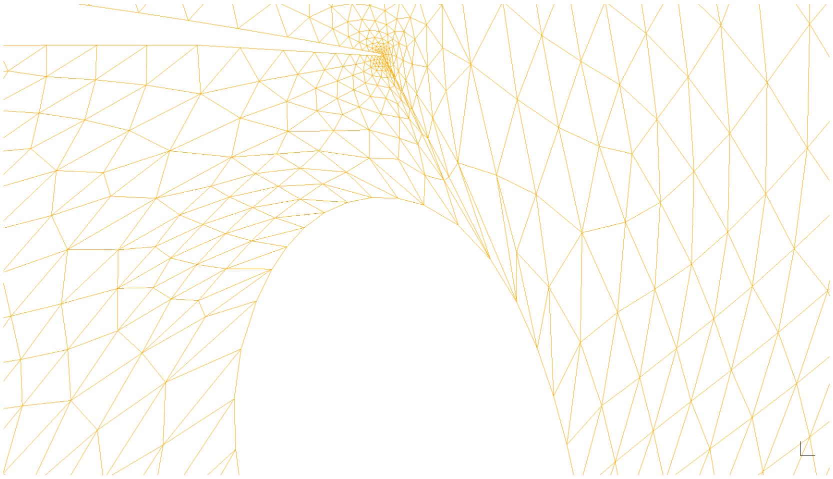


Figure 5: DGM

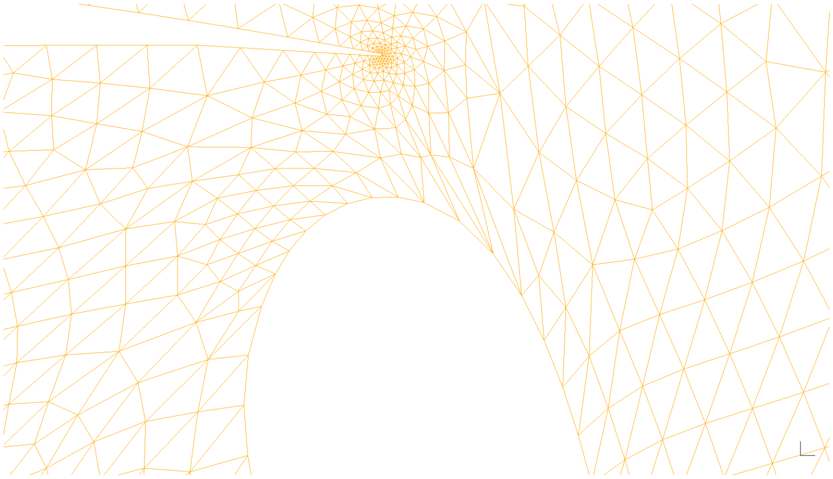


Figure 6: Linear elasticity

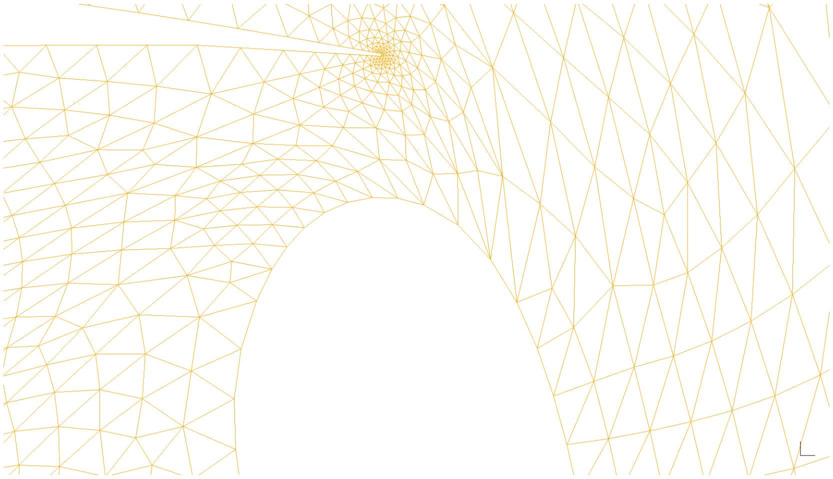
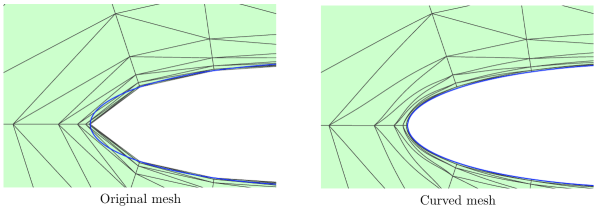


Figure 7: RBF



Original mesh

Curved mesh

Figure 8: Example of requirement of interior mesh curving, taken from [5].

References

[1] A. de Boer, M.S. van der Schoot, and H. Bijl. Mesh deformation based on radial basis function interpolation. *Computers and Structures*, 85:784–795, 2007.

[2] C. Farhat, C. Degand, B. Koobus, and M. Lesoinne. Torsional springs for two-dimensional dynamic unstructured fluid meshes. *Comput. Methods Appl. Mech. Engrg.*, 163:231–245, 1998.

[3] Ralf Hartmann and Tobias Leicht. High-order unstructured grid generation and discontinuous Galerkin discretization applied to a 3D high-lift configuration. 53^{rd} AIAA Aerospace Sciences Meeting (SciTech 2015), AIAA 2015-0819, Kissimmee, Florida, 2015.

[4] X. Liu, N. Qin, and H. Xia. Fast dynamic grid deformation based on delaunay graph mapping. *J. Comput. Phy.*, 211:405–423, 2006.

[5] P.-O. Persson and J. Peraire. Curved mesh generation and mesh refinement using lagrangian solid mechancics. *47th AIAA Aerospace Sciences Meeting including The New Horizons Forum and Aerospace Exposition*, 2009.

[6] P.-O. Persson and J. Peraire. High-order unstructured grid generation and discontinuous galerkin discretization applied to a 3d high-lift configuration. *Proc. of the 47th AIAA Aerospace Sciences Meeting*, 2009.

[7] L. Wang, D.J. Mavriplis, and W.K. Anderson. Unsteady discrete adjoint formulation for high-order discontinuous galerkin discretizations in time-dependent flow problems. *48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, AIAA 2010-637, 2010.

[8] Y. Wang, N. Qin, and N. Zhao. Delaunay graph and radial basis function for fast quality mesh deformation. *J. Comput. Phy.*, 294:149–172, 2015.

[9] Z.Q. Xie, R. Sevilla, O. Hassan, and K. Morgan. The generation of arbitrary order curved meshes for 3d finite element analysis. *Comput. Mech.*, 51:361–374, 2012.