Computer-Aided Design 67–68 (2015) 107–117

Contents lists available at ScienceDirect

Computer-Aided Design

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

Quality guaranteed all-hex mesh generation by a constrained volume iterative fitting algorithm[☆]

Hongwei Lin*, Sinan Jin, Hongwei Liao, Qun Jian

Department of Mathematics, State Key Lab. of CAD&CG, Zhejiang University, Hangzhou, 310027, China

HIGHLIGHTS

- An iterative algorithm is developed to fill a triangular mesh with an all-hex mesh.
- The Jacobian values of the all-hex mesh are guaranteed to be positive.
- The convergence of the iterative algorithm is proved.

ARTICLE INFO

Article history:
Received 23 May 2014
Accepted 23 May 2015

Keywords:
Solid modeling
Hexahedral mesh generation
Volume subdivision fitting
Guaranteed quality

ABSTRACT

The hexahedral mesh (hex mesh) is usually preferred to the tetrahedral mesh (tet mesh) in finite element methods for numerical simulation. In finite element analysis, a valid hex mesh requires that the scaled Jacobian value at each mesh vertex is larger than 0. However, the hex mesh produced by lots of prevailing hex mesh generation methods cannot be guaranteed to be a valid hex mesh. In this paper, we develop a constrained volume iterative fitting (CVIF) algorithm to fill a given triangular mesh model with an all-hex volume mesh. Starting from an initial all-hex mesh model, which is generated by voxelizing the given triangular mesh model, CVIF algorithm fits the boundary mesh of the initial all-hex mesh to the given triangular mesh model by iteratively adjusting the boundary mesh vertices. In each iteration, the movements of the boundary mesh vertices are diffused to the inner all-hex mesh vertices. After the iteration stops, an all-hex volume mesh that fills the given triangular mesh model can be generated. In the CVIF algorithm, the movement of each all-hex mesh vertex is constrained to ensure that the scaled Jacobian value at each mesh vertex is larger than 0, etc. Therefore, the all-hex mesh generated by the CVIF algorithm is guaranteed to be a valid all-hex mesh.

1. Introduction

In finite element methods for numerical simulation, the hexahedral mesh (hex mesh) is usually preferred to the tetrahedral mesh (tet mesh) owing to the reduced error and smaller number of elements [1,2]. However, generating a hex mesh with desirable qualities often requires significant geometric decomposition. Therefore, hex mesh generation can be extremely difficult to perform and automate. As a result, it requires considerable user interactions and may require days or even weeks in the case of complex shapes [3].

Moreover, it is well known that a valid hex mesh in finite element analysis should satisfy the requirement that, the scaled Jacobian value at each mesh vertex is larger than 0 [4]. Unfortunately, there is little work which can generate an all-hex mesh with the quality guarantee stated above. On the other hand, though boundary representation models, especially triangular mesh models, are popular in current computer graphics and computer aided design applications, lots of existing all-hex mesh generation algorithms need a tet mesh model as an input [5,6]. So they cannot handle triangular mesh models directly, and it is inconvenient.

In this paper, we develop a *constrained volume iterative fitting* algorithm (abbr. CVIF) which can *fill* a given triangular mesh model using an all-hex volume mesh, with guaranteed quality that the scaled Jacobian value at each mesh vertex is larger than 0. Given a triangular mesh model, we first construct an initial all-hex mesh model by voxelizing the given model, and extract the boundary quadrilateral mesh of the initial all-hex mesh model. Then, the initial all-hex mesh model is fitted to the given triangular mesh model by the CVIF algorithm. In each iteration of the CVIF algorithm, there are two steps:

(i) The adjustment of the vertices of the boundary quadrilateral mesh, and
(ii) the diffusion of the movement of the boundary mesh vertices to the inner mesh vertices.

☆ This paper has been recommended for acceptance by Jaroslaw Rossignac.

* Corresponding author. Tel.: +86 571 87951860 8304; fax: +86 571 88206681.
E-mail address: hwlin@zju.edu.cn (H. Lin).

http://dx.doi.org/10.1016/j.cad.2015.05.004
0010-4485/

In the above two steps, the movement of the mesh vertices is so constrained that the scaled Jacobian value at each mesh vertex after movement is larger than 0. In this way, the mesh quality of the all-hex mesh generated by our algorithm is guaranteed.

Specifically, the iterative adjustments of the boundary quadrilateral mesh vertices make up of the *constrained surface iterative fitting* algorithm (abbr. CSIF), and we show its convergence in this paper.

The structure of this paper is as follows. In Section 2, we briefly review related work. In Section 3, we develop the constrained volume iterative fitting algorithm. After presenting some results and discussions in Section 4, we conclude the paper in Section 5.

2. Related work

In this section, we will briefly review previous work related to our method, including hex mesh generation, subdivision fitting, and volume subdivision.

Hex mesh generation: There is a great deal of literature on the generation of volume meshes including tet [7,8] and hex meshes. In this paper, we focus on hex mesh generation. According to Owen's classification [9], hex mesh generation methods can be categorized into three classes, i.e., direct, indirect, and structured methods. Usually, the quality of the generated hex volume mesh should be improved by postprocessing [10,11].

Starting with a quadrilateral boundary surface mesh, *direct methods* generate a hexahedron for each quadrilateral according to a heuristically advancing-front approach. However, when the algorithmic heuristics are exhausted, no additional hexahedra can be placed. Consequently, this will leave void regions in the generated hex mesh [12,13].

Indirect methods first generate a tet mesh, and then convert it to a hex mesh by tetrahedral decomposition or combination. The disadvantage of these methods is that the quality of resultant hex mesh can be very poor owing to the high valence nodes [14,15].

A structured hex mesh is a mesh whose inner vertex valence is only six. A popular *structured method* for hex mesh generation is known as *mapping* [16], by which a map from the given solid with six surfaces to a cuboid is constructed. A cuboid has a trivial hex mesh, and the hex mesh in the given solid can be generated by inverse mapping. Although the mapping method can generate a high-quality hex mesh, it can only deal with solids of relatively simple shape, i.e., those with six boundary surfaces.

To deal with complex solids, a *submapping* method has been developed [17]. This submapping algorithm decomposes the given solid into separate mappable subregions while ensuring that the constraints within each subregion are consistent with the adjacent subregions.

The recently proposed hex mesh generation methods based on the PolyCube are also submapping methods that focus on the construction of the mapping [5]. A PolyCube [18] is a solid formed by combining a number of cubes with the same orientation; hence, it has a trivial hex mesh. By devising a mapping between the PolyCube and the input model, the sub-mapping method transfers the hex mesh in the PolyCube to the input model. Therefore, the quality of the hex mesh is heavily related to the shape of the PolyCube and the mapping.

In [19], the method of fundamental solutions is employed to design a harmonic volumetric mapping. In [20], the given model is first decomposed into the direct product of a surface and curve and then parameterized; subsequently, the mapping between the model and the PolyCube is constructed. Moreover, a volumetric deformation method is utilized to construct the correspondence between the given model and its PolyCube [6]. However, computing the PolyCube as well as a low-distortion mapping between it and the given model for general shapes remains an open problem [20,21]. Recently, some work is devoted to calculate a desirable polycube. In Ref. [22], the polycube is constructed using a variational method, by deforming an input triangle mesh through minimizing the l^1 norm of the mesh normals. In Ref. [23], a constrained discrete optimization technique is developed to make the generated polycube balance parameterization distortion against singularity count.

It should be pointed out that, though the mapping and sub-mapping methods can produce high quality hex mesh, they need a tet mesh as input. Therefore, the mapping and sub-mapping methods cannot be employed directly to transform a triangular mesh model into an all-hex mesh.

On the other hand, some hex mesh generation methods employ the frame field to guide the construction of mapping from the input tet mesh to the resulted hex mesh [24–26].

For more work on hex mesh generation, we refer the reader to excellent surveys [1,9].

Subdivision surface fitting: The limit surface of approximating subdivision schemes, such as Catmull–Clark scheme [27] will shrink, especially when the initial control mesh is sparse. Usually, this problem is solved by making the approximating subdivision surface fit the vertices of the initial mesh, by either global methods [28], or local methods [29].

Recently, some new methods, such as *progressive interpolation* (abbr. PI) and *geometric interpolation* (abbr. GI), have been proposed for subdivision surface fitting. They adjust the vertices of the control mesh iteratively, depending on either parametric distance in PI or geometric distance in GI, and the limit subdivision surface fits the initial mesh. The convergence of PI has been shown for the Loop [30], Doo–Sabin [31], and Catmull–Clark schemes [32]. On the other hand, Ref. [33] develops a geometric interpolation algorithm for the Loop subdivision scheme. Moreover, Ref. [34] presents a geometric approximation algorithm for the Loop subdivision surface by distributing the difference vector for each data point to the related control vertices. Given that the geometric interpolation (approximation) algorithm must compute the closest point on the limit surface for each data point in each iteration, it incurs great computational costs.

Different from existing subdivision fitting algorithms which take the limit surface of subdivision as the approximating surface, in this paper, we develop a constrained surface iterative fitting algorithm which takes the mesh surface after finite time subdivisions as the approximating surface.

Volume subdivision: Similar to the two-dimensional (2D) subdivision scheme, the volume subdivision scheme can generate a sequence of increasingly dense volume meshes by recursive subdivision starting from a coarse volume mesh. The volume subdivision is mainly applied to model deformation, and its smoothness analysis is difficult to handle.

Thus far, only a few volume subdivision schemes have been developed. To our best knowledge, the first volume subdivision scheme was developed in [35] for subdividing hex meshes. This scheme is an extension of the Catmull–Clark subdivision scheme. Furthermore, Bajaj et al. developed the MLCA subdivision rule and analyzed its smoothness [36]. The MLCA subdivision rule can be applied in any dimension, including 2D quadrilateral surface mesh, and 3D hex volume mesh. In Ref. [37], Pascucci introduced a subdivision scheme for 3D and higher dimensional meshes without the excessive vertex proliferation, which can be generalized to meshes of any dimension and with cells of virtually any type. Recently, inspired by the $\sqrt{3}$ subdivision scheme, an adaptive subdivision scheme for unstructured tetrahedral meshes was presented in [38], which generates only tetrahedra and supports adaptive refinement.