Advances in Applied Mathematics and Mechanics
Adv. Appl. Math. Mech., Vol. 7, No. 3, pp. 338-356

DOI: 10.4208/aamm.2014.m614
June 2015

A Hybrid Method for Dynamic Mesh Generation Based on Radial Basis Functions and Delaunay Graph Mapping

Li Ding, Tongqing Guo and Zhiliang Lu*

Department of Aerodynamics, Nanjing University of Aeronautics and Astronautics, 29 Yudao Street, Nanjing 210016, China

Received 10 May 2014; Accepted (in revised version) 26 August 2014

Abstract. Aiming at complex configuration and large deformation, an efficient hybrid method for dynamic mesh generation is presented in this paper, which is based on Radial Basis Functions (RBFs) and Delaunay graph mapping. Based on the computational mesh, a set of very coarse grid named as background grid is generated firstly, and then the computational mesh can be located at the background grid by Delaunay graph mapping technique. After that, the RBFs method is applied to deform the background grid by choosing partial mesh points on the boundary as the control points. Finally, Delaunay graph mapping method is used to relocate the computational mesh by employing area or volume weight coefficients. By applying different dynamic mesh methods to a moving NACA0012 airfoil, it can be found that the RBFs-Delaunay graph mapping hybrid method is as accurate as RBFs and is as efficient as Delaunay graph mapping technique. Numerical results show that the dynamic meshes for all test cases including one two-dimensional (2D) and two three-dimensional (3D) problems with different complexities, can be generated in an accurate and efficient manner by using the present hybrid method.

AMS subject classifications: 65Z05

Key words: Dynamic mesh method, hybrid method, background grid, radial basis functions, delaunay graph mapping.

1 Introduction

Dynamic mesh is critical for many Computational Fluid Dynamics (CFD) problems such as flutter, static aeroelastics, optimization design of high lift system, and so on. Grid regeneration at every time step is often encountered for a dynamic system. However,

*Corresponding author.
Email: luzl@nuaa.edu.cn (Z. L. Lu)

 ©2015 Global Science Press

https://doi.org/10.4208/aamm.2014.m614 Published online by Cambridge University Press

it is not only extremely time-consuming but also very inconvenient. Therefore, efficient methods are of great importance to generate dynamic mesh with high mesh quality.

With the development of CFD, more and more dynamic mesh generation methods have been proposed, such as Transfinite Interpolation (TFI) [1–4], spring analogy [5–7], inverse distance weighted (IDW) [8, 9], elasticity method [10–12], radial basis functions (RBFs) [13–15], Delaunay graph mapping [16], and so on. Each method has its own advantages and disadvantages. Basically, RBF is time consuming but may lead to better quality mesh in relation to skewness. Delaunay mapping method is efficient but the skewness of the deformed mesh is more difficult to control. When the deformation is large or the geometry shape is complex, the mesh quality cannot be kept.

Recently, more and more hybrid methods have been developed. Ding [17] applied RBFs and TFI for complex multi-block structured grid. Spekreijse [18] developed a grid deformation algorithm for multi-block structured grid, which is based on volume spline interpolation and TFI. Zhou [19] combined the spring analogy with Delaunay graph mapping to generate the dynamic mesh with large deformation. Although various dynamic mesh generation methods have been continuously improved and have gotten wide applications, dynamic mesh generation is still very challenging in the cases of complex configuration, large grid number and large deformation.

In this paper, a hybrid method for dynamic mesh generation is developed by combining RBFs with Delaunay graph mapping. In order to improve the Delaunay graph mapping to deal with complex configuration and large deformation well, the background grid used in Delaunay graph mapping, which only consists of mesh points on the boundary, is replaced by a set of fairly coarse unstructured background grid. The background grid contains moderate mesh points in the interior flow field and maintains the same mesh distributions as the given computational mesh on the wall surface. Then RBFs method is utilized to deform the background grid by choosing partial points on the boundary as the control points, and Delaunay graph mapping method is used to deform the computational mesh by employing area or volume weight coefficients. Different methods are used for a moving NACA0012 airfoil, and the comparison indicates that RBFs-Delaunay graph mapping hybrid method is as accurate as RBFs and is as efficient as Delaunay graph mapping. The present hybrid method is applied to one 2D and two 3D dynamic mesh problems with different complexities, and the numerical results demonstrate that the dynamic meshes for all test cases can be generated in an accurate and efficient manner.

2 RBFs and Delaunay graph mapping

2.1 RBFs

RBFs [20, 21] use the interpolation function f to describe the displacement in the whole physical space, and f can be approximated by a weighted sum of the basis functions

https://doi.org/10.4208/aamm.2014.m614 Published online by Cambridge University Press

$$f(\mathbf{x})=\sum_{i=1}^{N}a_i\cdot\varphi(\|\mathbf{x}-\mathbf{x}_i\|)+\psi(\mathbf{x}), \tag{2.1}$$

where $\mathbf{x}_i=[x_i,y_i,z_i]$ are the coordinates of control points with known displacements, N is the number of control points, φ is a given basis function with respect to the Euclidean distance $\|\cdot\|$ and $\psi=b_0+b_1x+b_2y+b_3z$ is a linear polynomial. The coefficients a_i and b_0, b_1, b_2, b_3 are evaluated from the collocation condition and additional requirement, which are formulated as follows

$$f(\mathbf{x}_i)=\mathbf{d}_i,\quad \sum_{i=1}^{N}a_i=\sum_{i=1}^{N}a_ix_i=\sum_{i=1}^{N}a_iy_i=\sum_{i=1}^{N}a_iz_i=0, \tag{2.2}$$

where $\mathbf{d}_i$ is the discrete known displacement on the boundary. Then, the coefficients a_i and b_0, b_1, b_2, b_3 can be obtained by solving the following equation system

$$\begin{bmatrix}\mathbf{M} & \mathbf{P}\\ \mathbf{P^T} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{a}\\ \mathbf{b}\end{bmatrix}=\begin{bmatrix}\mathbf{d}\\ \mathbf{0}\end{bmatrix}, \tag{2.3a}$$

$$\mathbf{M}=\begin{bmatrix}\varphi_{11} & \varphi_{12} & \cdots & \varphi_{1n}\\ \varphi_{21} & \varphi_{22} & \cdots & \varphi_{2n}\\ \vdots & \vdots & \ddots & \vdots\\ \varphi_{n1} & \varphi_{n2} & \cdots & \varphi_{nn}\end{bmatrix},\quad \mathbf{P}=\begin{bmatrix}1 & x_1 & y_1 & z_1\\ 1 & x_2 & y_2 & z_2\\ \vdots & \vdots & \vdots & \vdots\\ 1 & x_n & y_n & z_n\end{bmatrix}, \tag{2.3b}$$

$$\mathbf{a}=\begin{bmatrix}a_1\\ a_2\\ \vdots\\ a_n\end{bmatrix},\quad \mathbf{b}=\begin{bmatrix}b_0\\ b_1\\ b_2\\ b_3\end{bmatrix},\quad \mathbf{d}=\begin{bmatrix}d_1\\ d_2\\ \vdots\\ d_n\end{bmatrix}, \tag{2.3c}$$

where $\varphi_{ij}=\varphi(\|\mathbf{x}_i-\mathbf{x}_j\|)$. There are several well-established methods to solve Eq. (2.3a), such as the fast iterative techniques [22] and methods based on partition of unity [23].

After the interpolation function f is determined, the displacement $\mathbf{d}_P$ for any interior point $\mathbf{x}_P=[x_P,y_P,z_P]$ can be calculated by

$$\mathbf{d}_P=f(\mathbf{x}_P). \tag{2.4}$$

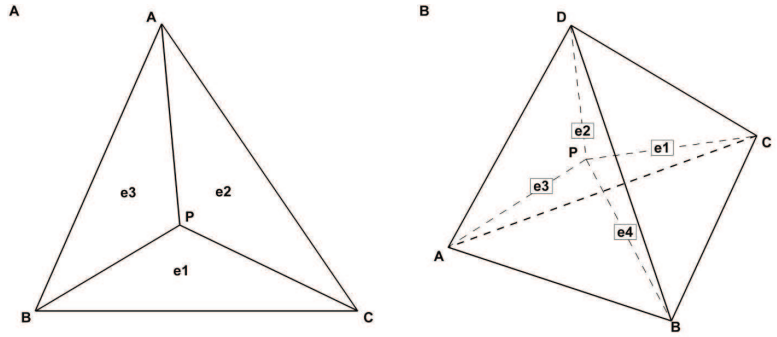
Boer has discussed the effects of different basis functions on mesh quality in [24]. RBFs with the spline type function $f(\mathbf{x})=\|\mathbf{x}\|^3$ are adopted in this work.

2.2 Delaunay graph mapping

The algorithm of Delaunay graph mapping can be written in a uniform weighted sum as

$$U=\sum_{i=1}^{N}W_iU_i,\quad \sum_{i}^{N}W_i=1, \tag{2.5}$$

https://doi.org/10.4208/aamm.2014.m614 Published online by Cambridge University Press



(a) (b)

Figure 1: Mesh point P located in a grid cell: (a) for 2D and (b) for 3D.

where U_i is the function value, W_i is the weight coefficient.

Due to simplicity and uniqueness, Delaunay graph mapping which takes area ratios or volume ratios as the weight coefficients are widely used.

2D cases

Fig. 1(a) illustrates a mesh point P located in a 2D triangle cell ABC, and the three weight coefficients W_1, W_2, W_3 are computed by using area ratios

$$W_i=\frac{S_i}{S},\quad i=1,2,3, \tag{2.6a}$$

$$S_1=\frac{1}{2}\begin{vmatrix} x_P & y_P & 1\\ x_B & y_B & 1\\ x_C & y_C & 1\end{vmatrix},\qquad S_2=\frac{1}{2}\begin{vmatrix} x_P & y_P & 1\\ x_C & y_C & 1\\ x_A & y_A & 1\end{vmatrix}, \tag{2.6b}$$

$$S_3=\frac{1}{2}\begin{vmatrix} x_P & y_P & 1\\ x_A & y_A & 1\\ x_B & y_B & 1\end{vmatrix},\qquad S=\frac{1}{2}\begin{vmatrix} x_A & y_A & 1\\ x_B & y_B & 1\\ x_C & y_C & 1\end{vmatrix}. \tag{2.6c}$$

3D cases

Fig. 1(b) illustrates a mesh point P located in a 3D tetrahedral element $ABCD$, and the four weight coefficients W_1, W_2, W_3, W_4 are computed by using volume ratios

$$W_i=\frac{V_i}{V},\quad i=1,2,3,4, \tag{2.7a}$$

$$V_1=\frac{1}{6}\begin{vmatrix} x_P & y_P & z_P & 1\\ x_B & y_B & z_B & 1\\ x_C & y_C & z_C & 1\\ x_D & y_D & z_D & 1\end{vmatrix},\qquad V_2=\frac{1}{6}\begin{vmatrix} x_P & y_P & z_P & 1\\ x_C & y_C & z_C & 1\\ x_D & y_D & z_D & 1\\ x_A & y_A & z_A & 1\end{vmatrix}, \tag{2.7b}$$

https://doi.org/10.4208/aamm.2014.m614 Published online by Cambridge University Press

$$V_3=\frac{1}{6}\begin{vmatrix} x_P & y_P & z_P & 1\\ x_D & y_D & z_D & 1\\ x_A & y_A & z_A & 1\\ x_B & y_B & z_B & 1 \end{vmatrix}, \qquad V_4=\frac{1}{6}\begin{vmatrix} x_P & y_P & z_P & 1\\ x_A & y_A & z_A & 1\\ x_B & y_B & z_B & 1\\ x_C & y_C & z_C & 1 \end{vmatrix}, \tag{2.7c}$$

$$V=\frac{1}{6}\begin{vmatrix} x_A & y_A & z_A & 1\\ x_B & y_B & z_B & 1\\ x_C & y_C & z_C & 1\\ x_D & y_D & z_D & 1 \end{vmatrix}. \tag{2.7d}$$

After weight coefficient W_i is obtained, the displacement vector $\Delta\mathbf{x}_P$ of the mesh point P can be calculated by

$$\Delta\mathbf{x}_P=\sum_{i=1}^{N} W_i\Delta\mathbf{x}_i, \tag{2.8}$$

where $N=3$ is for a 2D case and $N=4$ is for a 3D case, and $\Delta\mathbf{x}_i$ is the known displacement vector of the grid cell vertexes.

3 Hybrid RBFs-Delaunay graph mapping method for dynamic mesh generation

For a given computational mesh, the efficiency and quality of a dynamic mesh generation method are most important. The traditional Delaunay graph mapping method is efficient but not very accurate, while RBFs method is accurate but not efficient. The present work is motivated to combine the advantages of two dynamic mesh methods.

Based on Delaunay graph mapping algorithm, the background grid is needed. However, there are some differences between the background grid used in this paper and the traditional Delaunay graph. The RBFs method is applied to deform the background grid, so a few interior nodes are added which can help improving the deformation capacity of background grid. The initial background grid is generated based on few boundary points for reducing the computed amount of RBFs interpolation and it is very convenient for choosing the control points for RBFs interpolation. Then the background grid is regenerated partly for ensuring the areas of computational mesh and background grid are conformable. However, it will not increase the calculated amount because this step does not add any interior point. After the background grid is moved, the computational mesh can be deformed easily through Delaunay graph mapping method. The process of the current hybrid method mainly includes the following six steps:

Step 1 Generating the background grid;

Step 2 Choosing the control points for RBFs interpolation;

Step 3 Regenerating the background grid around the geometric shape;

https://doi.org/10.4208/aamm.2014.m614 Published online by Cambridge University Press

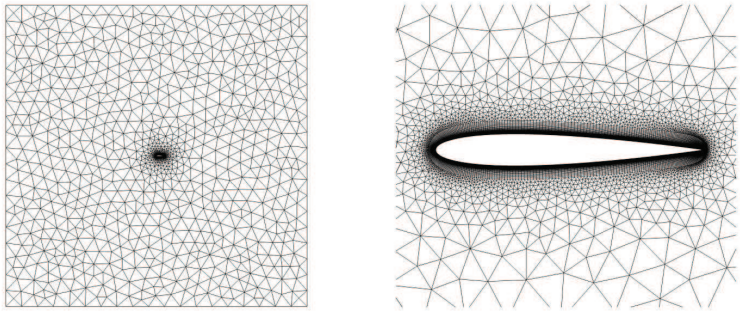
Step 4 Locating the computational mesh;

Step 5 Deforming the background grid;

Step 6 Relocating the computational mesh.

For a specific dynamic mesh problem like static aeroelastic numerical computation for an aircraft, steps 1-4 are done before numerical iteration and only steps 5-6 need to be repeated at every time step to generate the dynamic mesh.

In order to describe the RBFs-Delaunay graph mapping hybrid method in detail, a NACA0012 airfoil rotating 30 degrees about its quarter-chord point is considered. The initial hybrid grid for computation is shown in Fig. 2. The *O*-type structured grid and unstructured grid are generated around the airfoil and in the remaining space, respectively. In total, there are 29,701 mesh points, 26,520 quadrangle cells and 5,878 triangle cells.



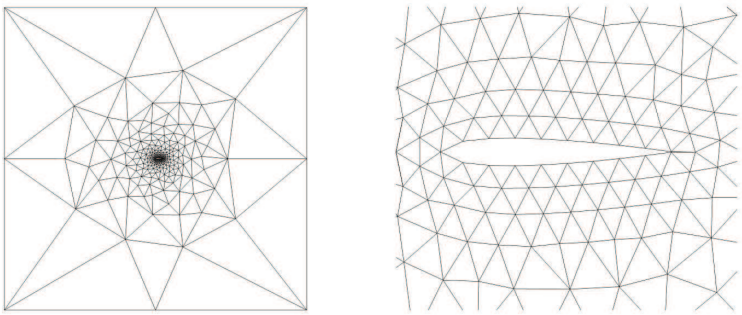
(a) (b)

Figure 2: Initial hybrid grid of NACA0012 airfoil.

3.1 Generating background grid

Most of dynamic mesh methods are applied by using a single computational grid. In 2006, Liu et al. [16] proposed a fast dynamic mesh deformation method based on Delaunay graph mapping, and recently it has become popular for unstructured dynamic mesh generation. On mapping the computational grid to the background one with a certain algebraic method, dynamic computational mesh can be generated by utilizing the deformation of background grid. The background grid used in Delaunay graph mapping is composed of all points on wall surface and a few boundary points in far field and symmetric plane. The lack of interior points has resulted in some serious problems: deformation cannot be accurately spread from boundary to interior space and dynamic mesh quality will be very bad in the case of large deformation. This problem can be well solved by adding some interior points into Delaunay graph. In this work, the background grid used in Delaunay graph mapping is replaced by fairly coarse unstructured background grid. Based on the computational mesh, the curve dimension can be reduced to an order

https://doi.org/10.4208/aamm.2014.m614 Published online by Cambridge University Press



(a) (b)

Figure 3: Background grid of NACA0012 airfoil.

of magnitude on geometry surface and 3∼5 on far-field boundary, regenerating it, and the new mesh is the background grid. As shown in Fig. 3, there are 308 mesh points and 543 triangle cells.

3.2 Choosing control points for RBFs interpolation

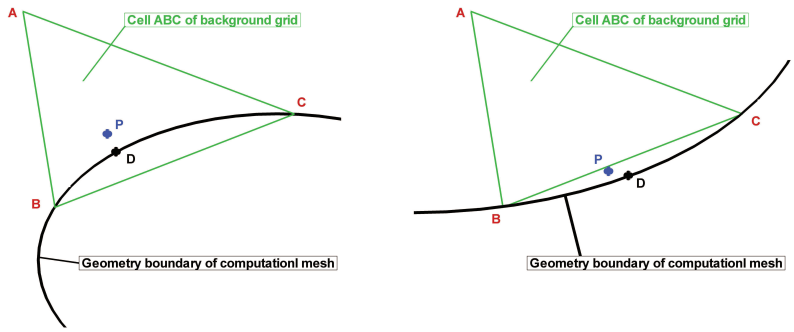
RBFs method is very desirable for scattered data interpolation and has been increasingly applied to dynamic mesh generation. A severe disadvantage of RBFs is that the computational effort is extremely large for a complex problem with a large grid number, and some attempts have been made to improve its efficiency [25–27], such as reducing size of equation system, decreasing the number of control points and so on.

For the present hybrid method, all boundary nodes are taken as the control points of RBFs interpolation. There are two main reasons: first, RBFs method is just used to deform the background grid, and the node number of background grid is small so that the calculated amount of RBFs interpolation is very small; second, for a RBFs interpolation system, taking points with known value as control points is the best fit, more or less control points will cause the loss of interpolation precision.

3.3 Regenerating background grid around geometric shape

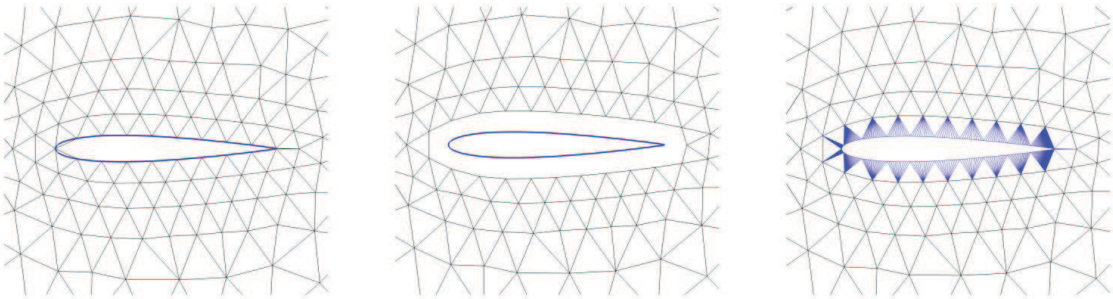
In the previous Section 3.1, we have talked about the generation of background grid, however, this background grid cannot be applied directly. Since the grid size between computational mesh and background grid varies greatly, and the shape of geometry boundary could not always be straight, and two inevitable problems appear. Take Fig. 4 as an example. The black point D is a normal point at the geometry boundary of computational mesh, and the blue point P is a random point of computational mesh which is nearby point D. For the left picture, the boundary of computational mesh is convex, and the displacements of point P which belongs to cell ABC only depend on the deformations of point A, B and C. On the other hand, the deformation of point D is known because

https://doi.org/10.4208/aamm.2014.m614 Published online by Cambridge University Press



(a) (b)

Figure 4: Two inevitable problems of the initial background grid.



(a) (b) (c)

Figure 5: Regenerating background grid around NACA0012 airfoil.

it is on the boundary. This means that the deformations of two closely spaced points are determined by two different ways, and the mesh may intersect in this area. For the right picture, the point P even does not belong to any unit.

In order to solve the above two problems, the area which the background grid covers must be strictly the same as the area covered by the computational mesh. Here, those mesh units which are around the geometry boundary are deleted, and the remaining mesh along with the computational mesh boundary points are loaded to a mesh generation software, unstructured mesh can be generated easily in the empty area. Fig. 5 illustrates the main procedures of background grid regeneration, and there are 697 mesh points and 932 triangle cells finally.

It should be mentioned that the Delaunay graph mapping method can be applied for triangle cells and tetrahedron cells only. In 3D cases, when the boundary mesh type of computational mesh is quadrangle or other non-triangle shape, the regenerated background grid cell close to wall boundary may be an arbitrary polyhedron. So, it has to be divided into several tetrahedrons. As shown in Fig. 6, pyramid $ABCDE$ is divided into two tetrahedrons $ABCE$ and $ACDE$.

https://doi.org/10.4208/aamm.2014.m614 Published online by Cambridge University Press

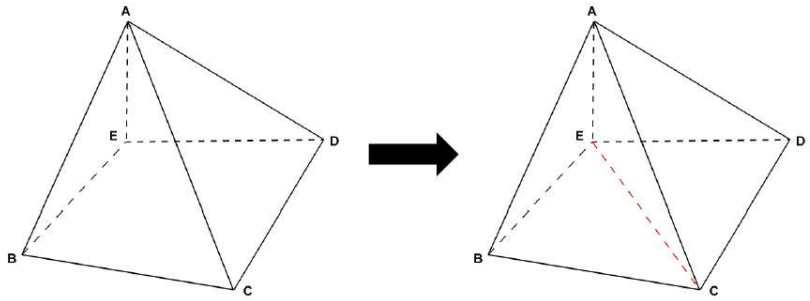


Figure 6: Treatment of non-tetrahedron background grid cell.

3.4 Locating the computational mesh

For a set of computational mesh and its corresponding background grid, different rules may be used to locate the computational mesh points in the background grid, but these rules must satisfy that the positions of the mesh points in the graph are uniquely defined by a one-to-one mapping. In this paper, we apply the area coefficients to define the points for 2D and volume coefficients to define points for 3D meshes [16]. Based on Eq. (2.6a) or Eq. (2.7a), a mesh point is judged to locate in a background grid cell if all weight coefficients W_i are nonnegative. Before calculating the weight coefficients, it is better to check whether the point is in the rectangular region which is determined by the minimum and maximum x, y, z coordinates of the cell or not. This can help reducing the computational time significantly.

3.5 Deforming background grid

Usually dynamic mesh generation obeys the rules that far-field boundary is fixed and wall boundary moves according to the prescribed motion or known deformation. By choosing all boundary grid points as the control points, deformation of background grid points is efficiently computed with RBFs. Fig. 7 shows the deformed background grid.

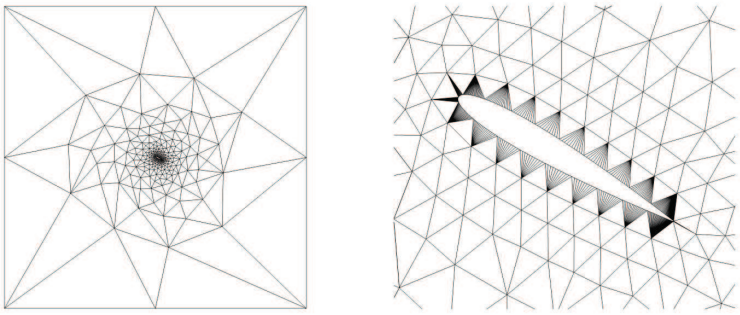
3.6 Deforming computational mesh

After background grid has been deformed, deformation of computational grid points is fast computed with the Delaunay graph mapping method as expressed in Eq. (2.8). Fig. 8 shows the deformed computational grid.

4 Comparison between dynamic mesh generation methods

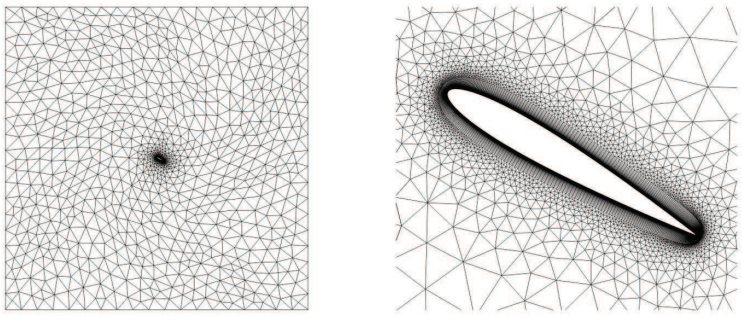
In this paper, three dynamic mesh generation methods including Delaunay graph mapping technique, RBFs method and present RBFs-Delaunay graph mapping hybrid method

https://doi.org/10.4208/aamm.2014.m614 Published online by Cambridge University Press



(a) (b)

Figure 7: Deformed background grid of NACA0012 airfoil.



(a) (b)

Figure 8: Deformed computational grid of NACA0012 airfoil.

are compared by the same dynamic mesh problem through three ways: CPU-time, mesh quality and CFD numerical simulation.

4.1 CPU-time

Time cost is an important index for a dynamic mesh method. In this paper, all CPU times are taken from a single 2.93GHz Intel processor, and all meshes are moved in one step.

4.2 Mesh quality

There are some mesh checking methods, such as angle skewness, quality, aspect ratio, determinant and so on. In order to check a single mesh cell, a parametric criterion for angle skewness is defined as

$$Q=1.0-\max\left(\frac{\theta_{Max}-\theta_A}{180-\theta_A},\frac{\theta_A-\theta_{Min}}{\theta_A}\right), \tag{4.1}$$

https://doi.org/10.4208/aamm.2014.m614 Published online by Cambridge University Press

where θ_{Max} and θ_{Min} are the largest included angle and the smallest one respectively, and θ_A is the average included angle of the checked cell. For example, θ_A equals to 60 degrees for a triangle cell.

In order to enable mesh quality inspection to focus on the mesh surrounding a geometric shape, which is most important for numerical simulation, a parametric criterion for the weighted average quality is introduced for the whole mesh as follows

$$\overline{Q}=\frac{\sum_{i=1}^{Ncell}Q_i(1-\tilde{V}_i^p)}{\sum_{i=1}^{Ncell}(1-\tilde{V}_i^p)}, \tag{4.2}$$

where Q_i is the cell quality of a mesh cell, $Ncell$ is the total mesh number, p is the power parameter with a value of 2 in this work, and $\tilde{V}_i$ is the normalized area or volume parameter calculated by

$$\tilde{V}_i=\frac{V_i-V_{Min}}{V_{Max}-V_{Min}}, \tag{4.3}$$

where V_i is the cell area or cell volume, V_{Min} and V_{Max} represent the minimum value and maximum value of V, respectively.

4.3 CFD numerical simulation

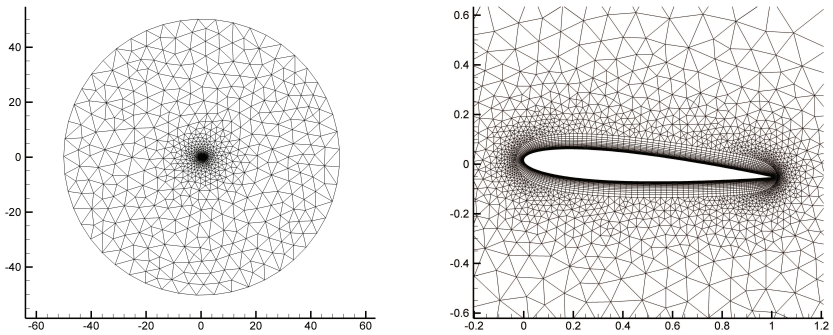
For a moving boundary flow problem, three main factors may affect the final result: the solver, number of iterations and the dynamic mesh method. In order to study the effects of dynamic mesh methods on aerodynamic force solely and purely, a series of steady flow simulations with same calculation condition and different deformation are conducted. Fluent 14.0 [28] is applied as the solver in this paper.

4.4 Example for comparison

Here, the NACA0012 airfoil is taken as the example for comparison. The computational mesh shown in Fig. 9 is generated through a hybrid way including 10,388 mesh points, 7,878 quadrangle cells, 4,769 triangle cells. There are 203 points on the geometry boundary and 50 points at the far field boundary. The far field boundary is designed as a circle with radius of 50 times the chord length and the center of the circle is the quarter-chord point of the airfoil. Thus, when the far field boundary is rotated around its center, the solution field can remain unchanged.

The simulation condition: the Mach number is 0.15, the Reynolds number per chord is 6 million, the reference temperature is 300 degree Kelvin, and the angle of attack is 4 degree.

https://doi.org/10.4208/aamm.2014.m614 Published online by Cambridge University Press



(a) (b)

Figure 9: Hybrid mesh of NACA0012 airfoil for comparison.

4.5 Comparison of results

Table 1 gives the comparison among three dynamic mesh methods on CPU-time. Rotating the far field boundary about its center along the counterclockwise direction step 5 degrees, the weighted average mesh quality and aerodynamic performances are performed in Fig. 10. Fig. 11 shows the mesh quality distribution of deformed meshes at 30 degrees rotation angle with three dynamic mesh methods.

Table 1: Comparisons among three dynamic mesh methods on CPU-time.

Dynamic mesh method	Delaunay graph mapping	RBFs	RBFs-Delaunay graph mapping
CPU-time/millisecond(ms)	<1	17	<1

As shown in Table 1, Fig. 10 and Fig. 11, the CPU times required by the present hybrid method and Delaunay graph mapping are of the same order of magnitude. The dynamic mesh quality and flow simulation results of RBFs-Delaunay graph mapping hybrid method is comparable with RBFs. For Delaunay graph mapping method, the mesh intersects when the rotation angle is greater than 45 degrees, and meshes keep well even though the angle is 90 degrees for RBFs and RBFs-Delaunay graph mapping methods. Comparisons indicate that RBFs-Delaunay graph mapping hybrid method is as accurate as RBFs and is as efficient as Delaunay graph mapping.

5 Test cases

The present hybrid dynamic mesh method has been applied to one 2D and two 3D mesh deformation problems with different complexities: a GA (W)-1 two-element airfoil, a wing with fuel tank and a wing-body-nacelle combination of a DLR-F6 airplane. Based on the given initial computational grids, the dynamic meshes for three test cases have been generated in an accurate and efficient manner. All CPU times are taken from a sin-

https://doi.org/10.4208/aamm.2014.m614 Published online by Cambridge University Press

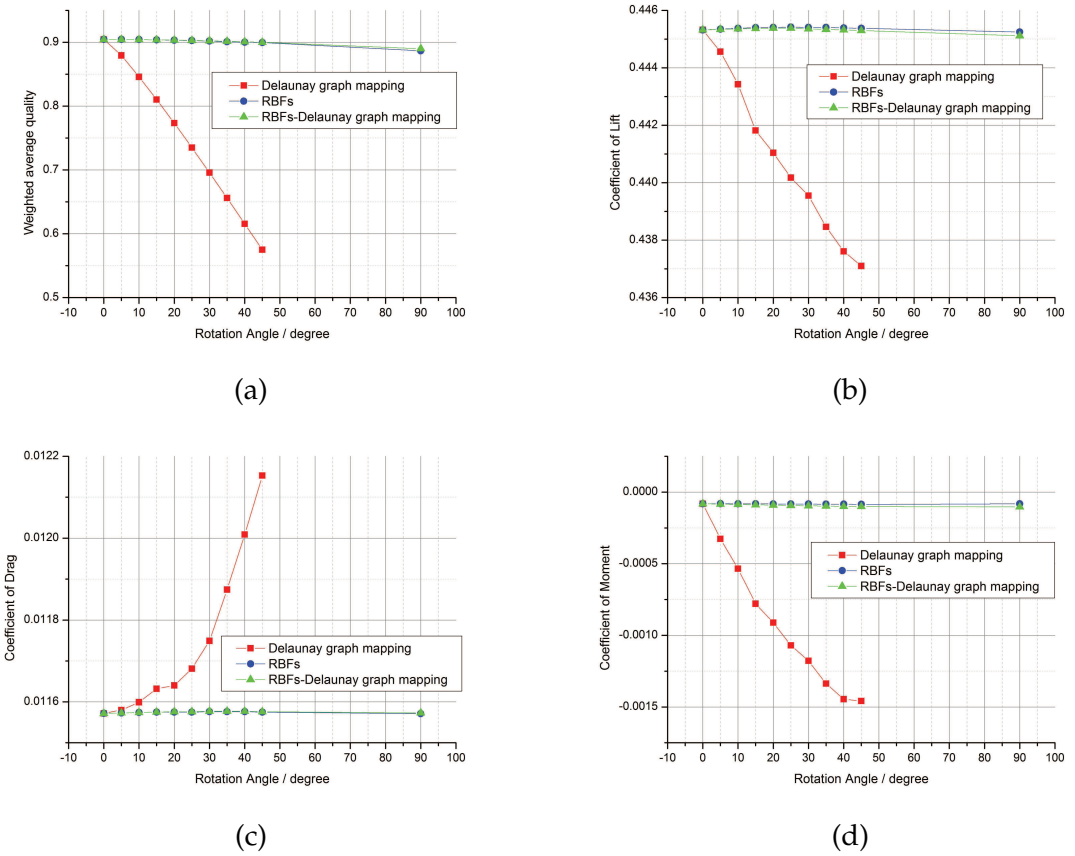


Figure 10: Weighted average mesh quality and aerodynamic performances of NACA0012 airfoil changing with deformation.

gle 2.93GHz Intel processor, and all mesh qualities are evaluated by using the weighted average as formulated in Eq. (4.2).

5.1 Test case 1: GA (W)-1 two-element airfoil

High lift device (HLD) plays an important role in aircraft design, and one key problem for HLD design with CFD is dynamic mesh generation. Test case 1 is a GA (W)-1 two-element airfoil where flap rotates 30 degrees about its leading edge. The initial hybrid viscous mesh is shown in Fig. 12, and it contains 41,738 grid points, 35,661 quadrangle cells and 11,407 triangle cells. The corresponding background grid contains 2,552 grid points and 4,425 triangle cells, just shown in Fig. 13. Fig. 14 shows the mesh quality distributions for the initial and deformed meshes. For this case, the CPU-time cost by RBFs-Delaunay graph mapping hybrid method is 3ms; the values of weighted average quality are 0.941 and 0.893 respectively for the initial and deformed meshes.

https://doi.org/10.4208/aamm.2014.m614 Published online by Cambridge University Press

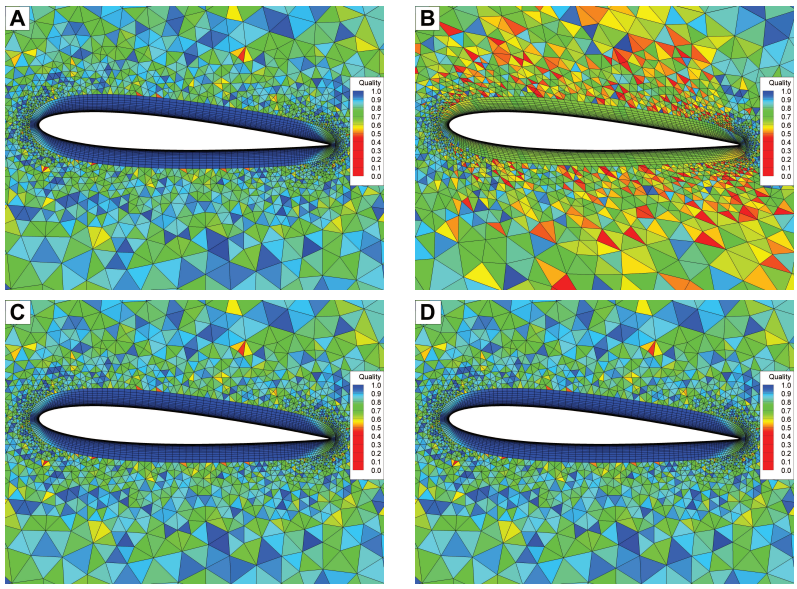
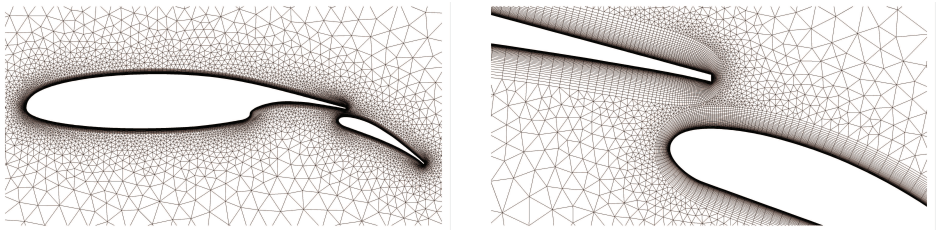


Figure 11: Mesh quality distribution of NACA0012 airfoil: (A) initial mesh, (B) deformed mesh using Delaunay graph mapping, (C) deformed mesh using RBFs and (D) deformed mesh using RBFs-Delaunay graph mapping hybrid method.



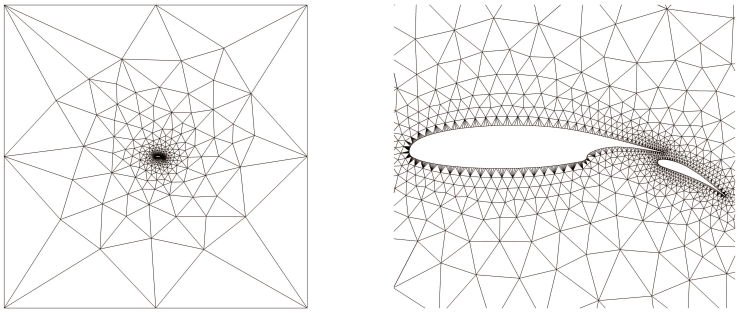
(a) (b)

Figure 12: Initial hybrid grid of GA (W)-1 two-element airfoil.

5.2 Test case 2: Wing with fuel tank

Test case 2 is a wing with fuel tank. The wing is undergone a bending-torsion coupling deformation, where bending deformation and torsion angle at the wingtip are 0.32m and 30 degrees respectively. Fig. 15 shows the initial and deformed configurations. For this wing-fuel tank combination, the initial unstructured grid contains 268,086 grid points and 1,526,302 tetrahedron cells. Background grid at geometry surface and symmetry plane is performed in Fig. 16. Fig. 17 shows mesh quality distributions for the initial and deformed meshes. For this case, the CPU-time needed by RBFs-Delaunay graph mapping hybrid method is 50ms, and the dynamic mesh quality value of 0.580 is very comparable with the initial mesh quality value of 0.592.

https://doi.org/10.4208/aamm.2014.m614 Published online by Cambridge University Press



(a) (b)

Figure 13: Background grid of GA (W)-1 two-element airfoil.

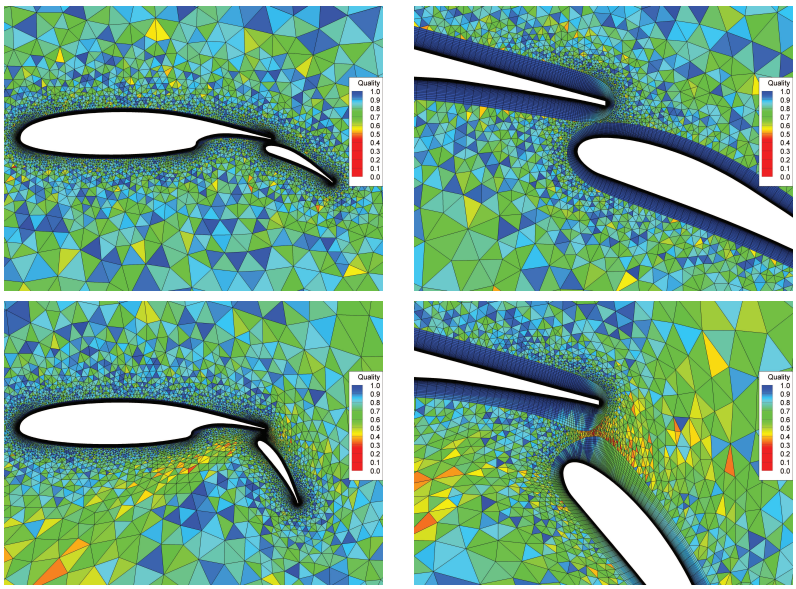


Figure 14: Mesh quality distributions for GA(W)-1 two-element airfoil: the upper is initial mesh and the lower is deformed mesh.

5.3 Test case 3: wing-body-nacelle combination

Test case 3 is a wing-body-nacelle combination of a DLR-F6 airplane. The wing is undergone a vertical bending deformation and the maximum deformation at the wingtip is 50% of the semi-wingspan. Fig. 18 gives the initial and deformed configurations. For this wing-body-nacelle combination, the initial viscous mesh contains 1,200,992 grid points, 6,818,718 tetrahedron cells and 58,210 pyramid cells. Fig. 19 is the corresponding background grid, it can be easily seen that all structured domains at surface were triangulated. Fig. 20 shows the mesh quality distributions for the initial and deformed meshes. For this case, the CPU-time required by RBFs-Delaunay graph mapping hybrid method is 600ms,

https://doi.org/10.4208/aamm.2014.m614 Published online by Cambridge University Press

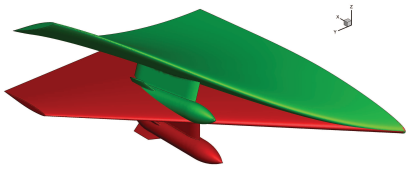
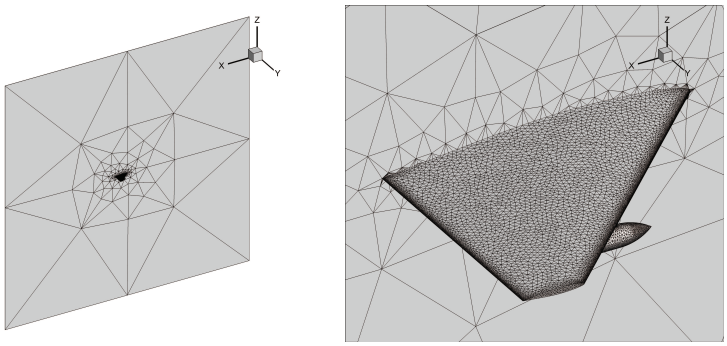


Figure 15: Configurations of wing with fuel tank: the red is Initial shape and the green is deformed shape.



(a) (b)

Figure 16: Background grid at geometry surface and symmetry plane of wing with fuel tank.

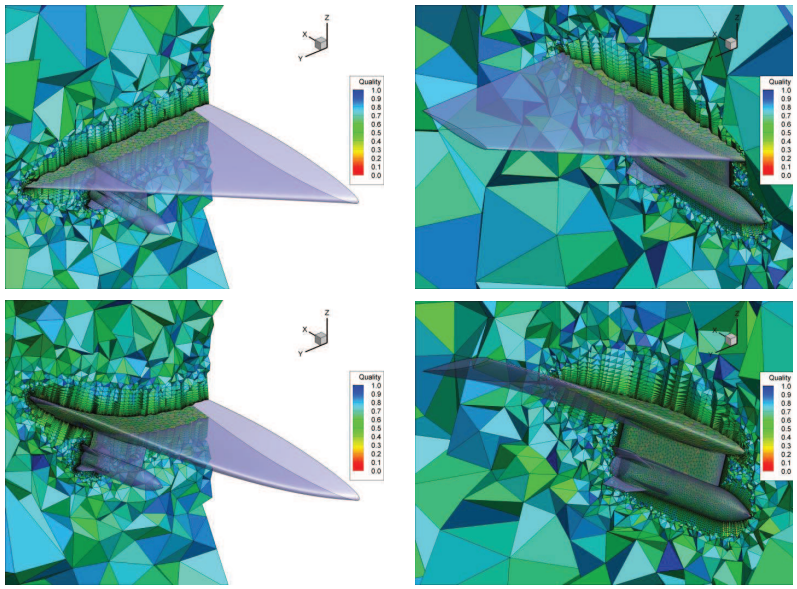


Figure 17: Mesh quality distributions for wing with fuel tank: the upper is initial mesh and the lower is deformed mesh.

and the dynamic mesh quality value is 0.465, which is nearly equal to the initial mesh quality value of 0.472.

https://doi.org/10.4208/aamm.2014.m614 Published online by Cambridge University Press

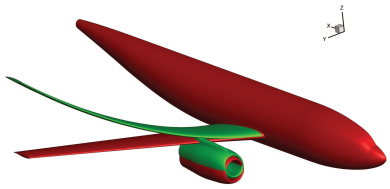
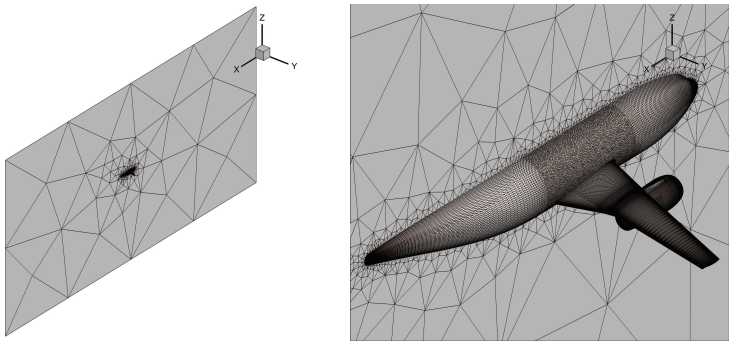


Figure 18: Configurations of DLR-F6: the red is Initial shape and the green is deformed shape.



(a) (b)

Figure 19: Background grid at geometry surface and symmetry plane of DLR-F6.

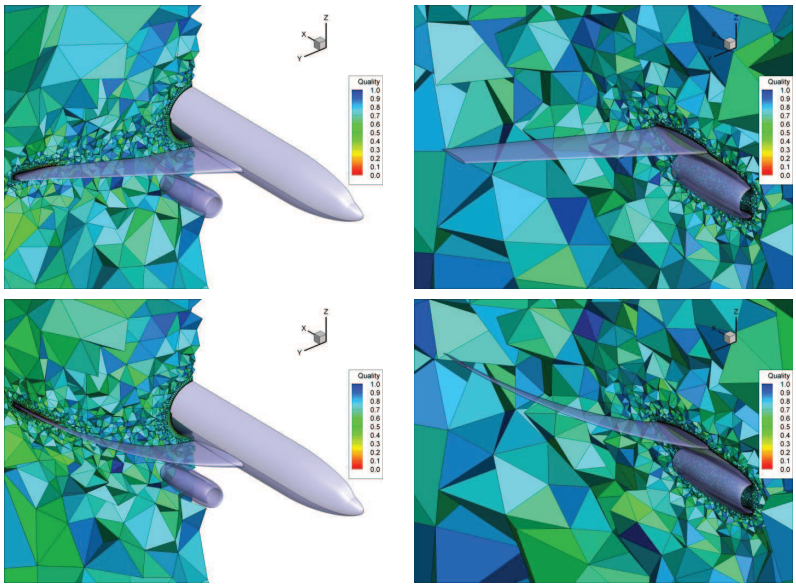


Figure 20: Mesh quality distributions for DLR-F6: the upper is initial mesh and the lower is deformed mesh.

6 Conclusions

A novel hybrid dynamic mesh method is proposed in this work, which combines the most favorable elements of RBFs and Delaunay graph mapping. The unstructured back-

https://doi.org/10.4208/aamm.2014.m614 Published online by Cambridge University Press

ground grid is carefully designed to improve Delaunay graph mapping technique to deal with complex configuration and large deformation well. By using the present RBFs-Delaunay graph mapping hybrid method, the dynamic mesh with millions of grid cells can be achieved in several seconds and the dynamic mesh quality is comparable to the initial mesh. In the future work, the present hybrid mesh deformation method will be applied to unsteady aerodynamic problems with unsteady movement, aeroelastic problems and optimization design problems of airplane.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (Grant No. 11372135), the National Basic Research Program of China ("973" Project) (Grant No. 2014CB046200) and the Fundamental Research Funds for the Central Universities (Grant No. NS2013005).

References

[1] C. Byun and G. P. Guruswamy, *A parallel, multi-block, moving grid method for aeroelastic applications on full aircraft,* AIAA Paper 98-4782, September 1998.
[2] J. Reuther, A. Jameson, J. Farmer, L. Martinelli and D. Saunders, *Aerodynamics shape optimization of complex aircraft configurations via an adjoint formulation,* AIAA Paper 96-0094, January 1996.
[3] W. T. Jones and J. Samareh-Abolhassani, *A grid generation system for multi-disciplinary design optimization,* AIAA Paper 95-1689, June 1995.
[4] W. Huang, Z. L. Lu and T. Q. Guo et al., *Numerical method of static aeroelastic correction and jigshape design for large airliners,* Science China Tech. Sci., 55(9) (2012), pp. 2447–2453.
[5] J. T. Batina, *Unsteady Euler algorithm with unstructured dynamic mesh for complex-aircraft aeroelastic analysis,* AIAA J., 29(3) (1991), pp. 327–333.
[6] V. Venkatakrishnan and D. J. Mavriplis, *Implicit method for the computation of unsteady flows on unstructured grids,* J. Comput. Phys., 127(2) (1996), pp. 380–397.
[7] C. Farhat, C. Degand, B. Koobus and M. lesoinne, *Torsional springs for two-dimensional dynamic unstructured fluid meshes*, Comput. Methods Appl. Mech. Eng., 163 (1998), pp. 231–245.
[8] J. A. S. Witteveen, *Explicit and robust inverse distance weighting mesh deformation for CFD,* AIAA 2010-165, January 2010.
[9] J. A. S. Witteveen and H. Bijl, *Explicit mesh deformation using inverse distance weighting interpolation,* AIAA 2009-3996, June 2009.
[10] P. Z. Bar-Yoseph, S. Mereu and S. Chippada, *Automatic monitoring of element shape quality in 2D and 3D computational mesh dynamics*, Comput. Mech., 27(5) (2001), pp. 378–395.
[11] E. J. Nielsen and W. K. Anderon, *Recent improvements in aerodynamic design optimization on unstructured meshes,* AIAA J., 40(6) (2002), pp. 1155–1163.
[12] E. F. Sheta, H. Q. Yang and S. D. Hanchi, *Solid brick analogy for automatic grid deformation for Fluid-Structure interaction,* AIAA 2006-3219, June 2006.

https://doi.org/10.4208/aamm.2014.m614 Published online by Cambridge University Press

[13] R. FRANK, *Scattered data interpolation: Tests of some methods,* Math. Comput., 38 (1982), pp. 181–200.
[14] Z. M. WU, *Multivariate compactly supported positive definite radial functions,* Adv. Comput. Math., 4 (1995), pp. 283–292.
[15] F. BERNAL AND G. GUTIERREZ, *Solving delay differential equations through BRF collocation,* Adv. Appl. Math. Mech., 1 (2009), pp. 257–272.
[16] X. Q. LIU, N. QIN AND H. XIA, *Fast dynamic grid deformation based on Delaunay graph mapping*, J. Comput. Phys., 211(2) (2006), pp. 405–423.
[17] L. DING, Z. L. LU AND T. Q. GUO, *An efficient dynamic mesh generation method for complex multi-block structured grid,* Adv. Appl. Math. Mech., 6(1) (2014), pp. 120–134.
[18] S. P. SPEKREIJSE, B. B. PRANANTA AND J. C. KOK, *A simple, robust and fast algorithm to compute deformations of multi-block structured grids,* NLR-TP-2002-105, 2002.
[19] X. ZHOU, S. X. LI AND B. CHEN, *Spring-interpolation approach for generating unstructured dynamic meshes,* Acta Aeronautica Et Astronautica Sinica, 31(7) (2010), pp. 1389–1395.
[20] M. D. BUHMANN, *Radial basis functions,* Acta Numer., 9 (2000), pp. 1–38.
[21] C. SHU, H. DING AND K. S. YEO, *Local radial basis function-based differential quadrature method and its application to solve two-dimensional incompressible Navier-Stokes equations,* Comput. Methods Appl. Mech. Eng., 192 (2003), pp. 941–954.
[22] A. C. FAUL AND M. J. D. POWELL, *Proof of convergence of an interactive technique for thin plate spline interpolation in two dimensions,* Adv. Comput. Math., 11 (1999), pp. 183–192.
[23] H. WENDLAND, *Fast evaluation of radial basis functions: methods based on partition of unity,* Approximation Theory X: Wavelets, Splines, and Applications, Vanderbilt University Press, 2002, pp. 473–483.
[24] A. DE BOER, M. S. VAN DER SCHOOT AND H. BIJL, *Mesh deformation based on radial basis function interpolation,* Comput. Struct., 85 (2007), pp. 784–795.
[25] M. BOTSCH AND L. KOBBELT, *Real-time shape editing using radial basis functions*, Comput. Graphics Forum, 24(3) (2005), pp. 611–621.
[26] T. C. S. RENDALL AND C. B. ALLEN, *Efficient mesh motion using radial basis functions with data reduction algorithms,* J. Comput. Phys., 228 (2009), pp. 6231–6249.
[27] T. C. S. RENDALL AND C. B. ALLEN, *Improved radial basis function fluid-structure coupling via efficient localized implementation*, Int. J. Numer. Methods Eng., 78(10) (2009), pp. 1188–1208.
[28] ANASYS Software Corporation, ANASYS FLUENT 14.0 User Manual, Printer in U.S.A, 2011.

https://doi.org/10.4208/aamm.2014.m614 Published online by Cambridge University Press