Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Engineering 00 (2017) 000–000

Procedia Engineering

www.elsevier.com/locate/procedia

Draft - 26th International Meshing Roundtable

Lepp-WCentroid method for tetrahedral mesh improvement

Pedro A. Rodriguez-Moreno[a,*], Maria-Cecilia Rivara[b]

[a] *Universidad del Bio-Bio, Departamento de Sistemas de Información, Av. Collao 1202, Concepción, 4051381, Chile*
[b] *Universidad de Chile, Departamento de Ciencias de la Computación, Avenida Beauchef 851, Santiago, 8370456, Chile*

Abstract

In this paper we propose an improved Lepp-centroid algorithm for the refinement of 3D triangulations which quickly increases the proportion of better tetrahedra in the mesh. This generalizes and improves previous Lepp-bisection algorithm in three-dimensions. We also study the practical behavior of the algorithm. The algorithm is a simple and fast mesh refinement tool for adaptive finite element methods. This has also great potential to be combined with local mesh improvement operations to quickly eliminate bad quality tetrahedra.

Keywords: Longest edge bisection; Lepp centroid; Lepp-bisection; terminal edge midpoint; terminal star centroid.

1. Introduction

In this paper we propose an improved Lepp-WCentroid algorithm for the refinement of 3D triangulations which highly increases the proportion of better tetrahedra in the mesh. The algorithm only performs local refinement operations which maintain conforming meshes throughout the whole refinement process.

It is known that longest edge algorithms for triangulation refinement in 2-dimensions maintain the triangulation quality (smallest angles bounded) while assure that the proportion of better triangles increases as the refinement proceeds [1,4,5]. Even when these properties have not been yet proved in 3-dimensions, practical evidence supports the conjecture that these properties also hold in 3-dimensions. A discussion on a 3D longest edge algorithm and a study on its practical behavior was presented by Rivara and Levin [2]. A different bisection algorithm that assures that a finite number of similar tetrahedra is produced, was discussed (in theory and practice) by Liu and Joe [6], including a comparison with pure 3D longest edge algorithm.

Later, improved and simpler Lepp based algorithms which completely avoid nonconforming issues were developed [3,4,7–9]. In particular, Lepp bisection algorithm in 3-dimensions, is an efficient reformulation of the longest edge algorithm of reference [2]. This performs longest edge bisections of a set of tetrahedra that share a common longest edge (terminal edge). The Lepp searching path (longest edge propagating path) is used to find the terminal edges associated to each processing tetrahedra.

* Corresponding author. Tel.: +56-41-311-1522 ; fax: +56-41-311-1040.
E-mail address: prodrigu@ubiobio.cl (Pedro A. Rodriguez-Moreno)

1877-7058

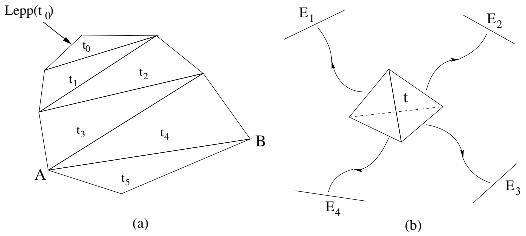


Fig. 1. (a) Lepp in 2-dimensions: Lepp(t_0) = $\{t_0, t_1, t_2, t_3, t_4, t_5\}$, AB is the terminal edge; (b) Lepp(t) in 3-dimensions has several terminal edges E_i.

This paper explores a new algorithm that maintains the simplicity of the Lepp bisection algorithm in 3-dimensions by avoiding the bisection patterns that keep some bad tetrahedra in the mesh. This is achieved by selecting a centroid C_w of the set of tetrahedra that share a common longest edge (terminal edge), which is simply inserted in the mesh as a perturbation of the terminal edge bisection operation. The centroid C_w generalizes the terminal triangles centroid selection used in Lepp Delaunay algorithms in 2-dimensions [4,8].

2. Lepp-WCentroid algorithm in 3-dimensions

2.1. Lepp and terminal edge concepts

In two dimensions, Lepp(t), the longest edge propagating path of a triangle t [4,7], is a sequence of increasing triangles that allows to find a unique local largest edge in the mesh (terminal edge) shared by two terminal triangles (one triangle for a boundary terminal edge). For an illustration see Fig. 1 (a). In 3-dimensions Lepp(t) corresponds to a multidirectional searching process that allows to find a set of terminal edges (see Fig. 1 (b)).

Definition E is a terminal edge in a tetrahedral mesh τ if E is the longest edge of every tetrahedron that shares E. In addition we call terminal star TS(E) to the set of tetrahedra that shares a terminal edge E.

Definition For any tetrahedron t_0 in τ, the Lepp(t_0) is recursively defined as follows:
(a) Lepp(t_0) includes every tetrahedron t that shares the longest edge of t_0 with t, and such that longest edge of t is greater than longest edge of t_0.
(b) For any tetrahedron t_i in Lepp(t_0), this Lepp(t_0) also contains every tetrahedron t that shares the longest edge of t_i and where the longest edge of t is greater than the longest edge of t_i.

Note that Lepp(t_0) is a 3D submesh with a finite and variable number of associated terminal-edges and terminal stars as schematically shown in Fig. 1 (b)

For each tetrahedron t to be refined, the Lepp-bisection algorithm finds Lepp(t) and an associated set W of terminal edges. Then for each terminal edge E in W, the longest edge bisection of every tetrahedron of the terminal star $TS(E)$ is performed, which is a local refinement operation that maintains a conforming mesh. This procedure is repeated until the target tetrahedron t is refined.

The Lepp WCentroid algorithm selects the WCentroid for point insertion defined as follows:

Serial3DLeppWCentroidAlgorithm(τ, S)
Input: τ mesh of tetrahedra; S set of tetrahedra to be refined
Output: refined mesh τ_f
while $S \neq \phi$ **do**
  For each tetrahedron $t_0 \in S$
  while t_0 remains in the mesh **do**
    Compute Lepp(t_0) and set of terminal edges W
    for each terminal edge L in W **do**
      if L is an interior edge **then**

Compute Wcentroid C_w
Refine terminal star TS(L) by joining C_w with the exterior faces of TS(L)
else
Perform longest edge bisection of the tetrahedra in TS(L)
end if
end for
end while
end while

Definition Let assume we have an interior terminal edge E with associated terminal star TS(E) (the set of tetrahedra that share a common longest edge E). Then we define the Wcentroid of TS(E) as the average of the centroids of the tetrahedra that belong to TS(E). Then the following properties hold:

Proposition 1. If E is an interior terminal edge E, then C_w is in the interior of the terminal star TS(E).

Proof. Each C_i is inside a tetrahedron t_i of TS(E) with interior edge E, which implies that the simple average the C_i must belong to TS(E). □

The following property supports the elimination of sliver tetrahedra (flat tetrahedra with good triangular faces situated in the equator of its circunsphere), as well as needle, cup and wedge tetrahedra:

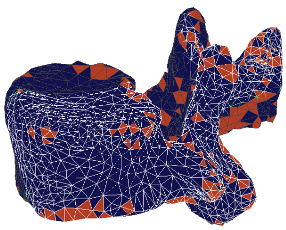
Proposition 2. (a) Given a sliver tetrahedra t_s, with longest edge equal to a terminal edge. Then the simple insertion of C_w in the mesh destroys t_s and produces better tetrahedra. (b) The centroid C_i of any tetrahedra t_i in a TS(E) which is both farther from E and from the tetrahedra faces, contributes most to the C_w computation.

Proof. Part (a) follows from the fact that the centroid of t_s is both close to E and close to the faces of t_s. Consequently the centroids of the other tetrahedra in TS(E) contribute the most to the C_w computation. □

3. Empirical testing

In three-dimensions, we have used the volume quality measure $q(t) = CV(t)/L^3$, where $V(t)$ is the volume of the tetrahedron t, L is the length of the longest edge and C is a constant such that q=*1* for the equilateral tetrahedra which implies $0 \leq q(t) \leq 1$. It is worth noting that every bad quality tetrahedron t has $q(t)$ close to 0 for this measure.

We present results for initial non quality meshes of the angel geometry and the spine geometry, respectively (see Figure 2) [11]. To test the improvement properties of the Lepp centroid algorithm and to compare it with Lepp bisection algorithm, we have performed several improvement iterations as follows: given a quality tolerance parameter Q_{tol}, the first iteration refines all the tetrahedra t with $q(t) < Q_{tol}$. Then the *i-th* iteration refines all the non quality tetrahedra remaining in the mesh *i-1*, for $i > 1$. Figures 3 and 4 allow to visualize the improvement behavior for both algorithms in the 6th iteration for $Q_{tol} = 0.2$.


(a)

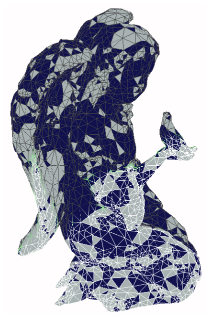

(b)

Fig. 2. Input meshes.

Figures 5 and 6 illustrates the Lepp WCentroid algorithm behavior for different Q_{tol} values for the Angel and Spine geometry, respectively.

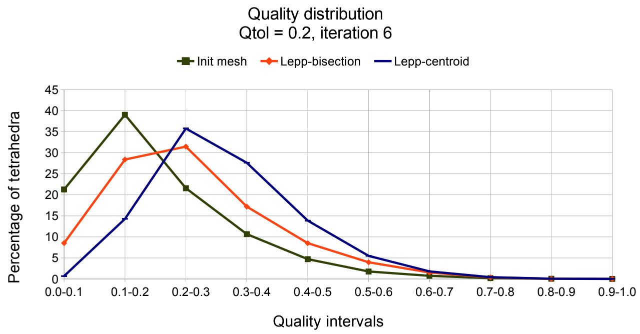


Fig. 3. Angel geometry. Quality distribution comparison: Lepp Wcentroid versus Lepp bisection algorithm.

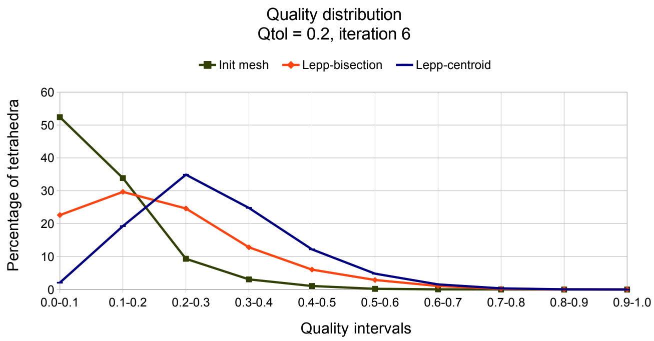


Fig. 4. Spine geometry. Quality distribution comparison: Lepp Wcentroid versus Lepp bisection algorithm.

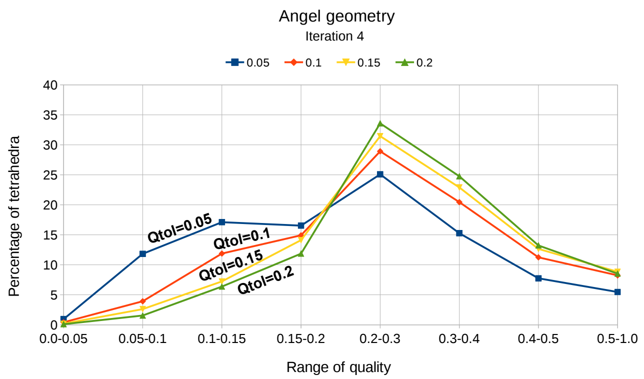


Fig. 5. Angel geometry. Lepp Wcentroid algorithm, quality distribution comparison of iteration 4.

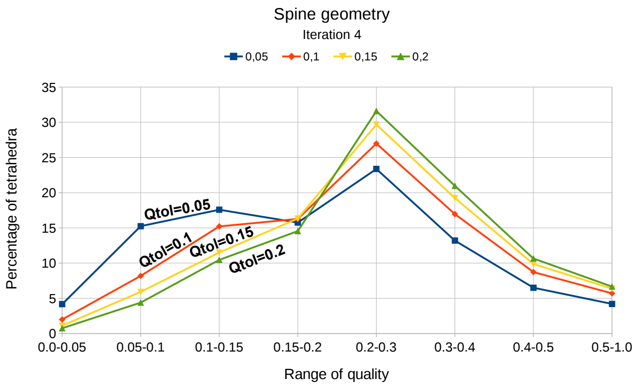


Fig. 6. Spine geometry. Lepp Wcentroid algorithm, quality distribution comparison of iteration 4.

The Lepp Wcentroid algorithm eliminates the bad quality tetrahedra producing smaller sized meshes as compared with Lepp bisection algorithm. This is very appropriate for adaptive multigrid finite element applications [1]. Furthermore the empirical study performed shows that the algorithm has great potential for fast tetrahedral mesh improvement. For bad quality meshes, we suggest to run some iterations with $Q_{tol} = 0.05$ to eliminate the worst quality tetrahedra combined with standard mesh improvement techniques.

Acknowledgements

Work partially supported by Departamento de Ciencias de la Computación, Univ. de Chile, Departamento de Sistemas de Información and Research Group GI150115/EF, Univ. del Bio-Bio. Powered@NLHPC: This research was partially supported by the supercomputing infrastructure of the NLHPC (ECM-02).

References

[1] M.C. Rivara. Algorithms for refining triangular grids suitable for adaptive and multigrid techniques. Int. J. Numer. Methods Eng., 20(4) (1984) 745756.

[2] M.C. Rivara, C. Levin. A 3-d refinement algorithm suitable for adaptive and multi-grid techniques. *Comm. in Appl. Numer. Meth.*, 8(1992) 281–290.

[3] M.C. Rivara, M. Palma, New lepp-algorithms for quality polygon and volume triangulation: Implementation issues and practical behavior, Trends in Unstructured Mesh Generation, AMD 220(1997) 1-8.

[4] M.C. Rivara. New longest-edge algorithms for the refinement and/or improvement of unstructured triangulations. Int. J. Numer. Methods Eng., 40(18) (1997) 3313–3324.

[5] C. Bedregal, M.C. Rivara, Longest-edge algorithms for size-optimal refinement of triangulations, Computer-Aided Des. 46(2014) 246–251.

[6] A. Liu, B. Joe. Quality local refinement of tetrahedral meshes based on bisection. SIAM Journal on Scientific Computing, 16(6) (1995) 1269–1291.

[7] M.C. Rivara. Lepp-bisection algorithms, applications and mathematical properties. Appl. Numer. Math., 59(9) (2009) 2218–2235.

[8] M.C. Rivara, C. Calderon. Lepp terminal centroid method for quality triangulation. Computer-Aided Design, 42(1) (2010) 5866.

[9] P.A. Rodriguez, M.C. Rivara. Multithread lepp-bisection algorithm for tetrahedral meshes. In Josep Sarrate and Matthew Staten, editors, Proceedings of the 22nd International Meshing Roundtable, Springer International Publishing, (2014) 525540.

[10] C. Bedregal, M.C. Rivara, New results on Lepp-Delaunay algorithm for quality triangulations, Procedia Engineering, 124(2015) 317-329.

[11] H. Si, TetGen, A Delaunay-based quality tetrahedral mesh generator, ACM Trans. on Math. Softw. 41(2015) 1–36. http://wias-berlin.de/software/tetgen/