ANISOTROPIC MESH GRADATION CONTROL*

Xiangrong Li[1†] Jean-François Remacle[2] Nicolas Chevaugeon[2]
Mark S. Shephard[1]

[1]*Scientific Computation Research Center, CII-7011, 110 8th Street, Rensselaer Polytechnic Institute, Troy, NY 12180-3590, U.S.A.*
[2]*Department of Civil Engineering, Place du Levant 1, 1348 Louvain-la-Neuve, Belgium*
[†]*Corresponding author: xli@solidworks.com*

ABSTRACT

The paper presents an *a priori* procedure to control the element size and shape variation for meshing algorithms governed by anisotropic sizing specifications. The field of desired element size and shape is represented by a background structure. The procedure consists in replacing the initial field with a smoothed one that preserves anisotropic features and smaller element sizes. The smoothness of the resulting field can be controlled by a prescribed threshold value γ_0. Examples are included to show the application in three dimensional anisotropic adaptive simulation, as well as the effect of γ_0.

Keywords: tensor field, anisotropic, mesh gradation, mesh adaptation

1. INTRODUCTION

To reduce computation time and memory usage without sacrificing accuracy, in general a well-graded anisotropic mesh is required [1, 2] (see figure 1 for an example). One of the most important aspect to generating such a desired finite element mesh is specifying a desired element size and shape in space [3, 4, 5, 6]. Sizing function and tensor field [7] have been used to represent this desired shape and size distribution, and many authors have described approaches to specify the scalar or tensor field from various factors, *e.g.*, error norms [8, 9, 10, 2], surface curvature/proximity to other surfaces [7, 11, 12, 13], user defined sources [14], *etc*. Emphasis also has been given to the conformity criteria between the field and the mesh [7, 15, 16]. The scalar/tensor field can be considered as a transformation that defines a *transformed* space (or Riemannian space), where all desired elements are unitary and equilateral. However, one technical issue related to the field specification and its conformity criteria remains not fully solved. In particular, due to the complexity and variety in both geometry and physics, the field defined as above may include abrupt change in size, shape or both, and the mesh conforming to the field may be in poor element shape and unlikely suitable for computation purpose. Figure 2 depicts a simple two dimensional example to demonstrate this issue, where a small element size is specified around the arc and a large global mesh size is specified anywhere else. Figure 2(a) shows the mesh conforming to the specified mesh size. Poor elements have to be created to connect short edges on the arc with long interior edges. To obtain the meshes as illustrated in figure 2(b), either determining the sizing and gradation during the meshing process [17, 18] (*i.e.*, the field will not be fully respected), or modifying the field by an *a priori* procedure is required.

It has been common, especially in adaptive simulations, that the tensor field is defined as a piecewise interpolation over a background structure covering the entire domain, where the background structure could be the mesh for previous solution [6, 4], the current mesh being modifying [15], an octree [19] or *etc*. General *a priori* mesh gradation control is possible for such field representation since the interpolant or nodal size and shape of the field can be locally modified based on

*THIS WORK WAS SUPPORTED BY THE ASCI FLASH CENTER AT THE UNIVERSITY OF CHICAGO, UNDER CONTRACT B341495, BY U.S. ARMY RESEARCH OFFICE THROUGH GRANT DDAD19-01-0655 AND THE DOE SCIDAC PROGRAM THROUGH AGREEMENT DE-FC02-01-ER25460.

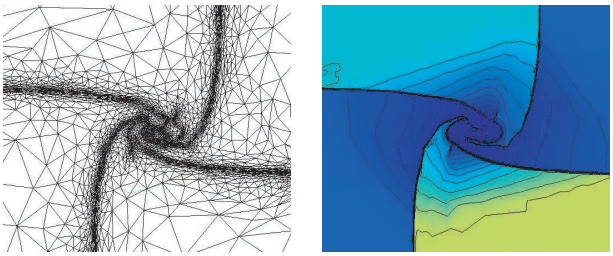


Figure 1: Graded anisotropic mesh (left) that captures evolving discontinuous solution field (right) in solving a four contact Riemann problem.

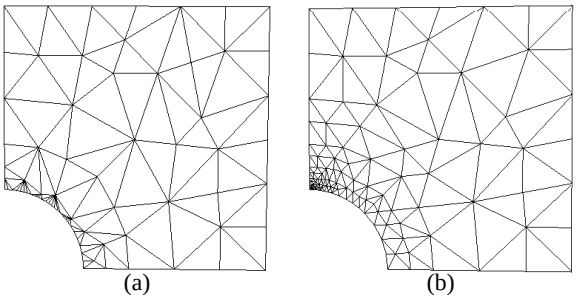


Figure 2: Two dimensional example to illustrate the need for mesh gradation control. (a) Poorly shaped mesh conforming a given sizing function. (b) Graded mesh conforming to a modified sizing function.

neighboring size and shape information.

For scalar field and isotropic mesh gradation control, such *a priori* procedures have been presented by Löhner [14, 20], Borouchaki *et al.* [21] and Owen *et. al.* [13]. Löhner utilizes a tetrahedral background mesh to provide sizing information to an advancing front tetrahedral mesher. To maintain a desired growth ratio, the desired mesh size attached to vertices of the background mesh is adaptively adjusted by applying a geometric growth formula. The background mesh can be refined if it can not well representing the mesh size field. In the work by Borouchaki *et al.*, two measures related to the gradient of scalar fields are proposed, and mesh size values attached to vertices of a background mesh are corrected to limit the proposed measures. Both Löhner and Borouchaki use piecewise linear interpolant. Owen *et al.* use a natural neighbor interpolation method to alleviate the abrupt variation of the field. Borouchaki *et al.* have proposed a simple anisotropic extension of their isotropic procedure by considering one specific direction [21]. This procedure does improve the mesh gradations, but tends to not maintain the desired level of mesh anisotropy.

This paper discusses an *a priori* anisotropic mesh gradation control procedure that explicitly accounts for preserving anisotropy. It can be considered as a supplement to papers in reference [22] and [2] related to anisotropic mesh size field definition. In section 2, the notion and its geometric significance of a directional mesh gradation measure is given. In section 3, we present a three dimensional *a priori* procedure that preserves element size and anisotropy. Section 4 provides example meshes to show this *a priori* procedure, and its application in three dimensional adaptive anisotropic simulations.

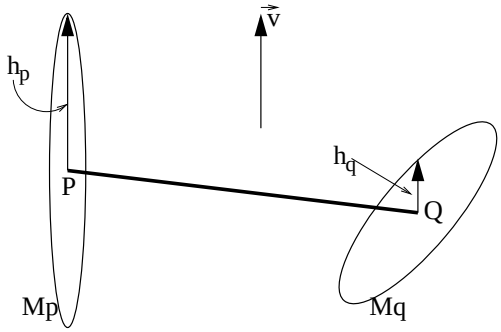


Figure 3: Definition of directional mesh gradation measure γ. Mesh tensors are indicated by ellipses.

2. A DIRECTIONAL MESH SIZE GRADATION MEASURE

In this section, we give an anisotropic mesh gradation measure that can evaluate the smoothness quality of a given mesh tensor field.

Definition Let M_i (i=p,q) be the 2×2 or 3×3 symmetric positive definite tensor specifying the desired mesh size and shape at point P and Q, and $\vec{v}$ be a unitary direction vector, as depicted in figure 3. The mesh size gradation measure related to point P and Q in direction $\vec{v}$ is:

$$\gamma(\vec{v}) = e^{\frac{|h_p(\vec{v}) - h_q(\vec{v})|}{L_{pq}}} \tag{1}$$

where L_{pq} is the distance between the two points, and $h_i (i = p, q)$ is the desired edge length of tensor M_i in direction $\vec{v}$, *i.e.* [15, 22]:

$$h_i(\vec{v}) = \frac{1}{\sqrt{\vec{v}\, M_i\, \vec{v}^T}} \tag{2}$$

To illustrate the significance of this measure, let us construct two neighboring mesh edges, $\mathbf{PA}$ and $\mathbf{AB}$, along edge $\mathbf{PQ}$ that satisfy the local mesh tensor field defined by M_p and M_q. As illustrated in figure 4(a), h_p and h_q are the desired mesh edge length along $\mathbf{PQ}$ computed in terms of equation (2), and $|x_a - x_p|$ and $|x_b - x_a|$ are the length of edge $\mathbf{PA}$ and $\mathbf{AB}$ respectively. Figure 4(b) shows the two mesh edges in the *transformed* space. Both are unitary since they perfectly match the tensor field [7, 10, 15]. For linearly interpolated mesh size in each direction, the desired edge length

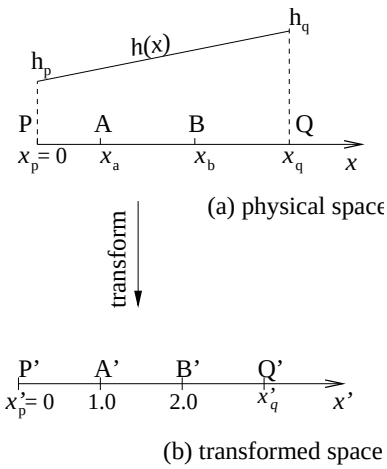


Figure 4: The significance of γ.

along $\mathbf{PQ}$ at position x is:

$$h(x, \vec{\mathbf{PQ}}) = \frac{h_q - h_p}{L_{pq}}\, x \;+\; h_p \quad (h_q \geq h_p) \qquad (3)$$

Let x be the coordinates, x' be the corresponding coordinate in the *transformed* space, and $x_p = 0$ is transformed into $x'_p = 0$. The mapping between the two spaces is:

$$x' = \int_0^x \frac{1}{h(x, \vec{\mathbf{PQ}})} dx = \frac{1}{C} ln(\frac{C}{h_p} x + 1) \qquad (4)$$

with C defined as $(h_q - h_p)/L_{pq}$. Plugging $x'_p = 0$, $x'_a = 1$ and $x'_b = 2$ into equation (4), the length of edge $\mathbf{PA}$ and $\mathbf{AB}$ can be derived:

$$|x_a - x_p| \;=\; \frac{h_p(e-1)}{C}\, e^{\frac{h_q - h_p}{L_{pq}}} \qquad (5)$$

$$|x_b - x_a| \;=\; \frac{h_p(e-1)}{C}\, e^{\frac{2(h_q - h_p)}{L_{pq}}} \qquad (6)$$

Thus the ratio of two neighboring edges is

$$\gamma(\vec{v}) = \frac{x_b - x_a}{x_a - x_p} = e^{\frac{h_q - h_p}{L_{pq}}}$$

Above equation indicates that the measure represents the ratio of desired edge length variation on edge $\mathbf{PQ}$. In particular, a mesh tensor field with $\gamma = 1$ in all directions is a constant field. A mesh satisfying the tensor field of $\gamma_x = 2$ (γ_x is the γ in x axis) should consist of edges in length series: l_0, $2l_0$, $4l_0$ ..... $2^n l_0$ (l_0 is the length of an edge on x axis). Figure 5 shows a 2D mesh of the 1×1 domain approximately satisfying a mesh tensor field with $\gamma_x = 1$ and $\gamma_y = 1.24$. It can be seen that edge length in x axis does not change while the number of elements in y axis decreases from 32 to 4, increasing the edge length in y direction at a ratio of 1.24.

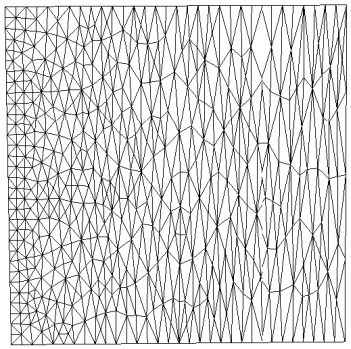


Figure 5: A 2D mesh satisfying a mesh tensor field of $\gamma_x = 1$, $\gamma_y = 1.24$.

Since it is not possible to consider all directions (that are infinite) at a point, we select directions in terms of the principle direction of the given mesh tensor at either P or Q as $\vec{v}$. In particular, we identify three situations in terms of the geometric shape a tensor can represents: sphere, spheroid and ellipsoid. If the mesh tensor has three identical eigenvalues, the tensor is degenerated into a scalar and the computation of h_i $(i = p, q)$ and γ is independent of directions. If the mesh tensor has two identical eigenvalues, its geometric shape is a spheroid and only the direction associated with the different eigenvalue need respecting. If the tensor has three different eigenvalues, all the three principle directions are respected.

This definition of $\gamma(\vec{v})$ is consistent with the H-shock introduced in reference [21] in the case of isotropic. Since, in terms of equation (4), the measuring length of edge $\mathbf{PQ}$ in the *transformed* space is:

$$L'_{pq} = |x'_q - x'_p| = \frac{L_{pq}}{h_q - h_p}\, ln(\frac{h_q}{h_p}) \qquad (7)$$

The exponential term of equation (1) can be replaced with $ln(\frac{h_q}{h_p})/L'_{pq}$, then:

$$\gamma = e^{ln(\frac{h_q}{h_p})/L'_{pq}} = (h_q/h_p)^{\frac{1}{L'_{pq}}} \qquad (8)$$

which is the definition of H-shock. The definition of equation (1) is of our favor since it avoids the concept of measuring length in *transformed* space.

Note that this measure describes the smoothness property of the mesh tensor field. It does not ensure if there is enough geometric space to create the desired mesh, which could be determined using equation (7). For example, if $L'_{pq} < 1$, there is not enough space; If $L'_{pq} = 1$, there is space for one edge and there are space for two if $L'_{pq} = 2$ and etc.

3. PROCEDURE OF MESH TENSOR FIELD SMOOTHING

3.1 Overview

Given a piecewise mesh tensor field defined on vertices of a background structure, our goal is to ensure the smoothness quality of the field by checking and, if necessary, modifying the discrete tensors so that the directional mesh gradation measure γ associated with any edge of the background structure is less than or equal to a given threshold value, thus the mesh satisfying the modified mesh tensor field has controlled gradation. This section proposes a mesh tensor smoothing procedure that respects the directionality and smaller sizes the given mesh tensor field describes.

A mesh tensor can be modified by changing its principle directions $\mathbf{e}_i$ (i=1,2,3) and the desired mesh size h_i in each principle direction, which relates to the eigenvalue of the tensor as: $\lambda_i = 1/h_i^2$. To respect anisotropy and the smaller mesh size, three assumptions are adopted in the proposed procedure:

- If a smaller mesh size is close to a large one, the large size is reduced.
- If a tensor of high aspect ratio [1] is close to a low aspect ratio tensor, the directions of the higher aspect ratio tensor are preserved and the direction of the lower one may be adjusted.
- If two high aspect ratio tensors are close, all principle directions are respected.

Although reducing the larger mesh size will increase the number of elements, it is conservative and will not lose accuracy in analysis. Therefore the strategy of reducing the large instead of increasing the smaller is adopted.

Figure 6 and 7 give two simple two dimensional examples to demonstrate the concept of the second and the third assumption, where mesh tensors attached onto point P and Q are indicated by ellipses and referred to as M_p and M_q, while the principle directions of M_p and M_q are illustrated by the axis of local coordinate systems. In figure 6(a), the aspect ratio of M_p and M_q is 10 and 1.1, respectively. To capture the anisotropy tensor M_p represents and make smooth mesh size variation possible, the direction of tensor M_q is adjusted to align with the principle directions of M_p and reduce its size in x axis as shown in figure 6(b). In figure 7, M_p and M_q have the same aspect ratio but perpendicular stretching directions. To capture anisotropy represented by both, all direction information should be maintained, however, the size in y axis of M_p and that in x axis of M_q are reduced to allow smoothness mesh size variation.

[1] Given a mesh tensor, the directional desired length distribution follows a ellipsoidal surface [5, 15]. Its aspect ratio R is defined as the maximum desired length to the minimum length. Clearly, $R \geq 1$.

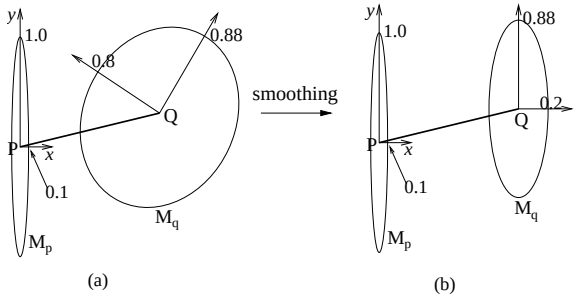


Figure 6: A 2D example to illustrate the need for the adjustment of both direction and size.

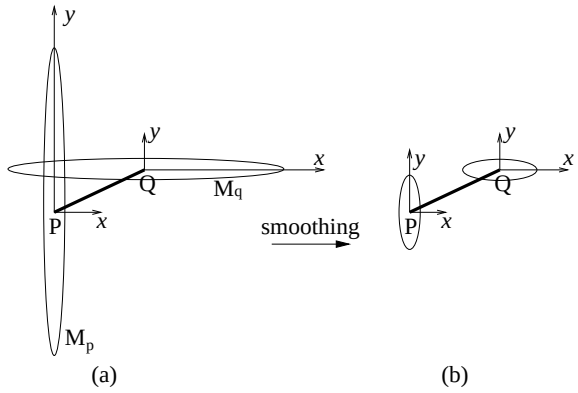


Figure 7: A 2D example of preserving direction and reducing size.

The subsections that follow are organized as follows: Section 3.2 introduces an algorithm that adjusts principle directions in terms of relative aspect ratio of two neighboring mesh tensors. Section 3.3 presents the method to perform directional larger mesh size reduction if any directional γ value exceeds a prescribed threshold. Section 3.4 presents the overall algorithm.

3.2 Selection of directions

Consider the two mesh tensors attached onto the end vertices of edge $\mathbf{PQ}$, we capture anisotropic features by preserving the stretching direction(s) of the higher aspect ratio tensor while allowing the principle direction(s) of lower aspect ratio tensor adjustable in terms of a parameter referred to as "anisotropy respect factor" in this context.

Definition Let R_p and R_q be the aspect ratio of the tensor at neighboring point P and Q, and $R_p \geq R_q$, the anisotropy respect ratio related to point P and Q is the value:

$$\alpha = \frac{(R_q - 1)\, R_p}{(R_p - 1)\, R_q} \tag{9}$$

Equation (9) has been defined such that α is a value in inter-

val $[0, 1]$ with $\alpha = 0$ if one of the mesh tensors is isotropic and $\alpha = 1$ if the two mesh tensors have the same aspect ratio. This property is ideal for the adjustment of eigenvectors of the mesh tensors at point P and Q. Note that, when both mesh tensors are isotropic, *i.e.*, $R_p = R_q = 1$, α is not defined ($\alpha = \frac{0}{0}$). However, this does not cause a problem since computing α is unnecessary in case of isotropic.

Equation (10) gives the formula to adjust eigenvectors of the less anisotropic mesh tensor based on α, where $\mathbf{e}_i^p$ and $\mathbf{e}_j^q$ are the eigenvector of mesh tensor at point P and Q with $R_p \geq R_q$, and $\mathbf{e}_j^q|_{new}$ is the adjusted eigenvector at point Q. It ensures the mesh tensor with strong anisotropy is maintained with respect to both. In the case tensor M_q is isotropic, simply set its principle direction the same as that of M_p.

$$\mathbf{e}_j^q|_{new} = (1 - \alpha)\, \mathbf{e}_i^p + \alpha\, \mathbf{e}_j^q \tag{10}$$

If both mesh tensor M_p and M_q have two identical eigenvalues, *i.e.*, the geometric representation of both tensors is a spheroid, we assume that the modified tensor remains spheroidal, then only one direction needs respecting and $i = j = 1$.

If at least one of the mesh tensors has three different eigenvalues, *i.e.*, the geometric representation of a mesh tensor is an ellipsoid, three directions can be obtained for each tensor, therefore, $i, j = 1, 2, 3$ and properly relating the principle direction $\mathbf{e}_i^p$ with that of $\mathbf{e}_j^q$ is required in order to apply equation (10).

The relation between index i, j can be determined by minimizing the maximal angle between direction $\mathbf{e}_i^p$ and $\mathbf{e}_j^q$ ($i, j = 1, 2, 3$). Since the desired mesh size along a direction is the same as that in its opposite direction, the angle between the two directions is in interval $[0, \pi/2]$ (flip the direction of $\mathbf{e}_i^p$ or $\mathbf{e}_j^q$ if the angle between them is greater than $\pi/2$). Figure 8 depicts the matching of index i and j for a two dimensional example. In this example, $\mathbf{e}_i^p$ and $\mathbf{e}_i^q$ (i=1,2) represent the principle directions of the mesh tensor at point P and Q. The dashed line is parallel to $\mathbf{e}_1^p$. It is drawn to show β_j (j=1,2), the angle between $\mathbf{e}_1^p$ and $\mathbf{e}_j^p$. Since $\beta_2 < \beta_1$, in this example, $\mathbf{e}_1^p$ is related to $\mathbf{e}_2^q$. The two remaining directions, $\mathbf{e}_2^p$ and $\mathbf{e}_1^q$, are related obviously. In the ambiguous situation where $\beta_1 = \beta_2$, we simply select one of the directions.

Figure 9 depicts a situation where M_p has two identical eigenvalue thus one characteristic direction $\mathbf{e}_1^p$ while M_q has three different eignvalues thus three characteristic direction $\mathbf{e}_1^q$, $\mathbf{e}_2^q$, $\mathbf{e}_3^q$. The three dashed vectors indicate the principle directions of tensor M_q originated at point P. To apply equation (10), we first compute the anlges between $\mathbf{e}_1^p$ and $\mathbf{e}_j^q$ (j=1,2,3), and determine that $\mathbf{e}_1^p$ should be related to $\mathbf{e}_1^q$ since the angle between them is the smallest. Then, direction $\mathbf{e}_i^p$ (i=2,3) is obtained by projecting $\mathbf{e}_i^q$ onto the plane perpendicular to $\mathbf{e}_1^p$ so that the anlge between $\mathbf{e}_i^p$ and $\mathbf{e}_j^q$ (i,j=2,3) is minimized.

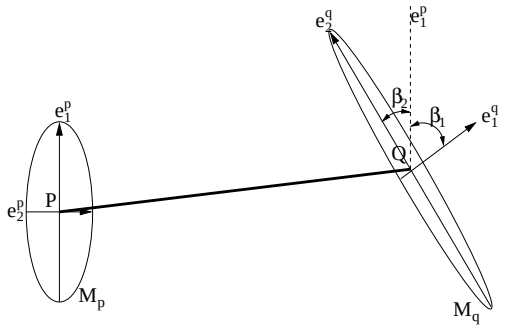


Figure 8: 2D example of principle direction matching between tensor M_p and M_q.

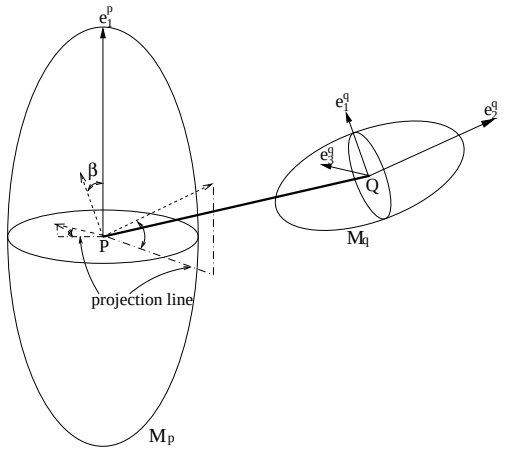


Figure 9: Matching of principle directions for one spheroid one ellipsoid case.

3.3 Directional adjustment of mesh size(s)

For the mesh size represented by a mesh tensor, in general there are three sizing components, one for each principle direction of the tensor. This section discusses the algorithm that checks the smoothness of mesh tensor variation and, if necessary, reduce all or part of the three sizing components. Special cases where the number of sizing components is degenerated into one or two components are also addressed.

Consider a mesh sizing component h_i^p of mesh tensor M_p, and a nearby mesh tensor M_q. The algorithm for checking and possibly reducing M_p consists of four steps:

- get h_i^q, the corresponding directional desired mesh size associated with mesh tensor M_q.
- compute the directional mesh gradation measure γ in terms of equation (1).
- if $\gamma > \gamma_0$ and $h_p > h_q$ (γ_0 is the prescribed threshold

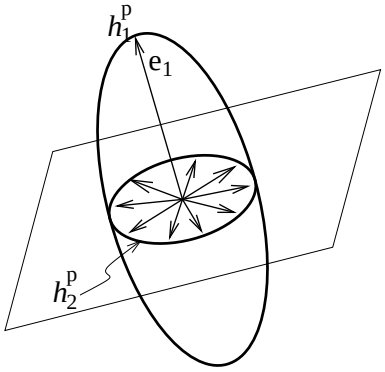


Figure 10: Illustration of the two sizing components of a spheroidal mesh tensor. One (h_1^p) is associated with direction $\mathbf{e}_1$. Another (h_2^p) is associated with all possible directions orthogonal to $\mathbf{e}_1$.

value), reduce h_p to make $\gamma = \gamma_0$.

- repeat the above steps for all sizing components of M_p.

We identify three situations in computing h_i^q: (i) when both M_p and M_q are isotropic, h_i^q is simply the degenerated scalar value of M_q. (ii) When both geometric representations of M_p and M_q are spheroidal, the number of sizing components is degenerated into two and one of the sizing components, h_2^p as depicted in figure 10, is associated with a plane (any direction on the plane has the same desired size). To make the reduced tensor remain spheroidal, h_2^q should be the sizing component of M_q that is also associated with a plane. (iii) In all other situations, each sizing component h_i^p is associated with a unique direction, thus we can determine a direction associated with tensor M_q in terms of equation (10) and compute h_i^q using equation (2).

Let h_p' be the reduced size of h_i^p. To make $\gamma = \gamma_0$ after the reduction, we have

$$\gamma_0 = e^{(h_p' - h_i^q)/L_{pq}}$$

Therefore

$$h_p' = L_{pq} ln(\gamma_0) + h_i^q \tag{11}$$

After all sizing components of tensor M_p are processed, M_p will be modified if any of its sizing components has been reduced. The new tensor are constructed as follows:

$$M_p^{new} = \left[\ \mathbf{e}_1\ \mathbf{e}_2\ \mathbf{e}_3\ \right] \begin{bmatrix} 1/h_1' & 0 & 0 \\ 0 & 1/h_2' & 0 \\ 0 & 0 & 1/h_3' \end{bmatrix} \left[\ \mathbf{e}_1\ \mathbf{e}_2\ \mathbf{e}_3\ \right]^T \tag{12}$$

where $\mathbf{e_i}$ (i=1,2,3) is the original principle directions of M_p or the adjusted directions given by equation (10), and h_i' is the reduced or original sizing component. For spheroidal tensor, h_3' is equal to h_2', $\mathbf{e}_2$ is an arbitrary direction orthogonal to $\mathbf{e}_1$ and $\mathbf{e}_3 = \mathbf{e}_1 \times \mathbf{e}_2$.

3.4 The anisotropic smoothing algorithm

Figure 11 describes the overall algorithm. The input is a threshold mesh gradation measure value γ_0 and a piecewise mesh tensor field defined on vertices of a background structure. The algorithm first traverses edges of the background structure once, processes each edge one by one and collects neighboring edges that need re-checking into a dynamically maintained list (whenever the tensor at a vertex is modified, all edges adjacent to that vertex need re-checking). Then, it repeatedly processes edges in the dynamic list until the list becomes empty. In line 16-20 of figure 11, a tagging process is included to efficiently (in the complexity of O(1)) ensure that edges in the dynamic list are unique.

When processing a specific edge $\mathbf{PQ}$ (line 3-20), the algorithm first identifies the isotropic case by computing aspect ratios and proceeds accordingly. The isotropic case is much simpler to process since no directional consideration is involved. For the anisotropic case, the algorithm first determines direction information as discussed in section 3.2, then loops over each mesh size component associated with a direction (or a plane in spheroidal situation), check and possibly reduce the current size component as discussed in section 3.3.

The small mesh size propagates when repeatedly processing edges of the dynamic list. Since we do not increase any directional size throughout the algorithm, no oscillation occurs during this process and the termination of the propagation is ensured.

Figure 12 depicts a 1D example to demonstrate the propagation. In this example, the background structure is shown by the horizon axis and the black dots, and the piecewise fields are indicated by poly-segments. The original piecewise size field is indicated by the cross symbols, which is 1.0 anywhere except a small size value of 0.2 at $x = 0.0$. The smoothed field that satisfies $\gamma \leq 3.0$ is indicated by circles and the dashed line. It can be seen that the small size propagation from $x = 0.0$ to $x = 0.6$, reducing the size at $x = 0.3$ and $x = 0.6$ to 0.53 and 0.86.

4. EXAMPLES

Three dimensional examples are given in this section to demonstrate the application of the *a priori* anisotropic mesh gradation control algorithm. In each example, an initial tetrahedral mesh goes through refinement and coarsening iterations to match a smoothed tensor field (see [22, 15] for details). The original tensor field is either specified as meshing attributes (the first two examples), or adaptively defined during adaptive simulations (the third example). To make the tensor field interrogation efficient and allow the applica-

```
1   initialize a dynamic edge list
2   Loop over edges of the background structure
3     Let PQ to be the current edge, and M_p, M_q the two mesh tensors
4     if both tensors at P and Q are isotropic
5       process the size at P or Q using the algorithm on page 1150 of ref. [21]
6     else
7       compute anisotropic respect ratio α
8       determine direction information (see section 3.2)
9       loop over size components of M_p and M_q
10        get the direction(s) associated with the current component
11        check and, if required, reduce the size of current component (see section 3.3)
12      if any size component of M_p or M_q has been reduced
13        construct a new mesh tensor
14        replace the original tensor at P or Q with the new one
15        for all edges bounded by the reduced mesh tensor
16          if the current edge has not been tagged being in the dynamic list
17            tag the edge
18            insert the edge into the dynamic edge list
19          remove edge PQ from the list
20          clear the tag
21  process the edges in the dynamic list in the same way until the list is empty
```

Figure 11: Mesh tensor field smoothing algorithm.

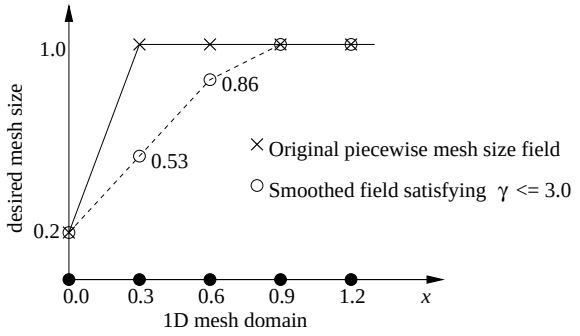


Figure 12: One dimensional example of small mesh size propagation.

tion of the gradation control algorithm, we use the evolving mesh as "background mesh", and represent the original sizing specification as a piecewise field attached to vertices of the evolving mesh. The tensor field attached to the initial mesh is pre-processed by the gradation control algorithm. During the evolution of the "background mesh", the tensor field is locally adjusted to respect the original specification. In particular, when new vertices are created in refinement, we first compute the tensors at these locations in terms of the given meshing attributes, then smooth them using a local version [2] of the mesh gradation control algorithm (the first two examples), or interpolate these tensors based on their neighbors, but reset the tensor field by the error indicator every 3-5 mesh adaptation iterations and re-smooth it (the third example). Except projecting new vertices onto curved boundaries [23], no other vertices are moved in our meshing algorithm to avoid the possible diffusion of the tensor field.

[2] The local algorithm is the same as that in figure 11 except that the input is a list of mesh edges connected to new vertices instead of the whole mesh.

4.1 Planar discontinuities in cubic domain

Figure 13(a) shows an initial tetrahedral mesh (40 tets and 27 vertices) over a $1 \times 1 \times 1$ cubic domain. The original tensor field is specified to have strong jumps at $x = 0.5 \pm 0.01$ and $z = 0.5 \pm 0.01$ as follows:

$$M(x,y,z) = \begin{bmatrix} 1/h_x^2 & 0 & 0 \\ 0 & 1/h_y^2 & 0 \\ 0 & 0 & 1/h_z^2 \end{bmatrix} \tag{13}$$

with

$$h_x = \begin{cases} 0.005 & \text{if } |x-0.5| \le 0.01 \\ 0.25 & \text{otherwise} \end{cases} \tag{14}$$

$$h_y = 0.25 \tag{15}$$

$$h_z = \begin{cases} 0.005 & \text{if } |z-0.5| \le 0.01 \\ 0.25 & \text{otherwise} \end{cases} \tag{16}$$

Figure 13 (b)-(f) show the result meshes conforming to a smoothed tensor field controlled by different γ_0. Figure 14 provides a slice of interior mesh faces and a close-up view to where the two discontinuities meet in the mesh of $\gamma_0 = 2.0$. Table 1 indicates the number of tetrahedra of these conforming meshes. It can be seen that small mesh size only propagates in one direction, *i.e.*, anisotropic features are preserved by the tensor field smoothing process, and the smaller γ_0,

Table 1: γ_0 vs. size of conformed meshes (example 1).

γ_0	1.5	1.75	2	3	4	8
# of tetrahedra	79,427	39,764	29,480	17,594	15,651	13,952
# of vertices	15,077	7,775	5,740	3,535	3,137	2,839

the further the propagation, the more the resulting elements. Also it can be seen that elements become isotropic on xz plane (*i.e.* needle-like in 3D) where two anisotropic features meet.

4.2 Boundary layers in intersected pipes

Figure 15 shows a quarter of two intersected cylinders and a coarse initial mesh consisting of 61 tetrahedra and 35 vertices. The radius of both cylinders is 50mm and the length is 300mm, 400mm respectively. To generate a mesh with boundary layers along the cylindrical surfaces, we specify the tensor field as meshing attributes of the geometry model as follows:

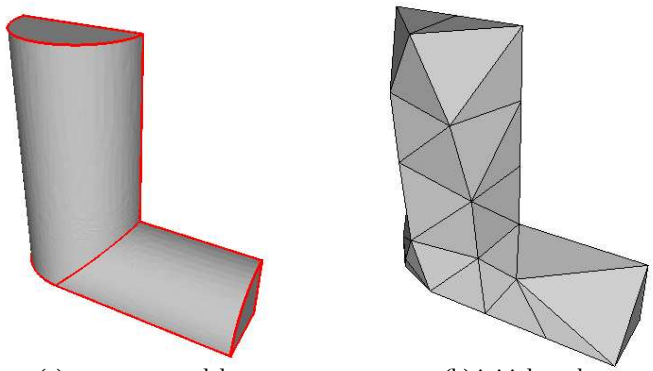
- On both cylindrical surfaces, the desired edge length is 25mm in tangential and axial directions, and 1mm in normal direction, *i.e.*, given any point on cylindrical surface, the tensor at the point is specified as:

$$\left[\begin{array}{ccc} \mathbf{e}_r & \mathbf{e}_\theta & \mathbf{e}_z \end{array}\right] \left[\begin{array}{ccc} 1 & 0 & 0 \\ 0 & 1/25^2 & 0 \\ 0 & 0 & 1/25^2 \end{array}\right] \left[\begin{array}{ccc} \mathbf{e}_r & \mathbf{e}_\theta & \mathbf{e}_z \end{array}\right]^T \tag{17}$$

  where $\mathbf{e}_r$, $\mathbf{e}_\theta$ and $\mathbf{e}_z$ are the base vectors in normal, tangential and axial direction of the cylindrical surface.

- Anywhere else the desired edge length is isotropic and is 25mm.

Figure 16(a)(b) show the mesh conforming to the smoothed tensor field with γ_0=1.6. Boundary vertices are automatically placed onto the geometry boundary during mesh adaptation [23]. Figure 16(c) shows the interior mesh by hiding all tetrahedra in front of the square plane. Figure 16(d)(e) provide two close-ups. One shows details of the boundary layer, while another shows the elements where two boundary layers meet. It can clearly be seen that boundary layers have been generated, propagated inward and smoothly connected to the interior isotropic elements. Also note the element size changing along the intersection curve of the two cylindrical surface in figure 16(a). This is caused by the changing of the relative normal directions between the two cylindrical surfaces. At the bottom where the two cylindrical surfaces are tangent to each other, no size is reduced in tangential and axial directions, while directional element size reductions are applied when the two normal directions are not aligned. Figure 17 shows the tetrahedral meshes conforming to the smoothed field with γ_0=1.25, 2.0 and 3.0. Table 2 indicates the mesh size increase with respect to different γ_0. Again it can be seen that, the depth of the boundary layer is controlled by the specified γ_0 value, and the closer to 1, the further the inward propagation, the more the result elements.



(a) geometry model. (b) initial mesh.

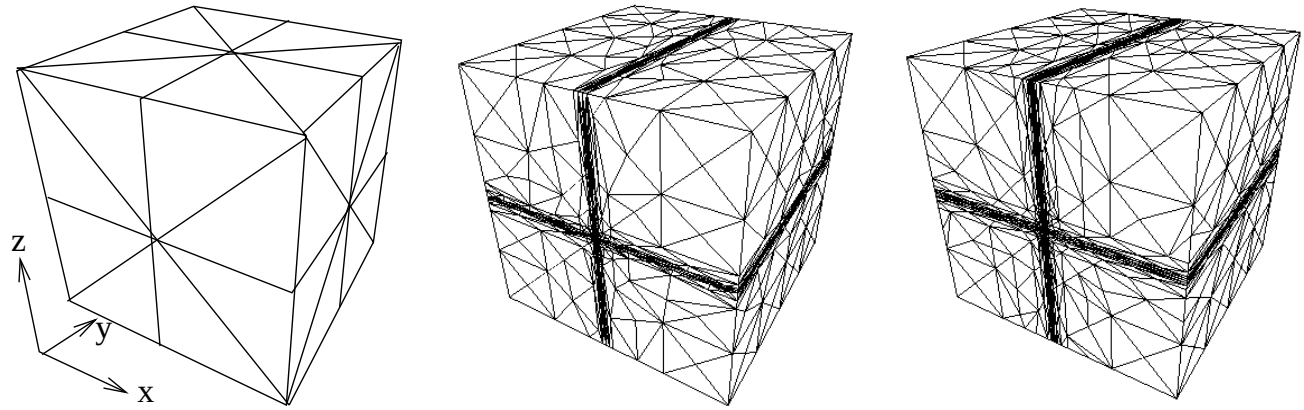
Figure 15: A quarter of two intersected pipes and its initial mesh.

Table 2: γ_0 vs. size of conformed meshes (example 2).

γ_0	1.25	1.6	2	3
# of tetrahedra	341,608	80,263	6,733	3,326
# of vertices	60,820	14,609	35,625	16,731

4.3 Cannon blast simulation

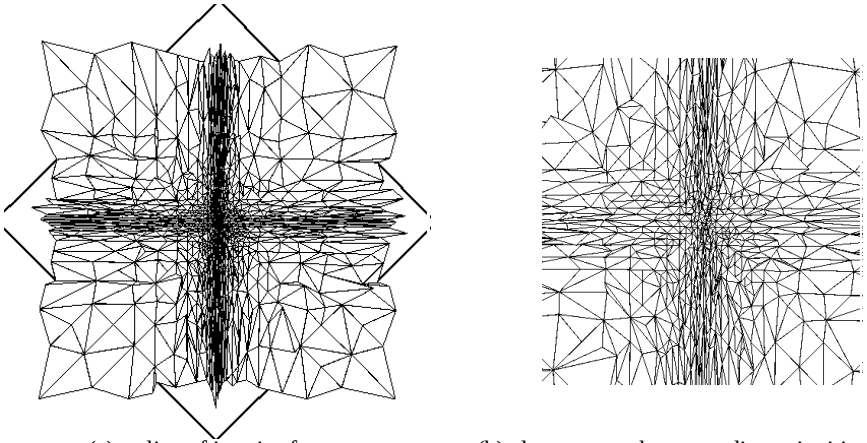
This example shows the application of the *a priori* procedure in 3D adaptive simulation of cannon blast problem governed by Euler's equation. Figure 18 shows a perforated cannon (idealized tube with a hexagonal cross section and with holes) inside a box domain. Figure 19 shows the evolving mesh and density field when the shock inside the cannon passed half of the perforated holes after 700 cycles of solutions and mesh adaptations. Figure 19(a) shows a slice of mesh faces intersecting the cut plane and Figure 19(b) shows the density contour surfaces. Figure 19(c) provides a close-up to the slice mesh faces and 19(d) provides a close-up to density contour near the perforated holes. During the adaptive simulation, anisotropic mesh tensor fields are adaptively specified in terms of the second derivatives of the evolving density field and a discontinuity detect, then smoothed using the anisotropic mesh gradation procedure with γ_0=3.0. Details of this adaptive simulation can be found in reference [2, 24].



(a) initial mesh. (b) $\gamma_0 = 8.0$. (c) $\gamma_0 = 4.0$.

(d) $\gamma_0 = 2.0$. (e) $\gamma_0 = 1.75$. (f) $\gamma_0 = 1.5$.

Figure 13: Initial tetrahedral mesh of cubic domain and conforming tetrahedral meshes.

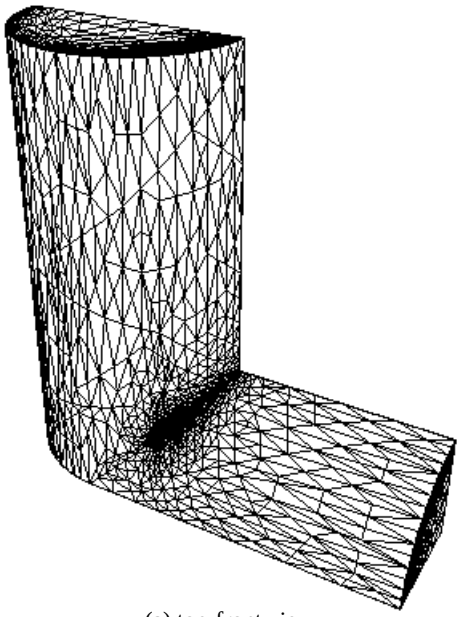


(a) a slice of interior faces. (b) close-up to where two discontinuities meet.

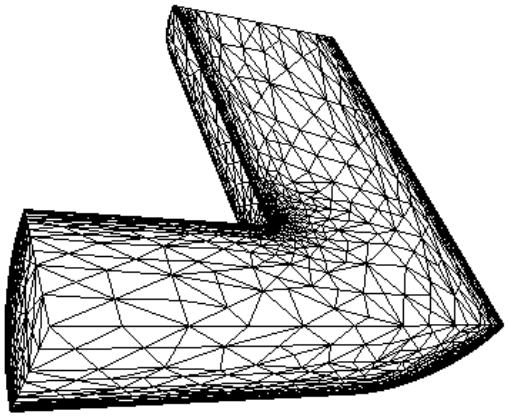
Figure 14: The interior view of a conforming tetrahedral mesh (γ_0=2.0).

5. CONCLUSION

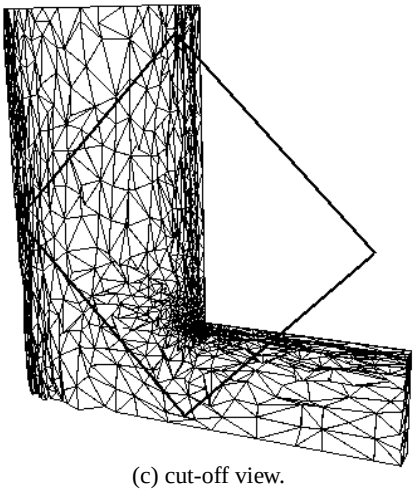
An *a priori* anisotropic mesh gradation algorithm has been proposed in this paper. Examples in three dimensional meshing and adaptive simulations have shown the effectiveness of the algorithm.



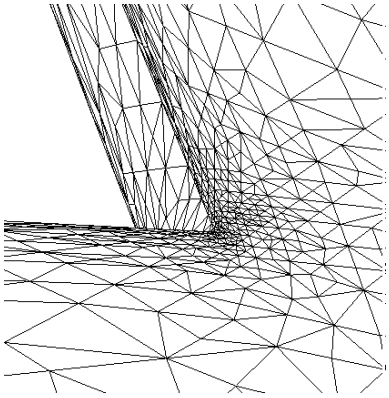
(a) top-front view.



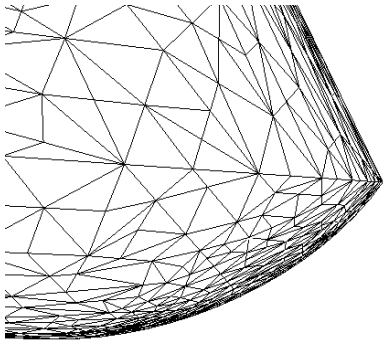
(b) bottom-back view.



(c) cut-off view.



(d) close-up.



(e) close-up.

Figure 16: Tetrahedral mesh conforming to the smoothed field with γ_0=1.6.

References

[1] Jansen K.E., Shephard M.S., Beall M.W. "On Anisotropic mesh generation and quality control in complex flow problems." *Tenth International Meshing Roundtable*. 2001.

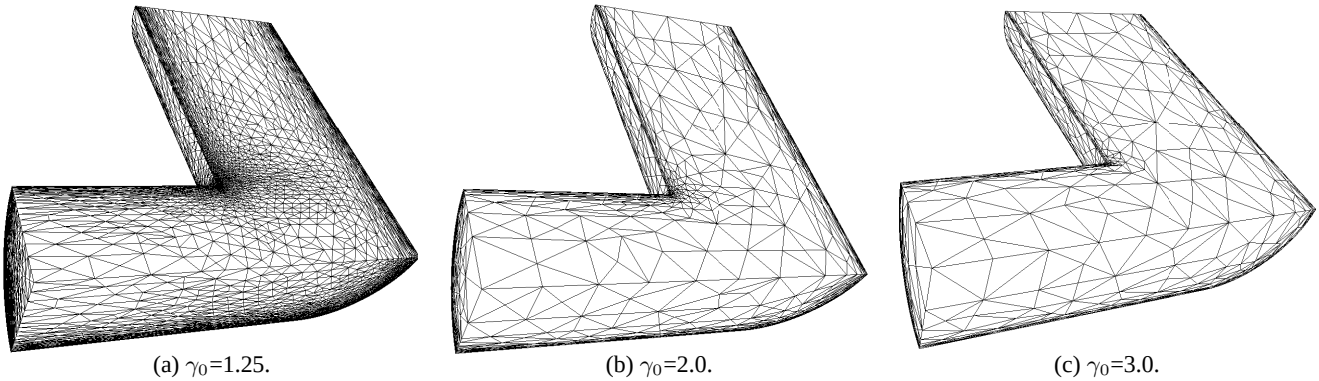
[2] Remacle J.F., Li X., Shephard M.S., Flaherty J.E. "Anisotropic Adaptive Simulation of Transient Flows using Discontinuous Galerkin Methods." *International Journal for Numerical Methods in Engineering*, 2003. Accepted.

[3] Löhner R. "Automatic unstructured grid generators." *Finite Elements in Analysis and Design*, vol. 25, 111–134, 1997.

[4] Borouchaki H., George P., Hecht F., Laug P., Saltel. "Delaunay mesh generation governed by metric specifications - Part I: Algorithms and Part II: Applications." *Finite Elements in Analysis and Design*, vol. 25, 61–83, 85–109, 1997.

[5] Yamakaw S., Shimada K. "High quality anisotropic mesh generation via ellipsoial bubble packing." *Ninth International Meshing Roundtable*. 2000.

[6] Peraire J., Peiro J., Morgan K. "Adaptive remeshing for three dimensional compressible flow computation."



(a) γ_0=1.25. (b) γ_0=2.0. (c) γ_0=3.0.

Figure 17: Conforming tetrahedral meshes at different mesh gradation level.

Journal of Computational Physics, vol. 103, 269–285, 1992.

[7] Frey P.J., George P.L. *Mesh Generation*. HERMES Science Europe Ltd., 2000.

[8] Li L.Y., Bettess P., Bull J.W. "Theoretical formulations for adaptive finite element computations." *Communications in Numerical Methods in Engineering*, vol. 11, 857–868, 1995.

[9] Kunert G. "Toward anisotropic mesh construction and error estimation in the finite element method." *Numerical Meth. in Partial Differential Equations*, vol. 18, 625–648, 2002.

[10] George P.L., Hecht F. "Non isotropic grids." J. Thompson, B.K. Soni, N.P. Weatherill, editors, *CRC Handbook of Grid Generation*, pp. 20.1–20.29. CRC Press, Inc, Boca Raton, 1999.

[11] Gursoy H.N., Patrikalakis N.M. "An automatic coarse and fine surface mesh generation scheme based on MAT Part I: algorithms." *Engineering With Computers*, vol. 8, 121–137, 1992.

[12] Cunha A., Canann A., Saigal S. "Automatic boundary sizing for 2D and 3D meshes." *AMD Trends in Unstructured Mesh Generation, ASME*, vol. 220, 65–72, 1997.

[13] Owen S.J., Saigal S. "Surface mesh sizing control." *International Journal for Numerical Methods in Engineering*, vol. 47, 497–511, 2000.

[14] Löhner R. "Extensions and improvements of the advancing front grid generation technique." *Communications in Numerical Methods in Engineering*, vol. 12, 683–702, 1996.

[15] Li X. *Mesh Modification Procedures for General 3-D Non-manifold Domains*. Ph.D. thesis, Rensselear Polytechnic Institute, August, 2003.

[16] Labbe P., Dompierre J., Vallet M.G., Guibault F., Trepanier J.Y. "A measure of the conformity of a mesh to an anisotropc metric." *Tenth International Meshing Roundtable*. 2001.

[17] Weatherill N.P., Hassan O. "Efficient three-dimensional Delaunay triangulation with automatic point creation and imposed boundary constraints." *International Journal for Numerical Methods in Engineering*, vol. 37, 2005–2039, 1994.

[18] Garimella R.V., Shephard M.S. "Boundary layer mesh generation for viscous flow simulations." *International Journal for Numerical Methods in Engineering*, vol. 49, 193–218, 2000.

[19] Zhu J., Blacker T., Smith R. "Background overlay grid size functions." *Eleventh International Meshing Roundtable*. 2002.

[20] Löhner R. "Adaptive remeshing for transient problems." *Computer Methods in Applied Mechanics and Engineering*, vol. 75, 195–214, 1989.

[21] Borouchaki H., Hecht F., Frey P.J. "Mesh gradation control." *International Journal for Numerical Methods in Engineering*, vol. 43, no. 6, 1143–1165, 1998.

[22] Li X., Shephard M., Beall M. "3-D anisotropic mesh adaptation by mesh modifications." *Computer Methods in Applied Mechanics and Engineering*, 2003. Submitted.

[23] Li X., Shephard M., Beall M. "Accounting for curved domains in mesh adaptation." *International Journal for Numerical Methods in Engineering*, vol. 58, 247–276, 2003.

[24] Chevaugeon N., Hu P., Li X., Xin S., Flaherty J.E., Shephard M.S. "Application of discontinuous Galerkin methods applied to shock and blast problems." 2003. In preparation.

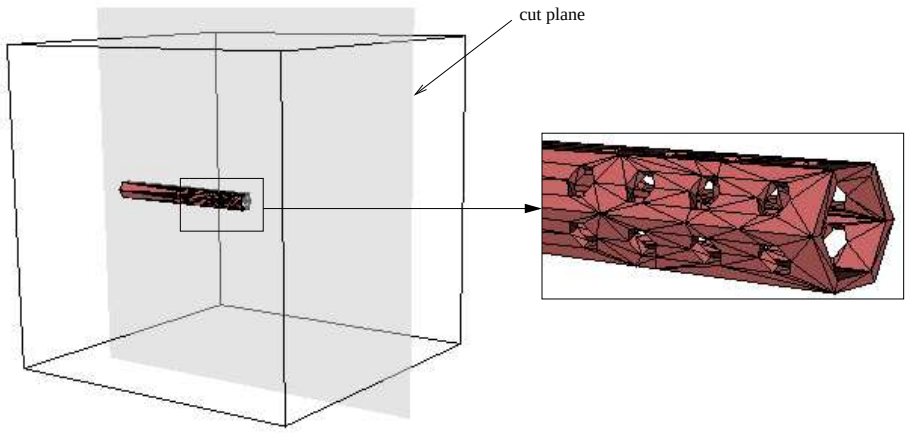
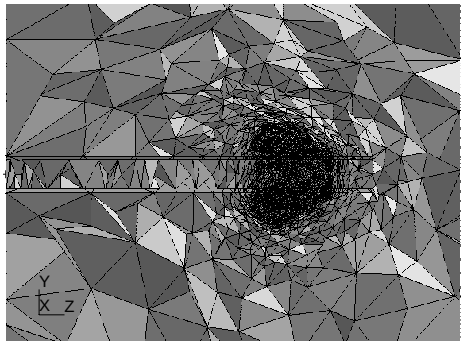
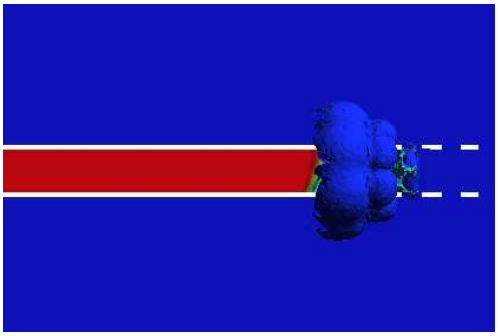


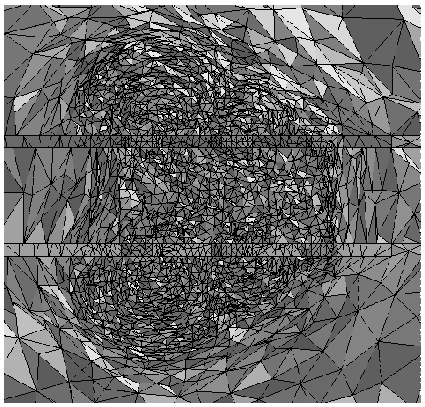
Figure 18: Analysis domain: a cannon with 24 perforated holes inside a box.



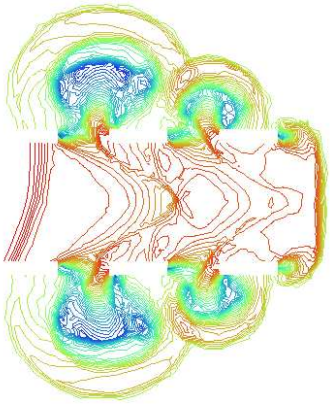
(a) a slice of the tetrahedral mesh intersecting the cut plane.



(b) contour surfaces of the density field.



(a) mesh close-up.



(b) density contour on cut plane.

Figure 19: Result mesh and density distribution after 700 adaptive cycles.