Finite Elements in Analysis and Design 132 (2017) 27–41

Contents lists available at ScienceDirect

Finite Elements in Analysis and Design

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

Steiner-point free edge cutting of tetrahedral meshes with applications in fracture

P. Areias[a,c,*], T. Rabczuk[b]

[a] Department of Physics, University of Évora, Colégio Luís António Verney, Rua Romão Ramalho, 59, 7002-554 Évora, Portugal
[b] Institute of Structural Mechanics, Bauhaus-University Weimar, Marienstraße 15, 99423 Weimar, Germany
[c] CERIS/Instituto Superior Técnico, University of Lisbon, Portugal

ARTICLE INFO

Keywords:
3D edge-based division
Local mesh refinement
Tetrahedral cutting
3D fracture

ABSTRACT

Realistic 3D finite strain analysis and crack propagation with tetrahedral meshes require mesh refinement/division. In this work, we use edges to drive the division process. Mesh refinement and mesh cutting are edge-based. This approach circumvents the variable mapping procedure adopted with classical mesh adaptation algorithms. The present algorithm makes use of specific problem data (either level sets, damage variables or edge deformation) to perform the division. It is shown that global node numbers can be used to avoid the Schönhardt prisms. We therefore introduce a nodal numbering that maximizes the trapezoid quality created by each mid-edge node. As a by-product, the requirement of determination of the crack path using a crack path criterion is not required. To assess the robustness and accuracy of this algorithm, we propose 4 benchmarks. In the knee-lever example, crack slanting occurs as part of the solution. The corresponding Fortran 2003 source code is provided.

1. Introduction

Division of tetrahedra with applications to fracture can make use of mesh adaptivity algorithms using edge-based division. 3D adaptivity is a classical subject and is discussed in mesh generation textbooks, cf. [11]. However, several aspects, such as the general case of edge-based division and the use of global node numbering to improve the division quality were not previously addressed. In addition, applications to fracture are infrequent and make use of specific crack front cases. We here address these issues and show applications to fracture, including slanting (see, e.g. [14] for an erosion-based algorithm predicting slanting).

Several discretization-based 3D applications make use of tetrahedron division:

- Level-set based operations and mesh creation/adaptivity for large deformations, including biomechanics applications [4].
- Visualization [13,9].
- Fracture [2,26].
- Surgery modeling [18,20].
- Biomechanics [4].

In terms of objective for the mesh division, we focus here on the required algorithms and a number of applications. It is known that tetrahedron mesh subdivision based on edges or faces generates five distinct members of the polyhedron family: tetrahedra, square pyramids, triangular prisms (both pentahedra) and octahedra. Tetrahedrization of square pyramids and octahedra can be made compatible with neighbour elements[1] for any give face-based criteria. However, triangular prisms can degenerate in the so-called Schönhardt prism, which is non-tetrahedrizable. Some Authors have been inserting nodes inside the original tetrahedron to deal with Schönhardt prisms (cf. [24]). The reason for this ad-hoc procedure is that two tasks are simultaneously being performed: mesh improvement and tetrahedrization. Of course these are equally important and here we address them separately. In terms of tetrahedron division, past work has dealt with two distinct families of methods: edge-based [25] and face-based [22] (Table 1).

Using global numbering (use the maximum node number), it is a simple matter to show that triangular prisms can be made tetrahedrizable, as will be addressed later. Prototype quality of triangles and tetrahedra (inverse relations given by P.L. George [12]) is given by:

* Corresponding author at: Department of Physics, University of Évora, Colégio Luís António Verney, Rua Romão Ramalho, 59, 7002-554 Évora, Portugal.
E-mail address: pmaa@uevora.pt (P. Areias).

[1] In the sense of a face

http://dx.doi.org/10.1016/j.finel.2017.05.001
Received 22 December 2016; Received in revised form 6 May 2017; Accepted 7 May 2017
0168-874X/

Table 1
Case selection as a function of number of marked edges and topology (neighborhood relations).

Number of marked edges	Description	Case	Local relations between local element numbering and case numbering
0	Single case	**#1**	Any order
1	Single case	**#2**	N1 and N2 are on the marked edge
2	A node shares two marked edges	**#3**	N3 does not share a marked edge N4 shares two marked edges
	Otherwise	**#4**	N1-N2 corresponds to the smallest local marked edge
3	A node shares 3 unmarked edges	**#5**	N3 is the corresponding node
	A node shares three marked edges	**#6**	N4 is the corresponding node
	Otherwise	**#7**	N1 and N2 have both two marked edges
4	One node contains 3 marked edges	**#8**	N3 shares one marked edge and N4 shares three
	Otherwise	**#9**	Both N1 and N2 are on the unmarked edge
5	Single case	**#10**	Both N1 and N2 are on the unmarked edge
6	Single case	**#11**	Any order

$$Q_{\text{triangle}} = \frac{\beta_{\text{triangle}} A_{\text{triangle}}}{\sum_{i=1}^{3} l_i^2}, \quad Q_{\text{triangle}} \in [0, 1] \tag{1}$$

$$Q_{\text{tetrahedron}} = \frac{\beta_{\text{tetrahedron}} V_{\text{tetrahedron}}}{\sum_{i=1}^{6} l_i^3}, \quad Q_{\text{tetrahedron}} \in [0, 1] \tag{2}$$

where $\beta_{\text{triangle}} = 4\sqrt{3}$ and $\beta_{\text{tetrahedron}} = 36\sqrt{2}$. In (1)–(2) A_{triangle} is the area of the triangle and $V_{\text{tetrahedron}}$ is the volume of the tetrahedron. The edge lengths are given by l_i with $i = 1, 2, 3$ for the triangle and $i = 1, \ldots, 6$ for the tetrahedron. The evaluation of mesh quality, the corresponding arithmetic average is used. We now discuss the division of tetrahedra based on edges.

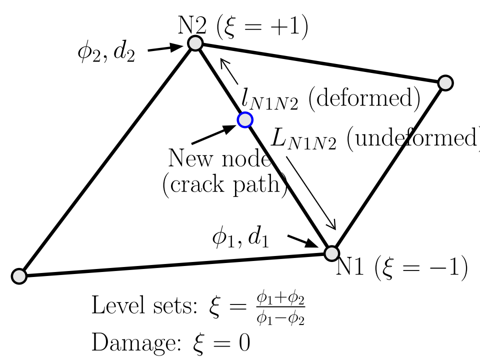

Fig. 1. Edge division based on level set or damage value.

2. Edge-based cutting with pre-ordering

2.1. Marking edges and calculating the crack intersection point

To identify the corresponding edges and mark for splitting, we have several choices, according to the main goal. Using two nodes of a given edge, let them be $N1$ and $N2$, we have a local edge coordinate ξ such as $N1$ corresponds to $\xi = -1$ and $N2$ to $\xi = +1$. If the surface is known from a function $\phi(\boldsymbol{x}) = 0$ such that $\boldsymbol{x}$ are the coordinates of a given point, we use an affine relation to obtain the marked edge and corresponding local coordinate ξ:

$$\xi = \frac{\phi_1 + \phi_2}{\phi_1 - \phi_2} \tag{3}$$

where ϕ_1 and ϕ_2 are the images of ϕ in nodes $N1$ and $N2$. If $\xi \in [-1, 1]$, the corresponding edge is marked. We now introduce two additional strategies for edge marking. In the case of edge length, we use the following criterion for marking:

$$l_{N1N2} \geq \frac{f_m}{2}(l_{\max} + l_{\min}) \tag{4}$$

where l_{N1N2} is the deformed edge length and $l_{\max}$ and $l_{\min}$ are the maximum and minimum *deformed* edge lengths, respectively. In addition, the typical value of the parameter f_m adopted here is 0.75 and is introduced to avoid excessive remeshing. For the damage problem, we have (see Fig. 1):

$$(1 - d)L_{N1N2} + dh_{\min} < f_m l_{N1N2} \tag{5}$$

where d is the indicative damage value at the edge, L_{N1N2} is the *undeformed* edge length and $h_{\min}$ is the minimum edge size, which is considered problem data. In (5), f_m is introduced to incorporate the effect of l_{N1N2} and damage in the same inequality. A typical value of f_m is 1.5 (Table 2).

Table 2
Index sets for tetrahedrization of #PRI, #PYR, #TET and #OCT. #TET indicates a tetrahedron, #PYR a pyramid, #PRI a triangular-base prism and #OCT a octahedron.

Solid: added edges	**Indices**
#PRI: 4-6, 1-3, 1-6	1, 2, 6, 3
	1, 6, 4, 3
	1, 6, 5, 4
#PRI: 4-6, 2-4, 1-6	1, 2, 6, 4
	1, 6, 5, 4
	2, 6, 4, 3
#PRI: 4-6, 2-4, 2-5	1, 2, 5, 4
	2, 3, 6, 4
	2, 6, 5, 4
#PRI: 3-5, 1-3, 1-6	1, 2, 6, 3
	1, 3, 5, 4
	1, 6, 5, 3
#PRI: 3-5, 1-3, 2-5	1, 3, 5, 4
	2, 5, 1, 3
	2, 6, 5, 3
#PRI: 3-5, 2-4, 2-5	1, 2, 5, 4
	2, 3, 5, 4
	2, 6, 5, 3
Solid: added edges	**Indices**
#PYR: 1-3	1, 2, 5, 3
	1, 3, 5, 4
#PYR: 2-4	1, 2, 5, 4
	2, 5, 4, 3
#TET:	1, 2, 3, 4
#OCT: 3-5	1, 3, 4, 5
	1, 3, 5, 6
	3, 4, 5, 2
	3, 5, 6, 2
#OCT: 4-6	1, 4, 6, 3
	1, 6, 4, 5
	2, 4, 6, 5
	2, 6, 4, 3