

ANISOTROPIC MESH GRADATION CONTROL *

Xiangrong Li^{1†}

Jean-François Remacle²

Nicolas Chevaugeron²

Mark S. Shephard¹

¹Scientific Computation Research Center, CII-7011, 110 8th Street, Rensselaer Polytechnic Institute, Troy, NY 12180-3590, U.S.A.

²Department 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 criterion 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 regular. However, one technical issue related to the field specification and its conformity criterion remains

not fully solved. In particular, due to the complexity and variety in both geometry and physics, the defined field 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 mesh 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, which can be the mesh for previous solution [6, 4], the evolving mesh [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 neighboring size and shape information.

*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.

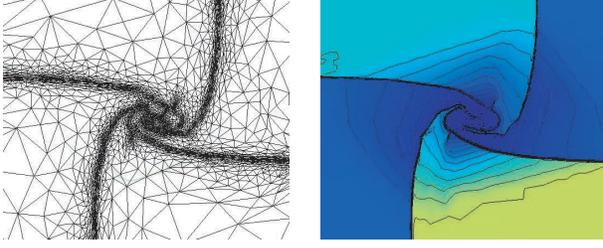


Figure 1: Graded anisotropic mesh (left) that captures evolving discontinuous solution field (right) in solving a four contact Riemann problem.

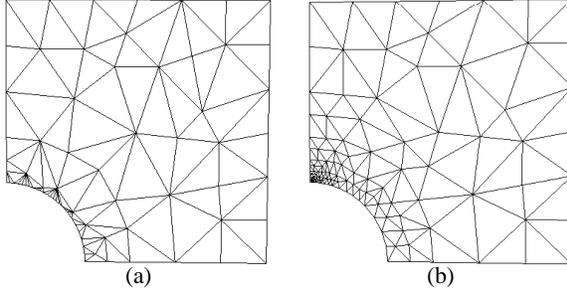


Figure 2: Two dimensional example to illustrate the need for mesh gradation control. (a) Poorly shaped mesh conforming a given sizing function. (b) Graded mesh conforming to a modified sizing function.

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 represent 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 interpolation. 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 geometric significance of a directional mesh gradation measure are given. In section 3, we present a three dimensional *a priori* procedure

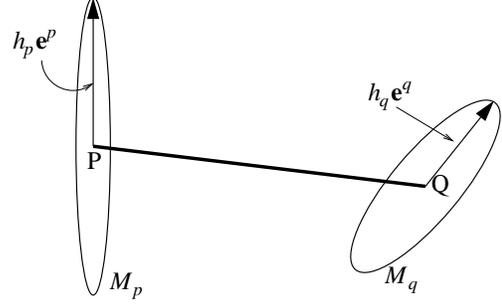


Figure 3: Definition of directional mesh gradation measure γ . Mesh tensors are indicated by ellipses.

that preserves element size and anisotropy. Section 4 provides three dimensional example meshes to show how the *a priori* mesh gradation control is accomplished.

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 \mathbf{e}^i be a unitary direction vector associated with M_i (see figure 3). The mesh size gradation measure related to point P and Q between direction pair $(\mathbf{e}^p, \mathbf{e}^q)$ is:

$$\gamma(\mathbf{e}^p, \mathbf{e}^q) = e^{\frac{|h_p(\mathbf{e}^p) - h_q(\mathbf{e}^q)|}{L_{pq}}} \quad (1)$$

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

$$h_i(\mathbf{e}^i) = \frac{1}{\sqrt{\mathbf{e}^i M_i \mathbf{e}^{iT}}} \quad (2)$$

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 (see figure 4(a)). h_p and h_q are the desired mesh edge length along \mathbf{PQ} computed in terms of equation (2) (*i.e.* \mathbf{e}^p and \mathbf{e}^q are parallel to \mathbf{PQ}), and $|x_a - x_p|$ and $|x_b - x_a|$ are the length of edge \mathbf{PA} and \mathbf{AB} . 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, the desired edge length along \mathbf{PQ} at

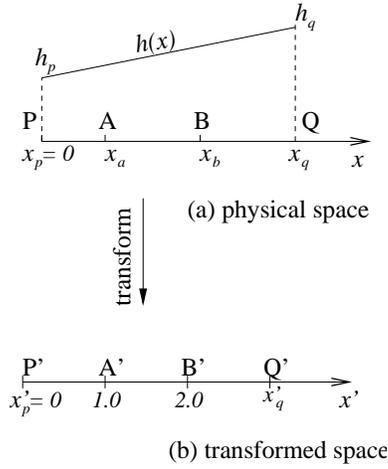


Figure 4: Illustration to the significance of γ .

position x is:

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

Let x be the coordinate, 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:

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

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 **PA** and **AB** can be derived:

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

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

Thus the ratio of two neighboring edges is

$$\gamma(\vec{PQ}, \vec{PQ}) = \frac{|x_b - x_a|}{|x_a - x_p|} = e^{\frac{h_q-h_p}{L_{pq}}} \quad (7)$$

which is the definition for γ . Therefore, a mesh tensor field with $\gamma = 1$ in all directions should be a constant field. A mesh satisfying the tensor field of $\gamma_x = 2$ (γ_x is the γ between x axes at two points) should consist of edges in length series: $l_0, 2l_0, 4l_0, \dots, 2^n l_0$ (l_0 is the length of an edge on x axis). Figure 5 shows a 2D mesh of a 1×1 domain approximately satisfying a mesh tensor field with $\gamma_x = 1$ and $\gamma_y = 1.24$ between two points along a horizontal line. It

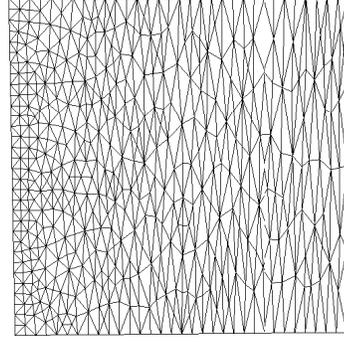


Figure 5: A 2D mesh satisfying a mesh tensor field of $\gamma_x = 1$, $\gamma_y = 1.24$ between two horizontal points.

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.

The direction pair can be determined in terms of the eigenvectors of the given mesh tensors at two considered points. We identify three situations: if both mesh tensors are isotropic (three identical eigenvalues), both are degenerated into a scalar and the computation of h_i ($i = p, q$) and γ is independent of direction pairs. If both mesh tensors have two identical eigenvalues, their geometric shapes are spheroids, thus the two polar directions (the direction associated with the different eigenvalue) consist of a pair. If any tensor has three different eigenvalues, all eigenvectors are respected, and three direction pairs have to be properly determined (see section 3).

In case isotropic, this definition of γ is consistent with the H-shock introduced in reference [21]. Since, in terms of equation (4), the measuring length of edge **PQ** in the *transformed* space is:

$$L'_{pq} = |x'_q - x'_p| = \frac{L_{pq}}{h_q - h_p} \ln\left(\frac{h_q}{h_p}\right) \quad (8)$$

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

$$\gamma = e^{\ln(\frac{h_q}{h_p})/L'_{pq}} = (h_q/h_p)^{\frac{1}{L'_{pq}}} \quad (9)$$

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.

It should be noted that this measure just describes the smoothness property of the mesh tensor field, and does not ensure if there is enough geometric space to create the desired mesh, which should be determined using equation (8).

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 directionality and smaller size.

A mesh tensor can be modified by changing its principle direction 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 ratio¹ is close to a low aspect ratio tensor, the directions of the higher aspect ratio tensor are preserved and the direction(s) 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 axes 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 smooth mesh size variation.

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

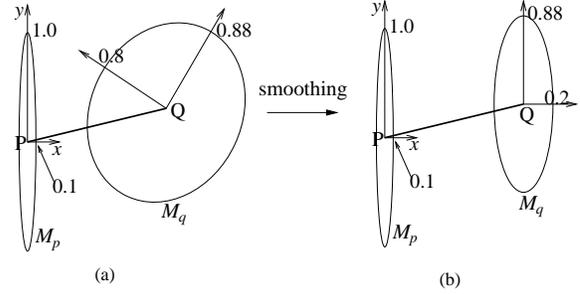


Figure 6: A 2D example to illustrate the need for the adjustment of both direction and size.

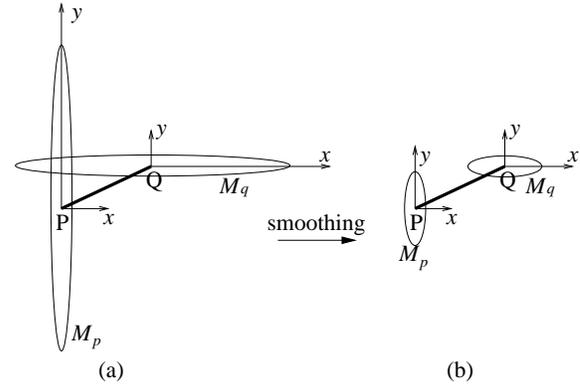


Figure 7: A 2D example of preserving direction and reducing size.

The subsections that follow are organized as follows: Section 3.2 discusses the adjustment of principle directions. Section 3.3 presents the method for directional larger mesh size reduction. Section 3.4 presents the overall algorithm.

3.2 Selection of directions

Consider the two mesh tensors attached onto the end vertices of edge 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:

$$\alpha = \frac{(R_q - 1) R_p}{(R_p - 1) R_q} \quad (10)$$

Equation (10) has been defined such that α is a value in interval $[0, 1]$ 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 mesh tensor’s

Table 1: Selection of eigenvector pairs ($R_p \geq R_q$).

Case	M_p	M_q	Selection of eigenvector pair(s)
1	any	sphere	no needs
2	spheroid	spheroid	a pair of polar directions
3	ellipsoid	spheroid	three pairs (see figure 9)
4	ellipsoid	ellipsoid	three pairs (see figure 8)

eigenvectors. Note that, $\alpha = \frac{0}{0}$ (when both mesh tensors are isotropic) does not cause a problem since computing α is unnecessary.

Equation (11) gives the formula to adjust the 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 case tensor M_q is isotropic, simply set its eigenvectors the same as these of M_p .

$$\mathbf{e}_j^q|_{new} = (1 - \alpha) \mathbf{e}_i^p + \alpha \mathbf{e}_j^q \quad (11)$$

Table 1 lists the selection of \mathbf{e}_i^p and \mathbf{e}_j^q based on geometry shapes of the two mesh tensors in the application of equation (11). If one of the two mesh tensors is spherical (case 1), the selection of eigenvector pair is not needed. If both mesh tensors are spheroidal (case 2), we assume that M_q remains spheroidal after the direction adjustment, thus the two polar directions should match. In case 3 and case 4, we select eigenvector pairs by minimizing the maximal angle between eigenvectors. Note that the angle between the two eigenvectors should be in interval $[0, \pi/2]$ since the desired mesh size along a direction is the same as that in its opposite direction.

Consider figure 8 for the eigenvector pair determination in case 4, where \mathbf{e}_i^p and \mathbf{e}_j^q ($i=1,2$) represent the eigenvectors 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^q . Since $\beta_2 < \beta_1$ in this setting, \mathbf{e}_1^p matches \mathbf{e}_2^q . The two remaining directions, \mathbf{e}_2^p and \mathbf{e}_1^q , are related obviously. If ambiguous situation where $\beta_1 = \beta_2$ occurs, we simply match \mathbf{e}_1^p with either directions.

Figure 9 illustrates the eigenvector pair selection for case 3, where M_q has two identical eigenvalue thus a polar direction \mathbf{e}_1^q while M_p has three different eigenvalues thus three principle direction $\mathbf{e}_1^p, \mathbf{e}_2^p, \mathbf{e}_3^p$. The three dashed vectors indicate the eigenvectors of M_p originated at point Q . To determine eigenvector pairs, we first match \mathbf{e}_1^q with one of the eigenvectors of M_p (\mathbf{e}_1^p in the example since the angle between is the smallest). Then, eigenvector \mathbf{e}_i^q ($i=2,3$) is obtained by projecting \mathbf{e}_i^p onto the equatorial plane of M_q which minimizes the maximal angle.

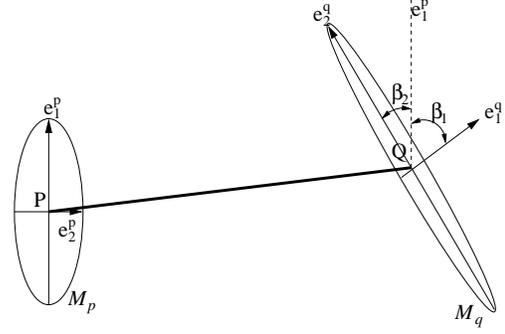


Figure 8: 2D example of eigenvector pair selection between tensor M_p and M_q .

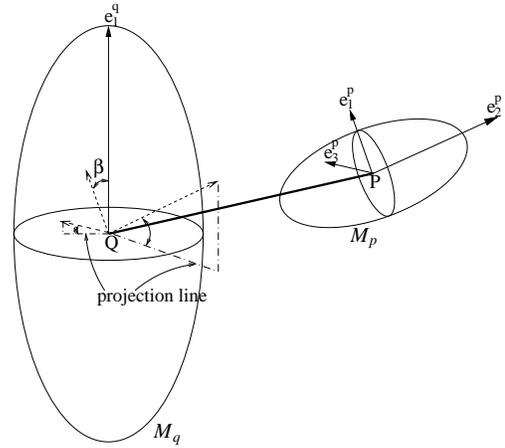


Figure 9: Eigenvector pair selection for one spheroid one ellipsoid case.

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. This section discusses the algorithm to check 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).

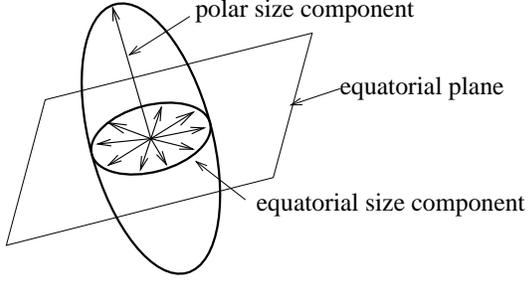


Figure 10: Illustration to polar and equatorial sizing components of a spheroidal mesh tensor.

- if $\gamma > \gamma_0$ and $h_i^p > h_i^q$ (γ_0 is the prescribed threshold value), reduce h_i^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 shapes of M_p and M_q are spheroidal, the number of sizing components is degenerated into two: polar component and equatorial component (see figure 10). The equatorial components represents the desired size in any direction orthogonal to the polar direction. To make the reduced tensor remain spheroidal, the polar and equatorial sizing component should match respectively. (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 (11) and compute h_i^q using equation (2).

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

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

Therefore

$$h_i'^p = L_{pq} \ln(\gamma_0) + h_i^q \quad (12)$$

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 is constructed as follows:

$$\begin{bmatrix} \mathbf{e}_1 & \mathbf{e}_2 & \mathbf{e}_3 \end{bmatrix} \begin{bmatrix} 1/h_1'^2 & 0 & 0 \\ 0 & 1/h_2'^2 & 0 \\ 0 & 0 & 1/h_3'^2 \end{bmatrix} \begin{bmatrix} \mathbf{e}_1 & \mathbf{e}_2 & \mathbf{e}_3 \end{bmatrix}^T \quad (13)$$

where \mathbf{e}_i ($i=1,2,3$) is the original eigenvector of M_p , or the adjusted one given by equation (11), 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 $\mathbf{e}_3 = \mathbf{e}_1 \times \mathbf{e}_2$.

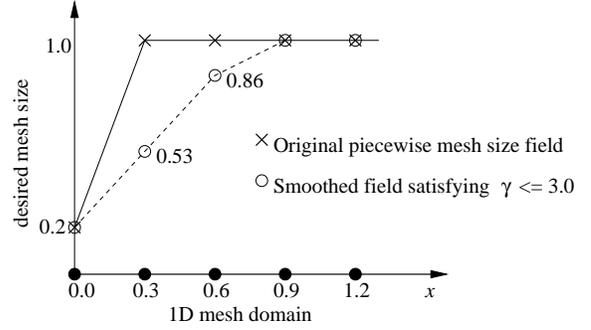


Figure 12: One dimensional example of small mesh size propagation.

3.4 The anisotropic smoothing algorithm

Figure 11 describes the overall algorithm. The input is a threshold 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 **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 an equatorial plane), checks and possibly reduces 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.

```

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  $\alpha$ 
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 attached onto PQ
21 process the edges in the dynamic list in the same way until the list is empty

```

Figure 11: Mesh tensor field smoothing algorithm.

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 application 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.

²The 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:

$$M(x, y, z) = \begin{bmatrix} 1/h_x^2 & 0 & 0 \\ 0 & 1/h_y^2 & 0 \\ 0 & 0 & 1/h_z^2 \end{bmatrix} \quad (14)$$

with

$$h_x = \begin{cases} 0.005 & \text{if } |x - 0.5| \leq 0.01 \\ 0.25 & \text{otherwise} \end{cases} \quad (15)$$

$$h_y = 0.25 \quad (16)$$

$$h_z = \begin{cases} 0.005 & \text{if } |z - 0.5| \leq 0.01 \\ 0.25 & \text{otherwise} \end{cases} \quad (17)$$

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 2 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 , 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.

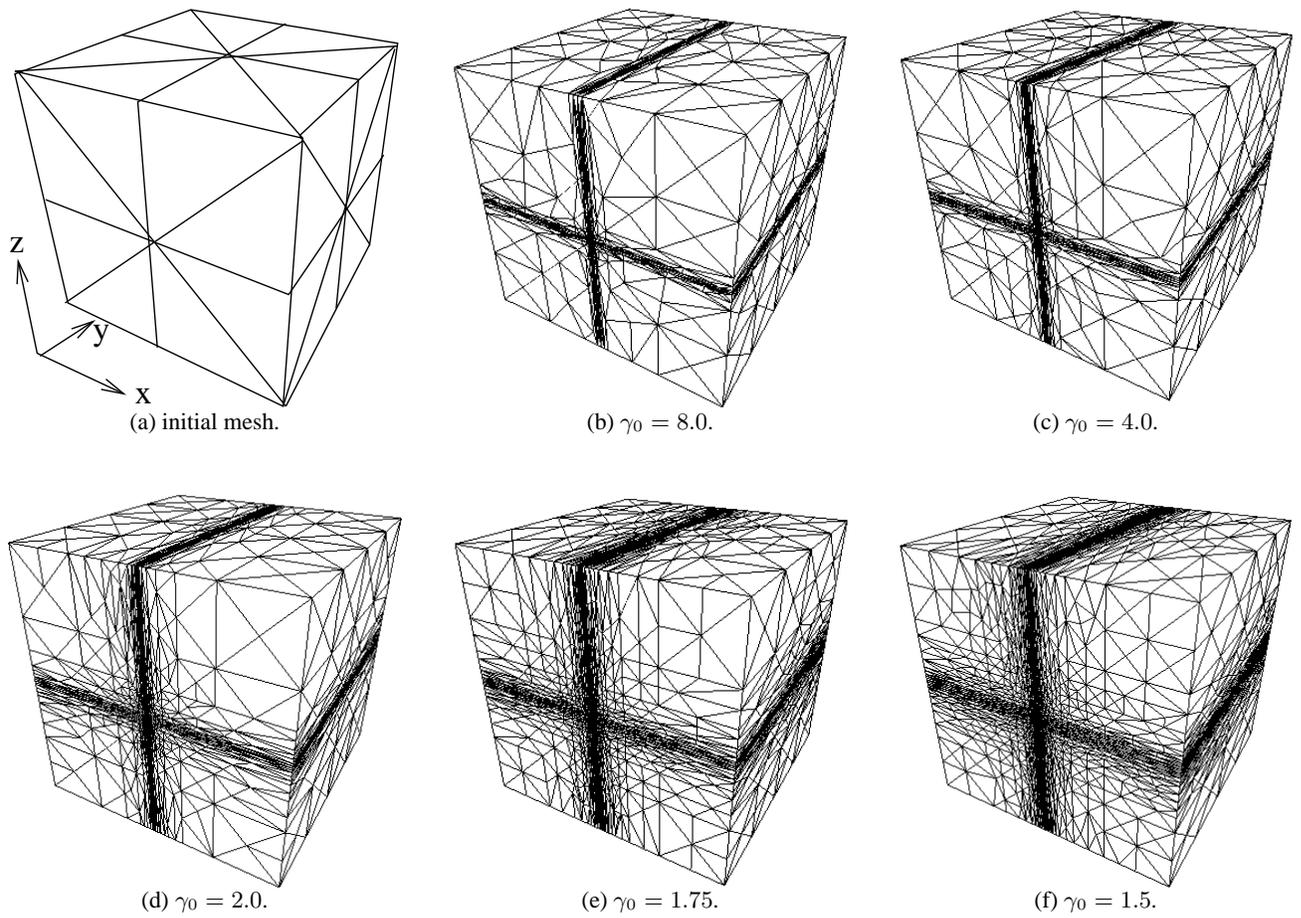


Figure 13: Initial tetrahedral mesh of cubic domain and conforming tetrahedral meshes.

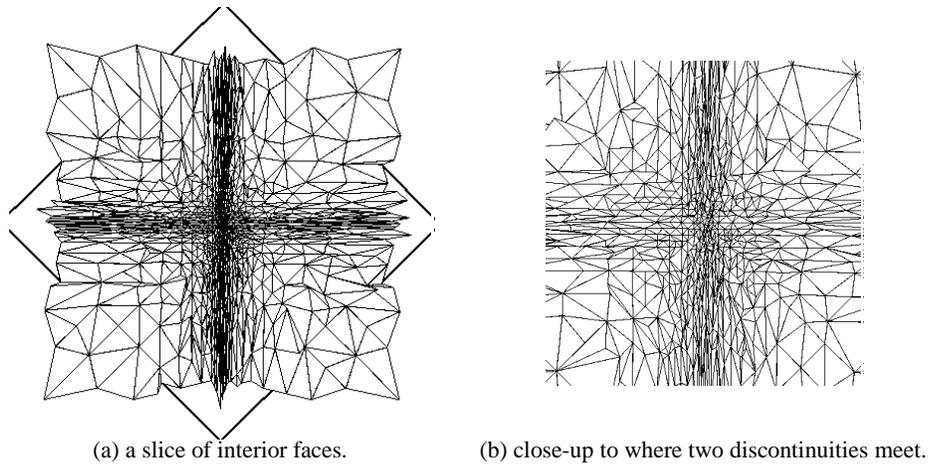


Figure 14: The interior view of a conforming tetrahedral mesh ($\gamma_0=2.0$).

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 ver-

tices. 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

Table 2: 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

Table 3: 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

as follows:

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

$$\begin{bmatrix} \mathbf{e}_r & \mathbf{e}_\theta & \mathbf{e}_z \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1/25^2 & 0 \\ 0 & 0 & 1/25^2 \end{bmatrix} \begin{bmatrix} \mathbf{e}_r & \mathbf{e}_\theta & \mathbf{e}_z \end{bmatrix}^T \quad (18)$$

where \mathbf{e}_r , \mathbf{e}_θ 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 $\gamma_0=1.6$. 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, showing details of the boundary layers and 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 $\gamma_0=1.25, 2.0$ and 3.0 . Table 3 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.

4.3 Cannon blast simulation

This example shows the application of the *a priori* procedure in 3D adaptive simulation of cannon blast problem gov-

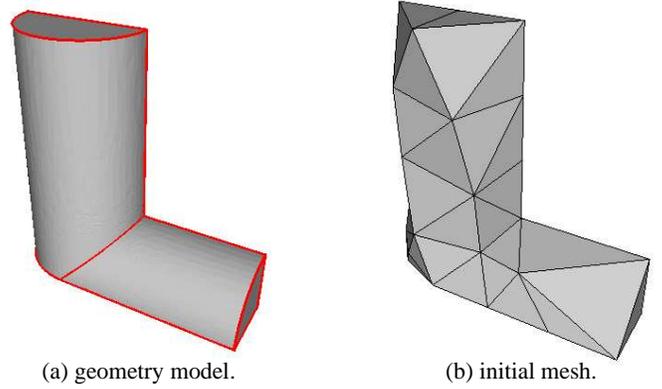


Figure 15: A quarter of two intersected pipes and its initial mesh.

erned 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 $\gamma_0=3.0$. Details of this adaptive simulation can be found in reference [2, 24].

5. CONCLUSION

This paper provides a straightforward way for the *a priori* control of anisotropic mesh gradation, which may smooth the variation of both eigenvalues and eigenvectors of a mesh tensor field. Examples in three dimensional meshing and adaptive simulations have shown how “abrupt” element size and shape may be specified and how the smoothing is effectively accomplished.

With this *a priori* mesh gradation control, probably it is useful to include the directional field of γ_0 as a part of the meshing attribute in generating desired meshes for fluid problems. Another possible extension is to determine the relationship between prescribed γ_0 value and the adapted mesh quality.

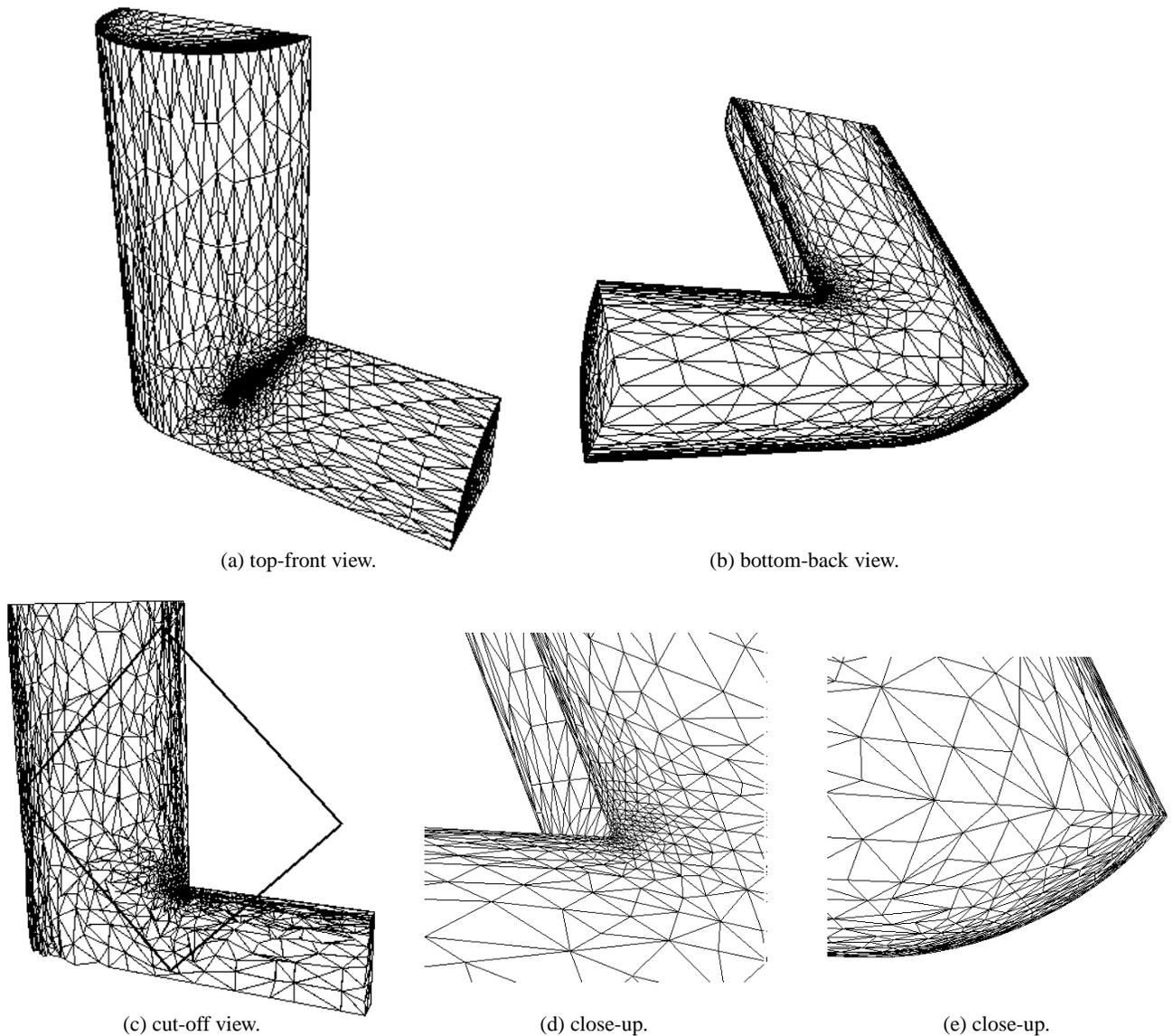


Figure 16: Tetrahedral mesh conforming to the smoothed field with $\gamma_0=1.6$.

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. In press.
- [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] Yamakawa S., Shimada K. "High quality anisotropic mesh generation via ellipsoidal bubble packing." *Ninth International Meshing Roundtable*. 2000.
- [6] Peraire J., Peiro J., Morgan K. "Adaptive remeshing for three dimensional compressible flow computation."

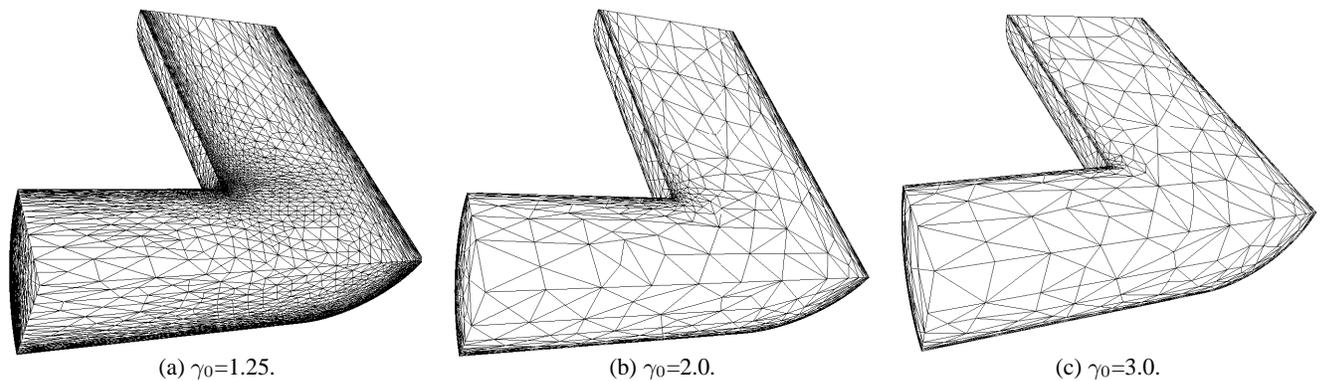


Figure 17: Conforming tetrahedral meshes at different mesh gradation level.

- 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] Guroy 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, Rensselaer 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 anisotropic 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.S., Beall M.W. “3-D anisotropic mesh adaptation by mesh modifications.” *Computer Methods in Applied Mechanics and Engineering*, 2004. Accepted.
- [23] Li X., Shephard M.S., Beall M.W. “Accounting for curved domains in mesh adaptation.” *International Journal for Numerical Methods in Engineering*, vol. 58, 247–276, 2003.
- [24] Chevaugnon 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.

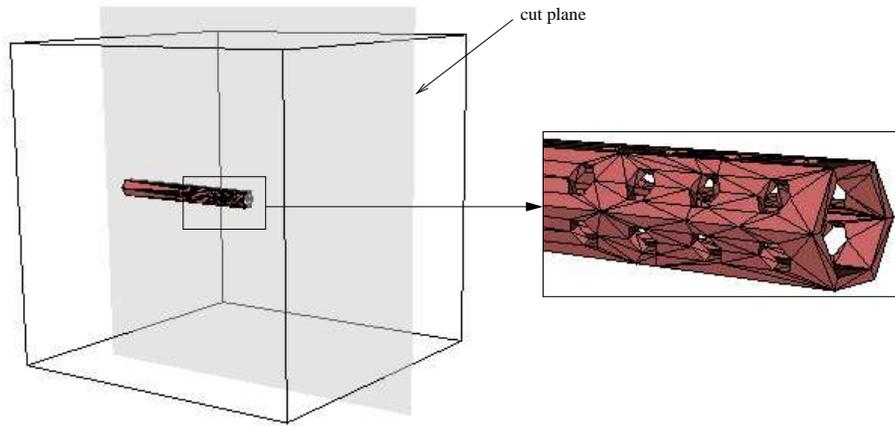
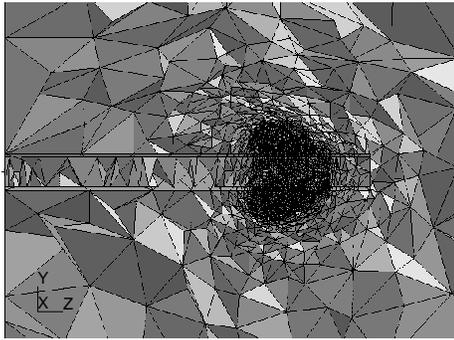
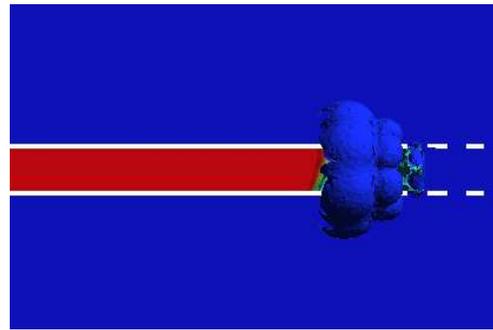


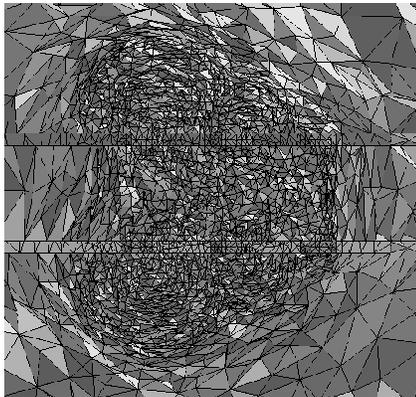
Figure 18: Analysis domain: a cannon with 24 perforated holes inside a box.



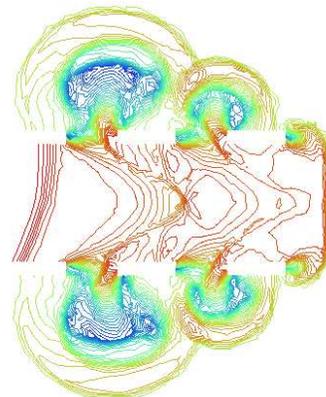
(a) a slice of the tetrahedral mesh intersecting the cut plane.



(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.