Int. Conf. on Computational Methods for Coupled Problems in Science and Engineering
COUPLED PROBLEMS 2007
M. Papadrakakis, E. Oñate and B. Schrefler (Eds)

MESH UPDATE TECHNIQUES: ROBUSTNESS AND EFFICIENCY

G. MARKOU[*] AND M. PAPADRAKAKIS[†]

[*]National Technical University of Athens (NTUA)
Institute of Structural Analysis & Seismic Research
Zografou Campus, Greece GR - 157 73 Athens
E-mail: markgeo@centreal.ntua.gr

[†] National Technical University of Athens (NTUA)
Institute of Structural Analysis & Seismic Research
Zografou Campus, Greece GR - 157 73 Athens
E-mail: mpapadra@central.ntua.gr

Key words: Mesh Update Techniques, Coupled Problems, Solution Algorithms.

Summary. In this paper a new mesh update technique is proposed and a numerical study for this method and other three spring analogy methods is conducted. For the solution of the moving mesh update problem a PCG-based algorithm is used, combined with the proposed mesh update technique. The PCG-based algorithm is also applied to the other three mesh update techniques, namely the torsional, the semi-torsional and the ball-vertex spring analogy schemes. The robustness of these four spring analogy methods is evaluated and their computational efficiencies are compared in 3D benchmark problems, including the AGARD wing 445.6. It is demonstrated that the ortho-semi-torsional spring analogy concept, in conjunction with a PCG type method for the solution of fictitious elasticity problems, provides robustness for substantially distorted meshes and computational efficiency for large scale problems.

1. INTRODUCTION

The numerical simulation of the dynamically updated unstructured three-dimensional meshes arises in many engineering applications i.e. moving boundary problems, bio-fluid mechanics (blood flow in human arteries), including fluid-structure interaction (FSI) problems like dynamic parachute air interaction, airfoil oscillation, flutter prediction, hydrodynamic design and other aeroelastic instability problems. In such applications, when some of the fluid domain boundaries undergo a motion with large amplitude, it becomes necessary to solve the flow equation based upon a regenerated or a moving mesh. If the mesh is not regenerated then it has to be dynamically updated.

2. THE SPRING ANALOGY CONCEPT

2.1 Torsional spring analogy method

The basic disadvantage of the linear spring analogy method is the lack of controlling collapse mechanisms due to the fact that the linear spring stiffness coefficients are not affected by angular or by volume changes of the element. To prevent element inversion on a 2D triangle, three torsional springs where attached, one at each vertex of the triangle and their stiffness coefficients were depended on the angle θ of the corresponding vertex. This method was extended to 3D applications by Farhat and Degand [1], where in order to control the volume of a tetrahedral, 12 triangles (three for each tetrahedral vertex) were implemented.

2.2 The ball-vertex spring analogy method

The ball-vertex spring analogy method considers the same linear springs with the linear spring analogy method. The basic idea of this method is to introduce additional linear springs that resist the motion of a mesh vertex towards its opposed face [2].

2.3 Semi-Torsional spring analogy method

The semi-torsional spring analogy method also considers the same springs with the linear spring analogy method along the edges. In this case, the stiffness coefficient of each linear spring is equal to the torsional stiffness coefficient of the facing angle. For 2D triangular elements, a semi-torsional stiffness coefficient of an edge *i-j* was proposed by Blom [3]. For the deformation of 3D unstructured meshes, Zeng and Ethier [4] proposed recently a semi-torsional spring analogy model of which its edge's stiffness is defined as the sum of its linear and semi-torsional stiffness.

2.4 The Ortho-Semi-Torsional spring analogy method (OST)

In the proposed OST spring analogy method we consider four additional imaginary linear springs as shown in Fig. 1, in addition to the springs of the semi-torsional method. Every tetrahedral node is projected on the opposite face and the distance of the corresponding projection forms a linear spring with its stiffness coefficient given by equation (1). This linear stiffness coefficient is subsequently transferred to the neighboring edge springs in such a way that the shortest linear spring benefits more from this imaginary spring. The additional stiffness coefficient k_{si} corresponding to node s is equal to:

$$k_{si} = \frac{1}{l_{si}} \tag{1}$$

where l_{si} is the length of distance si. This additional stiffness coefficient is divided into three parts (with the help of three linear distribution parameters), one for each neighboring edge. The linear allocation parameters are given by:

$$\lambda_{si,1} = \frac{l_{qs}}{l_{total}}, \quad \lambda_{si,2} = \frac{l_{ps}}{l_{total}}, \quad \lambda_{si,3} = \frac{l_{rs}}{l_{total}} \tag{2}$$

where $l_{total} = l_{qs} + l_{ps} + l_{rs}$. Following the calculation of the three distribution parameters $\lambda_{gi,j}$ of the vertex $g = p,q,r,s \ and \ j = 1,2,3$, the additional edge stiffness coefficients corresponding to each neighboring edge of node *s*, are obtained by:

$$k_{qs}^{+} = \left(\frac{k_{si}^{ortho}}{\left(\lambda_{si,1}\right)^{\kappa 1}} \right), \; k_{ps}^{+} = \left(\frac{k_{si}^{ortho}}{\left(\lambda_{si,2}\right)^{\kappa 1}} \right), \; k_{rs}^{+} = \left(\frac{k_{si}^{ortho}}{\left(\lambda_{si,3}\right)^{\kappa 1}} \right) \tag{3}$$

where k_{si}^{ortho} is given by equation (1) and κ_1 is a coefficient related to the closeness of the projection *si* to the neighboring edges. The total stiffness coefficient of the *qs* edge of the element is:

$$k_{qs}^{total} = \left[\left(\frac{k_{si}^{ortho}}{\left(\lambda_{si,1}\right)^{\kappa 1}} \right) + \left(\frac{k_{qi}^{ortho}}{\left(\lambda_{qi,1}\right)^{\kappa 1}} \right) \right]^{\kappa 2} + k_{qs}^{S-T} + k_{qs}^{L} \tag{4}$$

where coefficient κ_2 affects the contribution of the additional linear spring stiffness coefficients correspondingly to the total stiffness of the edge. The coefficient κ_1 is implemented to the shortest edge of the corresponding node. For almost equal edge lengths, then κ_1 is applied to all three edges. This is controlled by a tolerance parameter ε_l:

$$\frac{l_{qs} - l_{ps}}{l_{total}}, \; \frac{l_{qs} - l_{rs}}{l_{total}}, \; \frac{l_{rs} - l_{ps}}{l_{total}} \leq \varepsilon_l \tag{5}$$

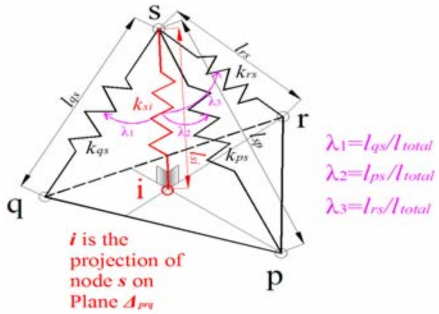

Figure 1: Projection of node *s* on the opposite face – Creation of the k_{si} imaginary linear spring coefficient.

If ε_l tolerance parameter is satisfied then the ratio importance coefficient κ_1 is implemented to the corresponding linear springs. In sections 3, of the numerical results, the following values are implemented: κ_1 =2, κ_2 = 1.5 and ε_l = 5%.

3. NUMERICAL APPLICATIONS

In this section, the superiority of the proposed OST spring analogy method and the computational performance of the adopted PCG-SSOR algorithm were studied. For this purpose, two 3D benchmark problems were considered. The first test problems consisted of unevenly structured grids of tetrahedral elements and were used to compare the computational efficiency of the proposed algorithm and the implementation adopted in [4], as well as to compare the computational performances of the four spring analogy methods. The 2nd benchmark problem was the AGARD wing 445.6.

4. CONCLUSIONS

In this work, an improved mesh update technique has been presented, the ortho-semi-torsional (OST) spring analogy method, which combines the simplicity of the semi-torsional spring analogy method and the robustness of the torsional spring analogy method. The proposed mesh update technique retains the advantages of both previously mentioned mesh update techniques without inheriting their weaknesses. Furthermore, a SSOR-PCG algorithm is implemented for the solution of the fictitious elastostatic problem. This solution algorithm was proved capable in improving the convergence properties for all four mesh update methods and performed considerably better than previously implemented solution methods for the mesh update problem. The importance of implementing a well formulated preconditioning scheme for solving iteratively a multi-step elastostatic problem, was also shown.

5. ACKNOWLEDGEMENTS

The first author acknowledges the financial support of the John Argyris Foundation.

REFERENCES

[1] C. Degand and C. Farhat, “A three-dimensional torsional spring analogy method for unstructured dynamic meshes”, Comput. Struct. 80 (2002) 305.

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

[3] F.J. Blom, “Considerations on the spring analogy”, Int. J. Numer. Methods Fluids 32 (2000) 647.

[4] D. Zeng and C.R. Ethier, “A semi-torsional analogy model for updating unstructured meshes”, G.E. Schneider (Ed.), Proceedings of the Ninth Annual Conference of the CFD Society of Canada, Waterloo, Ont., Canada, 5-27-2001, vol. 1, The CFD Society of Canada, 5-27-2001, p. 113.

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