Quasi-structured anisotropic quad-dominant mesh adaptation using metric-orthogonal approach

Lucille-Marie Tenkes, Adrien Loseille, Frederic Alauzet

Quasi-structured anisotropic quad-dominant mesh adaptation using metric-orthogonal approach

Lucille-Marie Tenkes*, Adrien Loseille† and Frédéric Alauzet‡
GAMMA Team, INRIA Saclay Ile-de-France, Palaiseau, France

We present a strategy for the generation of mixed-element quasi-structured meshes. This strategy is based on the tools of metric-based mesh adaptation. Using metric-orthogonal point placement, we can generate a pattern of points following the underlying structure of the metric-field, from which we generate a quasi structured triangular or mixed-element mesh. This paper presents some enhancement of the adaptation loop towards the generation of quad-dominant meshes. Especially, since this method highly depends on the prescribed metric-field, we present a tailored gradation process that favors the formation of structured elements. We also explore two processes to recover the quadrilaterals: an indirect method based on combining right triangles from a preliminary orthogonal triangular mesh, and a quadrilateral detection at the point-placement step.

I. Introduction

Fluid dynamics simulations are expected to capture several phenomena of various nature like shocks, boundary layer or turbulence. The accuracy of the simulation strongly depends on the mesh features. For example, schemes used for boundary layers are often suited for structured meshes that are aligned with the boundary of the domain. More generally, some node-centered schemes show better convergence on regular structured meshes.[6] In this context, quasi-structured meshes can be considered, to combine the advantages of adapted unstructured meshes, ensure alignment with the boundaries of the domain, and retrieve some structured parts in the mesh, see[2,22,23] for example. Our approach relies on the tools of metric-based mesh adaptation[9,16,17] to generate such quasi-structured meshes, the point being to eventually generate quad-dominant meshes with fully structured elements where they are relevant (shocks, boundary layers) and unstructured elements elsewhere. Using this approach, the mixed element mesh generation is also fully automated.
In our strategy, the standard adaptation process is modified to generate a quasi-structured-mesh, by exploiting the structure from the eigenvectors of the metric. Indeed, in hex-dominant or quad-dominant mesh generation, a directional field is often necessary and a few methods use the directional information contained in the metric field for this purpose, for example in,[7,11,13] it is used to move the vertices of an existing mesh, to recover some structure. In our work, the directional field used at the point-creation step. We apply a metric-orthogonal point-placement,[15,19] which creates a structured point distribution. From this distribution, there are two ways to generate a mixed-element mesh.
The first method is referred to as an *a posteriori* strategy. The points of the list are inserted in a triangular mesh using cavity-based operators.[18] The resulting mesh is mainly constituted with right-angled triangles. From this preliminary quasi-structured mesh, a mixed-element mesh can be recovered. There are numerous ways of recovering quadrilaterals from a triangular mesh, such as frontal approaches and matchings.[3] The method presented in this paper focuses on the formation of quadrilaterals with the strongest anisotropy ratio and the best alignment with the metric field. It takes advantage of the specific pattern of metric-orthogonal adapted meshes, which favors right-angled triangles with high aspect ratio in some areas, for example in shocks or boundary layers. The main drawback of this *a posteriori* approach is that the directional

*PhD student, lucille-marie.tenkes@inria.fr
†Researcher, adrien.loseille@inria.fr
‡Senior Researcher, frederic.alauzet@inria.fr

information has to be recovered while it is available during the metric-orthogonal point-placement step, in the local re-mesher.
Therefore, another approach is to use straightforwardly this information during the mesh generation process. This method is said *a priori*, since the aim is to detect the quadrilaterals before the insertion step. The point-placement step is identical to the *a posteriori* method, except a graph is built on the fly based on the point propositions and filtering. Using this structure, some topological and geometrical operations can be performed before the insertion step, to maximize the number and the quality of the quadrilaterals already formed in the graph. This graph is not the final mesh, it is meant to preliminary detect the points and edges forming quadrilaterals and enforce these edges at the insertion step, that is modified for this purpose.
Standard metric-based adaptation highly relies on the quality of the provided metric field. Consequently, in the adaptation loop, the metric field computed from the solution undergoes a gradation correction to ensure smoother size transitions. There are numerous approaches to this process. Our work uses the same framework as,[1] which extends H-shock control[4] to anisotropic metric fields. The algorithm described in this paper has two main steps: a so-called growth process computing a minimal correction from one point to the others, and an intersection steps to gather all these corrections. In this research, we show how the growth process has been modified to be better suited to quasi-structured and mixed-element mesh generation.
This paper unfolds as follows. We first recall the metric-based framework for standard and metric-orthogonal mesh adaptation in Section II. The gradation correction algorithm and the modifications to design a structured mesh aware process are detailed in Section III. The *a posteriori* quadrilateral recombination and results are analyzed in Section IV and the *a priori* strategy and preliminary results on quadrilateral detection are presented in Section V.

II. Metric-based mesh adaptation and metric-orthogonal point-placement

We first recall the basic grounds of metric-based mesh adaptation. Then, we describe the modifications brought to the algorithm to generate quasi-structured meshes using the same tools. More complete information about mesh adaptation is detailed in.[9,16,17]

A. Notion of metric

The notation $\mathcal{M}$ in this paper refers to a metric or a metric field. A metric tensor in $\mathbb{R}^n$, $n \in \{2,3\}$, is a symmetric, positive-definite tensor of size n. A vectorial space E supplied with a metric is called an Euclidian metric space. The particular metric space $(\mathbb{R}^n, I_n)$, where I_n is the identity tensor, is referred to as the physical space. In an Euclidian metric space, this tensor determines a scalar product $\langle \cdot, \cdot \rangle_{\mathcal{M}}$ as

$$\forall x, y \in E, \quad \langle x, y \rangle_{\mathcal{M}} = x^T \mathcal{M}\, y.$$

By extension, a norm $\| \cdot \|_{\mathcal{M}}$ and the associated distance $d_{\mathcal{M}}$, angles and volumes can be defined. To visualize a metric field, we consider its unit sphere

$$\{x \in E \mid \|x\|_{\mathcal{M}} = 1\}.$$

In the metric space $(E, \mathcal{M})$, this pictures a unit circle. However, in the physical space (E, I_n), the geometrical representation of this sphere is an ellipse (ellipsoid in $\mathbb{R}^3$), the shape and orientation of which are given by

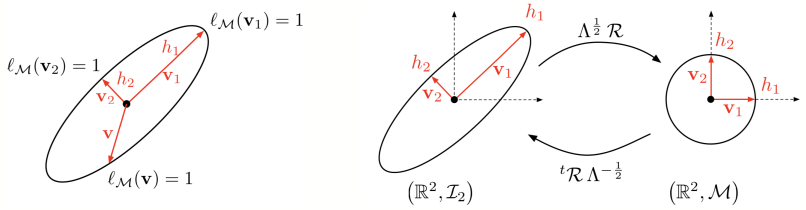

Figure 1. Visualization of the unit sphere of a metric field in the metric space $(\mathbb{R}^n, \mathcal{M})$ and in the physical space $(\mathbb{R}^n, \mathcal{I}_n)$.

the eigenvalues and eigenvectors of $\mathcal{M}$, see Figure 1. Indeed, since it is a symmetric positive definite tensor, it has positive eigenvalues and orthogonal eigenvectors. The corresponding orthonormal basis determines the orientation of the ellipsoid. The length of the i-th axis is $h_i = \sqrt{\lambda_i}^{-1}$, where λ_i is the corresponding eigenvalue.

In the case of an Euclidian metric space $(E, \mathcal{M})$, we assume that the metric is constant. A varying continuous metric field $\mathcal{M}(\cdot)$ can be applied to a continuous manifold Ω to define a Riemannian metric space $(\Omega, \mathcal{M}(\cdot))$. In a Riemannian space, the previously mentioned scalar product does not hold, though the notions of length and distance can be redefined. Given an edge $\mathbf{pq}$, the length is computed using a straight line parametrization $\gamma(t) = \mathbf{p} + t\mathbf{pq},\ t \in [0,1]$, as $\ell_{\mathcal{M}}(\mathbf{pq}) = \int_0^1 \|\gamma(t)\|_{\mathcal{M}} \mathrm{d}t = \int_0^1 \sqrt{\mathbf{pq}^T \mathcal{M}(\mathbf{p} + t\mathbf{pq})\mathbf{pq}}\mathrm{d}t$. The notions of angles and volume can also be re-defined.

In the context of mesh adaptation, the notion of unit edge is useful, naming an edge that has its length between $1/\sqrt{2}$ and $\sqrt{2}$ in the metric field. By extension, if all the edges of a mesh are unit, it is called a unit mesh.

B. Standard mesh adaptation

This section briefly recalls the main ideas of metric-based adaptation, more complete information can be found in.[5,9,12,16,17]

The aim of mesh adaptation is to find the best repartition of the degrees of freedom for a specific solution. This is a discrete problem, which is difficult to solve. However, the problem can be rewritten as a continuous problem using metric fields. In this context, it is possible to generate the optimal metric field. To obtain the discrete mesh, a unit mesh is generated from the metric field. Our standard mesh adaptation algorithm is represented as a flow chart in Figure 2.

First, an initial solution is computed on an initial mesh. From this solution, a metric-field can be extracted. This metric field is smoothed during a gradation correction process (see Section III). At the re-meshing step, an adapted mesh is generated from this metric field. Then a new solution can be computed. If the accuracy of the solution on this mesh is satisfying, the process stops, otherwise the adapted mesh and the solution become the new input of the process.

C. Metric-orthogonal mesh adaptation

To generate quasi-structured and mixed-element meshes, this algorithm is slightly modified. The main modifications affect the gradation step and the re-meshing step. The flow chart of the modified algorithm is shown in Figure 3. The most important steps are highlighted in gray.

The re-meshing step algorithm is based on the metric-orthogonal point-placement. This is a quasi-frontal approach creating a list of points to be inserted in the adapted mesh. The process starts with the following initialization: a heap-list is created with the points from the boundary of the domain, associated with the directions and sizes from the metric at these points. The order of point-creation has a significant impact on the structure. Figure 5 shows the edge-graph connecting the proposed points in two configurations for the point-placement, either following a first-in first-out queue, or a heap-list. In both cases, new points have been created according to the main directions of the metric tensors, but we observe that a structured pattern clearly emerges using the heap-list, while the alignment in the smallest direction is not recovered using the first-in first-out order.

The point-placement is an iterative process. While the heap list is not empty, the first point of the heap list is removed. Since the heap list is sorted, this point is associated with the current smallest size in the metric. It proposes two new points in the direction associated with this smallest size, as depicted in Figure 4. This figure shows the four points that can be proposed by vertex P, the prescribed directions being u_1 and u_2. The blue lines show the mesh pattern that appears after the insertion of surrounding vertices, using this point-placement scheme. If the new point is inside the domain, and is not filtered by an existing point, it is added to the heap list.

At the end, this iterative process provides a list of points to be inserted in the mesh using cavity operators.[18] This strategy and some examples have been described in.[15,19] In the remainder, we focus on how to improve this process, especially for mixed-element mesh generation. Our first contribution concerns metric gradation control.

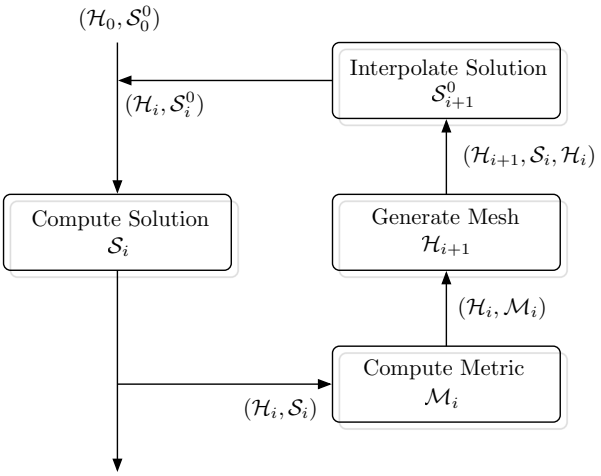

Figure 2. Standard mesh adaptation loop.

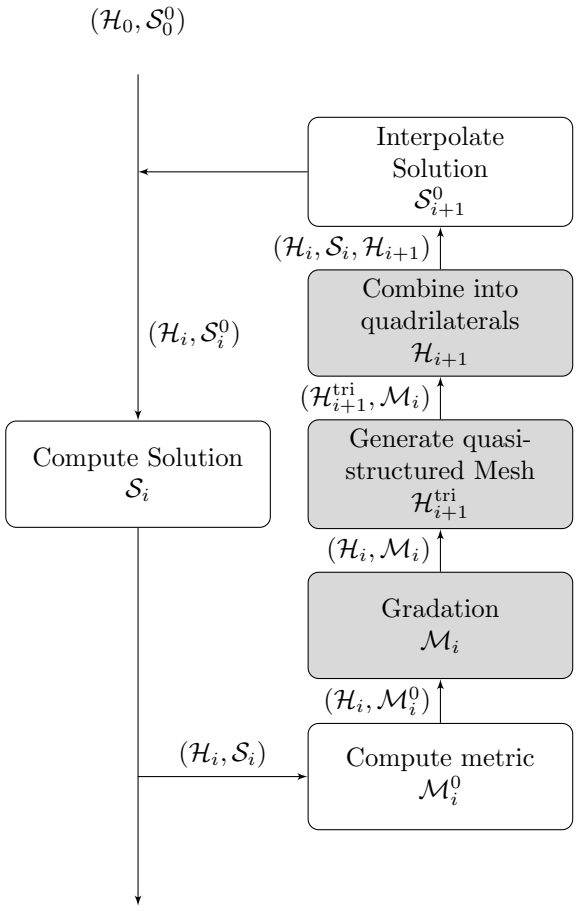

Figure 3. Modified algorithm for Mixed-element mesh adaptation

III. Orthogonality-aware metric gradation

The approach of size gradation control described in this paper is based on the control of the H-shock[4] and its extension to anisotropic metric fields described in.[1] The latter gives an efficient algorithm to smooth anisotropic metric fields. However, mixed-elements and orthogonal meshes do not raise the same issues as standard adapted meshes. Our work in this section was to determine which gradation process was the best for this purpose.

The first part of this section summarizes this algorithm for standard mesh adaptation. Then, to design an orthogonality-aware gradation, we modify the so-called growth step detailed in Section B. In,[1] three growth processes were described: physical-space growth, metric-space growth and a mixed space growth which is somehow a compromise between these two. Metric-space growth and mixed-space growth have proven to give better results in simulations according to,[1] mainly because this gradation correction process leads to a more anisotropic smoothed metric field than physical space growth. We show that for the purpose of quad-dominant mesh generation, better results are observed using physical space growth process since it leads to smoother size transitions. In addition, we design a fourth growth strategy, the directionally-constrained growth, that show even smoother transitions and better mixed-element meshes.

A. Overview on metric gradation and standard algorithm

To begin with, we recall size gradation control definitions and algorithm, see[1] and[4] for further details.

We first consider an isotropic metric field $\mathcal{M}(\mathbf{x}) = h(\mathbf{x})\ I$, and focus on gradation correction along an edge. Then, we present the algorithm to apply this process to the whole mesh.

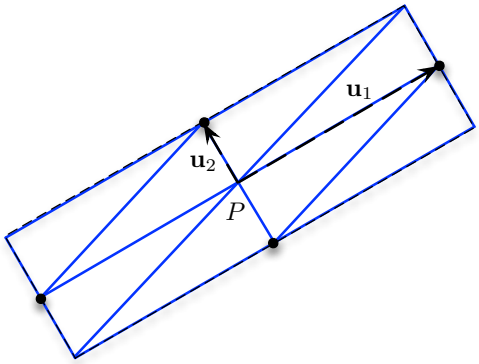

Figure 4. Metric-orthogonal point-placement from vertex P: four points are created following directions $\mathbf{u}_1$ and $\mathbf{u}_2$, this pattern favors the formation of right triangles, as suggest the blue edges.

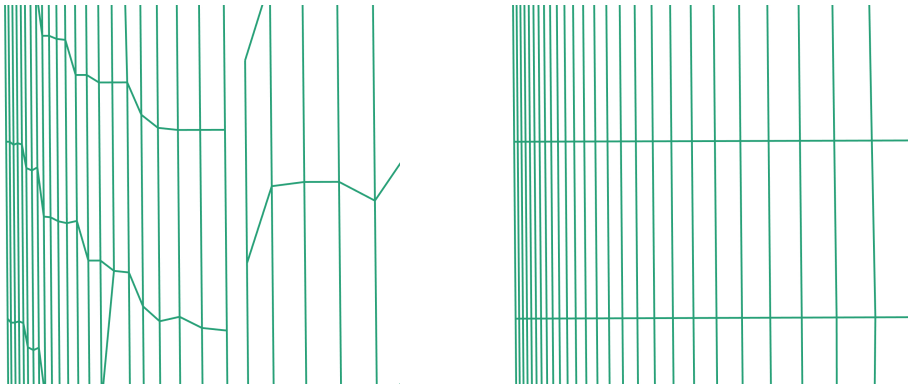

Figure 5. Comparison between a first-in, first-out and a sorted point-placement for the same prescribed metric field. This figure represents the structure of the point-placement, i.e. the edges connecting the proposed and filtered points.

Isotropic size gradation correction The idea is to bound a quantity called the H-shock or size gradation, measuring the spatial geometric progression of the size prescription in the metric. Given two vertices $\mathbf{p}$ and $\mathbf{q}$ with respective size prescriptions h_p and h_q, the H-shock $c(\mathbf{pq})$ is defined as

$$c(\mathbf{pq}) = \max\left(\frac{h_p}{h_q}, \frac{h_q}{h_p}\right)^{1/\ell_{\mathcal{M}}(\mathbf{pq})}.$$

The isotropic size correction consists in providing a minimal (optimal) reduction $\widetilde{\mathcal{M}}$ of $\mathcal{M}$ such that, for all points, the H-shock is bounded by a given threshold β. This correction regularizes the metric field and bounds the variations.

In what follows, it is assumed that $h_{\mathbf{p}} < h_{\mathbf{q}}$ so $\mathbf{p}$ is the vertex that corrects the size at $\mathbf{q}$. The corrected size is denoted by $\tilde{h}_q$.

In this case, the correction consists in computing a ratio $r(\mathbf{pq}) = h_p/\tilde{h}_q = \beta^{\ell_{\mathcal{M}}(\mathbf{pq})}$. The computation of $\ell_{\mathcal{M}}(\mathbf{pq})$ depends on the chosen interpolation law.[1] Indeed, since the metric field is only known at the vertices of the mesh, we need to interpolate to have a continuous metric field in the whole domain. For example, using the linear interpolation on h, $r(\mathbf{pq}) = 1 + \ell_p(\mathbf{pq})\ln(\beta)$. To write a similar relation on metrics instead of sizes, we introduce the coefficient $\eta^2(\mathbf{pq}) = r(\mathbf{pq})^{-2}$.

Therefore, the metric of spanned size constraints from $\mathbf{p}$ to $\mathbf{q}$, denoted by $\widetilde{\mathcal{M}}_{\mathbf{p}}(\mathbf{q})$ is obtained from $\mathcal{M}_{\mathbf{p}}$ as

$$\widetilde{\mathcal{M}}_{\mathbf{p}}(\mathbf{q}) = \eta^2(\mathbf{pq})\mathcal{M}_{\mathbf{p}}. \tag{1}$$

The process above applies to an edge $\mathbf{pq}$. To perform a gradation correction on the metric defined at every vertex of the mesh, we use the following algorithm that has two steps, as illustrated in Figure 6.

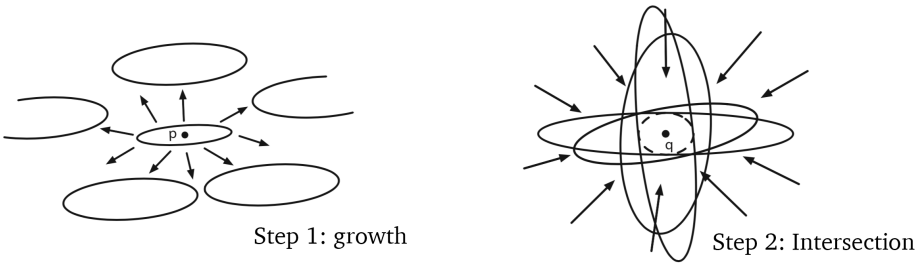

Figure 6. Simplified representation of metric gradation correction algorithm.

1. **Growth**: each vertex $\mathbf{p}$ computes a correction metric to all other vertices $\mathbf{q} \neq \mathbf{p}$.
2. **Correction**: the final corrected metric is the intersection of all the correction metrics.

To avoid the quadratic complexity of this algorithm, the implementation of gradation correction used in our work is an approached iterative process, as described in.[1] Corrections are applied edge-wise during a loop on the edges.
For the isotropic case, the spanned sizes from the growth step are given by

$$\tilde{h}_p(\mathbf{q}) = r^2(\mathbf{pq})h_p \quad \forall \mathbf{p}, \forall \mathbf{q} \neq \mathbf{p},$$

and the final correction is

$$\tilde{h}_q = \min\left(\min_{\mathbf{p}\neq\mathbf{q}} \tilde{h}_p(\mathbf{q}), h_q.\right)$$

Anisotropic size gradation correction We consider an anisotropic metric field $\mathcal{M}(\cdot)$. Anisotropy raises some difficulties, since different size constraints in all directions are involved. A choice must be made to determine how the size constraints are spanned during the growth step to bound the size variation correctly. Several options are considered and detailed in Section B. The correction step is likely more complicated in the anisotropic case. A metric intersection is applied to gather the corrections of all vertices. In an exact process, this intersection is obtained by solving an optimization problem to compute the John Ellipsoid.[10,14] In our case, we approximate the intersection with simultaneous reduction.[8] Consequently, the result depends on the order of the corrections.

B. Growth processes for anisotropic gradation control

There are several ways to span an anisotropic metric in a domain in a similar way as the isotropic case, to bound size variation. We describe four approaches of this growth process. In Section C, we analyze the impact of this choice on the structure of the resulting metric-orthogonal and mixed-element meshes. Figure 7 pictures these four growth processes. The spanning of the metric at the center of the squared domain is represented on a hidden uniform mesh for each approach.

Metric-space growth This growth is a direct extension of the isotropic growth modeled by Equation (1). A scalar growth factor $\eta^2(\mathbf{pq})$ is applied.

$$\widetilde{\mathcal{M}}_{\mathbf{p}}(\mathbf{q}) = \eta^2(\mathbf{pq})\mathcal{M}_{\mathbf{p}},$$

The aspect ratio doesn't change as shown in Figure 7 (top left frame), so the growth is homogeneous in the metric space.

Physical-space growth This growth process is modeled by a growth tensor, so that the growth depends on the direction. In the following relation, the orthonormal basis of the eigenvectors of $\mathcal{M}_p$ is denoted by $\mathcal{R}$:

$$\widetilde{\mathcal{M}}_{\mathbf{p}}(\mathbf{q}) = \mathcal{R}^T\mathcal{N}(\mathbf{pq})\Lambda\mathcal{R},$$

with $\mathcal{N}(\mathbf{pq}) = \text{diag}(\eta_i^2(\mathbf{pq}))$. The coefficients $\eta_i^2(\mathbf{pq})$ depend on the eigenvalues λ_i of $\mathcal{M}_p$ and on the chosen metric interpolation law. For example, using a linear-on-h interpolation law,

$$\eta_i^2(\mathbf{pq}) = \left(1 + \frac{\ell_p(\mathbf{pq})}{\sqrt{\lambda_i}} \ln(\beta)\right)^{-2}.$$

The growth is faster in the direction of the largest eigenvalue (smallest size). As a result, the spanned metric tends to become isotropic in the physical space as it gets far from the source, as observable in Figure 7 (top right frame).

Mixed-space growth This growth consists in combining both previous strategies using a scalar growth factor η_M corresponding to the metric space growth contribution, and a growth tensor $\mathcal{N}_P$ corresponding to the physical space growth effect, weighted by a coefficient t

$$\widetilde{\mathcal{M}}_{\mathbf{p}}(\mathbf{q}) = \mathcal{R}^T(\eta_M^2)^{1-t}\mathcal{N}_P(\mathbf{pq})^t\Lambda\mathcal{R}.$$

This process is illustrated on the bottom-left frame in Figure 7. The spanned metric far from the source appears slightly more anisotropic than in the physical-space growth (for this plot, the parameter t is 0.25).

Directionally-constrained growth The aim of this growth strategy is to slow the growth in the direction of the largest size while the other size increases, until the ratio between these sizes reaches a certain threshold $\rho_{\min}$, close to 1.
Let $\mathcal{M}_p$ be the metric at vertex $\mathbf{p}$ and λ_0, λ_1 its eigenvalues. We assume $\lambda_0 \leq \lambda_1$, i.e. $h_1 \leq h_0$. $\widetilde{\mathcal{M}}_{\mathbf{p}}(\mathbf{q})$ is the grown metric tensor at vertex $\mathbf{q}$ and $\tilde{\lambda}_0 = \tilde{h}_0^{-2}$, $\tilde{\lambda}_1 = \tilde{h}_1^{-2}$ its eigenvalues. The metric $\widetilde{\mathcal{M}}_{\mathbf{p}}(\mathbf{q})$ has to satisfy that its aspect ratio is smaller than a coefficient $\rho_{\min}$, i.e.

$$\frac{\tilde{h}_0}{\tilde{h}_1} \leq \rho_{\min} \;\Leftrightarrow\; \frac{\tilde{\lambda}_1}{\tilde{\lambda}_0} \leq \rho_{\min}^2.$$

Thus, directionally-constrained growth is modeled as

$$\widetilde{\mathcal{M}}_{\mathbf{p}}(\mathbf{q}) = \mathcal{R}^T\widetilde{\Lambda}\mathcal{R},\ \widetilde{\Lambda} = \text{diag}(\tilde{\lambda}_i) \tag{2}$$

with the smallest size (largest eigenvalue) having a physical-space growth with a coefficient η_1

$$\tilde{\lambda}_1 = \eta_1^2(\mathbf{pq})\lambda_1, \tag{3}$$

and

$$\tilde{\lambda}_0 = \begin{cases} \eta_0^2(\mathbf{pq})\lambda_0 & \text{if } \rho_{\min}\tilde{h}_1 > \tilde{h}_0 \\ \min\left(\lambda_0, \frac{\tilde{\lambda}_1}{\rho_{\min}^2}\right) & \text{else.} \end{cases} \tag{4}$$

To take into account the difference between the directions the following model is preferred

$$\widetilde{\mathcal{M}}_{\mathbf{p}}(\mathbf{q}) = \mathcal{R}^T\ \widetilde{\Lambda}_{DC}^t\ \widetilde{\Lambda}_{\text{phy}}^{1-t}\ \mathcal{R} \tag{5}$$

where

- $\widetilde{\Lambda}_{DC}$ denotes the tensor gathering the eigenvalues of the directionally-constrained correction as modeled first,
- $\widetilde{\Lambda}_{\text{phy}}$ denotes the tensor gathering the eigenvalues of the physical-space correction, and
- t quantifies the alignment between the edge or direction $\mathbf{pq}$ and the direction of the smallest size, denoted $\mathbf{e}_{\max}$

$$t = \frac{|\mathbf{pq} \cdot \mathbf{e}_{\max}|}{\|\mathbf{pq}\|_2\|\mathbf{e}_{\max}\|_2}.$$

In the representation of this process in Figure 7, bottom-right frame, the chosen threshold is $\rho_{\min} = 1.1$. Looking close to the center, in comparison with physical-space growth, the spanning metric becomes almost isotropic before growing in the small direction, as expected.

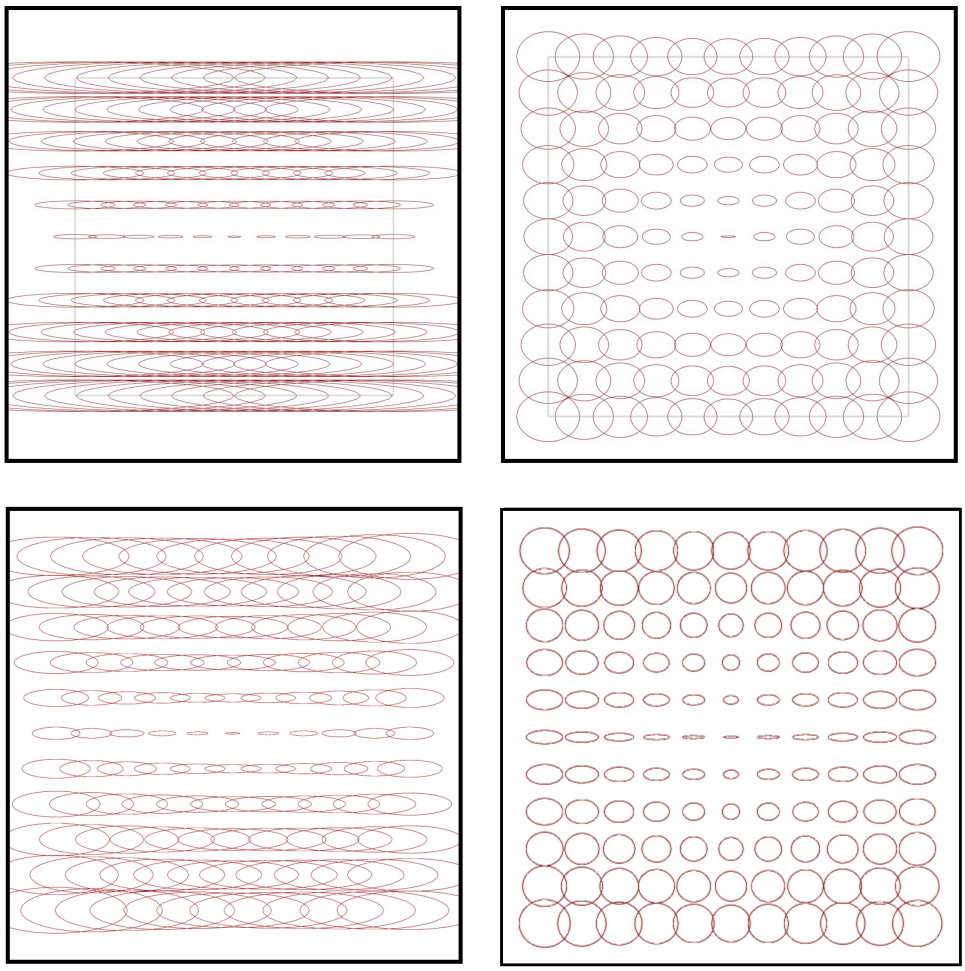

Figure 7. Comparison of four growth processes using the same threshold $\beta = 1.1$. From top to bottom and left to right: metric-space growth, physical-space growth, mixed-space growth and directionally-constrained growth.

C. Structure-aware gradation

This section analyzes the size gradation correction described previously for the purpose of orthogonal or quad-dominant mesh generation. After pointing out the important aspects that should be corrected by a this gradation, we present some measurements to quantify the performance of the correction process. The difference between the several strategies is illustrated on a simple test case. More complex test cases are presented in Section IV.

1. Structure and growth strategies

In standard mesh adaptation, metric-space growth and mixed-space growth have proven to be the best options for the gradation algorithm. However, for mixed-element adaptation, the main goal is to maximize the number, alignment and quality of the quadrilaterals. To do so, abrupt size jumps must be avoided since they lead to obtuse angles at the size transition, as illustrated in Figure 8. Using the *a posteriori* approach detailed in Section V, the corresponding triangles either block the formation of quadrilaterals or are combined into poor quality elements.
Since physical-space growth smoothes the directions as well as the sizes, it has shown to reduce the number of size jumps. To amplify this effect, we have introduced the directionally-constrained growth described earlier, and as expected it gives even better results concerning the number, alignment and quality of the

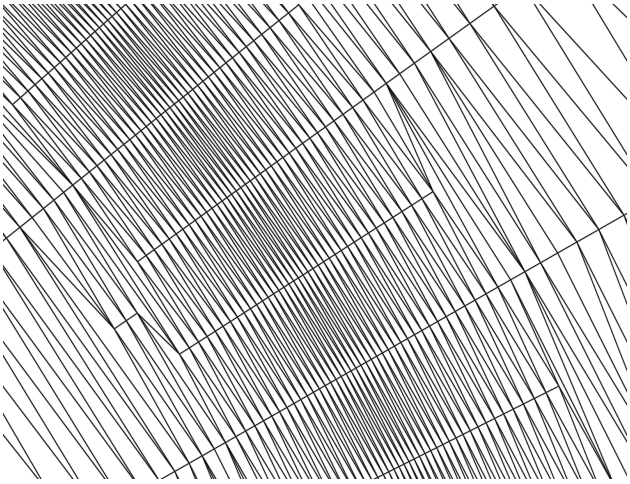

Figure 8. Example of transition obtuse angles

quadrilaterals. The drawback of these two growth strategies is that there is less anisotropy in the mesh, leading to a larger mesh size.

2. Measurement of the growth strategies efficiency

The various growth processes have been compared on some test cases to evaluate their ability to preserve structure in the mesh. To measure their performance, we use the following quantities.

QUADRILATERAL QUALITY We use the angle-based quadrilateral quality from.[21] The quality of a quadrilateral q is

$$\kappa(q) = \max\left(1 - \frac{2}{\pi}\max_k\left(\left|\frac{\pi}{2} - \alpha_k\right|\right), 0\right),$$

where $\alpha_k,\ k \in 1, 4$ denote the angles of the quadrilateral.

SIZE JUMPS INDICATOR This quantity is evaluated at each vertex of the quasi-structured triangular mesh. Let $\mathbf{p}$ be a vertex. Its size jump indicator is the ratio between the largest and the smallest area of the surrounding triangles

$$J(\mathbf{p}) = \frac{\max_{K\ni\mathbf{p}}|K|}{\min_{K\ni\mathbf{p}}|K|},$$

where $|K|$ denotes the area of the triangle $|K|$.
Another useful indicator is the angle repartition in the quasi-structured mesh.

3. Line test case

This first simple test case is an initial discontinuous metric on a rectangular domain $\Omega = [0\ 1] \times [0\ 1]$, to which a metric-space growth and a physical-space growth gradation correction are applied. It shows how these two strategy behave facing an abrupt initial size variation and how directions and orthogonality are propagated from a limited anisotropic area to an isotropic area.
The initial metric field is

$$\forall \mathbf{p} = (x, y) \in \Omega \quad \mathcal{M}(\mathbf{p}) = \begin{cases} \text{diag}(h_1, h_2) & \text{if } y = 0 \\ h_0\ I & \text{otherwise.} \end{cases}$$

The prescribed sizes are $h_1 = 0.1,\ h_2 = 0.001$, and $h_0 = 0.5$.
A gradation correction with $\beta = 1.1$ is applied. The resulting metric-orthogonal meshes are displayed in Figures 9 and 10. The histograms comparing the angle repartition and size jump indicators are shown in

Figure 11. As expected, this example shows clearly the difference between the two methods. The physical space growth seems to favor the formation of quadrilaterals in this case. Indeed, the histograms show that the proportion of size jumps and obtuse angles is higher in the metric-space mesh. This is corroborated by the corresponding quad-dominant meshes, since the proportion of quadrilaterals is 93.8% using the metric space growth and 70.5% using the physical space growth.

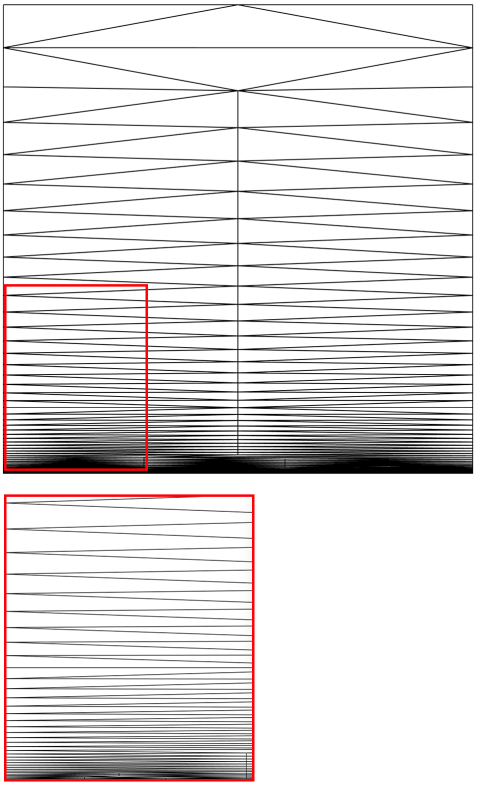

Figure 9. Resulting metric-orthogonal mesh using a metric-space growth gradation for the line case, large scale view and close-up view.

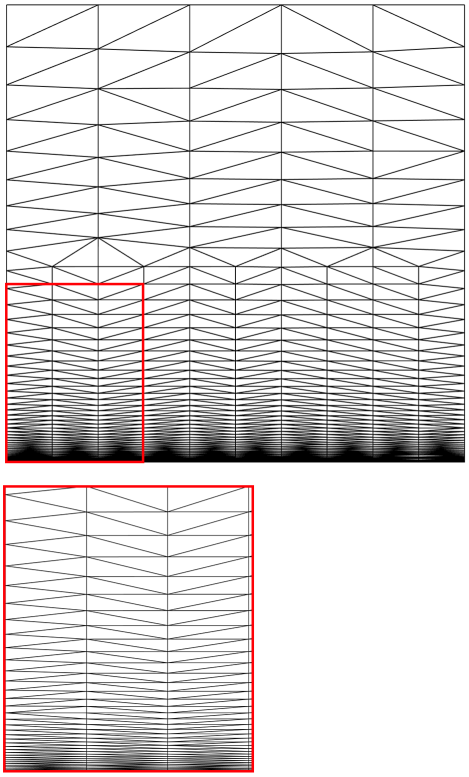

Figure 10. Resulting metric-orthogonal mesh from a physical-space growth gradation for the line case, large scale view and close-up view.

IV. Mixed-element mesh adaptation : *a posteriori* method

In this section, we describe the algorithm of quadrilateral recovery by combination from the metric-orthogonal adapted mesh. Then, this process is coupled with the gradation correction from last section and illustrated on a few examples to discuss the method.

A. Quadrilateral combination method

The strategy presented here classifies as an indirect method for quadrilateral mesh generation. Our algorithm is quite close to[3] or,[20] which sort the mesh edges according to the quality of the quadrilateral formed by the two surroundings triangles, then form the best quadrilaterals. These algorithms give satisfying results for unstructured meshes. Though, we want to take advantage of the specific pattern of metric-orthogonal point placement. The chosen quadrilateral combination procedure aims at forming in priority the quadrilaterals with the strongest aspect ratio and following the smallest size.

1. Add the triangles to a heap-list, sorted by aspect ratio
2. Remove first triangle from the heap-list until it reaches a threshold aspect ratio r_1

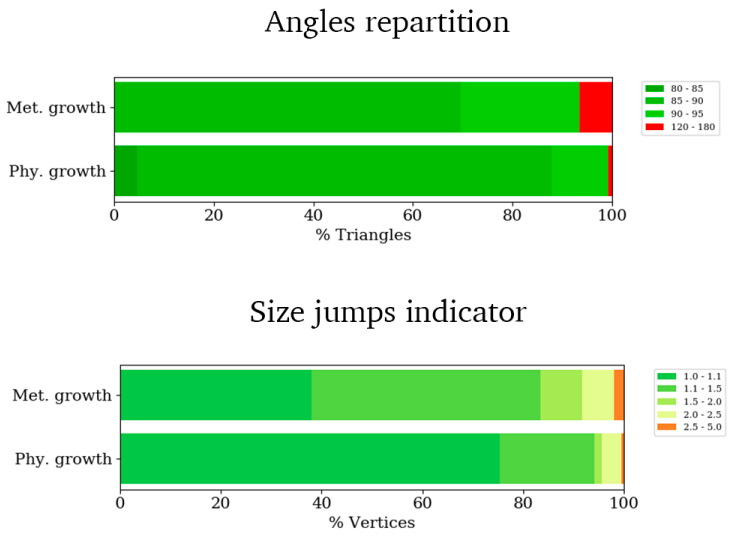

Figure 11. Comparison of the angle repartition and size jump indicator of metric-orthogonal meshes using a metric-space growth and a physical-space growth gradation, for the line case.

(a) Form quadrilaterals from this triangle in the direction of the smallest metric size prescription, until the quality of the formed quadrilaterals reach a threshold r_2

The result of this strategy is compared to the one described in[3] in Figures 12 and 13. It is observable that, when the topology is favorable, our strategy can completely recover the structured elements where there are mostly orthogonal triangles. The other approach mostly relies on a geometrical criterion so it is likely to leave some isolated triangles. Due to our threshold, our algorithm didn't form quadrilaterals far from the boundary layer. It is not very suited yet to fully unstructured mesh, but we consider to couple it with a classical algorithm like[3] or a perfect matching.[21]

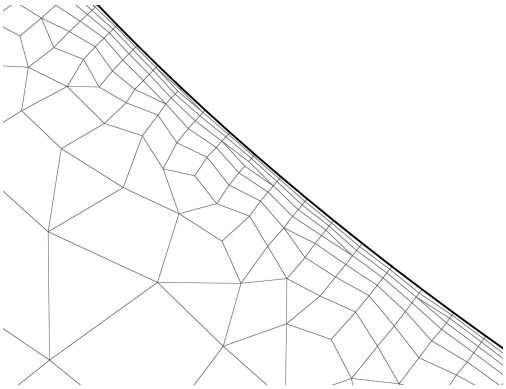

Figure 12. Quadrilateral combination using the approach of Borouchaki and Frey[3] on a metric-orthogonal adapted mesh from a RANS simulation, close-up view on the boundary layer.

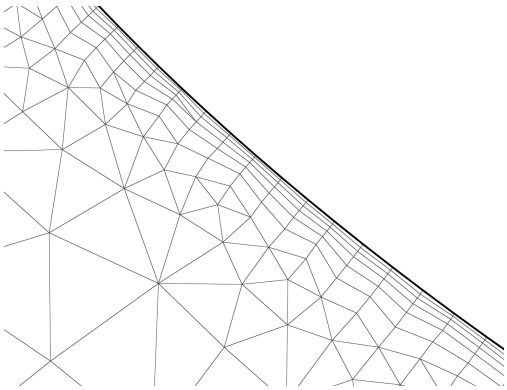

Figure 13. Quadrilateral combination using our approach on a metric-orthogonal adapted mesh from a RANS simulation, close-up view on the boundary layer.

However, experience has shown that the accuracy of our simulations was more affected by isolated triangles in the structured parts than by the overall proportion of quadrilaterals. We chose not to get rid of these isolated triangles similarly to,[3,20,21] since it tends to break the alignment in metric-orthogonal meshes. Our isolated triangles are mostly a consequence of size transitions, which we reduced using a specific gradation correction as described in previous Section III. They are also due to unfavorable connectivity, that we expect

to correct at the point-placement step, similarly to the *a priori* strategy.

B. Circle test case

This example is meant to study how metric curvature is handled by the process. Our method is applied to an analytical metric field representing a curved anisotropic feature having the shape of a circle

$$\mathcal{M}(x,y) = \begin{pmatrix} h_1^{-2}\cos^2\theta + h_2^{-2}\sin^2\theta & (h_1^{-2} - h_2^{-2})\cos\theta\sin\theta \\ (h_1^{-2} - h_2^{-2})\cos\theta\sin\theta & h_1^{-2}\sin^2\theta + h_2^{-2}\cos^2\theta \end{pmatrix},$$

with $\theta = \arctan(x,y)$, $h_1 = \min(0.002 \times 5^{\alpha}, h_{\max})$, $h_2 = \min(0.05 \times 2^{\alpha}, h_{\max})$, $h_{\max} = 0.1$ and $\alpha = 10 \times |0.75 - \sqrt{x^2+y^2}|$.
Here, we take $\beta = 1.2$. To simulate an exponential growth, this coefficient is multiplied at each iteration by a coefficient δ (here $\delta = \beta = 1.2$). As presented in,[1] the dependency on mesh topology creates some 'rays' that are tangent to the curved areas, leading the metric field to be wrongfully modified, and so are the resulting meshes. To overcome this issue, the threshold β is multiplied by a coefficient at each iteration.
All growth processes give satisfying results : metric-space growth gives 90.9% quadrilaterals and physical-space growth gives 92.3%. Directionally-constrained growth gives 81.4% quadrilaterals but the histograms are better in the latter case, see Figure 17. Physical-space and directionally-constrained growth show less size jumps in the radial direction. Consequently, quadrilateral-only areas are more easily formed, as observed in Figures 14 and 16, which is more suitable for numerical simulations. The results for metric-space growth are somehow not as good as the mesh obtained without gradation, which is not very surprising since the initial metric is quite smooth and bounded, so the improvement due to size gradation correction is limited. However, directionally-constrained gradation seem to give considerably better results on quadrilateral quality as shown in the histograms in Fig 17, in spite of the loss of anisotropy that cannot be avoided with this method.

C. Adapted turbulent flow simulation around a NACA airfoil

We consider a turbulent flow simulation on a NACA0012 modeled by the Reynolds-Averaged Navier-Stokes equations, with the Spalart-Allmaras one equation turbulence model. The simulated flow is subsonic with a Mach number $M = 0.5$, a Reynolds Number $Re = 10^5$ and an incidence angle $\alpha = 0$. The adapted meshes are shown in Figures 18, 19, 20 and the angles and size jumps histograms are presented in Figure 21. This case is again better handled by the metric gradation with physical-space growth and directionally-constrained growth. The effect is particularly observable in the boundary layer region, where the smoothness of the transitions is essential. The small size prescribed at the boundary is propagated throughout almost the entire boundary layer, whereas numerous transitions break the alignment when using the metric-space growth gradation.

D. Improvement of the *a posteriori* method

The mixed-element mesh generated using this method can be improved in various ways, through geometric or topologic optimizations to correct the "wavy pattern" that appear in the quadrilateral area. However, these optimization steps are redundant with some operations of the generation step. Besides, some information on orthogonality is given by the point-placement step and is lost in the output quasi-structured triangular mesh. In next section, we describe a structure on which *a priori* optimization and an *a priori* quadrilateral generation can be based.

V. A priori approach: strategy and preliminary results

As seen in section II, metric-orthogonal point-placement is used to create a list of points to be inserted in the mesh. A connectivity graph can be created using the placement and the filtering step. This is not a mesh but it can be used, in this approach, to

- detect quadrilaterals beforehand,
- ensure the corresponding edges are in the final mesh,

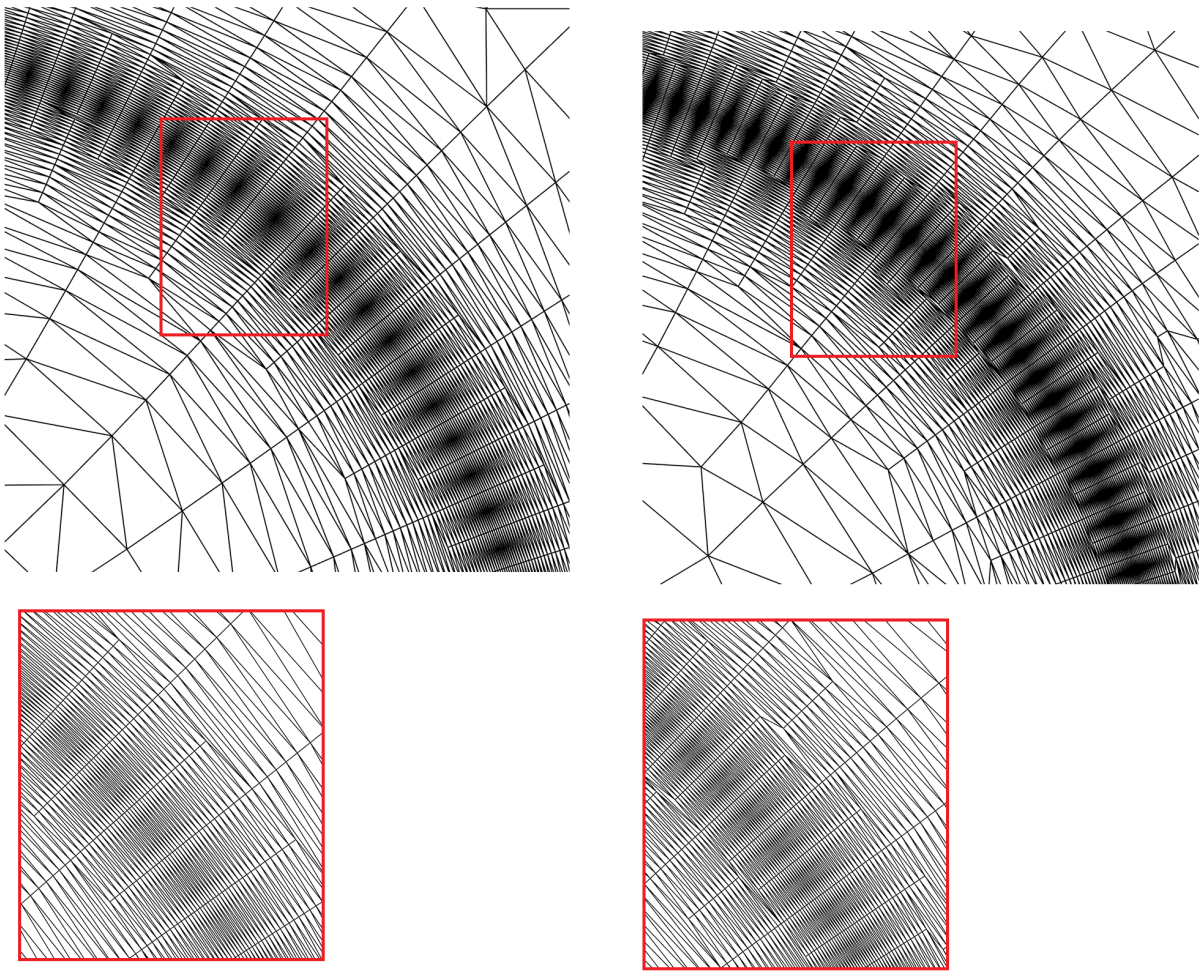

Figure 14. Metric-orthogonal mesh using physical-space gradation, for the circle test case.

Figure 15. Metric-orthogonal mesh using metric-space gradation, for the circle test case.

- perform preliminary topological optimization,
- improve quadrilateral quality.

We first detail the creation of the connectivity edge graph, then present some options for mesh optimization and quadrilateral detection.

A. Edge graph construction algorithm

1. **Initialization**: add the vertices and corresponding directions from the boundary (initial front) to a heap list sorted by the associated size to the directions.
2. **Point creation and graph construction**: While the heap list is not empty
   (a) Remove the point $\mathbf{p}$ at the top of the heap list. It is associated with a size h and direction $\mathbf{e}$.
   (b) In this direction, create two points

$$\mathbf{p}_i = \mathbf{p} + (-1)^i h\ \mathbf{e},\ i = 1, 2$$

   (c) If a point is valid and not filtered by an existing point, it is added to the heap list. The link $[\mathbf{p}\ \mathbf{p}_i]$ is added to the graph.

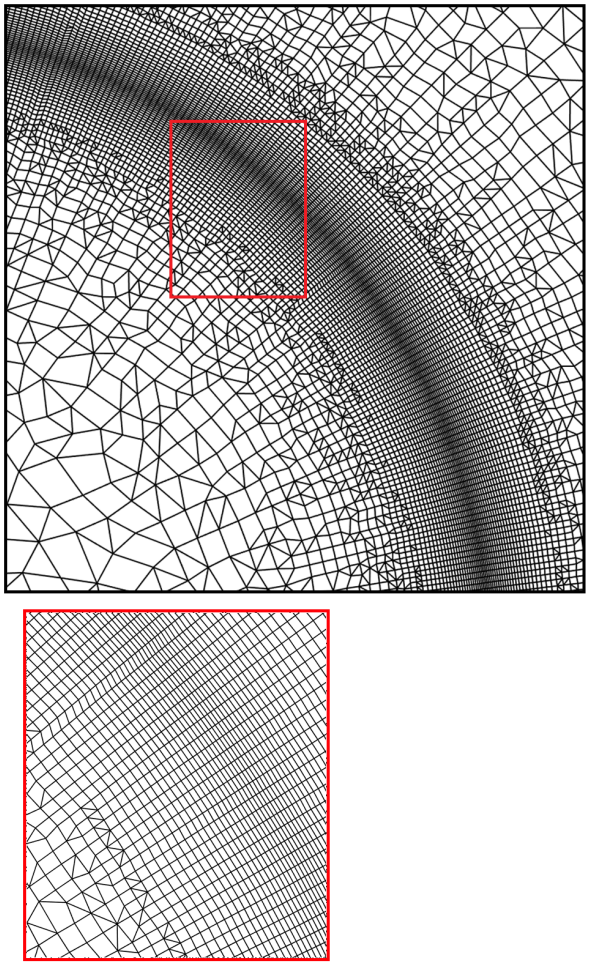

Figure 16. mixed-element mesh from directionally-constrained gradation, for the circle test case.

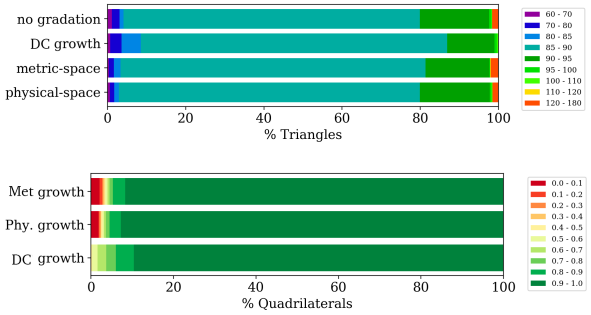

Figure 17. Circle: Angles (top) and quadrilateral quality (bottom) in mixed-element meshes resulting from directionally constrained (DC), metric space and physical space gradation processes.

(d) If a point is filtered by an existing point **q**, the link [**p q**] is added to the graph.

3. **Link the edges**: build a chained list for each vertex of all the edges it is linked to

4. **Graph optimization and quadrilateral recovery**

B. Connectivity and geometric optimization in the edge graph

Optimization of the connectivity can be first applied on the edge graph to recover the cartesian pattern induced by metric-orthogonal point-placement. The edges representing point propositions follow well-defined directions, whereas the edges from filtering only give information on the connectivity. We perform an optimization by moving the vertices along the point proposition edges to align the filtering ones. A simple example is shown in Figure 22.

C. Quadrilateral detection

Once the topology is corrected, we can extract the quadrilaterals from the graph. The idea is to detect these quadrilaterals before the insertion, to save this information and ensure the vertices and edges that belong to quadrilaterals will appear in the final mesh.

1. Loop on the edges of the graph.

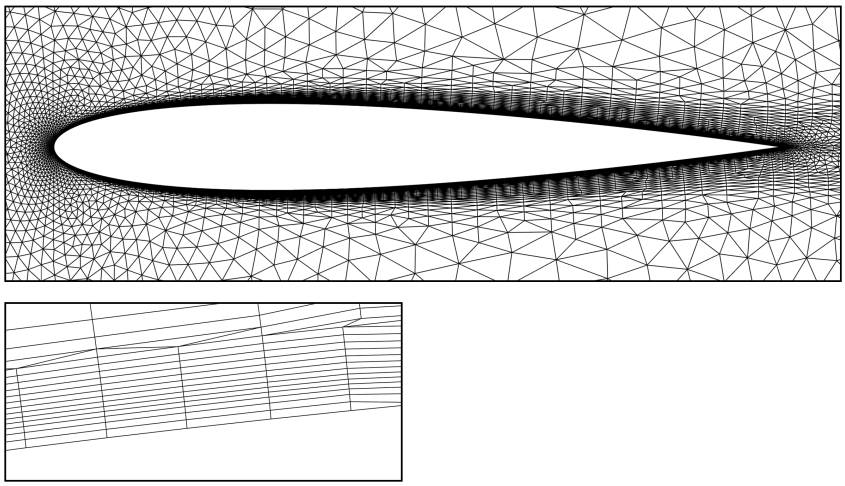

Figure 18. Mixed-element mesh from physical-space gradation, adapted viscous fluid simulation.

2. For each edge, form the two quadrilaterals (or one quadrilateral if the edge is on the boundary) using the connectivity information and some geometric information.

3. Check the validity and the quality of the quadrilaterals and keep it if it is valid and over a chosen quality threshold.

4. If the quadrilateral is valid, mark the vertices and edges.

VI. Conclusion

In this paper, we have described a method to generate quad-dominant adapted meshes, using some tools of metric-based adaptation. This method relies on the metric-orthogonal point-placement, which provides a list of points following a cartesian pattern if the directions are well-defined and if the metric field is smooth enough. To regularize the metric-field, a gradation correction process is applied beforehand. For the purpose of quasi-structured or quad-dominant meshing, the gradation correction process can be upgraded to favor smooth transitions and reduce obtuse angles. The improvement was illustrated on a few examples.
To recover the quadrilaterals, the most intuitive strategy is the combination of the orthogonal triangles favored by the point-placement, into quadrilaterals. To achieve this step, we need to retrieve directional and geometrical information, which can be redundant with the point-placement step. Moreover, the final mesh would often need optimization to improve the alignment, and the tested methods have not proven to be satisfactory. For example, the efficiency of geometric optimization operators is likely to be limited if the mesh connectivity is not optimal. As a complementary approach, we have set up an edge-graph structure at the point-placement step, to help optimize the orthogonality and the alignment as a pre-processing step. At this point, we can access and modify the mesh connectivity and geometry more easily than on the final mesh. This method still needs development but it seems more encouraging than *a posteriori* optimizations for structured meshes.

References

[1]Frédéric Alauzet. Size gradation control of anisotropic meshes. *Finite Elements in Analysis and Design*, 46:181–202, 02 2010.
[2]David Bommes, Bruno Lévy, Nico Pietroni, Enrico Puppo, Claudio Silva, Marco Tarini, and Denis Zorin. Quad-Mesh Generation and Processing: A Survey. *Computer Graphics Forum*, 32(6):51–76, 2013.
[3]Houman Borouchaki and Pascal Frey. Adaptive Triangular-Quadrilateral Mesh Generation. Research Report RR-2960, INRIA, 1996.
[4]Houman Borouchaki, Frédéric Hecht, and Pascal J. Frey. Mesh gradation control. *International Journal for Numerical Methods in Engineering*, 43(6):1143–1165, 1998.
[5]M. J. Castro-Díaz, F. Hecht, B. Mohammadi, and O. Pironneau. Anisotropic unstructured mesh adaption for flow simulations. *International Journal for Numerical Methods in Fluids*, 25(4):475–491, 1997.

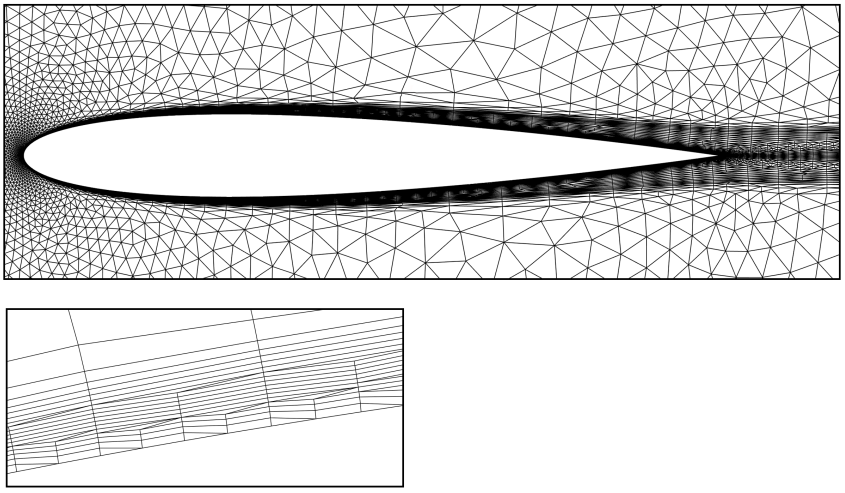

Figure 19. Mixed-element mesh from metric-space gradation, adapted viscous fluid simulation.

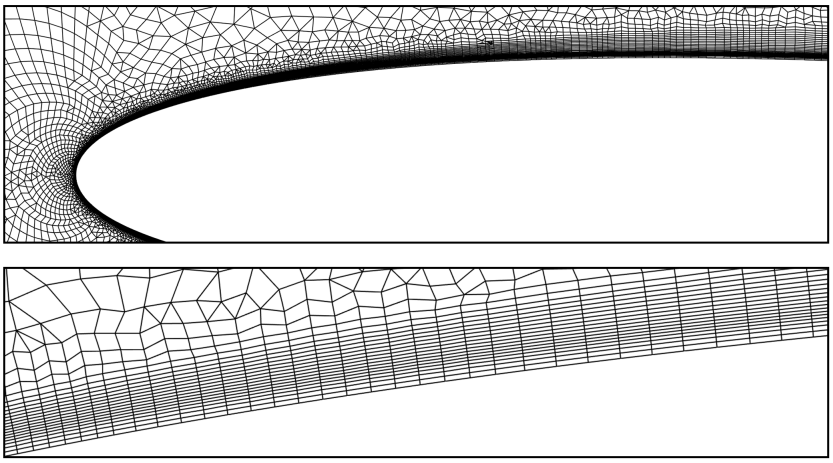

Figure 20. Mixed-element mesh from directionally-constrained gradation, adapted viscous fluid simulation.

[6]Boris Diskin and James L. Thomas. Comparison of node-centered and cell-centered unstructured finite-volume discretizations: Inviscid fluxes. *AIAA Journal*, 49(4):836–854, 2011.

[7]Dirk Ekelschot, Marco Ceze, Anirban Garai, and Scott M. Murman. *Robust metric aligned quad-dominant meshing using Lp centroidal Voronoi tessellation.*

[8]Paul Louis George, Houman Borouchaki, Frederic Alauzet, Patrick Laug, Adrien Loseille, and Loic Marechal. *Meshing, Geometric Modeling and Numerical Simulation, Volume 2: Metrics, Meshes and Mesh Adaptation.* John Wiley & Sons, 2019.

[9]F. Hecht, B. Mohammadi, F. Hecht, and B. Mohammadi. *Mesh adaption by metric control for multi-scale phenomena and turbulence.*

[10]F. John. Extremum problems with inequalities as subsidiary conditions. *Studies and essays presented to R. Courant on his 60th birthday*, pages 187–204, 1948.

[11]Bruno Lévy and Yang Liu. Lp centroidal voronoi tessellation and its applications. *ACM Trans. Graph.*, 29(4), July 2010.

[12]X. Li, M. S. Shephard, and M. W. Beall. 3d anisotropic mesh adaptation by mesh modification. *Computer methods in applied mechanics and engineering*, 194(48-49):4915–4950, 2005.

[13]Yang Liu, Wenping Wang, Bruno Lévy, Feng Sun, Dong-Ming Yan, Lin Lu, and Chenglei Yang. On Centroidal Voronoi Tessellation–Energy Smoothness and Fast Computation. *ACM Transactions on Graphics*, 28(4):Article 101, August 2009.

[14]Adrien Loseille. *Adaptation de maillage anisotrope 3D multi-échelles et ciblée à une fonctionnelle pour la mécanique des fluides. Application à la prédiction haute-fidélité du bang sonique.* Theses, Université Pierre et Marie Curie - Paris VI, December 2008.

[15]Adrien Loseille. Metric-orthogonal anisotropic mesh generation. *Procedia Engineering*, 82:403 – 415, 2014. 23rd International Meshing Roundtable (IMR23).

[16]Adrien Loseille and Frédéric Alauzet. Continuous mesh framework part i: Well-posed continuous interpolation error. *SIAM*

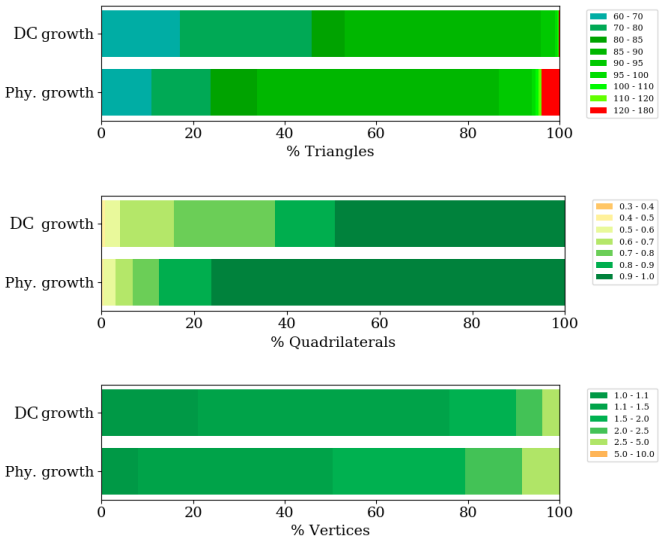

Figure 21. Angle repartition, quality and size jump repartition for physical-space and directionally-constrained (DC) gradation, adapted viscous fluid simulation.

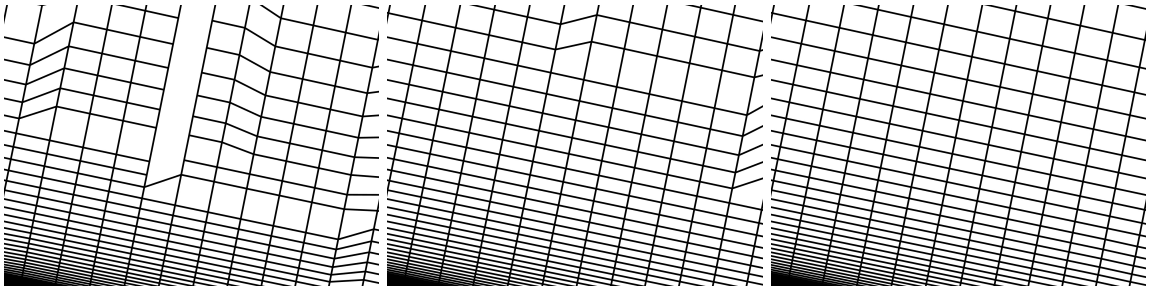

Figure 22. Iterative *a priori* optimization illustrated on a simple case (metric from the line example).

J. Numerical Analysis, 49:38–60, 01 2011.
[17]Adrien Loseille and Frédéric Alauzet. Continuous mesh framework part ii: Validations and applications. *SIAM J. Numerical Analysis*, 49:61–86, 01 2011.
[18]Adrien Loseille and Rainald Lohner. *Cavity-Based Operators for Mesh Adaptation.*
[19]D. Marcum and F. Alauzet. 3d metric-aligned and orthogonal solution adaptive mesh generation. *Procedia Engineering*, 203:78 – 90, 2017. 26th International Meshing Roundtable, IMR26, 18-21 September 2017, Barcelona, Spain.
[20]Ernst Rank, Manfred Schweingruber, and Markus Sommer. Adaptive mesh generation and transformation of triangular to quadrilateral meshes. *Communications in Numerical Methods in Engineering*, 9(2):121–129, 1993.
[21]J.-F. Remacle et al. Blossom-quad: A non-uniform quadrilateral mesh generator using a minimum-cost perfect-matching algorithm. *International Journal for Numerical Methods in Engineering*, 89:1102–1119, 2012.
[22]Mahkame Sharbatdar and Carl Ollivier Gooch. *Anisotropic Mesh Adaptation: Recovering Quasi-structured Meshes.*
[23]Jasmeet Singh and Carl F. Ollivier Gooch. *Advancing Layer Surface Mesh Generation.*