A study on generation of three-dimensional M-uniform tetrahedral meshes in practice

Lennard Kamenski and Hang Si

Weierstrass Institute, Mohrenstr. 39, 10117 Berlin, Germany
lennard.kamenski@wias-berlin.de, hang.si@wias-berlin.de

1 Introduction

Mesh generation and mesh adaptation are essential processes for numerical methods like finite element and finite volume simulations. The quality of the mesh will tremendously affect the accuracy and efficiency of the methods. Research shows that the actual meaning of the mesh quality depends on the real problem (isotropic or anisotropic features) and the objectives (minimizing the interpolation error or improving the conditioning number).

The use of a metric tensor field for mesh quality and mesh size specification is now a widely used tool for mesh generation and mesh adaptation. We adopt the concept of an *M-uniform mesh* [3], which considers any adaptive mesh as a uniform one in some metric depending on the quantity of interest. In other words, an *M-uniform* mesh is an ideal mesh for a given metric which may be a combination of several criteria, like element quality, element size, and element orientation, etc. With this concept, the meshing problem can be formalized as: *How to generate the M-*uniform *mesh for a given metric.* It is a very challenging problem in theory.

Tetrahedral mesh generation has been studied extensively for decades. There are established methods for mesh generation and mesh adaptation, see, e.g., [2]. Robust software implementations exist and some of them are freely available. Most of the codes provide the option to take a user-defined mesh sizing function or a metric tensor field as input and generate an adapted tetrahedral meshes with respect to it. A practical question naturally arises: *How close are meshes computed by some common mesh generators to being M-uniform?* So far, only few works addressed this question in theory [5, 6]. Numerical experiments suggest that, at least in two dimensions, meshes that are close to be M-uniform can be generated in practice [3, 5, 6].

The purpose of this research note is to present a practical study of the above question in three dimensions. We compare meshes generated with two freely available programs, `Mmg3d` [1] and `TetGen` [7], by using isotropic mesh sizing functions. This study provides a fair justification of the state-of-the-art metric-based methods for adaptive mesh generation.

2 Mesh conformity measure

We consider an M-uniform mesh approach and look at any adaptive mesh as a uniform one in some metric M. An M-uniform mesh satisfies the equidistribution and alignment conditions [4],

$$|K| \det(M_K)^{\frac{1}{2}} = \frac{1}{N} \sum_{\tilde{K} \in \mathcal{T}_h} \left|\tilde{K}\right| \det(M_{\tilde{K}})^{\frac{1}{2}} \quad \forall K \in \mathcal{T}_h, \tag{1}$$

$$\frac{1}{d} \operatorname{tr}\left((F_K')^T M_K F_K' \right) = \det\left((F_K')^T M_K F_K' \right)^{\frac{1}{d}} \quad \forall K \in \mathcal{T}_h, \tag{2}$$

where $F_K \colon \hat{K} \to K$ is the affine mapping from a reference element $\hat{K}$ to a mesh element K, F_K' is the Jacobian matrix of F_K, M_K is an average of M over K and N is the number of mesh elements.

In practice, however, it is more realistic to expect that the generated mesh is only quasi-M-uniform,

$$|K| \det(M_K)^{\frac{1}{2}} \le C_{eq} \frac{1}{N} \sum_{\tilde{K} \in \mathcal{T}_h} \left|\tilde{K}\right| \det(M_{\tilde{K}})^{\frac{1}{2}} \quad \forall K \in \mathcal{T}_h, \tag{3}$$

$$\frac{1}{d} \operatorname{tr}\left((F_K')^T M_K F_K' \right) \le C_{ali} \, |K|^{\frac{2}{d}} \det(M_K)^{\frac{1}{d}} \quad \forall K \in \mathcal{T}_h, \tag{4}$$

where $C_{eq} \ge 1$ and $C_{ali} \ge 1$ are some constants independent of K, N, and $\mathcal{T}_h$ (note that conditions (3) and (4) with $C_{eq} = C_{ali} = 1$ imply (1) and (2)).

The important question is how does the difference of the generated (quasi-M-uniform) mesh to the desired (M-uniform) mesh affects, for example, the interpolation error? Fortunately, the bound on the interpolation error depends continuously on the difference of the desired and the actually constructed mesh. For example, for any quasi-M-uniform mesh satisfying (3) and (4) the interpolation error in the H^1 semi-norm is bounded by

$$|u - u_h|_{H^1(\Omega)} \le C_{ali}^{\frac{d+1}{2}} C_{eq}^{\frac{d+2}{2d}} \mathcal{E},$$

where $\mathcal{E}$ is the interpolation error bound for the M-uniform mesh [5, Theorem 2.1]. (using a different norm instead of $|\cdot|_{H^1}$ will change the powers of C_{eq} and C_{ali} but the essential statement remains the same).

Thus, it is natural to define the values C_{eq} and C_{ali} as the mesh size and shape conformity measures (or $C_{ali}^{\frac{d+1}{2}} C_{eq}^{\frac{d+2}{2d}}$ as the global conformity measure).[1]

[1] Another mesh conformity measure was developed in [6] which can compare two meshes in order to determine which of them is closer to the prescribed metric. However, it doesn't have the direct connection to the interpolation error estimates as the measures defined in this section.

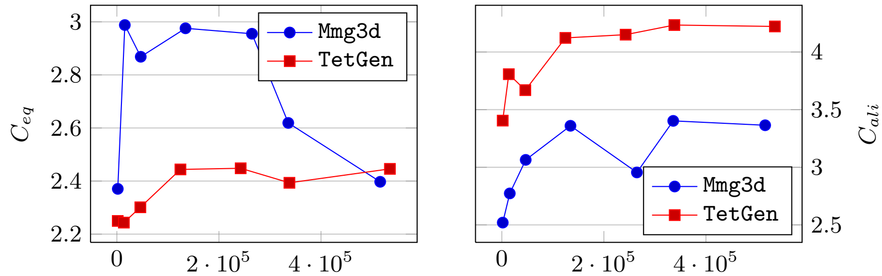


Figure 1: Uniform meshes: C_{eq} and C_{ali} vs. number of elements

3 Employed mesh generators

TetGen [7] is C++ program for generating quality tetrahedral meshes for 3D polyhedral domains. It implements the constrained Delaunay refinement algorithm [8] to efficiently refine and improve the mesh quality and the mesh adaptivity. Starting from a given polyhedral domain and a scalar mesh sizing function (defined on the nodes of a background mesh), TetGen generates a good quality isotropic tetrahedral mesh whose mesh size conforms to the given sizing function. For our tests we use the version 1.5 of the program.

Mmg3d [1] is a freely available tetrahedral re-meshing program for anisotropic mesh adaptation. It is based on local mesh modifications and an anisotropic version of Delaunay kernel for vertex insertion. Starting from a given tetrahedral mesh, it produces quasi-uniform meshes with respect to a metric tensor field.

4 Numerical results and discussion

First, we check the most simple case of uniform meshes of the unit cube $[0,1]^3$ with Mmg3d and TetGen. In this case, the metric M is given by the identity matrix,

$$M = I,$$

rescaled accordingly to the software input requirements to produce a mesh with a predefined number of elements. Figure 1 shows the values of C_{eq} and C_{ali} for a series of grids. We see that both C_{eq} and C_{ali} always stay bounded and small with $C_{eq} < 3$ and $C_{ali} < 4.5$. Interestingly, TetGen seems to do a better job on producing elements of proper size (at least for grids with smaller number of elements) whereas Mmg3d seems to produce elements that are closer to the uniform shape.

For the second example we use adaptive meshes of the unit cube with

$$M = \left(0.1 + 1000 \cdot e^{-100\left|(x-0.5)^2+(y-0.5)^2+(z-0.5)^2-0.25^2\right|}\right) \times I,$$

which is rescaled accordingly to produce meshes with a predefined number of elements. A proper mesh should be very fine near the surface of the sphere with the origin in the center of the cube and the radius of 0.25 (see Figs. 2a and 2b). As in the example with uniform meshes, C_{eq} and C_{ali} stay small with the increasing number of elements (Fig. 2c); both `TetGen` and `Mmg3d` produce very similar results with $C_{eq} \approx C_{ali} \approx 6$ for finer grids.

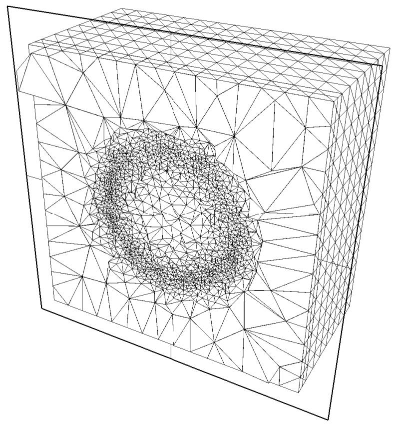
A typical histogram for the element-wise size and shape conformity measures ($C_{eq,K}$ and $C_{ali,K}$) is shown in the Fig. 2d for a mesh generated by `TetGen` (results for `Mmg3d` are similar). Here, instead of $C_{eq,K}$ we show $\log C_{eq,K}$ so that elements of optimal size have $\log C_{eq,K} = 0$, a positive value means that the element is bigger than prescribed by the metric and a negative value means that the element is smaller than prescribed. Note that $C_{ali,K} \geq 1$ by definition and its optimal value is one. Figure 2d shows that most of the elements have size and shape close to the optimal with larger deviations appearing only for a small number of elements.

Thus, we can conclude that the state-of-the-art methods for isotropic tetrahedral mesh generation and mesh adaptation are able to produce meshes that are close to be uniform in the prescribed metric.

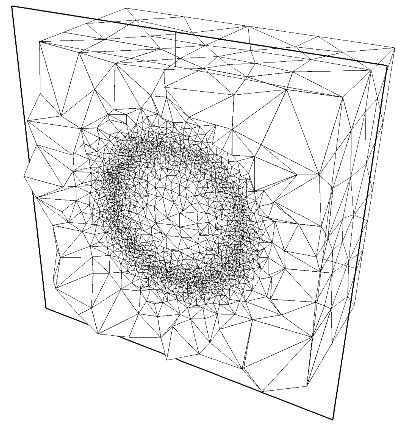
In the future, tests using anisotropic metric fields are planned.

References

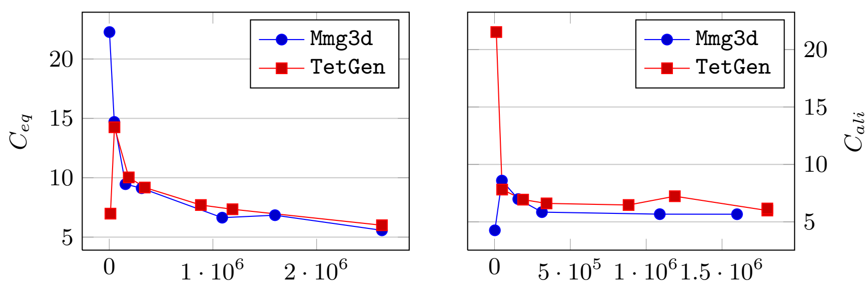
1. C. Dobrzynski. MMG3D: User Guide. Rapport Technique RT-0422, INRIA, Mar. 2012. `http://www.math.u-bordeaux1.fr/~cdobrzyn/logiciels/mmg3d.php`.
2. P. J. Frey and P. George. *Mesh Generation - Application to Finite Elements*. Hermes Science Publishing, Oxford, UK, 1st edition, 2000. ISBN 1-903398-00-2.
3. W. Huang. Metric tensors for anisotropic mesh generation. *J. Comput. Phys.*, 204(2):633–665, 2005.
4. W. Huang. Anisotropic mesh adaptation and movement. In T. Tang and J. Xu, editors, *Adaptive computations: Theory and Algorithms*, Mathematics Monograph Series 6, chapter 3, pages 68–158. Science Press, Beijing, 2007.
5. L. Kamenski and W. Huang. How a nonconvergent recovered Hessian works in mesh adaptation. arXiv: 1211.2877 [math.NA] (submitted).
6. P. Labbé, J. Dompierre, M. Vallet, F. Guibault, and J.-Y. Trépanier. A universal measure of the conformity of a mesh with respect to an anisotropic metric field. *Int. J. Numer. Methods Engrg.*, 61(15):2675–2695, 2004.
7. H. Si. `http://tetgen.org`, 2013.
8. H. Si. Adaptive tetrahedral mesh generation by constrained Delaunay refinement. *Int. J. Numer. Methods Engrg.*, 75(7):856–880, 2008.



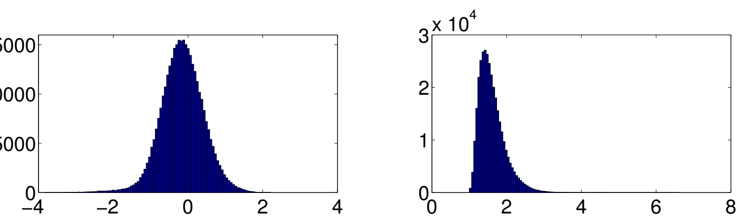
(a) Mmg3d mesh, 312 700 elements



(b) TetGen mesh, 342 990 elements



(c) C_{eq} and C_{ali} vs. number of elements



(d) histograms for $\log C_{eq,K}$ (left) and $C_{ali,K}(right)$ (TetGen mesh, 342 990 elements)

Figure 2: Adaptive meshes