Computers and Structures 136 (2014) 24–33

Contents lists available at ScienceDirect

Computers and Structures

journal homepage: www.elsevier.com/locate/compstruc

Vertex-Ball Spring Smoothing: An efficient method for unstructured dynamic hybrid meshes

T.J. Lin [a], Z.Q. Guan [a,*], J.H. Chang [a], S.H. Lo [b]

[a] State Key Laboratory of Structural Analysis for Industrial Equipment, Dept. of Engineering Mechanics, Dalian University of Technology, Dalian 116024, China
[b] Dept. of Civil Engineering, The University of Hong Kong, Pokfulam, Hong Kong, China

ARTICLE INFO

Article history:
Received 30 August 2013
Accepted 23 January 2014
Available online 18 February 2014

Keywords:
Mesh deformation
Dynamic mesh
Ball-vertex method
Vertex-ball spring system

ABSTRACT

Spring analogy approach is one of the most popular dynamic mesh deformation methods. In the ball-vertex method, perpendicular linear springs are introduced to deal with the element collapse problem. However, it is not very efficient as a large system of linear equations has to be resolved. In order to overcome this difficulty, the Vertex-Ball Spring Smoothing algorithm (VerBSS) is proposed in this paper. Following the mesh smoothing concept, a sub-spring system derived from the "ball-vertex" model is built and solved on a node by node basis using an $\boldsymbol{LDL}^{\mathrm{T}}$ solver. Interior nodes are smoothed layer by layer in an iterative manner to achieve the best result. Parallel scheme is also introduced in the smoothing process for further improvement of the efficiency. Numerical examples in two and three dimensions show that VerBSS is much more efficient than the ball-vertex method, and is capable of dealing with practical engineering objects with complex geometries subject to large deformations. VerBSS can be applied to complicated mesh topologies as well, not only to 2D/3D dynamic mesh, but also to the hybrid dynamic mesh.

1. Introduction

In modern numerical simulation of engineering applications, we often need to deal with many unsteady flow problems, such as free surface flows, bio-fluid mechanics problems, forced vibration and fluid–structure interaction problems. The mesh of the solution domain needs to be updated at each time step of the numerical simulation process when flow parameters and geometries of the computational field are changing with time. The dynamic mesh approach is one of the most popular methods to solve this class of problems, especially for multi-block structures moving in unsteady flow with irregular boundary displacement specifications. There are generally three ways to dynamically update unstructured meshes [1,2]. (1) Remeshing: according to the changes in the domain, local or global mesh is regenerated by mesh generator, in which the flow properties of the new mesh are obtained by interpolation. However, generating new meshes will not only increase the computational cost, but will also bring additional errors [3] in the computation. (2) Mesh deformation: through nodal repositioning, the shape and the size of the elements are changed while keeping nodal connectivity intact. It is simple to implement and it does not require topological modifications. Nevertheless, it is difficult to be applied in large displacement problems. (3) Combination of the two approaches: compared to mesh deformation, remeshing is much more computationally expensive, especially for the three-dimensional applications.

Mesh deformation method is mainly divided into three categories: the PDE-based mapping, the pseudo-material and the spring analogy method. Lohner [4] and Helenbrook [5] proposed Laplacian or bi-harmonic equation methods, which are based on PDE. This kind of methods employs potential equations to determine how nodes should be repositioned. By avoiding mesh degenerating and maintaining the size of elements, they can efficiently solve the problem of intersection between mesh edges. Due to the restriction of non-coupling property between displacement variables, PDE-based method is not suitable for complex geometries and large-scale problems. It is usually applied to small-scale mesh or optimization problems. For the pseudo-material approach [6,7], the basic idea is to map the fluid domain to a pseudo-material and to calculate the mesh deformation following the classical laws of mechanics governed by a set of boundary displacement conditions. The main drawback of the pseudo-material approach is its low computational efficiency. The most widely used mesh deformation technique is the spring analogy method, which was first introduced by Batina [8], in order to deform a mesh around a pitching airfoil. By this method, a network of springs connecting all vertices (nodes) in the mesh is created, that is, each edge in

* Corresponding author. Tel.: +86 13609843062.
E-mail address: guanzhq@dlut.edu.cn (Z.Q. Guan).

http://dx.doi.org/10.1016/j.compstruc.2014.01.028
0045-7949/© 2014 Elsevier Ltd. All rights reserved.

the mesh is replaced by a fictitious spring whose stiffness is inversely proportional to its length. In this method, the equilibrium length of the spring is equal to the initial length of the edge. When the boundary moves, the interior nodes will be repositioned so that the spring system will converge to a new static equilibrium. Spring analogy is more often applied to aero-elastic calculations [9,10]. The method is also applied to structured meshes as well, for example, by Nakahashi and Deiwert [11]. The system works well as long as the displacement is small compared to the element size. However, in some applications, the fluid domain boundary undergoes a motion with relatively large amplitude. It does indeed fail, when local invalid elements occur at some parts of the mesh. The reason why it fails has been studied for years. First of all, the resulting equation from the spring system is elliptic; it means that local perturbations only have local impact. Boundary perturbation cannot propagate into the interior solution domain effectively. Therefore, the spring analogy can only solve problems of small deformations. Some researchers have made improvements to the spring analogy method. By increasing the stiffness near the boundary, Blom [12] relieved the localization of deformation and enhanced the deformation capacity of the mesh. Secondly, the method lacks control mechanisms for element collapse. The spring method only considers the stretching effect of the spring, the stiffness of the edges is not related neither to the areas of the connected triangles nor to the angles of these triangles. To address this issue, torsional springs were added to the linear springs by Farhat et al. [13,14] in order to prevent the collapse of the elements. Later, Bottasso et al. [15] proposed a ball-vertex method which is arithmetically less complex than the torsional spring analogy method. The method introduces additional perpendicular linear springs that restrict the motion of the vertex towards its corresponding opposite face. These additional springs effectively confine each vertex within the polyhedral ball that encloses it. The resulting linear equation is solved by a Gauss–Seidel solver. When dealing with repeated cycles of severe deformations, such as airfoil oscillation, the ball-vertex method shows a remarkably consistent behavior, which seems to be more robust than the torsional spring. The method can be implemented with only a little additional computational cost comparing with the spring analogy method. The combination of the additional springs and the original springs can greatly improve the deformation capacity of the spring analogy method. However, it is still rather time consuming as a large system of linear equilibrium equations has to be resolved. Recently, a novel method [16–18] based on the creation of a background graph of the original mesh is developed. The mesh movement is carried out using the background graph with ease and efficiency. The mesh is then mapped back onto the deformed graph to provide the new mesh. Zhang [19] and Lin [20] et al. recommended improved methods for two-dimensional mesh deformation by adding new nodes into the background graph. Although the mesh update time is reduced, difficulties occur when it is extended to three-dimensional applications.

In this paper, a dynamic mesh deformation method named VerBSS (Vertex-Ball Spring Smoothing Method) is proposed. The algorithm is found to be robust for substantially distorted mesh and the solution strategy based on the $\boldsymbol{LDL}^{\mathrm{T}}$ solver can significantly improve the computational efficiency.

2. The spring-based methods

2.1. Spring analogy method

The classical edge spring analogy method is easy to implement. A basic spring system model is shown in Fig. 1. The edges of the mesh are considered as springs with stiffness inversely proportional to its length, and the spring system is in a balanced state. Given a mesh edge e_{ij}, which is connected by vertices i and j, the force on vertex i exerted by vertex j can be written as

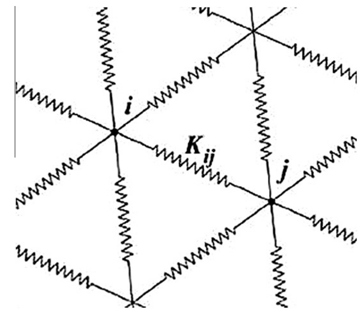

Fig. 1. Spring system.

$$\boldsymbol{f}_{ij}^{\text{Edge}} = k_{ij}(\boldsymbol{u}_j - \boldsymbol{u}_i) \cdot \boldsymbol{n}_{ij}\boldsymbol{n}_{ij} = -\boldsymbol{f}_{ji}^{\text{Edge}} \tag{1}$$

where k_{ij} is the stiffness of edge e_{ij}. $\boldsymbol{n}_{ij}$ is the unit vector from i to j. The displacement of vertex i and j are denoted by $\boldsymbol{u}_i$ and $\boldsymbol{u}_j$, respectively. Considering the effect of all its n vertices connected to vertex i, the equilibrium of node i is given by

$$\sum_{j=1}^{n} \boldsymbol{f}_{ij}^{\text{Edge}} = 0 \tag{2}$$

The elastic problem is solved for each interior node of the mesh and a large linear system $\boldsymbol{Kx} = \boldsymbol{b}$ is established. The global matrix $\boldsymbol{K}$ is formed with the edge spring stiffness k_{ij}, the vector $\boldsymbol{x}$ represents the displacements of the mesh vertices (nodes), and the vector $\boldsymbol{b}$ is formed based on the given boundary conditions.

By solving the linear system, the displacements of the interior nodes are obtained. A new mesh can be created by updating the nodal coordinates of the mesh according to the displacement vector.

2.2. Ball-vertex method

The disadvantage of the spring analogy method is the lack of collapse controlling mechanisms. That is, element inversions cannot be detected by the spring system. Node penetrations may occur as shown in Fig. 2. This is usually the case when large amplitude of motion is involved. To prevent elements inversion, additional perpendicular linear springs are added in the ball-vertex method.

As is shown in Fig. 3 [21], a polyhedral cavity can always be defined around an interior vertex. In the case of a simplicial mesh, for a given vertex i, we can construct the region G_i by all the elements adjacent to vertex i. The ball B_i of vertex i is defined as the face of G_i. B_i is the set of the elements' edges (in two dimensions) or triangles (in three dimensions) facing the vertex i. The boundary entities (edges or triangles) of the ball are shown with green lines in Fig. 3(a) and (c). Notice that in the non-simplicial case, we can firstly decompose the quadrilateral or hexahedral elements adjacent to vertex i into triangular or tetrahedral elements in the way that fictitious edges used for the decomposition may not pass through vertex i, and then obtain the region G_i and ball B_i which are used in the same way as the simplicial cases. The bounding entities of the ball are not part of the original elements, thus, we represent these entities using green dotted lines in Fig. 3(b) and (d). Perpendicular springs that connect node i and its projection points on the face of the ball are constructed as depicted in Fig. 4. The vertex can be restricted by the new springs not to leave its ball.

For the sake of simplicity, we only take a single tetrahedron for consideration. Let T_{ijkl} denote a tetrahedron whose vertices are respectively i j k and l, and p is the projection of vertex i onto its opposite face (see Fig. 4(c)). Perpendicular spring S_{ip} is constructed