ARTICLE IN PRESS

Available online at www.sciencedirect.com

SCIENCE DIRECT®

ELSEVIER

Journal of Computational and Applied Mathematics ■■■ (■■■) ■■■–■■■

JOURNAL OF COMPUTATIONAL AND APPLIED MATHEMATICS

www.elsevier.com/locate/cam

Recent progress in robust and quality Delaunay mesh generation☆

Qiang Du[a,*], Desheng Wang[b,c]

[a]*Department of Mathematics, Penn State University, University Park, PA 16802, USA*
[b]*Department of Mathematics, University of Xiangtan, Hunan, PRChina*
[c]*Civil and Computational Engineering Centre, School of Engineering, University of Wales Swansea, Singleton Park, Swansea SA2 8PP, UK*

Received 15 August 2004; received in revised form 16 March 2005

Abstract

In this paper, some current issues of Delaunay mesh generation and optimization are addressed, with particular emphasis on the robustness of the meshing procedure and the quality of the resulting mesh. We also report new progress on the robust conforming and constrained boundary recovery in three dimensions, along with the quality mesh generation based on Centroidal Voronoi tessellations. Applications to the numerical solution of differential equations and integrations with other softwares are discussed, including a brief discussion on the joint mesh and solver adaptation strategy.

Keywords: Mesh generation; Delaunay meshes; Robustness and quality; Mesh optimization; Boundary recovery; Centroidal Voronoi tessellations

1. Introduction

Mesh generation often forms a crucial part of the numerical solution procedure in many scientific and engineering problems ranging from flow simulations to structural analysis. The robustness, efficiency

☆ This work was supported in part by US NSF-DMS 0409297, NSF-DMR 0205232 and the China State Key Basic Research Project G199903280. Part of the work was completed while the authors were at the Lab for Scientific and Engineering Computing, Chinese Academy of Sciences, Beijing, China.
* Corresponding author.
E-mail address: qdu@math.psu.edu (Q. Du).

0377-0427/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.cam.2005.07.014

and quality are key issues to be addressed for all meshing procedures. Though subject to different interpretations, it is commonly viewed that the efficiency implies that the underlying meshing procedure can be implemented and completed with a low cost, either being measured in complexity or CPU time estimates. Robustness refers to the fact that the procedure can work for general geometry, can incorporate specified geometric features and handle degeneracy. Quality means the resulting mesh provides a good distribution of elements with nice geometric shapes and obey the sizing control. Naturally, these issues are often dealt with in an integrated fashion in modern meshing methodology.

In recent years, tremendous advances have been made in automatic unstructured mesh generations, in particular, the triangular and tetrahedral mesh generations. The advancing front techniques (AFT) [49,54–58,60,62,64,65], Octree methods [3,36,67,68,72] and Voronoi Delaunay-based methods [8,9,26–28,38,39,51,70,75,76] are some of the well-studied approaches in the unstructured mesh generation. In this paper, we focus on the popular Delaunay-based tetrahedral meshing methods. One of the key issues for such methods in relation to the robustness is the three-dimensional (3D) conforming and constrained boundary recovery [27,28,40,41,52,75,76] as the Delaunay-based methods usually first produce an initial triangulation that forms the convex hull of the boundary points which may not match with the prescribed boundary surface, that is, the triangulation may not satisfy the constraints (edges and faces in 3D) imposed by the surface triangulation. Thus, in real applications, one encounters the problem of recovering the boundary geometric constraints from the initially constructed triangulation, or simply, the problem of boundary recovery. Another important issue we address in this paper is on improving the quality and the functionality of the meshes for a given domain. There are many mesh optimization methods available in the meshing literature, for instance, methods such as geometric and topological optimization, vertex insertion and deletion, and global optimization. All of which may be applied or combined to offer improvements to mesh quality. Of course, the notion of mesh quality itself is an evolving concept and it should be linked to other requirements from real simulation demands.

This paper is not intended to be a survey of the subject but rather a brief account of several techniques for improving the robustness and quality of Delaunay-based meshing which are mostly proposed by the authors in the recent years. Some new procedures for both the conforming and the constrained boundary recovery in 3D spaces are discussed. Mesh optimization techniques based on the concept of centroidal Voronoi tessellation (CVT) are presented. In addition, works on the anisotropic centroidal Voronoi Delaunay triangulations and the joint mesh-solver adaptation strategy for the numerical solution of partial differential equations are mentioned. Numerical examples are provided to demonstrate the effectiveness of the various procedures discussed. Due to page limitations, we are not able to provide a comprehensive list of references. For a more in-depth discussion of the field of Delaunay meshing, we refer to [39,73] and the references cited therein.

2. Unstructured mesh generation methodology

For unstructured triangular or tetrahedral mesh generations, Octree, Delaunay and advancing front are three most popular techniques used in practice.

For a given domain, the Octree method (Quadtree in two-dimensional (2D)) [3,36,67,68,72] utilizes the recursive subdivision of a cube covering the domain via an Octree data structure. The constructed cubic cells consist of regular inner cells and irregular boundary cells which are then meshed into tetrahedral elements. The Octree technique is often used in accelerated geometric searching [73], and it may be combined with Delaunay and other meshing schemes [67]. Its recent applications include meshing of medical images and geometry modeling based on the iso-surface reconstruction [78].

The advancing front technique, first introduced by George [37] and generalized to 3D by Lo [54,55], places vertices in layers from the boundary then into the interior, and tetrahedral elements are then properly constructed.

The modern AFT algorithm is based on the modification by Peraire et al. [64] which allows the simultaneous generation of vertices and elements. The AFT method, due to the high-quality point distribution, is often used in many commercial meshing softwares, though it is less efficient in comparison with the Delaunay meshing. Recent attempts to improve the efficiency and to combine with the Delaunay meshing have been made by George et al. [12,35]. We note that an unresolved issue concerning the AFT technique is the front closeness, though various heuristic approaches, such as elements deletion, formation recording, and trial-and-deletion [56,65], have been proposed.

Based on the concepts of Voronoi tessellations and the dual Delaunay triangulations, Delaunay mesh generations have become the most popular mesh generation methods. Let $\{P_k\}$ be a finite set of points in R^d and for each k, the point set V_k is defined as: $V_k = \{p : \|p - P_k\| \leqslant \|p - P_j\|, j \neq k\}$. V_k is called the Voronoi cell of P_k. The collection of all the Voronoi cells $\{V_k\}$ covers the whole space and it is known as the Voronoi (or Dirichlet) Tessellation of the entire space with respect to the generators $\{P_k\}$. The Delaunay triangulation of $\{P_k\}$ is defined as the dual of the Voronoi tessellation [7,74]. Delaunay triangulation is optimal in many ways due to the fact that the circum-ball associated with each element does not contain any other point of the triangulation except for the degenerate cases.

To construct a Delaunay triangulation with respect to a given set of points, one of the most effective way is the incremental Delaunay insertion method introduced by Hermeline [43] and Watson [74], and studied by Hecht and George [9,41] and others [10,11,51,70,75,76]. To insert a new point into the current triangulation, the Delaunay kernel consists of the construction of three parts: *Base*, *Cavity* and *Ball*.

The usual Delaunay mesh generation starts from a boundary discretization given by a surface triangulation. And an initial Delaunay triangulation is constructed by the above Delaunay insertion procedure, followed by boundary recovery operations [41,76]. Interior points are generated and inserted into the current Delaunay tetrahedral mesh iteratively until the points distribution agrees well with the required sizing. Finally, optimizations can be performed for mesh quality improvement. For anisotropic cases, both in 2D and 3D, the Delaunay insertion and its kernel have been generalized using metric tensor for directional sizing control [9,38]. By some accounts, the almost linear efficiency of the Delaunay mesh generation in practice is a main advantage over the Octree and advancing front methods [8,10,11]. This is largely due to the local insertion procedure and the fast searching techniques [10,11]. On the other hand, robust boundary recovery and Delaunay insertion procedures play key roles in Delaunay meshing methods [40,71,75,76].

3. Robust Delaunay mesh generation

3.1. Robust Delaunay insertion

Given a point P to be inserted into an existing Delaunay mesh T, the classical Delaunay insertion procedure starts with the construction of the Base(P) which includes all tetrahedra containing P; and the Cavity(P) is obtained by enlarging the Base(P) with those elements whose circum-spheres containing

P; then all the interior edges and faces of the Cavity(P) are removed, and the boundary triangles of Cavity(P) and P form a new set of tetrahedral elements called the Ball(P). The Delaunay mesh may then be updated by $T_{\text{new}} = T - \text{Cavity}(P) + \text{Ball}(P)$. The robustness of this Delaunay insertion method mainly depends on the validity of the Cavity, i.e., the star-shapedness. In 3D, this turns to be very sensitive to round-off errors [10,11]. To assure the validity of the Cavity, various techniques were developed, such as the exact geometrical computation in normalized integers, tolerance specification and small perturbation of problematic points [10]. In [11], a correction procedure was proposed for the Cavity construction to guarantee the Cavity being star-shaped and it was also generalized to anisotropic cases [9,38].

3.2. *Robust three-dimensional boundary recovery*

For a given 3D domain, the input data of the Delaunay mesh generation procedure are often given by a surface triangulation of the boundary. Delaunay-based methods usually first produce an initial triangulation that forms the convex hull of the boundary vertices which may not always match with the prescribed boundary surface. This leads to the problem of recovering the boundary constraints from the initially constructed triangulation, or simply, the problem of boundary recovery. A robust boundary recovery is a necessary ingredient of a robust Delaunay meshing process. While such a problem has been successfully resolved in 2D spaces [8,75], it is still under active investigation in three dimension.

Roughly speaking, there are two types of 3D boundary recovery procedures. The first is the *conforming boundary recovery*, which applies edge/face splitting to recover a constraint as the concatenation of edges/faces. It usually requires the insertion of points to the missing constraints [27,46,52,71,76,77]. The second approach is the *constrained boundary recovery*, which does not allow extra points being added to the missing constraints during the recovery, and offers more robustness than the conforming recovery especially in mesh merging. Note that the resulting tetrahedral mesh after boundary recovery may not be strictly Delaunay. Moreover, due to the Schrondert configuration [40,41], it is well known that the success of constrained boundary recovery often relies on the insertion of interior Steiner points [28,40,41].

3.2.1. *Robust conforming boundary recovery*

For conforming boundary recovery, various traditional approaches share a common characteristics: adding points on a missing constraint (edge/face) to reconstruct the missing edge or face as a union of sub-edges or sub-faces [46,76,77]. Schewchuk [70] and Shephard [46] proposed Delaunay refinement methods to construct a triangulation conforming with the surface geometry by using local mesh modifications such as edge/face splitting to recover a constraint as the concatenation of edges/faces, while keeping the Delaunay property. Though the effectiveness of such conforming boundary recovery methods has been demonstrated in many cases, no theoretical proof is provided for their convergence. When adding a point to a constraint in the Delaunay method, some recovered constraints may be deleted in the refinement processes, causing redundancy. The refinement steps may also require the insertion of excessive number of points to the missing constraints, hence violating the local sizing specification (prescribed by the surface triangulation or by other methods).

In Du and Wang [27], an algorithm for conforming boundary recovery was presented and its convergence was rigorously proven. The method involves two stages, with the first stage consisting of three basic single-step local edge/face swaps which are able to recover a large portion of missing constraints. During this stage, the tetrahedra set connecting the missing item is compared with the configurations of the three basic swaps. Once a match is found, the corresponding swapping is performed to recover the

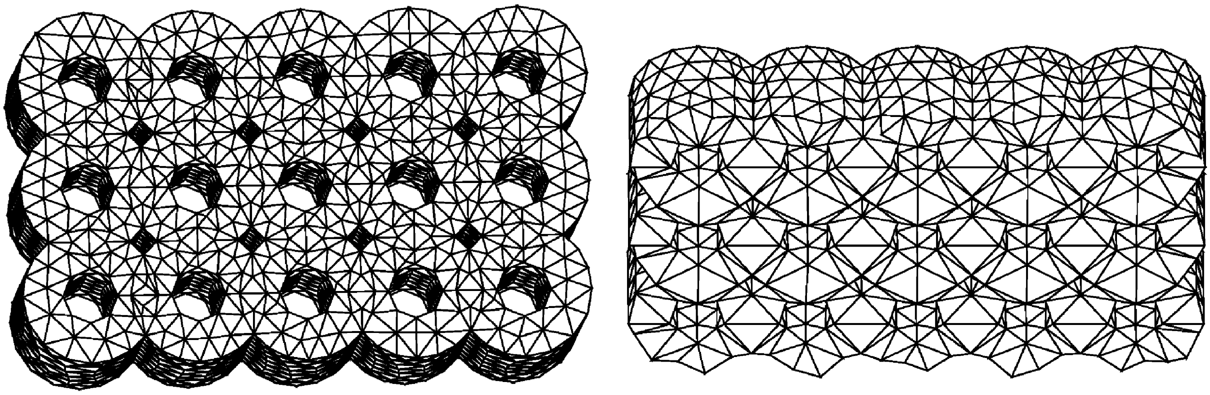

Fig. 1. Conforming boundary recovery of Delaunay meshing example: (left) the surface triangulation; (right) cutting view of the tetrahedral mesh.

missing item. In the second stage, a refinement method is used for the remaining missing items. Different from the works in [46,77] where one first recovers edges and then faces, the approach in [27] leads to, one by one, the simultaneous recovery of a missing face and its missing edges. For each missing edge of a missing face, intersection points of the edge with the initial triangulation of the boundary points are first located and then the nearest mid-intersection-points are added sequentially using a modified Delaunay insertion procedure until the recovery of the edge is achieved. When the missing edges of a missing face are all recovered, if the face is still missing, the intersection points of the face with the initial triangulation are then determined and they are added one by one by the modified Delaunay insertion procedure until the face is recovered.

The modified Delaunay insertion in [27] enjoys the following crucial property: when inserting a point to a constraint, no existing or recovered constraint is deleted. The use of local transformations in the first stage of the algorithm also greatly reduces the need for points insertion in the second stage, thus our method makes the boundary recovery simple and the mesh in tune with the local sizing specification. The idea of protecting recovered constraints has also been addressed by Wright and Jack [77], but the protection method used there only becomes viable with a consistent node ordering of the faces of the *inserted polyhedra*, or Cavity. The method proposed systematically in [27] via the modification of the Delaunay kernel is applicable for all cases and can also be easily implemented. Since the number of missing constraints are finite, the convergence of the boundary recovery procedure naturally follows. Moreover, it is shown to be very effective through many numerical examples, see Fig. 1 for a demonstration [27].

3.2.2. Robust constrained boundary recovery

For constrained boundary recovery of 3D Delaunay triangulation, George et al. [41] introduced ingenious techniques to re-establish the surface edges and faces through a series of edge/face swaps (or flips), in addition to some heuristic insertions of interior points (Steiner points). However, several examples in Baida [52] provided evidence that this heuristic method may fail in some situations. Later, Weatherill and Hassan in [76] suggested some possible remedy but their approach hinges on tackling the closeness problem of AFT. For a robust constrained boundary recovery, several issues have to be resolved, such as

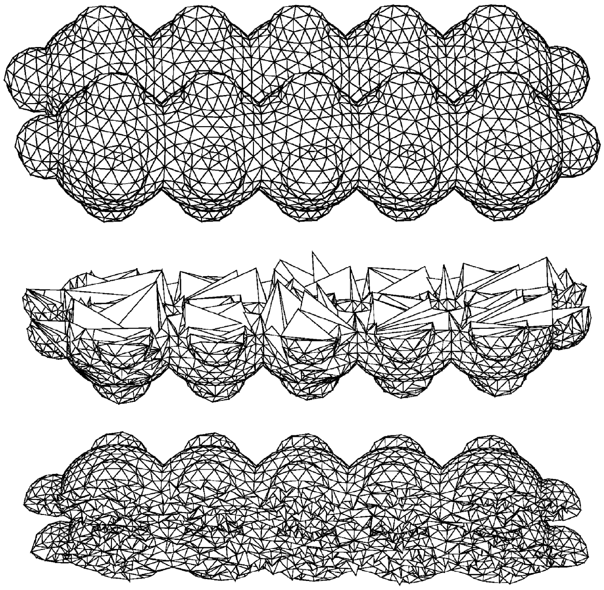

Fig. 2. A Delaunay mesh for molecular modeling: (top) the surface triangulation; (center) cutting view of the boundary Delaunay tetrahedral mesh; (bottom) cutting view of the final Delaunay tetrahedral mesh.

the placement of Steiner points and minimizing the number of added points [41]. In [28], we presented a new constrained boundary recovery algorithm which combines conforming boundary recovery with the splitting of inserted points to recover a missing constraint in a constrained manner. The convergence of the method was theoretically proved. Our approach is different from the previous works [41,76]. For each missing face, we apply the conforming recovery proposed in [27] by adding points to the missing edges or the interior of the face when necessary, followed by the splitting of each added point into two interior Steiner points located away from the missing face. The splittings are performed sequentially, producing some basic swappable configurations which can be used for a direct or gradual recovery of the missing edge or face in the constrained manner. The process is done by first splitting the points added to each missing edge of the missing face, one by one, into two interior Steiner points located away from the missing face until a complete constrained recovery of the missing edge. Once all the missing edges of the face are recovered, we then apply again splitting operations to the added points on the face sequentially until the recovery of the missing face. Such a splitting operation includes a directional perturbation and a constrained Delaunay insertion of a face-symmetric point. The cavity is appropriately chosen so that a basic flippable local tetrahedra set or configuration can be generated. For a missing edge, by flipping the local tetrahedra set, the edge or a part of it is recovered in a constrained manner; for a missing face with already recovered edges, the number of added points are gradually reduced until the complete recovery of the face. When a missing face is recovered in a constrained manner, for each point added in the initial conforming recovery procedure, there are two interior Steiner points positioned on the two sides of the recovered face. To complete the mesh generation process, we can either perform vertex suppression to delete these added points if such deletions are allowed or suitably reposition them with respect to the given sizing field. Figs. 2 and 3 are typical applications of the constrained recovery. More examples can be found in [28].

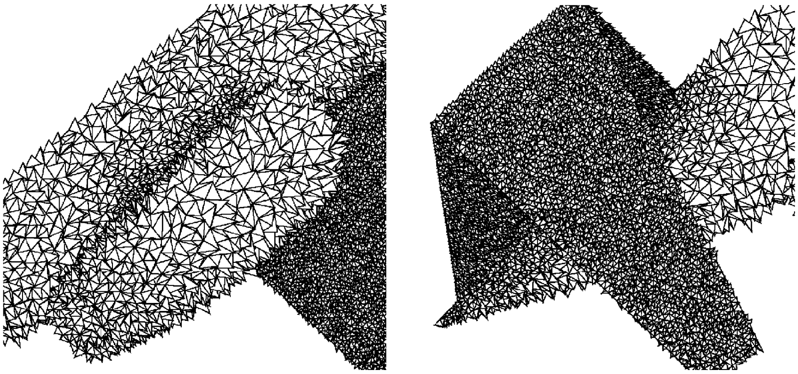

Fig. 3. Cutting view of the final Delaunay tetrahedral mesh of the Gulf2 model: (left) area around the engine and (right) area around a wing.

We note that in [40], George et al. also proposed similar ideas of addition-and-deletion, but their procedure requires post-processing via optimization. Though the mesh validity remains to be verified when an added point inside the domain is taken out, the proposed approach were numerically shown to mesh very pathological configurations with success [40].

4. Quality Delaunay meshing based on the CVTs

Due to the existence of the notorious Slivers, a Delaunay mesh often requires further improvement and optimization. Traditional approaches for the unstructured mesh optimization often fall into the following basic categories [1,6,13,14,34,44,45,47,61]: *geometric optimization*, meaning mesh smoothing or vertices relocation without changing the node connectivity, through strategies such as the Laplacian smoothing and its variants; *topological optimization*, consisting of local reconnections such as edges/faces flipping, while keeping node positions unchanged; and *vertex insertion or deletion*, referring to operations such as the sink insertion [33,51]. These techniques are often combined and performed in an iterative manner, and they form the core of the *classical optimization methods*. There have also been some studies on global optimization approaches, such as Winslow transforms, harmonic mappings and optimizing algebraic or geometric mesh quality measures [47].

Recent studies on the centroidal Voronoi tessellation (CVT) [19,20,22,26,32,29] have shown that CVTs often provide optimal point distributions, thus making CVT based mesh generation and optimization techniques very effective. Given a density function ρ defined on a region V, the *mass centroid* $\mathbf{z}^*$ of V is defined by

$$\mathbf{z}^* = \frac{\int_v \mathbf{y}\rho(\mathbf{y})\,\mathrm{d}\mathbf{y}}{\int_v \rho(\mathbf{y})\,\mathrm{d}\mathbf{y}}.$$

For a given set of points $\{\mathbf{z}_i\}_{i=1}^k$ in the domain Ω and a positive density function ρ defined on Ω, a Voronoi tessellation is a centroidal Voronoi tessellation (CVT) if $\mathbf{z}_i = \mathbf{z}_i^*$, $i = 1, \ldots, k$, i.e., the generators of the Voronoi regions are themselves the mass centroids of those regions. The dual Delaunay triangulation is called the centroidal Voronoi–Delaunay triangulation (CVDT) which often yields high-quality Delaunay meshes [20,26,32]. For any tessellation $\{V_i\}_{i=1}^k$ of the domain Ω and a set of points $\{z_i\}_{i=1}^k$ (independent of $\{V_i\}_{i=1}^k$) in Ω, we define the following *cost* (or *error* or *energy*) functional:

$$F(\{V_i\}_{i=1}^k, \{z_i\}_{i=1}^k) = \sum_{i=1}^k \int_{V_i} \rho(x)\|x - z_i\|^2 \, \mathrm{d}x.$$

The standard CVTs along with their generators are critical points of the cost functional.

The concept of CVT has been applied to mesh generation and optimization in isotropic 2D and 3D unstructured meshing [20,26,31], and it is also generalized to anisotropic and surface quality mesh generation [32]. Using the notion of the cost functional, both constrained CVT (CCVT) and its dual CCVDT have also been studied [20,26]. In [32], CVT has been generalized to the anisotropic case with a Riemannian metric and a one-sided distance.

The numerical construction of CVT and CVDT can be performed via either probabilistic or deterministic methods [19,53]. For studies on the probabilistic methods as well as their parallelization, we refer to [21,59]. Here, we apply a deterministic algorithm based on the popular Lloyd's method [18,19,53] which is an obvious iteration between constructing Voronoi tessellations and centroids. And it enjoys the property that the functional F is monotonically decreasing throughout the iteration. Improvements using multilevel ideas and linearization schemes for the CVT constructions have been recently studied in [16,17].

4.1. Application to quality mesh generation

The construction of CVDT (or CCVDT) through the Lloyd iteration can be also viewed as a smoothing process of an initial mesh. The CVDT concept provides a good theoretical explanation to its effectiveness: by successively moving generators to the mass centers (of the Voronoi regions), the cost functional is reduced. Here, *smoothing* means both the node-movement and the node reconnection. If the density function can be chosen according to the sizing function, the cost functional may be related to the distortion of the mesh shape and quality with respect to the mesh sizing. Thus, the process of iteratively constructing CVDTs, like the Lloyd's algorithm, contributes to the reduction of the global distortion of element shape and sizing. The final CVDT would have a minimal distortion, and hence shares good element quality with respect to the sizing distribution [20,26].

A practically useful property of the CVT and CVDT is the local equi-distribution of error (cost) [19,26]. It is not difficult to show that in the one dimensional case, there is a constant $c > 0$ such that $\int_{V_i} \rho(x)(x - x_i)^2 \, \mathrm{d}x \approx c$ for all i when the number of generators goes to infinity. This means, asymptotically speaking, the cost is equally distributed in the Voronoi intervals [19]. For the multidimensional CVT, the Gersho conjecture [42] predicts that asymptotically, as the number of generators becomes large, all Voronoi regions are approximately congruent to the same basic cell that only depends on the dimension, this in turn implies the local cost equi-distribution principle. The basic cell has been shown to be the regular hexagon in two dimensions [63], with the dual cell be the regular triangle, thus explaining why the CVDTs in 2D tend to provide high quality meshes. The conjecture remains open in three and higher dimensions [5,42] and recently in [31], more numerical evidence was provided to substantiate the claim that the basic 3D cell is the predicted truncated dodecahedron. It is thus practically prudent to take the advantage of the equi-distribution of the cost functional. If the cost can be related implicitly to the distortion of the elements quality [26], the equi-distribution principle can then be interpreted as the equi-distribution of the distortion of the elements quality. Hence, asymptotically, almost uniform triangulation/tetrahedralization can be generated. This idea has been applied to quality isotropic 2D and 3D mesh generation and optimization [20,26], and such an assertion is indeed numerically supported by our various meshing examples there. Successful generalizations to anisotropic and surface grid generation were made in [32].

Given a bounded domain and a prescribed element sizing, suppose a constrained boundary Delaunay triangulation/tetrahedralization of the domain with respect to the sizing has been generated and stored [26,32], the Lloyd iteration, interpreted as a natural optimization of an existing mesh, can be briefly described as follows:

Algorithm 4.1 (*The Lloyd iteration*). *Given a set of vertices in* R^d. (1) *Construct the Voronoi region for each of the interior points that are allowed to change their positions, and construct the mass center of the Voronoi region with a properly defined density function* $\rho(p)$ *derived from the sizing field* $H(p)$ ($\rho(p) = 1/H(p)^{2+d}$ *up to a constant scaling and here d is the space dimension number*).

(2) *Insert the computed mass centers into the constrained boundary Delaunay triangulation* (*tetrahedralization*) *through a constrained Delaunay insertion procedure* [10,11].

(3) *Compute the difference* $D = \sum_{i=1}^{k} \| P_i - P_{\text{imc}} \|^2$, $\{P_i\}$ *is the set of interior points allowed to change*, $\{P_{\text{imc}}\}$ *is the set of corresponding computed mass center*. *If D is less than a given tolerance*, *terminate*; *otherwise*, *return to step* 1.

The effectiveness of the above procedure has been demonstrated in a recent work [29].

4.1.1. Application to 2D meshing

In Du and Gunzburger [20], the concept of CVT was first applied 2D triangular grid generation and optimization. Numerical examples show that high-quality meshes can be constructed based on CVDTs. Also, the numerical solution of PDE on the CVDT was shown to provide higher accuracy than others [20], see also [25] for a related theoretical proof. Though the discussion is only preliminary, it opened the door of the applications of CVT to mesh generation and optimization. Our recent investigations on the effect of CVT-based optimization to 2D examples [29] demonstrate that the CVT-based mesh optimization is much more effective than the classical method consisting of edges Delaunay swapping and Laplacian smoothing. Also the final result is less sensitive to the initial points distribution and mesh topology than the classical counterpart. In Fig. 4, an almost equilateral triangular mesh is constructed via the Lloyd iteration from two totally different initial Delaunay meshes with bad qualities.

4.1.2. Application to 3D meshing

In [26], the centroidal Voronoi tessellation was applied to generate quality constrained Delaunay tetrahedral meshes from an initial Delaunay tetrahedral mesh of a 3D domain. A surface triangular mesh is taken as the input. Conforming boundary tetrahedralization, which includes the Delaunay triangulation of the boundary vertices and the boundary recovery described before, is first performed, followed by interior refinement through points generation and Delaunay insertion. The construction of the 3D constrained CVDT is then carried out via the application of the Lloyd iteration: the Voronoi regions of the interior vertices are computed from the Delaunay tetrahedralization and the mass centers of these Voronoi regions are computed; then these mass centers are inserted into the stored boundary Delaunay mesh to replace the original generators. If any generator is close to the boundary, a projection or merging technique is applied. The resulting converged mesh gives the constrained CVDT which is in better harmony with the specified

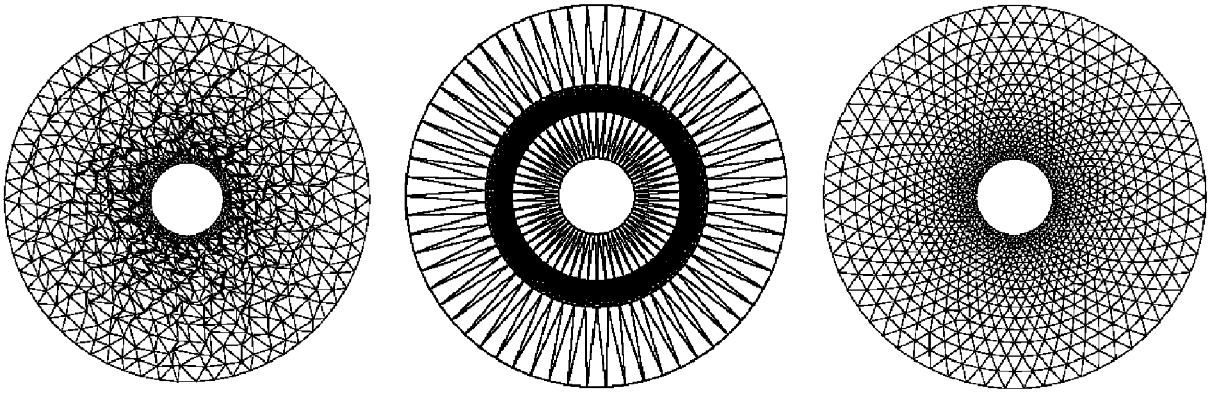

Fig. 4. Quality two-dimensional CVDT constructed from two different initial Delaunay Meshes: (left) with random perturbation, (center) with points clustered and (right) final CVDT.

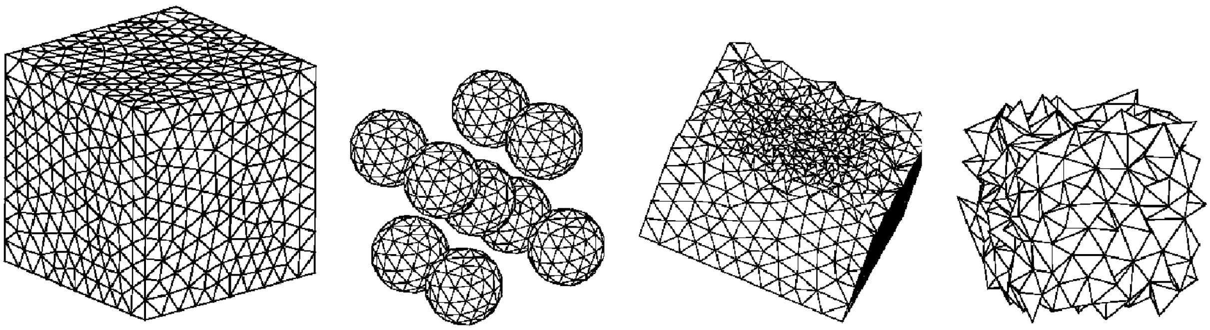

Fig. 5. Quality CVDT for a composite material simulation with inclusions. (Left pair) exterior surface triangulation of the unit box and the eight inclusions; (right pair) cutting views of the final CVDT.

sizing field. The mesh vertices and the overall mesh structure are both optimized. And almost all the slivers existing in the initial mesh are removed after the iteration which results in a dramatic enhancement of the mesh quality. For further improvement, simple local edges/faces flippings are performed to kill the remaining bad-shaped elements. Various numerical examples in [26] demonstrate that the proposed method is effective in quality tetrahedral mesh generation. Fig. 5 shows a high quality CVDT constructed for a composite material simulation with several inclusions.

The optimization effect can be partially understood through recent numerical studies on the Gersho's conjecture in three dimensions [31]. In a related work [29], the effect of CVT-based tetrahedral mesh optimization was also investigated through numerical examples along with comparisons with classical tetrahedral mesh optimization techniques and cogent arguments were made on viewing the CVT-based tetrahedral mesh optimization as a preferred choice for 3D tetrahedral mesh generation and optimization. The CVT-based optimized mesh enjoys higher quality, more structured topology and it is less sensitive to the initial mesh configuration. Fig. 6 shows the more-structured and higher-quality CVT-based tetrahedral mesh (right) for a femur head and the mesh optimized via a classical technique (left).

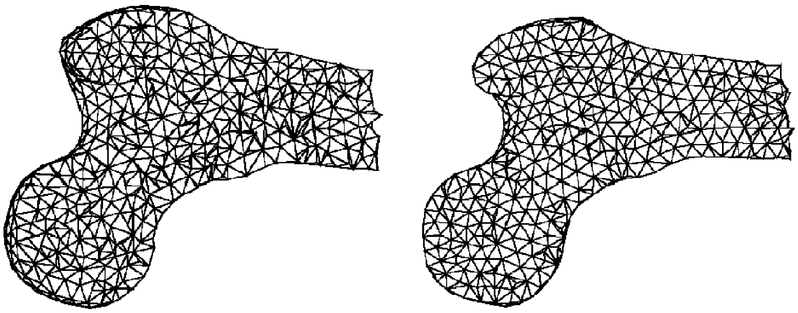

Fig. 6. Tetrahedral mesh generation for femur fracture simulation: (left) with classical optimization (right) with CVDT based optimization.

4.1.3. Application to anisotropic and surface meshing

The basic definition of the CVT can be extended to very broad settings ranging from abstract spaces to discrete point sets [19]. In [32], it was generalized to anisotropic cases by introducing a new and consistent definition for anisotropic centroidal Voronoi tessellations in the Euclidean space but related to a given Riemannian metric tensor which possesses anisotropy. By introducing a directional distance definition for any two points as a significant simplification of the classical Riemannian distance measure, the notion of anisotropic Voronoi region (AVR), anisotropic Voronoi tessellation (AVT) and the corresponding anisotropic Delaunay triangulation (ADT) can be suitably defined. Their definitions are different from the standard ones in [19] and they also differ from other popular definitions used in the literature [48,50]. Our approach leads to a straightforward definition of the mass centroids, thus providing a consistent definition of the anisotropic centroidal Voronoi tessellations (ACVTs). The ACVTs enjoy useful optimization properties that are naturally tied to the basic function approximation theory, and they reduce to the standard CVTs for isotropic Riemannian tensors. Applications and generalizations of the ACVTs have been given in [30] in relation to vector field simplication and representations.

When applied to surface tessellation and triangulation, our definition is also different from the notion of constrained CVTs discussed in [23] where the distance remains to be measured in the Euclidean metric and only the definition of the mass centroids reflects the surface geometry.

Even with the simplified notion of directional distances, the direct construction of anisotropic Voronoi tessellation is still computationally challenging due to the generality and the complexity of the Riemannian metric. In [32], the method of unit meshing proposed in [9] was used to provide an approximate construction of the ADT and subsequently the AVT. A key observation based on the computational experience is that the AVTs can often be well approximately by their visibility regions. We also extended the classical Lloyd method to compute the ACVT. The proposed algorithm in [32] was shown to be very effective through various examples.

A direct application of ACVT is the 2D ADT via the optimal anisotropic centroidal Voronoi–Delaunay triangulation (ACVDT). Under the Riemannian metric, the density function of ACVT is defined to be unit, which means that, asymptotically speaking, the dual triangulation ACVDT of the final converged ACVT have approximately unit-length edges, and accordingly, the triangulation is an almost regular anisotropic triangulation under the Riemannian metric. This is similar to our previous works in isotropic meshing [20,26]. Numerous anisotropic examples vindicate the above assertion. Another direct application of

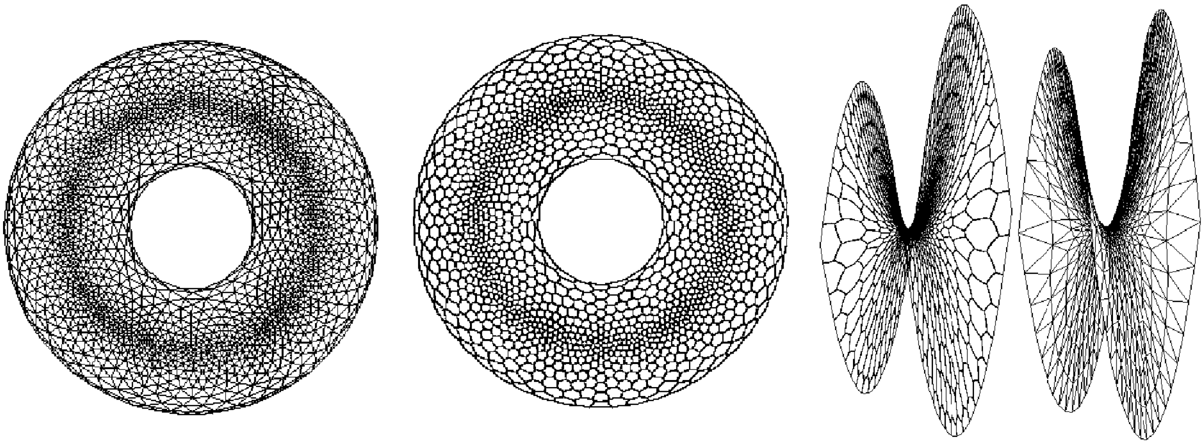

Fig. 7. Applications to quality surface mesh generation: (left pair) torus, (right pair) saddle.

the anisotropic CVT constructed through the Lloyd iteration is the effective surface tessellation and triangulation. For demonstration, we recall two examples in Fig. 7 for the applications of anisotropic CVT to quality anisotropic CVDT generation and quality surface mesh generation [32].

5. Software integration: the mesh-solver co-adaptation

The performance of finite element methods usually depends on the integration of many software components such as mesh generation and iterative solvers. The investigation on whether a given mesh can appropriately represent the numerical solution has been carried on throughout the historical development of finite element methods [2,4,6,15,66,69]. To assess mesh quality, a guiding principle has been that both the element shapes and sizes and the local solution behavior are to be related [15,66]. For a in-depth discussion on the relation between the mesh and the interpolation and approximation errors as well as the conditioning of local stiffness matrices, we refer to a recent report [69].

In general, finite element discretizations reduce the problem of solving a partial differential equation to the problem of solving a large, sparse system of algebraic equations, in particular, linear algebraic equations. The properties of finite element meshes thus affect both the accuracy of the numerical solutions as well the efficiency for obtaining them. In [24], we advocated a simple principle: the grid generation and optimization should not only be based on the behavior of the solution, but should also be related to the (linear) solver(s) used for finding the numerical solutions, and vice versa. Such a principle was coined as a joint mesh-solver adaptation strategy.

As an illustration, we considered in [24] a particular case where the numerical solutions of certain PDEs display anisotropic behavior and anisotropic grids are thus used. The model PDE problems in [24] are limited to second order linear elliptic equations in a simple 2D domain, but they are good representatives of more complicated practical problems. By considering different types of triangulation, and taking algebraic multigrid methods and conjugate gradient type methods as the linear solvers, we observed that the elimination of badly shaped element for the sake of improving the efficiency of the linear solver is often practically an even more pressing issue than generating suitable grids with good representations of the solution for the sake of achieving high accuracy in the approximation error. We discussed how to reconcile the need for a high-resolution scheme using anisotropic meshes and the need for efficient linear solvers for the resulting linear systems, and proposed modifications to both the meshing techniques as well as the iterative algorithms so that they can be co-adapted together to provide higher efficiency for the numerical solution on anisotropic finite element grids. Even though the examples in [24] are only in 2D, the principle of mesh-solver co-adaptation strategy would no doubt be useful also in 3D.

6. Conclusions and future work

Besides the issues we have discussed above, there are obviously many interesting problems to be studied in the future, for instance, the anisotropic 3D CVT and mesh generation, the mesh and solver joint adaptation in an anisotropic environment, and the generation of almost regular mesh or mesh suitable for special discretization schemes. We stress again that this paper simply provides a brief account to some of the works we have carried out in the last few years. Our discussion on the Delaunay mesh generation, in particular on their robustness and quality issues and their applications, is still limited. Issues related to the efficiency and simplicity in coding also require further consideration. We anticipate that future works will bring more progress on the fundamental algorithmic development, efficient and robust numerical code implementation and the construction of friendly user interface.

References

[1] N. Amenta, M. Bern, D. Eppstein, Optimal point placement for mesh smoothing, J. Algorithms 30 (1999) 302–322.
[2] E. D'Azevedo, B. Simpson, On optimal triangular meshes for minimizing the gradient error, Numer. Math. 59 (1991) 321–348.
[3] P. Baehmann, L. Scott, L. Wittchen, M. Shephard, K. Grice, M. Yerry, Robust geometrically based, automatic two-dimensional mesh generation, Internat. J. Numer. Methods Engrg. 24 (1987) 1043–1078.
[4] R. Bank, Mesh smoothing using a posteriori estimates, SIAM J. Numer. Anal. 34 (1997) 979–997.
[5] E. Barnes, N. Sloane, The optimal Lattice quantizer in three dimensions, SIAM J. Algebraic Discrete Methods 4 (1983) 30–41.
[6] M. Berzins, A solution-based triangular and tetrahedral mesh quality indicator, SIAM J. Sci. Comput. 19 (1998) 2051–2069.
[7] A. Bowyer, Computing Dirichlet tessellations, Comput. J. 24 (1981) 162–166.
[8] H. Borouchaki, P. George, Aspects of 2-D Delaunay mesh generation, Internat. J. Numer. Methods Engrg. 40 (1997) 1957–1975.
[9] H. Borouchaki, P. George, F. Hecht, P. Laug, E. Saltel, Delaunay mesh generation governed by metric specifications. Part I, Algorithms 25 (1997) 61–83.
[10] H. Borouchaki, P. George, S. Lo, Optimal Delaunay point insertion, Internat. J. Numer. Methods Engrg. 39 (1996) 3407–3437.
[11] H. Borouchaki, S. Lo, Fast Delaunay triangulation in three dimensions, Comput. J. Numer. Methods. Engrg. 128 (1995) 153–167.
[12] H. Borouchaki, L. Patrick, P. George, Parametric surface meshing using a combined advancing-front generalized Delaunay approach, Internat. J. Numer. Methods Engrg. 49 (2000) 233–259.
[13] G. Buscaglia, E. Dari, Anisotropic mesh optimization and its application in adaptivity, Internat. J. Numer. Methods Engrg. 40 (1997) 4119–4136.
[14] L. Chew, Guaranteed quality mesh generation for curved surfaces, in: Proceedings of the Ninth Annual Computer Geometry, 1993, pp. 274–280.
[15] P. Ciarlet, Finite Element Methods, North-Holland, Amsterdam, 1975.

[16] Q. Du, M. Emelianenko, Uniform convergence of a nonlinear energy-based multilevel quantization scheme, preprint, 2005.
[17] Q. Du, M. Emelianenko, Acceleration schemes for computing centroidal Voronoi tessellations, preprint, 2005
[18] Q. Du, M. Emelianenko, L. Ju, Convergence properties of the Lloyd algorithm for computing the centroidal Voronoi tessellations, SIAM J. Numerical Analysis, 2005, accepted for publication.
[19] Q. Du, V. Faber, M. Gunzburger, Centroidal Voronoi tessellations: applications and algorithms, SIAM Rev. 41 (1999) 637–676.
[20] Q. Du, M. Gunzburger, Grid generation and optimization based on centroidal Voronoi tessellations, Appl. Comput. Math. 133 (2002) 591–607.
[21] Q. Du, M. Gunzburger, L. Ju, Probablistic methods for centroidal Voronoi tessellations and their parallel implementations, J. Parallel Comput. 28 (2002) 1477–1500.
[22] Q. Du, M. Gunzburger, L. Ju, Meshfree, probabilistic determination of point sets and support regions for meshless computing, Comput. Methods Appl. Mech. Engrg. 191 (2002) 1349–1366.
[23] Q. Du, M. Gunzburger, L. Ju, Constrained centroidal Voronoi tessellations on general surfaces, SIAM J. Sci. Comput. 24 (5) (2003) 1499–1506.
[24] Q. Du, Z. Huang, D. Wang, Mesh and solver coadaptation in finite element methods for anisotropic problems, Numer. Methods for Differential Equations 21 (2005) 859–874.
[25] Q. Du, L. Ju, Finite volume methods and spherical centroidal Voronoi tessellations, SIAM J. Numer. Anal., accepted for publication.
[26] Q. Du, D. Wang, Tetrahedral mesh generation and optimization based on centroidal Voronoi tessellations, Internat. J. Numer. Methods Engrg. 56 (2003) 1355–1373.
[27] Q. Du, D. Wang, Boundary recovery for three dimensional conforming Delaunay triangulation, Comput. Methods Appl. Mech. Engrg. 193 (2004) 2547–2563.
[28] Q. Du, D. Wang, Constrained boundary recovery for three dimensional Delaunay triangulation, Internat. J. Numer. Methods Engrg. 61 (2004) 1471–1500.
[29] Q. Du, D. Wang, Mesh optimization based on centroidal Voronoi tessellation, Internat. J. Numer. Anal. Modelling, 2004, accepted for publication.
[30] Q. Du, X. Wang, Centroidal Voronoi tessellation based algorithms for vector fields visualization and segmentation, IEEE Proceedings of Visualization 2004 (VIS2004), Austin, TX, 2004, pp. 43–50.
[31] Q. Du, D. Wang, On the optimal centroidal Voronoi tessellations and the Gersho's conjecture in the three dimensional space, Comput. Math. Appl. 49 (2005) 1355–1373.
[32] Q. Du, D. Wang, Anisotropic centroidal Voronoi tessellations and their applications, SIAM J. Sci. Comput. 26 (2005) 737–761.
[33] H. Edelsbrunner, D. Guoy, Sink insertion for mesh improvement, Internat. J. Foundations Comput. Sci. 13 (2002) 223–242.
[34] P. Frey, H. Borouchaki, Geometric surface mesh optimization, Comput. Visualization Sci. 1 (1998) 113–121.
[35] P. Frey, H. Borouchaki, P. George, 3D Delaunay mesh generation coupled with an advancing-front approach, Comput. Methods Appl. Mech. Engrg. 157 (1998) 115–131.
[36] P. Frey, L. Marechal, Fast adaptive quadtree mesh generation, in: Proceedings of the Seventh International Meshing Roundtable, 1998, pp. 211–222.
[37] A. George, Computer implementation of the finite element method, Ph.D. thesis, Standford University, STAN-CS-71-208, 1971.
[38] P. George, Gamanic3d, adaptive anisotropic tetrahedral mesh generator, Technical Report, INRIA, 2002.
[39] P. George, H. Borouchaki, Delaunay triangulation and meshing, Application to Finite Elements Methods, Hermès, Paris, 1998.
[40] P. George, H. Borouchaki, E. Saltel, Ultimate robustness in meshing an arbitrary polyhedron, Internat. J. Numer. Methods Engrg. 58 (2003) 1061–1089.
[41] P. George, F. Hecht, E. Saltel, Automatic mesh generation with specified boundary, Comput. Methods Appl. Mech. Engrg. 92 (1991) 169–188.
[42] A. Gersho, Asymptotically optimal block quantization, IEEE Trans. Inform. Theory 25 (1979) 373–380.
[43] F. Hermeline, Triangulation automatique d'un polyèdre en dimension, N.R.A.I.R.O. 16 (1982) 211–242.
[44] N. Hitschfeld, L. Villablanca, J. Krause, M.C. Rivara, Improving the quality of meshes for the simulation of semiconductor devices using Lepp-based algorithms, Internat. J. Numer. Methods Engrg. 58 (2003) 333–347.

[45] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, W. Stuetzle, Mesh optimization, ACM Comput. Graphics 27 (1993) 19–26.
[46] B. Karamete, M. Beall, M. Shephard, Triangulation of arbitrary polyhedra to support automatic mesh generators, Internat. J. Numer. Methods Engrg. 49 (2000) 167–191.
[47] P. Knupp, Achieving finite element mesh quality via optimization of the Jacobian matrix norm and associated quantities, I—a framework for surface mesh optimization, Internat. J. Numer. Methods Engrg. 48 (2000) 401–420.
[48] F. Labelle, J. Shewchuk, Anisotropic Voronoi Diagrams and Guaranteed-Quality Anisotropic Mesh Generation, SoCG 2003, San Diego.
[49] C. Lee, Automatic metric advancing front triangulation over curved surfaces, Eng. Comput. 17 (2000) 48–74.
[50] G. Leibon, D. Letscher, Delaunay triangulations and Voronoi diagrams for Riemannian manifolds, in: Proceedings of the 16th Annual Symposium on Computational Geometry (HK), ACM, New York, June 2000, pp. 341–349.
[51] X. Li, Sliver-free three dimensional Delaunay mesh generation, Ph.D thesis, UIUC, 2000.
[52] A. Liu, M. Baida, How far flipping can go towards 3D conforming/constrained triangulation, in: Proceedings of the Ninth International Meshing Roundtable, 2000.
[53] S. Lloyd, Least square quantization in PCM, IEEE Trans. Inform. Theory 28 (1982) 129–137.
[54] S. Lo, Volume discretization into tetrahedra-I, Verification and orientation of boundary surfaces, Comput. Structures 36 (1991) 493–500.
[55] S. Lo, Volume discretization into tetrahedra-II, 3D triangulation by advancing front approach, Comput. Structures 36 (1991) 501–511.
[56] R. Lohner, Progress in grid generation via the advancing front technique, Engrg. Comput. 12 (1996) 186–199.
[57] R. Lohner, C. Juan, Generation of non-isotropic unstructured grids via directional enrichment, Internat. J. Numer. Methods Engrg. 49 (1) (2000) 219–232.
[58] R. Lohner, P. Parikh, Generation of three-dimensional unstructured grids by the advancing-front method, Internat. J. Numer. Methods Engrg. 8 (1988) 1135–1149.
[59] J. MacQueen, Some methods for classification and analysis of multivariate observations, in: L. LeCam, J. Neyman (Eds.), Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, I, University of California, 1967, pp. 281–297.
[60] D. Marcum, N. Weatherill, Unstructured grid generation using iterative point insertion and local reconnection, AIAA J. 33 (9) (1995) 1625.
[61] D. Mavriplis, Directional coarsening and smoothing for anisotropic Navier–Stokes problems, Electron. Trans. Numer. Anal. 6 (1997) 182–197.
[62] K. Nakahashi, D. Sharov, Direct surface triangulation using the advancing front method, AIAA paper 95-1686-CP, 1995.
[63] D. Newman, The Hexagon theorem, IEEE Trans. Inform. Theory 28 (1982) 137–139.
[64] J. Peraire, J. Peiro, L. Formaggia, K. Morgan, O. Zienkiewicz, Finite element Euler computations in three dimensions, Internat. J. Numer. Methods Engrg. 26 (1988) 2135–2159.
[65] A. Rassineux, Generation and optimization of tetrahedral meshes by advancing front technique, Internat. J. Numer. Methods Engrg. 41 (1998) 651–674.
[66] S. Rippa, Long and thin triangles can be good for linear interpolation, SIAM J. Numer. Anal. 29 (1992) 257–270.
[67] W. Schroeder, M. Shephard, A combined octree/Delaunay method for fully automatic 3D mesh generation, Internat. J. Numer. Methods Engrg. 29 (1990) 37–55.
[68] M. Shephard, M. Georges, Automatic three-dimensional mesh generation by the finite Octree technique, Internat. J. Numer. Methods Engrg. 32 (1991) 709–749.
[69] J. Shewchuk, What is a good linear finite element? Interpolation, conditioning, anisotropy and quality measures. CS report, UC Berlekey.
[70] J. Shewchuk, Tetrahedral mesh generation by Delaunay refinement, in: 14th Annual ACM Symposium on Computational Geometry, 1998, pp. 86–95.
[71] J. Schewchuk, Constrained Delaunay tetrahedralization and provably good boundary recovery, in: Proceedings of the11th International Meshing Roundtable, 2002, pp. 193–204.
[72] K. Tchon, K. Mohammed, G. Francois, C. Ricardo, Constructing anisotropic geometric metrics using octrees and skeletons, in: Proceedings of the 12th International Meshing Roundtable, 2003, pp. 293–304.
[73] J. Thompson, B. Soni, N. Weatherill, Handbook of Grid Generation, CRC Press LLC, Boca Raton, FL, 1999.

[74] D. Watson, Computing the *n*-dimensional Delaunay tessellation with applications to Voronoi polytopes, Comput. J. 24 (1981) 167–172.
[75] N. Weatherill, The integrity of geometrical boundaries in the two-dimensional Delaunay triangulation, Comput. Appl. Numer. Methods 6 (1990) 101–109.
[76] N. Weatherill, O. Hassan, Efficient three dimensional Delaunay triangulation with automatic point creation and imposed boundary constraints, Internat. J. Numer. Methods Engrg. 37 (1994) 2005–2039.
[77] J. Wright, A. Jack, Aspects of three-dimensional constrained Delaunay meshing, Internat. J. Numer. Methods Engrg. 37 (1994) 1841–1846.
[78] Y. Zhang, C. Bajaj, B. Sohn, 3D Finite Element Meshing from Imaging Data, Comput. Methods Appl. Mech. Eng., 2005, in press.