Proceedings of the International Association for Shell and Spatial Structures (IASS) Symposium 2009, Valencia
Evolution and Trends in Design, Analysis and Construction of Shell and Spatial Structures
28 September – 2 October 2009, Universidad Politecnica de Valencia, Spain
Alberto DOMINGO and Carlos LAZARO (eds.)

A motion mesh approach for fluid-structure interaction of spatial structures

Xiaping ZHANG[a], Dai ZHOU*,Xiangqiao CHEN[a]

*Professor, School of Naval Architecture, Ocean and Civil Engineering,
Shanghai Jiao Tong University
No. 800 Dongchuan Road, Shanghai 200240, People's Republic of China
Email: zhoudai@sjtu.edu.cn

[a] School of Naval Architecture, Ocean and Civil Engineering,
Shanghai Jiao Tong University

Abstract

Mesh motion strategy is one of the key problems in fluid-structure interaction analysis (FSI). One popular technique which often used to solve this problem is known as the spring analogy method. In this paper a new mesh update approach based on the spring analogy method is presented for the effective treatment of mesh moving boundary problem,which can avoid the generation of squashed invalid elements and maintain mesh quality by considering each element shape and grid scale in the definition of the spring stiffness. Meanwhile, the approach is applied to several 2D and 3D boundary correction problems for fully unstructured meshes and evaluated by a mesh quality indicator. With this application, it is demonstrated that the present approach can keep good mesh quality even under large motion of bodies. Finally, the computational robustness of the present approach on is highlighted.

Keywords: spatial structure, mesh motion, unstructured mesh, mesh quality, grid scale

1. Introduction

Since last decades, long-span spatial structures have been worldwidely used in a variety of public buildings. As being characterized by lightweight and flexibility, the great vibration of long-span spatial structures could be excited by wind action, which may also affect the wind pressure distribution on structural surface. Mesh motion strategy is one of the key problems in FSI analysis (Lopez *et al*. [12]). Already several mesh motion approaches for the dynamic mesh have been developed and show good computational results under circumstances of structural oscillation with small amplitude and small structural deformation (Batina [2]; Venkatakrishnan and Mavriplis [18]; Crumpton and Giles [6]; Nielsen and Anderson [14]; Tezduyar [17]). However, these approaches are still far from

being mature ones for geometrically complex and greatly moving and deforming body problems.

The re-meshing method (Löhner *et al*. [11]) which generates the mesh automatically at each time step on the time-dependent domain, is commonly used for moving boundaries problems if the mesh has some distorted and invalid elements. By the use of the re-meshing, the quality of the obtained mesh system could be improved. However, the re-meshing has serious problems, that is, it suffers from the loss of physical conservation law. In other words, it may locally reduce the computational accuracy due to drastic grid size variation. It also needs extra computational costs especially when solving a flow field about three-dimensional complex bodies (Douglass *et al*. [7]).

Common approach of the unstructured dynamic meshes is to use the spring analogy (Bottasso *et al.* [5]; Farhat *et al*. [9]; Markou *et al.* [13]), which is introduced by Batina [1], who used the method for a forced vibration problem of an airfoil. Many researchers have adopted the spring analogy to solve moving boundary problems. To give a few examples, the spring analogy is used for free surface problems by Slikkeveer *et al*. [16], store separation problems by Hassan *et al*. [10], forced vibration and fluid – structure interaction problems by Blom and Leyland [4], and aero-elastic calculation by Farhat *et al*. [8] and Piperno [15].

The initial spring analogy only uses the linear tension spring analogy. However, this approach can only prevent the nodes collision, cannot control the cell shape and often creates ill-conditioned cells for large movements. Then the torsional springs are designed to prohibit the interpenetration of neighboring triangles or tetrahedrons by the consideration of the internal angles of triangles or the angles of tetrahedrons' two faces,which show good robustness performance and extend the applications of spring analogy to two-dimensional and three-dimensional problems.

The lineal spring stiffness is a value related to grid scale, while the torsional spring stiffness related to element shape. Usually, different meshes in a flow field possess different grid scales. In order to ensure the computational precision in simulating flow field, the mesh near the walls and boundaries should be kept in small grid scale.

In this paper, we present a new mesh motion approach by introducing a grid scale parameter to the definition of the spring stiffness. This approach is similar to the lineal sring formulation with angle information incorporated into spring stiffness, and the springs of which have different matching parameters in 2D and 3D moving boundary problems. Moreover, the robustness of this approach is evaluated by several applications with greatly moving and deforming bodies.

2. Numerical approaches

2.1. Spring analogy

Spring analogy models can be categorized into two types: vertex model and segment model (Piperno [15]). Vertex springs are always under tension unless the spring length is zero, and are mainly used for mesh smoothing because the tension tends to pull the mesh towards equal inter-nodal distances. On the other hand, segment springs have zero tension at some

equilibrium length (before deformation) and nodal displacement on the boundary creates spring forces that subsequently displace internal nodes. The descriptions for these two kinds of springs are consistent (Blom [3]). However, segment spring models are better suited for moving internal nodes to follow a dynamically deforming computational domain. Only segment spring analogy is discussed in this paper.

The lineal spring stiffness k_{ij} for a given element edge *i-j* takes the following general form:

$$k_{ij} = \frac{1}{[(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2]^{\alpha}} \tag{1}$$

where (x_i, y_i, z_i) and (x_j, y_j, z_j) are the spatial coordinates of the two nodes connected by the edge *i-j*, and α is the coefficient. Node collision is prevented by selecting the spring stiffness as the inverse of the edge length, demonstrated by Blom [3] in 1D analysis. It is noted that the version of Eq. (1) takes the original form studied by a great deal of numerical applications when $\alpha = 1$.

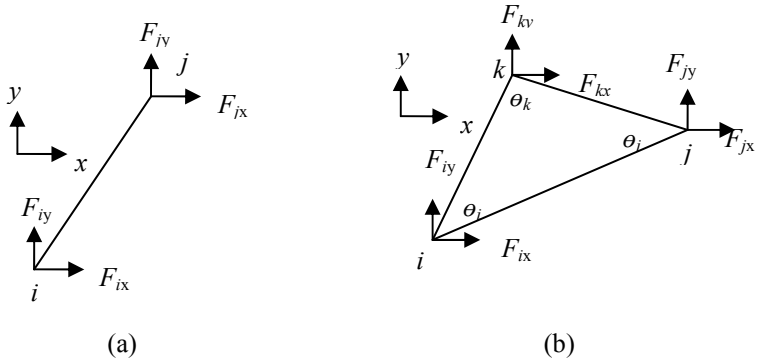


Figure 1: The spring simulation principle
(a) The lineal spring; (b) The torsional spring

Neither angular displacement nor area (volume) change of an element is associated with its edges' stiffness, which means that element inversion are not sensed by the lineal model. To prevent element inversion, a torsional spring is attached to each vertex in addition to the lineal spring. The stiffness of the torsional spring is defined as

$$C_i^{ijk} = \frac{1}{\sin^2 \theta_i^{ijk}} \tag{2}$$

where C_i^{ijk} is the torsional stiffness associated with the angle θ_i^{ijk} where the node *i* is as its vertex on the triangle Δ_{ijk}. By virtue of the definition for torsional stiffness, as θ_i^{ijk} approaches zero or π, the corresponding torsional stiffness increases rapidly and prevents element inversion.

A semi-torsional spring analogy is a approach which incorporates both edge and angle information into the spring stiffness. For 2D triangular elements, a semi-torsional stiffness of an edge *i-j* was proposed by Blom [3],

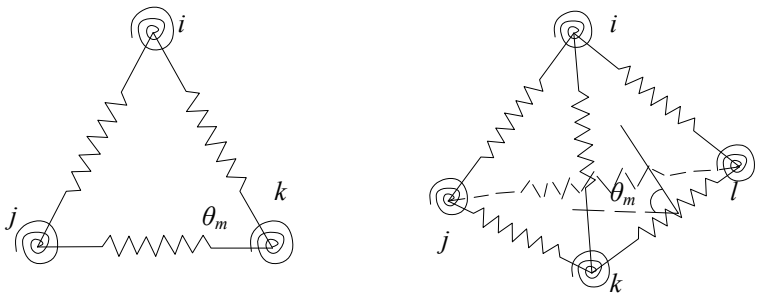
$$k_{ij}^{semi-torsional} = \frac{k_{ij}^{lineal}}{\theta} \tag{3}$$

where θ is the angle facing the edge *i-j* of the element.

Another kind of semi-torsional spring analogy model in which the stiffness of an edge *i-j* is defined as the sum of its lineal stiffness and its torsional stiffness proposed by Zeng and Ethier [19]:

$$k_{ij}^{semi-torsional} = k_{ij}^{lineal} + k_{ij}^{torsional} \tag{4}$$

In comparison with the semi-torsional model as in Eq. (3), the semi-torsional stiffness as in Eq. (4) has several advantages (Zeng and Ethier [19]). Here, we choose the Eq. (4) as the combination pattern of the spring stiffness.



(a) (b)

Figure 2: Definition of the spring stiffness
(a) 2D spring stiffness; (b) 3D spring stiffness

As shown in Fig. 2, the stiffness of an edge *i-j* is defined as:

$$k_{ij} = \lambda\left(\frac{1}{l_{ij}^{\alpha}} + \left(\sum_{m=1}^{NE_{ij}} \frac{1}{\sin^2 \theta_m^{ij}}\right)^{\beta}\right) \tag{5}$$

where NE_{ij} is the number of elements sharing edge *i-j*. In 2D mesh system, θ_m^{ij} is the angle the edge *i-j* facing and NE_{ij} = 2; while in 3D mesh system, θ_m^{ij} is the angle between two faces as shown in Fig. 2b, the value of NE_{ij} depends on the mesh generation while α and β are coefficients depended on dimension, where α = 2 and β = 4 in 2D problem, and α = 1

and $\beta = 3$ in 3D problem for a great deal of numerical applications. the parameter λ is the grid scale parameter, defined as

$$\lambda = \frac{l_0}{l_{ij}} \tag{6}$$

where l_0 is the largest scale of the model. From the Eq. (5) and Eq. (6), we can see that the stiffness is greater in smaller grid scale, so the higher density meshes near the moving boundaries cannot move easily and the mesh quality can be preserved. For this reason, the mesh can move uniform under different grid scales using the present stiffness definition.

2.2. Solution technique

The fictitious spring force $\mathbf{F}_{ij}^n$ acting on node i from i-j is:

$$\mathbf{F}_{ij}^n = k_{ij}^n(\boldsymbol{\delta}_j^n - \boldsymbol{\delta}_i^n) \tag{7}$$

where $\boldsymbol{\delta}_j^n$ and $\boldsymbol{\delta}_i^n$ are nodal displacements of both node j and i at step n, respectively:

$$\boldsymbol{\delta}_j^n = \mathbf{y}_j^n - \mathbf{y}_j^{n-1} \tag{8}$$

$$\boldsymbol{\delta}_i^n = \mathbf{y}_i^n - \mathbf{y}_i^{n-1} \tag{9}$$

The static equilibrium equation for node i at time step n is

$$\sum_{j=1}^{N_{ei}} k_{ij}^n(\boldsymbol{\delta}_i^n - \boldsymbol{\delta}_j^n) = 0 \tag{10}$$

where N_{ei} is the number of nodes directly connected to node i through fictitious springs. A system of equations is derived through applying the equilibrium conditions to all nodes in the mesh system. To linearize the system, the spring stiffness is computed using nodal coordinates of the previous time step. With displacements on boundary nodes prescribed, and nodal coordinates of the previous time step known, the system is solved iteratively for nodal displacements at internal nodes. Nodal coordinates are updated by using the nodal displacements and old coordinates:

$$\mathbf{y}_i^n = \mathbf{y}_i^{n-1} + \boldsymbol{\delta}_i^n \tag{11}$$

We implement the successive over-relaxation (SOR) algorithm to solve the linear system for nodal displacements at internal nodes. The system is linearized by using the spring constants based on the geometry of the previous time step. On the other hand, the stopping criterion for the SOR iteration was

$$\max\{\left|\delta_{1,i}^{k+1} - \delta_{1,i}^k\right|, \left|\delta_{2,i}^{k+1} - \delta_{2,i}^k\right|, \left|\delta_{3,i}^{k+1} - \delta_{3,i}^k\right|\} \le \varepsilon \tag{12}$$

where i ranged over all internal nodes, k denoted iteration, and δ_1, δ_2, δ_3, are three components of nodal displacement. Based on a variety of preliminary tests, we set ε between 5×10^{-6} and 5×10^{-5}, which can ensure robust convergence at an acceptable computational cost.

2.3. Mesh quality judgment

Since the static equilibrium equations for the mesh are elliptic, the principle of Saint Venant holds for the deformation of the mesh. Therefore, boundary displacement does not spread far into the mesh. Besides, the quality of mesh near the boundary is a guarantee of the computational precision of the convection term when simulating the flow field. Here, only the quality of meshes attaching to moving boundaries is measured in numerical applications, and we use mesh gradient as the yardstick of the mesh quality, defined as

$$R = \max\left[\frac{Q_{\max} - Q_e}{\pi - Q_e} \cdot \frac{Q_e - Q_{\min}}{Q_e}\right] \tag{13}$$

where, in 2D problem, Q_e is equilateral triangle's interior angle, $Q_e = \pi/3$, and in 3D problem, Q_e is the angle between two surfaces of regular tetrahedron, $Q_e = 2\arctan\frac{\sqrt{2}}{2}$. Q_{max} and Q_{min} are the maximum and minimum interior angles of one element respectively. The more R approaches zero, the better mesh quality becomes.

3. Numerical applications

3.1. 2D rolling NACA0012 airfoil

For the validation of the present approach, the grid movement of a two-dimensional NACA0012 airfoil is computed. The computational grid is shown in Fig. 3. The number of the grid nodes and triangle cells are 3875 and 7587, respectively. This airfoil is assumed to be forced to rotate around the quarter chord as follows,

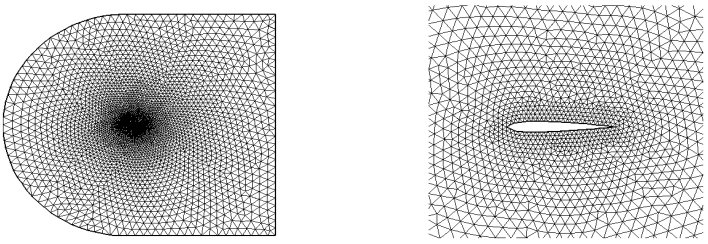
$$\alpha = \alpha_0 + \alpha_m \sin\frac{\pi}{2n} \tag{14}$$

where α, α_0, α_m and n are the instantaneous angle of attack, average angle of attack, the oscillation amplitude, and the steps of the movement, respectively. Here, α_0 and α_m are set to be 0° and 45°, which is very great amplitude. And n is set at 100.

Using the linear spring method, the maximum and minimum pitch angles can only achieve at 27°, and then invert element occurs as shown in Fig. 4a. Fig. 4b and 4c are the results of torsional spring method and the present approach respectively. From Fig. 4b, we can see that elements near the boundary distort dramatically by using the torsional spring method, where some elements are squashed heavily and some are elongated. In contrary, as Fig. 4c shown, the grid moves very smoothly as the airfoil pitches by using the present approach and elements near the airfoil keep in good quality even with the relatively great oscillation.

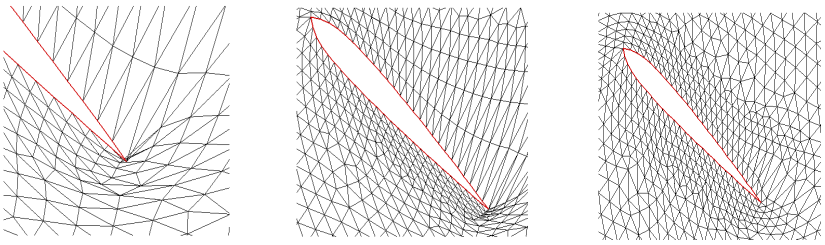
Fig. 5 shows the statistics of the mesh gradient criteria near the moving boundary. We divide the mesh gradient criteria into ten intervals as follows: 0~0.1; 0.1~0.2; ……0.9~1.0. The abscissa is the ten intervals while the ordinate represents the percentage of elements in each interval versus total elements. After the movement obtained by using the torsional spring method, the majority of mesh gradient R are greater than 0.5, that is, elements have great distortion. But, the majority of mesh gradient R is in 0.4~0.6 by using present

approach. The mesh around the greatly curved airfoil after the movement is shown in Fig. 6, where the mesh system is still fair around the region.



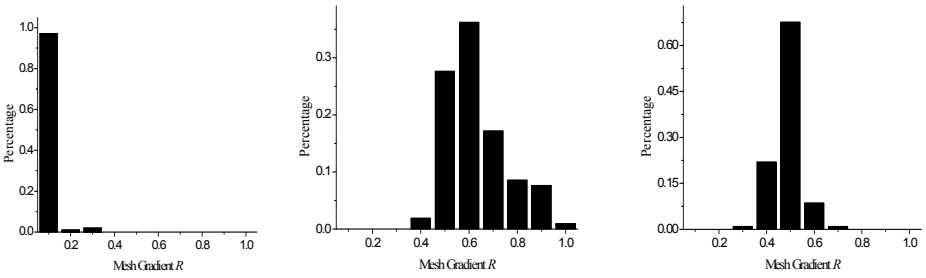
(a) (b)

Figure 3: Computational grid of NACA0012 airfoil
(a) Overview; (b) Close-up view



(a) (b) (c)

Figure 4: Close-up view of the airfoil
(a) Lineal spring method; (b) Torsional spring method; (c) The present spring approach



(a) (b) (c)

Figure 5: Statistics of the mesh gradient criteria
(a) Origin mesh; (b) Using torsional spring method; (c) Using the present spring approach

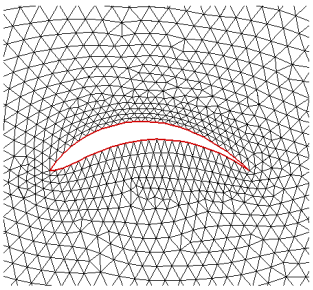


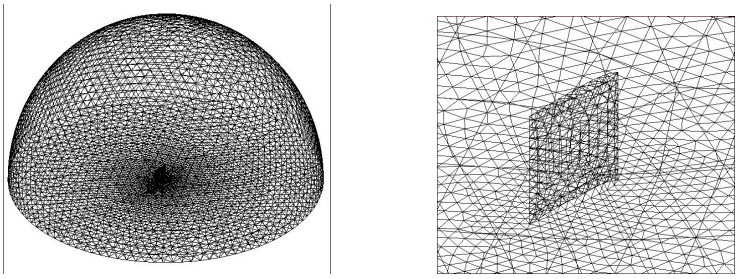
Figure 6: Close-up view of the curved airfoil

3.2. 3D rolling NACA0012 airfoil

Moreover, the proposed approach is also evaluated through another example of the rotation of a three-dimensional wing, where the computational grid is shown in Fig. 7. The number of the grid nodes and tetrahedral cells are 57924 and 308850, respectively. The outer boundary is a sphere whose radius is fifteen times of the airfoil length. This wing is assumed to be forced to rotate around the half chord by Eq. (14). Here, α_0, and α_m are set to be 0° and 30°, which is fairly great amplitude. And n is set at 100.

The results by using the linear spring analogy are shown in Fig. 8, where some invalid tetrahedrons are generated near the trailing edge at attack angle of about 17° and more rotation cannot continue. If using the torsional analogy, the rotary angle only achieves at about 26°. Fortunately, by using the present approach, the mesh system can move smoothly without any invalid tetrahedrons and the rotary angle of α_m can be achieved.

The results of mesh gradient R using the present approach are shown in Fig. 9. We can see that there are no invalid tetrahedrons near the boundary domain. Fig.10a~10c show the moved meshes by using the present approach at the rotary angles of 30° and **(-**30°**)** as well as under a great curve.



(a) (b)

Figure 7: Computational mesh of NACA0012 airfoil
(a) Overview; (b) Close-up view

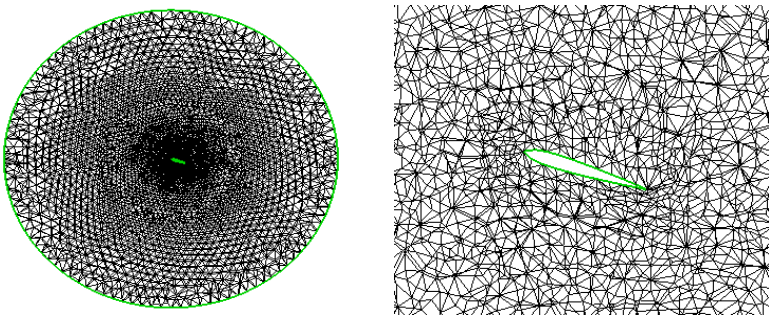
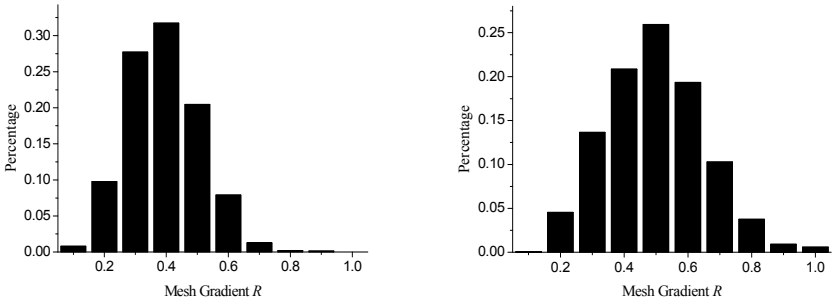
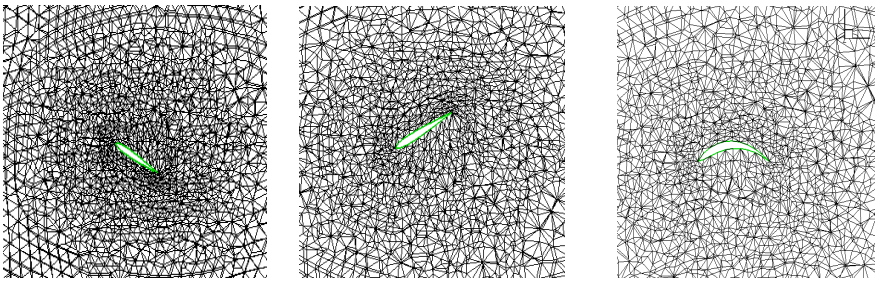


Figure 8: Computational mesh of NACA0012 airfoil after the pitching motion
By using linear spring method



(a) (b)

Figure 9: Statistics of the mesh gradient criteria
(a) Origin mesh; (b) Using the present approach



(a) (b) (c)

Figure 10: Moved mesh by the present approach
(a) At the rotary angle of 30°; (b) At the rotary angle of (−30°); (c) Under a great curve

4. Conclusion

In this paper, we develop grid scale parameters to the combination of the spring stiffness, in order to make the mesh move smoothly without generating invalid elements and keep them in good quality. The present motion mesh approach is applied to several typical problems and evaluated by the comparison with the lineal spring method and torsional spring mesh method. It is shown through a series of examples that the computation for the mesh movement fails under some cases or inhomogeneous mesh deformation if using the commonly-used approaches. However, by means of the present approach, the mesh system can convieniently keep in high quality and successfully move without much cost in CPU time even under severe deformation. Besides, it should be pointed out that, through extensive mesh testing of the spring analogy model, we can implement the mesh updating algorithm for solving the 2D/3D incompressible N-S equations in the arbitrary Lagrange–Euler (ALE) formulation.

Acknowledgement

Support from the Doctoral Disciplinary Special Research Project of Chinese Ministry of Education (No.200802480056) and the Key Project of Fund of Science Technology Development of Shanghai (No. 07JC14023) are acknowledged.

References

[1] Batina J.T., Unsteady Euler airfoil solutions using unstructured dynamic meshes. *AIAA Journal*, 1990; **28**(8);1381-1388.

[2] Batina J.T., Unsteady euler algorithm with unstructured dynamic mesh for complex-aircraft aerodynamic analysis. *AIAA Journal*, 1991; **29**(3):327-333.

[3] Blom F.J., Considerations on the spring analogy. *International Journal for Numerical Methods in Fluids*, 2000; 32(6); 647-668.

[4] Blom F.J. and Leyland P., Analysis of fluid–structure interaction on moving airfoils by means of an improved ALE method, in AIAA 1997. *Fluid Dynamics Conference*, Snowmass Village, CO, AIAA-1997-1770.

[5] Bottasso C.L., Detomi D. and Serra R., The ball-vertex method: a new simple analogy method for unstructured dynamic meshes. *Computer Methods in Applied Mechanics and Engineering*, 2005; **194**; 4244-4264.

[6] Crumpton P.I. and Giles M.B., Multi-grid aircraft computations using the OPlus parallel library. *Parallel Computational Fluid Dynamics*, 1996; **196**; 339-346.

[7] Douglass R.W., Carey G.F. and White D.R., Hansen G.A., Kallinderis Y., Weatherill N.P., Current views on grid generation: summaries of a panel discussion. *Numerical Heat Transfer*, 2002; **41**; 211-237.

[8] Farhat C., Lesoinne M. and Maman N., Mixed explicit/implicit time integration of coupled aeroelastic problems: three-field formulation, geometric conservation and distributed solution. *International Journal for Numerical Methods in Fluids*, 1995; **21**; 807-835.

[9] Farhat C., van der Zee K.G. and Geuzaine P., Provably second-order time-accurate loosely-coupled solution algorithms for transient non-linear computational aeroelasticity. *Computer Methods in Applied Mechanics and Engineering,* 2006; **195**(17-18); 1973-2001.

[10] Hassan O., Probert E.J. and Morgan K., Unstructured mesh procedures for the simulation of three-dimensional transient compressible inviscid flows with moving boundary components. *International Journal for Numerical Methods in Fluids*, 1998; **27**; 41-55.

[11] Löhner R., Yang C., Baum J.D., Luo H., Pelessone D., Charman C.M., The numerical simulation of strongly unsteady flows with hundreds of moving bodies. *International Journal for Numerical Methods in Fluids,* 1999; **31**; 113-120.

[12] Lopez E.J., Nigro N.M. and Storti M.A., Toth J.A., A minimal element distortion strategy for computational mesh dynamics. *International Journal for Numerical Methods in engineering,* 2007; **69**; 1898-1929.

[13] Markou G.A., Mouroutis Z.S., Charmpis D.C., Papadrakakis M., The ortho-semi-torsional (OST) spring analogy method for 3D mesh moving boundary problems. Computer Methods in Applied Mechanics and Engineering, 2007; 196; 747-765.

[14] Nielsen E.J. and Anderson W.K., Recent improvements in aerodynamic design optimization on unstructured meshes. *AIAA Journal*, 2002; **40**(6); 1155-1163.

[15] Piperno S., Explicit/implicit fluid-structure staggered procedures with a structural predictor and fluid subcycling for 2D inviscid aeroelastic simulations. *International Journal for Numerical Methods in Fluids*, 1997; **25**(10); 1207-1226.

[16] Slikkeveer P.J., Loohuizen E.P. and O'Brien S.B.G., An implicit surface tension algorithm for Picard solvers of surface-tension-dominated free and moving boundary problems. *International Journal for Numerical Methods in Fluids*, 1996; **22**; 851-865.

[17] Tezduyar T.E., Finite element methods for flow problems with moving boundaries and interfaces. *Archives of Computational Methods in Engineering*, 2001; **8**(2); 83-130.

[18] Venkatakrishnan V. and Mavriplis D.J., Implicit method for the computation of unsteady flows on unstructured grids. *Journal of Computational Physics*, 1996; **127**; 380-397.

[19] Zeng D.H. and Ethier C.R., A semi-torsional spring analogy model for updating unstructured meshes in 3D moving domains. *Finite Elements in Analysis and Design*, 2005; **41**; 1118-1139.