Automatic coarsening of three dimensional anisotropic unstructured meshes for multigrid applications

Youssef Mesri, Hervé Guillard, Thierry Coupez

Automatic coarsening of three dimensional unstructured meshes for efficient reservoir simulations

Youssef Mesri

IFP, 1 et 4 av. de Bois-Preau, B.P. 311, 92852 Rueil-Malmaison Cedex, France

Hervé Guillard

Inria, B.P. 93, 2004 Route des Lucioles, 06902 Sophia-Antipolis Cedex, France and Laboratoire Jean-Alexandre Dieudonné, University of Nice Sophia-Antipolis, Parc Valrose 06108 Nice Cedex, France

Thierry Coupez

CEMEF. Ecole des Mines de Paris. CNRS UMR 7635. B.P. 207, 06904 Sophia-Antipolis Cedex, France.

Abstract

This paper describes an algorithm designed for the automatic coarsening of three-dimensional unstructured simplicial meshes. This algorithm can handle very anisotropic meshes like the ones typically used to capture the boundary layers in CFD with Low Reynolds turbulence modeling that can have aspect ratio as high as 10^4. It is based on the concept of mesh generation governed by metrics and on the use of a natural metric mapping the initial (fine) mesh into an equilateral one. The paper discusses and compares several ways to define node based metric from element based metric. Then the semi-coarsening algorithm is described. Several application examples are presented, including a full three-dimensional complex model of an aircraft with extremely high anisotropy.

Keywords:
Mesh generation, Tetrahedral meshes, Coarsening, Anisotropy, Multigrid

Email addresses: youssef.mesri@ifpen.fr (Youssef Mesri), herve.guillard@inria.fr (Hervé Guillard), thierry.coupez@mines-paristech.fr (Thierry Coupez)

Algorithms, CFD, Boundary layers.

---

1. INTRODUCTION

In recent years upscaling has become increasingly important for converting highly detailed geological models to computional grids. These geological models usually require fine-scale descriptions of reservoir porosity and permeability on grids of tens of millions of cells to honor the known and inferred statistics of these reservoir properties. The goelogical grids The geological grids of this order are far too fine to be used as simulation grids. Even with today's computing power, most of the full-field reservoir models are of the order of 100000 a factor 100 less than the geological grids. Upscaling has been developed to bridge the gap between these two scales. Given a fine reservoir description scale and a simulation grid, an *upscaling algorithm* is designed to obtain suitable values for the porosity, permeability, and other property data for use in the coarse grid simulation. Many upscaling methods have been devloped, such as pressure-solver [35], renommalization [42], effective medium [42], power law averaging [44] harmonic/arithmetic mean, local averaging, and homogenization [37]; see the reviews of upscaling and pseudoization techniques by Christie [38] and Barker and Thibeau [36], for example. Here, we briefly mention a few of these techniques.

- For the single phase flow, the aim of upscaling is to preserve the gross features of flow on the simulation grid. An algorithm is needed to compute an *effective permeability*, which will result in the same total flow of the fluid through the coarse homogeneous grid as that obtained from the fine heterogeneous grid. In the *pressure-solver method* [35] for example, we set up a single phase flow computation with specific boundary conditions and then ask what value of effective permeability generate the same flow rate as the fine-scale computation. The results obtained depend on the assumptions made, particularly with regard to the boundary conditions. If no-flow boundary conditions are used, a diagonal effective permeability tensor can be derived and entered directly into a reservoir simulator. Alternatively, if periodic boundary conditions are employed, a full effective permeability tensor can be obtained [45].

- For two-phase flow, it is generally believed that upscaling of the absolute permeability alone is not enough to capture the effects of het-

erogeneous on two-phase fluid simulation [43, 40], particularly when the correlation length of the heterogeneity not represented on the flow simulation grid is significant compared with the well spacing. A multiphase upscaling technique must be used. The most obvious technique is the use of *pseudorelative permeabilities*, i.e., pseudos [41]. The role of pseudorelative permeabilities is to determine the flow rate of each fluid phase out of a gridblock. They relate the flow rate to the pressure gradients between the gridblock and its neighbors, given the average saturation in each gridblock. Both the flow rate and the pressure gradient depend on the details of the saturation distribution within the gridblock. Hence, to obtain a pseudorelative permeability curve, it is necessary to determine the saturation distribution within the block for any given average saturation [36]; see the review papers by Christie [38] and Barker and Thibeau [36] for the generation of pseudorelative permeabilities.

A major limitation in upscaling is that it usually gives an answer whitout any indication of whether the assumptions made in th obtaining the answer hold. No rigorous theory exists behind the upscaling process. Furthermore, some factors give rise to a concern about whether the upscaled values are good approximations; these include large-aspect ratio gridblocks, significant transport at an angle to the grid lines, and upscaled gridblocks close in size to a correlation length of a heterogeneous reservoir. Compared with single phase upscaling, multiphase upscaling is far less developed and understood.

Multigrid algorithms are among the most efficient methods to solve linear and non-linear algebraic problems in computational sciences. However, when non-structured simplicial meshes (triangles in 2D, tetrahedra in 3D) are used, these algorithms suffer from the necessity of constructing a sequence of meshes of different resolution of the same geometry. Two main strategies have been proposed to overcome this problem. The first family of methods uses an algebraic interpretation of the problems and relies only on the graph of the matricial problem corresponding to the PDE's that have to be solved. These methods that present several variations e.g. [32, 34] are extremely efficient for linear problems. However for non-linear ones, their utilisation is not obvious. The second family of methods are the so-called geometric multigrid methods where a sequence of real meshes of different mesh sizes are generated. These methods can be used for linear as well as non-linear problems. Geometrical methods can be classified into several families:

- Embedded grids: The simplest manner to build a hierarchy of embedded grids consists in first choosing a coarse grid and then to generate the finer level by element subdivision. Unfortunately, this approach has serious limitations. Indeed, as finer and finer grids are built, they become to some extent less and less unstructured since artificial "macro-element" corresponding to the coarse mesh elements are present in the fine triangulation. These macro-elements may influence the computation and show themselves as artificial internal boundaries in the results [6]. In three dimensions, another potential drawback of this method appears as the successive refinements of tetraedra produce a sequence of elements with deteriorating shape quality. Moreover, with this technique, the end user has a limited control on the final fine mesh.

- Volume agglomeration: Another idea closely related to algebraic methods is to use the volume agglomeration technique. In this approach, the discretisation on the fine grid is associated with control volume defined on a dual mesh. Coarser finite volume grids can then be built by agglomeration of the cells; different algorithms to realize this task are available(see e.g. [12, 23, 22]). The volume agglomeration method is certainly one of the simpler and most efficient to deal with non-structured meshes for first-order differential problems. A weakness of this approach lies in the difficulties to build consistent approximations for second-order differential operators on the coarse dual grids. Some solutions have been proposed [20, 18] but they still have to be fully validated for three dimensional real life problems.

- Non-nested coarse grid methods : An alternative to the agglomeration technique consists in building automatically a hierarchy of coarse grids whose elements are not embedded. In 2-D, the first algorithm proposed for this task was in [16]. It uses a Delaunay retriangulation associated with a MIS (Maximal Independent Subset) algorithm to delete nodes of the fine triangulation. Similar algorithms aimed at the same task have been proposed [8],[26, 27], [1]. The algorithm of [16] has been followed by 3-D extensions on non-structured isotropic tetrahedral meshes that proved to be very effective for inviscid steady computations [6] and other algorithms using closely related idea have also been proposed [29, 30].

However, to the best of our knowledge, none of these geometrical algorithms are fully satisfactory when the fine meshes are highly anisotropic as is the case for instance for the meshes used for Navier-Stokes computations with Low Reynolds Turbulence modeling. The goal of the present paper is thus to extend coarsening algorithms to this type of meshes. For the meshes we will consider in this work, the maximum aspect ratio of the elements can be very large (typically, more than 10^4 for boundary layers problems) and the multigrid smoothers are quite inefficient for these cases. The remedy that is known since a long time is to use semi-coarsening in the direction perpendicular to the maximal stretching.

For non-structured grids, this is not an easy task as one has to devise an automatic way to define this direction and then to coarsen the mesh in this direction only. We will realize this task by using the concept of mesh generation controlled by a metric map. Many authors (for example [9], [3],[4]) showed that the definition of a metric field simplifies the generation of adapted and anisotropic meshes : The metrics associated to the Riemannian space specify the size of the mesh and the direction of the stretching and then an adapted and anisotropic mesh in the Euclidean space can be represented as an isotropic and unitary mesh in the Riemannian space.

For semi-coarsening of non-structured meshes, the algorithm that we propose is therefore divided into two steps :

The first one is an analysis step where the current (fine) mesh is analysed to identify the directions and sizes of stretching. This is done by computing on each element, the metric tensor where this element is equilateral. Then, these tensors are averaged in such a way that on each node of the triangulation, a metric tensor representing the averaged sizes and stretching directions of the elements that contain this node is defined. Then, the eigenvalues of this metric tensor are modified in such a way that the edge lengths of the equilateral element for this tensor is twice the original edge length in the direction of minimal stretching.

The second step is a mesh generation step. We use the fine mesh and the metric field defined on its node as the background mesh and the size specification field as inputs of any existing mesh generation software using the concept of mesh generation governed by metric specifications. In this work,

we have used the MTC software [9, 15] but in principle any other mesh generation (Delaunay, frontal, etc) tool can be used for this purpose provided that it can handle anisotropic elements.

The remainder of this paper is organized as follows : Section 2 recall the main concept of a metric map and deal with the problem of metric interpolation. Then in section 3, we describe our directional semi-coarsening algorithm while section 4 gives a quick description of the mesh generator that we have used. Finally, in section 5, we demonstrate the effectiveness of our approach on several anisotropic meshes displaying huge aspect ratio (as large as 10^4).

2. MESHES AND METRICS

The concept of mesh generation governed by metric specifications is now of current use in this field [33, 25, 26, 10, 15, 24, 19]. In this section, we review the fundamental properties and definitions of a metric field that will be useful for our applications.
First, we recall the definition of a metric ([3]).

Definition 1: A Metric (also Tensor) $\mathcal{M}$ in $\mathbb{R}^{d\times d}$ is a symmetric definite positive real matrix with d the dimension of space. It is then a $d\times d$ invertible matrix, its eigenvalues are real and positive, and its eigenvectors form an orthogonal basis of $\mathbb{R}^d$. such that

$$\mathcal{M} = V(\mathcal{M})^t \Lambda(\mathcal{M}) V(\mathcal{M})$$

where $V(\mathcal{M})$ denotes the orthonormal matrix corresponding to the eigenvectors of $\mathcal{M}$ while $\Lambda(\mathcal{M})$ is the diagonal matrix of its eigenvalues ■

The orthogonalization property of the metric allows to define a scalar product of two vectors in $\mathbb{R}^d$ with respect to a metric $\mathcal{M}$:

$$(u,v)_{\mathcal{M}} = (u, \mathcal{M}v) = u^t \mathcal{M} v \ \in \ \mathbb{R} \tag{1}$$

we now define, the associated norm of a vector in $\mathbb{R}^d$ with respect to the metric $\mathcal{M}$:

$$\|u\|_{\mathcal{M}} = ((u,u)_{\mathcal{M}})^{1/2} \tag{2}$$

A useful graphical representation of a metric $\mathcal{M}$ associates to it the ellipsoid that represents the unit ball for the scalar product $(.,.)_{\mathcal{M}}$:

$$\mathcal{E}_{\mathcal{M}} = \{u/(u,u)_{\mathcal{M}}^{1/2} = 1\} \tag{3}$$

The equation (3) represents a general ellipsoid form, where the main axis are given by the eigenvectors of $\mathcal{M}$ and the radius of each axis is given by the inverse of the square root of the associated eigenvalues.
More details and definitions can be found in [3].

2.1. Notations

We will use the following notations:

- $V(\mathcal{M})$ an orthonormal matrix corresponding to eigenvectors of a Metric $\mathcal{M}$
- $\lambda_1(\mathcal{M}), \lambda_2(\mathcal{M}), .., \lambda_d(\mathcal{M})$ the eigenvalues corresponding to a metric $\mathcal{M}$.
- $h_1(\mathcal{M}), h_2(\mathcal{M}), .., h_d(\mathcal{M})$ the local size in each direction of the metric defined by: $h_i(\mathcal{M}) = \sqrt{1/\lambda_i(\mathcal{M})}$ for $i = 1, ..., d$.
- For any function f defined on $\mathcal{R}$, we also define the matrix $f(\mathcal{M})$ as the matrix
$$f(\mathcal{M}) = V(\mathcal{M})^t f(\Lambda(\mathcal{M}))V(\mathcal{M})$$
where $f(\Lambda(\mathcal{M}))$ is the diagonal matrix whose entries are $f(\lambda_i(\mathcal{M}))$

We will suppress the notation $(\mathcal{M})$ when no ambiguity will possible

2.2. Metric associated with a simplicial element

Let $\mathcal{T}_h$ be a tetrahedrization of a polygonal domain Ω defined by a set of nodes $\mathcal{N}$ and a set of simplicial elements (triangles in 2-D, tetrahedra in 3-D) $\mathcal{T}$ whose nodes belong to $\mathcal{N}$. It is quite obvious (and this fact is the basis of the concept of generation of a mesh governed by a metric field, see [3]) that the set of simplicial elements defined a P^0 field of metrics on Ω such that for this metric field each simplex is equilateral : for a given element T, define the set $E(T)$ of all edges of T, $E(T) = \{(k,l) : k \neq l, k, l \in \mathcal{N}, k, l \in T\}$. Then the element metric $\mathcal{M}_T$ on T should satisfy

$$\vec{x}_{kl}^T \mathcal{M}_T \vec{x}_{kl} = 1 \tag{4}$$

for all edges $(k,l) \in E(T)$ and $\vec{x}_{kl} = \vec{x}_l - \vec{x}_k$ where $\vec{x}_k$ are the coordinates of the node k. This condition gives a system of $1, 3$ or 6 linear algebraic equations (for 1D, 2D and 3D, resp.) for components $(\mathcal{M}_T)_{ij}, i \leq j$ of the metric matrix. The solution of this linear system is given in the following result :

Lemma 2.1. *The element metric matrix M_T can be computed as follows:*

$$M_T = C_M \cdot \left(\sum_{\substack{i,j=1 \\ i<j}}^{d+1} \vec{x}_{ij} \vec{x}_{ij}^T \right)^{-1}, \tag{5}$$

with $C_M = (d+1)/2$ and d the space dimension. ■

2.3. Metric associated to the nodes of a tetraedrization

A mesh generation algorithm governed by metric specifications uses a metric field to define at any point of the domain Ω the desired mesh size and stretching directions. However, this metric field has to be defined in some way. In some cases, it is possible to define this metric field by splitting Ω into several subdomains and defining analytically the metric field over the subdomains. However, in practice, it is more convenient to define this field on an initial mesh, sometimes called the background mesh. This background mesh is only useful for defining at any point of the domain the metric specification and do not have to be confused with the actual mesh that we want to generate. In our coarsening application, the background mesh will obviously be defined by the fine mesh. Using the result of section 2.2, it is clear that we can define a P^0 (constant by element) metric field on the background mesh. However, our coarsening algorithm will remove nodes of the fine triangulation and so thus will delete the elements of the background triangulation. It is thus much more convenient to define a P^1 metric field on the backgound mesh and thus to define the metric field at the nodes of the background (fine) triangulation. Moreover, to the best of our knowledge, the anisotropic mesh generation softwares currently in use or in development [3, 9] take as input a metric map defined only over the nodes of the background mesh because it is easier to transport P^1 fields during mesh generation process than P^0 fields. We thus have to consider the problem of defining from a P^0 tensor field a P^1 tensor field that, in some sense has to represent the same field : Given for each element, a metric that defines the length of edges and the stretching directions, we have to define a set of metrics defined on the nodes of the triangulation that will define for any point of the domain approximatively the same lengths and stretching directions. For this, we have considered three different strategies that are described and compared in the following sections.

2.3.1. Interpolation of element metrics to node

A first way is to consider a simple interpolation of the metrics defined on the elements that surround a common node. Let us denote $\mathcal{T}(i)$ the set of all elements T which contain a node i, $i \in T$. The nodal metric matrix of node i is simply obtained by:

$$\mathcal{M}_i = \left(\frac{1}{card(\mathcal{T}(i))} \sum_{T \in \mathcal{T}(i)} \mathcal{M}_T^{-1/2} \right)^{-2} \tag{6}$$

where $\mathcal{M}^{-1/2}$ is a function of the matrix based on the diagonalization of $\mathcal{M} = V.D.V^T$ calculated by taking $\mathcal{M}^{-1/2} = V.D^{-1/2}.V^T$ with a similar definition for $\mathcal{M}^{-2}$. The powers $-1/2$ and -2 have been chosen to give to the expression (6) the meaning of averaging the characteristic mesh size over the elements $T \in \mathcal{T}(i)$ adjacent to i.

The nodal metric defined by (6) has been constructed on the element metrics, and as such should take into account the local shape and direction of anisotropies of adjacent mesh elements. However, numerical experiments show that it does not give, in some cases, quite exact information on the average aspect ratio of the adjacent elements and that it oversizes the smallest characteristic mesh size (eg. the boundary layer thickness). This situation happens, for example, on a curved boundary layer with a high aspect order (order 10^4 or more).

For such a case, we propose a correction of the eigenvalues of the nodal metric matrix M_i computed in (6). For each metric matrix M, let us denote its diagonal matrix by D with eigenvalues ordered in an ascending order on the diagonal and an orthonormal matrix of corresponding eigenvectors by V,

$$M = V.D.V^T$$

With reference to expression (6) let us compute a modified diagonal matrix $\mathcal{D}_i$ as follows

$$\mathcal{D}_i = \left(\frac{1}{card(\mathcal{T}(i))} \sum_{T \in \mathcal{T}(i)} D_T^{-1/2} \right)^{-2} \tag{7}$$

The final nodal metric is thus given by

$$\mathcal{M}_i = V_i.\mathcal{D}_i.V_i^T \tag{8}$$

where V_i is the orthogonal matrix of eigenvectors of the original matrix $\mathcal{M}_i$ used in formula (6). This procedure thus replaces the eigenvalues of $\mathcal{M}_i$ by the corrected eigenvalues given by (7) that tries to have a better estimate of the aspect ratio of the elements at node i.

3. ANISOTROPIC COARSENING ALGORITHM

In this section we describe briefly the main steps of an automatic coarsening algorithm taking into account the mesh anisotropy and discuss some practical points regarding its implementation.

3.1. An anisotropic coarsening algorithm

The anisotropic coarsening algorithm follows three main steps :

1. Generate on each node of the finest mesh an initial nodal metric that reflects the size and stretching of elements belonging to this mesh.

2. Modify the initial metric to establish a corresponding coarsened mesh metric. This is done by modifying the eigenvalues λ_i associated to the metric, which will modify the mesh size in the desired direction which is the corresponding eigenvector V_i.

3. Provide the background (fine) mesh and the desired metric field to a mesh generation tool using metric specifications.

The modification of the eigenvalues necessary to specify the new metric is realized using the following algorithm :

Algorithm 1: Anisotropic semi-coarsening algorithm

Input: $\lambda_1, \cdots, \lambda_d$ eigenvalues of a $\mathcal{M}_k$

Output: Update $\lambda_1, \cdots, \lambda_d$ coarsened eigenvalues

For each $i \in \mathcal{N}$

Set: $h_k^i = (\lambda_k^i)^{-1/2}$, $k = 1, \ldots, d$, $h_1^i \leq \ldots \leq h_d^i$
and $h_0 = C_{cf}.h_1$.

$h_k^i = \max(h_k^i, \min\left(C_{cf} \cdot h_k^i, h_{k-1}^i\right))$.

Define new eigenvalues $\lambda_k \leftarrow h_k^{-2}$.

End for.

In this algorithm, C_{cf} is the coarsening factor (in all our applications, taken equal to 2). It is seen that this algorithm doubles the mesh size only in the direction where the mesh size is minimal in the region of high anisotropy where $h_1^i << h_2^i$ while it doubles it in the three space directions in the region of isotropy where $h_1^i \sim h_2^i \sim h_3^i$

3.2. Modified anisotropic coarsening algorithm

The previous anisotropic coarsening algorithm produces a semi-coarsening in the boundary layer region and a total coarsening in the isotropic region. However, as the different meshes are progressively coarsened, it may happen that this algorithm produces an extremely rapid change in the metric specifications. To understand this point, consider Figure **??**. It represents on the left, a configuration of a mesh zone between boundary layer and isotropic mesh zone. On the right, we have shown a representation of the metric at the nodes i and j respectively. Observe that the x-radius of the two ellipses are equal. The previous algorithm will identify node i as being in the isotropic region and consequently, will ask for a doubling of the radius of the ellipsis in the x and y directions. However, point j will be identified as being in the boundary layer and therefore, the metric will be modified only in the y-direction. The metric specification for the new coarsened meshes is therefore represented in figure **??**. It is clear that we have a conflict between the two opposite requirements given by these metrics. This situation has been studied in other context in [5] and mesh generation algorithms have difficulties to handle these cases. A solution is to allow a smooth evolution of the metric

field. There are many different ways to obtain a regular metric map. For instance, one can use smoothing of the metric field by averaging the different metric on an enlarged stencil around a node using the techniques described in section 2.3. Here, we simply suggest an improved algorithm that allows to coarsen smoothly the intermediate region between anisotropic and isotropic mesh regions.

This algorithm takes into account the specification of the nodal metric on the neighbors of a given node and do not allow a fast variation of the metric specifications. This algorithm is the following :

Algorithm 2: Modified anisotropic coarsening algorithm
Input: $\lambda_1, \cdots, \lambda_d$ eigenvalues of a $\mathcal{M}_k$
Output: Update $\lambda_1, \cdots, \lambda_d$ coarsened eigenvalues

For each $i \in \mathcal{N}$

Set: $h_k^i = (\lambda_k^i)^{-1/2}$, $k = 1, \ldots, d$, $h_1^i \leq \ldots \leq h_d^i$.

Set $h_0 = C_{CF} \cdot h_1$.

Store the initial size: $h^i_{k,\text{old}} = h_k^i$.

// compare the current node size specifications with its neighbors

if $\max(h_d^i, \min\left(C_{cf} \cdot h_d^i, h_{d-1}^i\right)) = C_{cf} \cdot h_d^i$

and $\exists j_0 < i \in V(i)$ such as
$h_1^{j_0} < h_1^i$. then $h_k^i = h_{k,old}^i, k = 1, \ldots d$

Define new eigenvalues $\lambda_k \leftarrow h_k^{-2}$.

End for

4. THE MTC MESH GENERATOR

The coarsening algorithm developped in this work can be used with any mesh generator using the concept of generation governed by a metric map (or alternatively by stretching directions). These mesh generators can use the Delaunay principle, advancing fronts or layers or any other kind of mesh

generation method. However, in practice, we have found that for three-dimensional geometries, the number of mesh generation softwares able to handle the extremely anisotropic meshes typical of CFD applications is extremely limited. In this section, we give a short description of the MTC mesh-generation tool that is indeed able to produce and modify meshes with extremely high aspect ratio. MTC is being developed at Ecole des Mines de Paris, Centre de Mise en Forme des Materiaux, Sophia Antipolis. It is based on the idea to improve iteratively, an initial unsatisfactory mesh by local improvements. The general algorithm can be expressed as follows, for further details see [9, 15].

4.1. Meshing and re-meshing processes in MTC

MTC mesh generator re-mesh the initial mesh iteratively by a local mesh optimization technique. The mesh optimization technique consists in local re-meshing of cavities formed by small clusters of elements in order to increase the "quality" of the elements of the cluster.

In the re-meshing process, two principles are enforced :

- **Minimal volume**, that insures the conformity of the mesh, with no overlaps of elements: let $\mathcal{T}_i(C)$ denote the i-th set of elements T filling the local cavity. Following the minimum volume principle we choose as an optimal (possibly not unique) re-triangulation of the cavity the one satisfying

$$\sum_{T \in \mathcal{T}_i(C)} |(Volume)(T))| \rightarrow \min, \tag{9}$$

  where the minimization is done over a small set $i = \{1, \ldots, I\}$ of possible triangulations $\mathcal{T}_i(C)$ of elements (Fig. **??** right) connecting the nodes on the border of the cavity, or other nodes like the cavity barycenter, with all boundary faces.

- **The geometrical quality** $Q(T)$, which is evaluated for each element. If the minimizer of (9) is not unique, this criterion picks among all admissible cavity re-triangulations the one improving the geometrical quality of the mesh by improving the quality of the worst element of the triangulation.

While the former criterion assures the conformity of the mesh, if the initial mesh was conform, the latter handles improvements of element shape, size, connectivity, etc., depending on the quality function $Q(T)$. Usually, the quality function $Q(T)$ is a function of the geometry of the element T and the prescribed background metric, which give together a measure for the element size and the element form (aspect ratio).

4.2. Definition of the quality function

Let $C_n = \{T_1, \cdots, T_n\} \subset T_h$ be a set of n elements. We define the quality of this set with respect to a given metric $M \in \mathbb{R}^{d\times d}$ as a real *n-vector*

$$Q_n = \{q(T_1), \cdots, q(T_n)\}$$

The quality $q(T)$ of an element T of C_n, measured in the metric M, is defined as the product of two factors :

$$q(T) = Q_F(T) \cdot Q_S(T) \tag{10}$$

The first one $Q_F(T)$ controls the shape of the element T. The best possible element being the equilateral element (in the metric M). It is defined by :

$$Q_F(T) = C_0 \frac{(Volume)(T)_{M(T)}}{h^d_{M(T)}} \tag{11}$$

where

- d is the space dimension,
- $(Volume)(T)_{M(T)}$ is the volume of T measured in the metric space, and is given by:

  $$(Volume)(T)_{M(T)} \;=\; (Volume)(T)\sqrt{det(M(T))}.$$

  The matrix $M(T)$ is obtained by averaging the nodal metric matrices on nodes of the element T,

  $$M(T) = \left(\frac{1}{d+1}\sum_{i\in T} M_i^{-1/2}\right)^{-2},$$

  and correct its eigenvalues by the same process as in Section 2.3.1.

- $h_{M(T)}$ is the average of lengths of edges of T measured by the metric $M(T)$,

$$h_{M(T)} = \left(\frac{2}{d(d+1)} \sum_{(i,j)\in T} (M(T)(\vec{x}_j - \vec{x}_i), (\vec{x}_j - \vec{x}_i)) \right)^{1/2} \tag{12}$$

- in (11) C_0 is such that 11 is equal to 1 when T is equilateral in the metric $M(T)$.

The second factor $Q_S(T)$ controls the *size* of the element in the metric M. Its definition is given by :

$$Q_S(T) = min(\frac{1}{h_{M(T)}}, h_{M(T)})^d \tag{13}$$

where $\vec{x}_i \in \mathbb{R}^d$ are the coordinates of the node i.

With these definition of the quality of a set of elements, we just need now a way to compare two different sets in order to pick the "best" one. This is done by defining a lexicographic order " $<''$ between two different sets of elements.

4.3. Algorithm (MTC iteration)

With the concept previously defined, the mesh generation in MTC will proceed by successive improvements of given mesh. Let us denote T_h, E_h, N_h, respectively the sets of all mesh elements, edges and nodes. Repeat for different cavities $C_0 \in T_h$ composed by a group of adjacent elements obtained as the nearest neighborhood of a node $n \in N_h$ or of an edge connecting 2 nodes $(n_1, n_2) \in E_h$, $n_1, n_2 \in N_h$.

1. Denote $N_0 \subset N_h$ and $E_0 \subset E_h$ respectively, the set of all nodes and all edges of the cavity C_0.

2. Denote $\partial E \subset E_0$ the set of all edges belonging to the border of the cavity C_0.

3. Evaluate the quality Q_0 of the set C_0 of elements $T \in C_0$ by a given function, e.g. 10.

4. For each $n \in N_0$ do:

• Connect the node n to each edge $e \in \partial E_0, n \notin e$ to get a set of elements $C_n = \{T_e, T_e = (e, n), \text{attempting to re triangulate the cavity } C_0\}$.

• Evaluate the quality Q_n of the set C_n by a given function, e.g. 10.

• If $Q_0 < Q_n$ (in the sense of Definition 4.2) then set $C_0 \leftarrow C_n, Q_0 \leftarrow Q_n$.

Until stagnation of the changes applied to the mesh.

4.4. Modified Quality function

In our application, we have found useful to re-define slightly some component of the MTC algorithm. This section details these modifications. First, we have re-defined the functions $Q_S(T)$ and $Q_F(T)$ in expression (10) in the following way. We take

$$Q_S(T) = \min_{(i,j)\in T} \left(h_{ij}^{M(T)}, \frac{1}{h_{ij}^{M(T)}} \right)^d,$$

instead of

$$Q_S^{orig}(T) = \min \left(h_{M(T)}, \frac{1}{h_{M(T)}} \right)^d,$$

This new definition avoid taking averages, because the mesh lengths may be very different on an element due to the anisotropy of the mesh.

The measure of the shape quality remains the same as in the original MTC code,

$$Q_F(T) = C_0 \frac{(Volume)_{M(T)}(T)}{h_{M(T)}^d},$$

Finally, we have found useful to add to the measure of the quality of an element (10) an additional function $Q_V(T)$ to have a better control on the element characteristic volume. Thus our definition of the quality of an element is now

$$Q(T) = Q_S(T) \cdot Q_F(T) \cdot Q_V(T)$$

This increase the sensitivity of $Q(T)$ to the non-respect of prescribed aspect ratio. The definition of $Q_V(T)$ that we use is

$$Q_V(T) = \min\left(\frac{1}{(Volume)_{M(T)}(T)}, (Volume)_{M(T)}(T)\right).$$

5. APPLICATIONS

5.1. Coarsening synthetic mesh

The coarsening process and development detailed in the previous sections, is applied here on a model problem in order to show how it works. Then in the following section, we will apply the semi-coarsening tool to industrial meshes with a very large aspect ratio.

Our model problem is a regular hexaedral domain Ω divided into 8 hexaedrons whose volumes are $V = \delta x \times \delta y \times \delta z$, we take $\delta x = \delta y$ and $\delta z << \delta x$. In the example given below, $\delta z = 10^{-3}\delta x$ but we have checked that the results does not depend on this ratio and that the results are unchanged even with $\delta z = 10^{-6}\delta x$. Each hexaedron is divided into two prisms, each being subdivided into three tetrahedra. Figure 7 (a) shows this tetrahedrization that contains 27 nodes and is composed of three node layers in the z direction. On each of these 27 nodes, the eigenvectors of the corresponding metric are approximately the three x, y, z axis with corresponding eigenvalues respectively equal to $(1/\delta x)^2, (1/\delta y)^2, (1/\delta z)^2$. The coarsening algorithm will change this metric for another one with the same eigenvectors but with eigenvalues corresponding to the length $\delta x, \delta y, 2\delta z$ then the mesh generator will try to construct a mesh corresponding to this wished metric. The result is shown on figure [7 (b)]. As expected, this mesh contains 18 nodes and the coarsening process has eliminated the intermediate node layer thus performing a semi-coarsening by a factor 2 in the z direction.

6. Conclusion

The present work has studied the capability of a semi-coarsening algorithm to handle extremely stretched meshes used for Navier-Stokes computations with low Reynolds turbulence modeling. This algorithm is able to generate a sequence of meshes with decreasing number of nodes in the direction perpendicular to the geometry. This sequence of meshes is thus suitable

for multigrid acceleration.

In particular in this work we have study the interpolation procedures and smoothing of the metric fields. Our work has relied on some recent results on metric interpolations [2, 11] as well as on some more simpler ideas based on averaging of the mesh edges length. In addition, we have proposed a way to handle the connection between isotropic regions in the far field and the anisotropic ones in the boundary layers. Then we have designed an anisotropic coarsening algorithm able to identify the directions of stretching and to remove nodes in the direction of minimal mesh size only. We have also performed several relevant tests for the semi-coarsening strategy. These experiments show that geometrical methods can now be used for multigrid acceleration even for highly anisotropic meshes employed for Navier-Stokes computations with turbulence modeling in an industrial context.

7. Acknowledgements

This work has been supported in part by a grant from TO BE COMPLETED

References

[1] Adams, M. *A parallel maximal independent set algorithm*, Proceedings 5^{th} Copper mountain conference on iterative methods, 1998.

[2] Arsigny, V., Fillard, P., Pennec, X., and Ayache, N., *Fast and Simple Computations on Tensors with Log-Euclidean Metrics* INRIA Research Report - NO 5584.

[3] Borouchaki, H., George, P.-L., Hecht, F., Laug, P., and Saltel, E., *Delaunay mesh generation governed by metric specifications. Part 1 : Algorithms*, Finite Elements in Analysis and Design, 25, 1997, pp. 61-83.

[4] Borouchaki, H., George, P.-L., and Mohammadi, B. *Delaunay mesh generation governed by metric specifications. Part 2 : Application examples*, Finite Elements in Analysis and Design, 25, 1997, pp. 85-109.

[5] Borouchaki, H., Hecht, F. and Frey, P. *Mesh Gradation Control*, International Journal for Numerical Methods in Engineering, Wiley, Vol 43, pp. 1143-1165, 1999

[6] Carre, G., Carte, G., Guillard, H., and Lanteri, S. *Multigrid strategies for CFD problems on unstructured meshes* in "Multigrid Methods VI", Krys Riemslagh (ed), Lecture Notes in Computational Science and Engineering, 14, pp. 1-11, Springer, 2000.

[7] Carte, G., Coupez, T., Guillard, H., and Lanteri, S., *Coarsening techniques in multigrid applications on unstructured meshes*, European congress on computational methods in applied sciences and engineering, ECCOMAS 2000, Barcelona, 2000.

[8] Chan, T., and Smith, B., *Domain decomposition and multigrid algorithms for elliptic problems unstructured meshes*, Electronic Transactions in Numerical Analysis, 2, pp. 171-182, 1994.

[9] Coupez, T., *Génération de maillage et adaptation de maillage par optimisation locale* Revue Européene des éléments finis 9 (2002), no. 4, pp. 403-423.

[10] Dompierre, J., Vallet, M., Fortin, M., Bourgault, Y., and Habashi, W. *Anisotropic mesh adaptation: towards a solver and user independent cfd.* , AIAA paper 97-0861 (1997)

[11] Fletcher, P.T., and Joshi, S., *Principal Geodesic Analysis on Symmetric Spaces: Statistics of Diffusion Tensors*, Proceedings of Workshop on Computer Vision Approaches to Medical Image Analysis (CVAMIA), 2004.

[12] Francescatto, J., and Dervieux, A., *A semi-coarsening strategy for unstructured multigrid method based on Agglomeration*, Int. J. Num. Meth. Fluids 26, pp. 927-957.

[13] Fréchet, M. *Les éléments aléatoires de nature quelconque dans un espace distancié*, Inst. H. Poincaré, (10), pp. 215-310,1948

[14] George, P.-L., and Borouchaki, H., *Premières expériences de maillage automatique par une méthode de Delaunay anisotrope en trois dimensions*, Technical report 0272, INRIA, 2002, http://www.inria.fr/rrrt/rt-0272.html.

[15] Gruau, C., and Coupez, T. *3D tetrahedral unstructured and anisotropic mesh generation with adaptation to natural and multidomain metric*, Comput. Meth. Appl. Mech. Engrg. 194, 48-49 (2005), pp. 4951-4976.

[16] Guillard, H., *Node-Nested Multi-Grid with Delaunay Coarsening* INRIA Research Report No 1898, 1993.

[17] Guillard, H., Janka, A. and Vanek, P. *Analysis of an Algebraic Petrov-Galerkin Smoothed Aggregation Multigrid Method*, Applied Numerical Mathematics 58 (2008), pp. 1861-1874

[18] Guillard, H. and Vanek, P *An Aggregation Multigrid Solver for Convection-Diffusion Problems on Unstructured Meshes*, University of Colorado in Denver, Center for Computational Mathematics, Report UCD-CCM No 130, 1996

[19] Huang, W. *Metric tensors for anisotropic mesh generation.* J. Comput. Phys. 204 (2005), pp. 633-665.

[20] Janka, A., *Multigrid methods for compressible fluids*, PhD in Applied Mathematics, University of Nice Sophia-Antipolis, 2002

[21] Karcher, H. *Riemannian center of mass and mollifier smoothing*, Communications on Pure and Applied Mathematics, 30(5), pp. 509-541, 1977

[22] Koobus, B., Lallemand, M.H., and Dervieux, A., *Unstructured Volume Agglomeration MG : Solution of Poisson Equation*, Int. J. Num. Mesh. Fluids 18, pp. 27-42.

[23] Lallemand, M.H., Stéve H., and Dervieux, A., *Unstructured multigriding by Volume Agglomeration: Current Status*, Computers and in Fluids 21, pp. 397-433.

[24] Li, X., Shephard, M., and Beall, M. *3D anisotropic mesh adaptation by mesh modification*, Comput. Meth. Appl. Mech. Engrg. 194, 48-49 (2005), pp. 4915-4950.

[25] Lohner, R., and Beum, J. *Adaptive h-refinement on 3D unstructured grids for transient problems*, Int. Numer. Meth. Fluids 14 (1992), pp. 1407-1419

[26] Mavriplis, D. *Adaptive meshing techniques for viscous flow calculations on mixed-element unstructured meshes*, AIAA paper 97-0857 (1997)

[27] Mavriplis, D. *Multigrid strategies for viscous ow solvers on anisotropic unstructured meshes*, Journal of Computational Physics, 145 (1998), pp. 141-165.

[28] Müller, J.D., *Coarsening 3-D Hybrid Meshes for multigrid methods*, Copper mountain conference, 1999.

[29] Müller, J.D., *Anisotropic adaptation and multigrid for hybrid grids*, Int. J. Num Meth. in Fluids, 2002, 40, pp. 445-455.

[30] Ollivier-Gooch, C.F. *Coarsening unstructured meshes by edge contraction*, International Journal for Numerical Methods in Engineering, 57(3), (2003), pp. 391-414

[31] Pennec, X. *Probabilities and statistics on Riemannian manifolds: basic tools for geometric measurements*, IEEE workshop on non-linear signal and image processing, 1999.

[32] Ruge, J., and Stuben, K. *Algebraic Multigrid Methods*: in Multigrid Methods (S. Mc Cormick Ed), Frontiers in Applied Math, Vol 5, SIAM, Philadelphia.

[33] Tam, A., Ait-Ali-Yahia, D., Robichaud, M.P., Moore M., Kozel V., and Habashi, W.G. *Anisotropic mesh adaptation for 3D flows on structured and unstructured grids*, Computer Methods in Applied Mechanics and Engineering , Volume 189, Number 4, 29 September 2000, pp. 1205-1230(26)

[34] Vanek, P., Mandel, J., and Brezina, M., *Algebraic Multigrid by Smoothed Aggregation for Second and Fourth Order Elliptic Problems*, Computing 56 (1996) pp. 179-196.

[35] Beggs, S.H., Carter, R.R., and Dranfield, P. *Zssigning effective values to simulator gridblock parameters for heterogeneous reservoirs*, SPE Reservoir Engineering J., Nov. 1989, pp. 455-465.

[36] Barker, J.W. and Thibeau, S. *A critical review of the use of pseudo relative permeabilities for upscaling*, SPE Reservoir Engineering J., May 1997, pp. 138-143.

[37] Amaziane, B., Bourgeat, A., and Koebbe, J. *Numerical simulation and homogenization of two-phase flow in heterogeneous porous media*, Transp. Porous media 6, pp. 519-539, 1991.

[38] Christie, M.A. *Upscaling for reservoir simulation* J. Pet. Tech., Nov. 1996, pp. 1004-1010.

[39] Deutsch, C.V. *Calculating effective absolute permeability in standstone/shale sequences*, SPE Research Engineering, September 1989, pp. 343-350.

[40] Durlofsky, L.J., Jones, R.C., and Milliken,W.J. *A new method for scale up of displacement processes in heterogeneous reservoirs*, Proc. Third European Conference on the Mathematics of Oil Recovery, Roros, Norway, 1994.

[41] Lake, L.W., Kasap, E., and Shook, M. *Pseudofunctions- The key to practical use of reservoir description*, in North Sea Oil and Gas Reservoirs-II, A. T. Buller, et al., eds., Graham and Trotman, London, 1990.

[42] King, P.R. *The use of renormalization for calculating effective permeability*, Transp. Porous media 4, pp. 37-44, 1989.

[43] Muggeridge, A.H. *Generation of effective relative permeabilities from detailed simulation of flow in heterogeneous porous media*, in Reservoir Characterization II, L.W. Lake et al., eds., Academic Press, San Diego, CA, pp. 197-225, 1991.

[44] Whitaker, S. *Flow in porous media I: A theoritical derivation of Darcy's law*, Transp. Porous media 1, pp. 3-25, 1986.

[45] White, C.D., and Horne, R.N. *Computing absolute transmissibility in the presence of fine-scale heterogeneity, SPE 16011*, in The SPE Symp, on Reservoir Simulation, San Antonio, TX, 1987

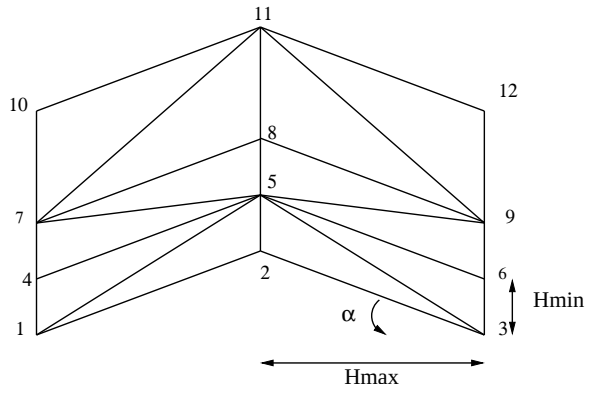
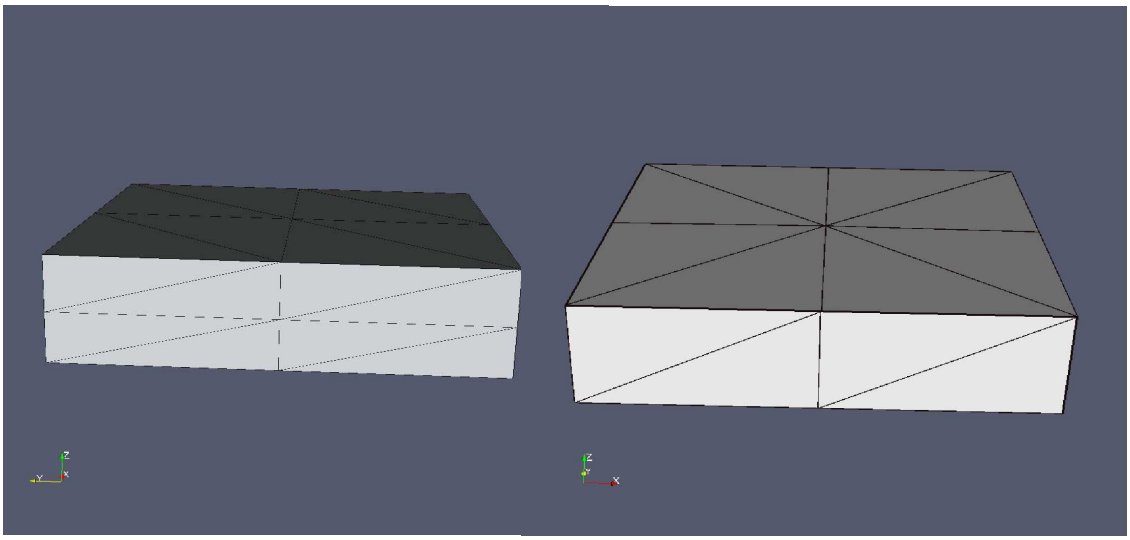


Figure 1: 2-D model boundary layer mesh



(a)Baseline mesh (b)Coarsened mesh

Figure 2: Semi-coarsening of a model boundary layer mesh. Note that the y-axis is scaled in order to be able to visualize the result.