V European Conference on Computational Fluid Dynamics
ECCOMAS CFD 2010
J. C. F. Pereira and A. Sequeira (Eds)
Lisbon, Portugal,14-17 June 2010

# AN L$^{\infty}$-L$^{P}$ SPACE-TIME ANISOTROPIC MESH ADAPTATION STRATEGY FOR TIME-DEPENDENT PROBLEMS

**F. Alauzet**[*] **and G. Olivier**[†]

[*†]INRIA, projet Gamma
Domaine de Voluceau - Rocquencourt B.P.105, 78135 Le Chesnay, France
e-mail: frederic.alauzet@inria.fr and geraldine.olivier@inria.fr

**Key words:** Anisotropic mesh adaptation, Hessian-based, Metric, CFD, Unsteady flow

**Abstract.** *This work presents a first extension of the multi-scale mesh adaptation theory, where the interpolation error is controlled globally in* $\mathbf{L}^p$*-norm, to unsteady flows. The extension follows the main lines of the transient fixed-point mesh adaptation scheme. To avoid to adapt the mesh at each solver time step, the simulation time frame is split into several adaptation sub-intervals and adapted meshes are generated for each sub-intervals. We thus propose a* $\mathbf{L}^{\infty}$-$\mathbf{L}^p$ *error estimate on each adaptation sub-intervals. To solve the non-linear problem of mesh adaptation, we consider a global fixed-point mesh adaptation algorithm that allows us to compute the space-time mesh complexity which is used in the metric normalization. This strategy is applied on a blast model problem.*

## 1 INTRODUCTION

The prediction of physical multidimensional processes involving discontinuities, when addressed with non mesh-adaptive approximations, shows generally a poor convergence rate to exact solution, this rate being typically limited to first-order or worse. Conversely, a good mesh-adaptive approximation should recover higher order accuracy, and more precisely enjoy the following Early Capturing (EC) property: *"a mesh adaptive method enjoys the Early Capturing property if it is able to converge to discontinuous solutions at the same rate as for smooth solutions".* The considered error sensor is the main term of the interpolation error of a typical solution field. The interpolation error is controlled globally in $\mathbf{L}^p$-norm, generally $\mathbf{L}^2$ is chosen, which provides a mesh size prescription depending on the Hessian of the selected solution field. A fixed point is applied for solving the non-linear interaction between mesh and sensor. It can be demonstrated that such an adaptive process is second-order convergent when the mesh is adapted for best $P_1$ interpolant. This property for second-order flow solver has been proved and illustrated in the case of steady inviscid flows [3]. In this work, we propose a first extension to unsteady flows.

This presentation discusses the contribution of anisotropic mesh adaptation to high-order convergence of *unsteady flow simulations* on complex geometries. Trying to recover the theoretical second-order convergence for time-dependent problems relies on four fundamental points:

- a $\mathbf{L}^p$-norm control of the interpolation error, as already applied for steady flows [3]
- a transient fixed-point mesh adaptation algorithm such as in [1]
- a $\mathbf{L}^\infty(0,T;\mathbf{L}^p(\Omega))$ space-time adaptation criterion to manage the space-time mesh on each adaptation sub-interval
- a $P_1$-conservative solution interpolation between two remeshing [2].

We discuss how a global fixed-point mesh adaptation algorithm for unsteady flows can be designed in order to satisfy the EC condition based on a $\mathbf{L}^\infty(0,T;\mathbf{L}^p(\Omega))$ space-time adaptation criterion. The $\mathbf{L}^\infty$ strategy consists in requiring that each mesh associated with adaptation sub-interval $[t_i, t_{i+1}]$ fulfills a specified $\varepsilon$-error criterion for solution state in $[t_i, t_{i+1}]$. Moreover, the size of each adaptation time interval $[t_i, t_{i+1}]$ can become an extra discretisation parameter to be specified in the algorithm for convergence analysis to improve the space-time convergence. A variation law of the time interval with respect to the prescribed error is exhibited to satisfy the EC condition. A blast model example is provided to illustrate the efficiency of the approach.

## 2 MULTI-SCALE ANISOTROPIC MESH ADAPTATION FOR STEADY PROBLEMS

For a given continuous function $u$, we denote by $\Pi_{\mathcal{H}} u$ the $P_1$-interpolation of $u$, that is the element of $V$ such that $\Pi_{\mathcal{H}} u(\mathbf{x}_i) = u(\mathbf{x}_i)$. In this study, we start with the usual assumption in metric-based methods: *mastering the* $\mathbf{L}^p$ *norm of the* $P_1$**-interpolation error** *of solution field $u$ is enough to control the global* **approximation error**.
We first recall the metric-based generation of anisotropic adapted meshes. Second, the analysis to seek for the best mesh for controlling the interpolation in $\mathbf{L}^p$ norm is given.

### 2.1 Metric-based generation of anisotropic adapted meshes

The metric-based generation of anisotropic adapted meshes uses the notion of Riemannian metric space [7]. For a computational domain $\Omega \subset \mathbb{R}^d$, a Riemannian metric space $(\mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\Omega}$ is a spatial field that defines at any point of $\Omega$ a metric tensor $\mathcal{M}(\mathbf{x})$, e.g. a $d \times d$ symmetric definite positive matrix. Then, it is possible for a mesh generator to work in this Riemannian metric space instead of the canonical Euclidean space where the dot product is defined locally by metric tensor $\mathcal{M}$: $\langle \mathbf{u}, \mathbf{v} \rangle_{\mathcal{M}} = \langle \mathbf{u}, \mathcal{M}\mathbf{v} \rangle$ for $(\mathbf{u}, \mathbf{v}) \in \mathbb{R}^3 \times \mathbb{R}^3$. In that case, the length of edge $\mathbf{ab}$ is computed using the straight line parameterization $\gamma(t) = \mathbf{a} + t\,\mathbf{ab}$, where $t \in [0, 1]$:

$$\ell_{\mathcal{M}}(\mathbf{ab}) = \int_0^1 \|\gamma'(t)\|_{\mathcal{M}}\,\mathrm{d}t = \int_0^1 \sqrt{{}^t\mathbf{ab}\;\mathcal{M}(\mathbf{a} + t\,\mathbf{ab})\;\mathbf{ab}}\,\mathrm{d}t\,,$$

and the volume of element $K$ is:

$$|K|_{\mathcal{M}} = \int_K \sqrt{\det \mathcal{M}(\mathbf{x})}\,\mathrm{d}\mathbf{x}\,.$$

It is important to note that, in a Riemannian metric space, computing the length of a segment (*i.e.,* an edge) differs from evaluating the distance between the extremities of this segment. Indeed, the straight line is no more the shortest path between two points, which is now given by a geodesic.

The main idea of metric-based mesh adaptation, initially introduced in [8], is to generate *a unit mesh* in the prescribed Riemannian metric space, e.g. a mesh of $\Omega \subset \mathbb{R}^3$ such that each edge has a unit length and each tetrahedron is regular for $(\mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\Omega}$:

$$\forall \mathbf{e},\;\; \ell_{\mathcal{M}}(\mathbf{e}) = 1 \quad \text{and} \quad \forall K,\;\; |K|_{\mathcal{M}} = \frac{\sqrt{2}}{12}\,.$$

As it is not possible to tesselate $\mathbb{R}^3$ with the regular tetrahedron, we look for a mesh such that all edges have a length close to unity and such that all elements are almost regular in the Riemannian metric space .

### 2.2 Multi-scale anisotropic mesh adaptation

We seek for the best mesh approximating the smooth solution $u$ of a steady scalar PDE over a domain $\Omega$ of $\mathbb{R}^d$. We simplify our investigation into the specification of a mesh that is optimal for the interpolation error.

Multi-scale mesh adaptation relies on the minimization of the interpolation error in the spatial norm $\mathbf{L}^p(\Omega)$, in contrast to the minimization of the $\mathbf{L}^\infty(\Omega)$ norm of interpolation error which is equivalent to equi-distribute the spatial error. Minimizing the $\mathbf{L}^p(\Omega)$ norm of the interpolation error is an active field of research [1, 5, 6, 9]. We follow now the continuous mesh formulation introduced in [11] and used in [3, 12, 14].

The considered problem of mesh adaptation consists in finding the mesh $\mathcal{H}$ of $\Omega$ that minimizes the linear interpolation error $u - \Pi_\mathcal{H} u$ controlled in $\mathbf{L}^p$ norm. The problem is thus stated in an *a priori* way:

$$\text{Find } \mathcal{H}_{\mathbf{L}^p} \text{ having } N \text{ nodes such that } E_{\mathbf{L}^p}(\mathcal{H}_{\mathbf{L}^p}) = \min_{\mathcal{H}} \|u - \Pi_\mathcal{H} u\|_{\mathbf{L}^p(\Omega_h)}\,. \tag{1}$$

As it, Problem (1) is a global combinatorial problem which turns out to be intractable practically. Indeed, both topology and vertices locations need to be optimized. This ill-posed problem can be reformulated in the continuous mesh framework [11, 12]. In this framework, we propose the following continuous model of a mesh. A continuous mesh $\mathbf{M}$ of a domain $\Omega$ is identified to a Riemannian metric space $\mathbf{M} = (\mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\boldsymbol{\Omega}}$ that, at each point $\mathbf{x}$ of $\Omega$, prescribes a density, a set of anisotropy directions and the stretching along these directions, this information being contained in metric tensor $\mathcal{M}(\mathbf{x})$. Continuous mesh $\mathbf{M}$ defines a class of equivalence which is the set of unit discrete meshes for $\mathbf{M}$. We also defines a continuous model of the linear interpolation operator $\Pi_\mathcal{H}$ denoted $\pi_\mathcal{M}$. It is then possible to set the well-posed global optimization problem of finding the optimal continuous mesh minimizing the continuous interpolation error in $\mathbf{L}^p$ norm:

$$\text{Find } \mathbf{M}_{\mathbf{L}^p} \text{ such that } \mathcal{E}_{\mathbf{L}^p}(\mathbf{M}_{\mathbf{L}^p}) = \min_{\mathbf{M}} \left( \int_\Omega (u(\mathbf{x}) - \pi_\mathcal{M} u(\mathbf{x}))^p \,\mathrm{d}\mathbf{x} \right)^{\frac{1}{p}}, \tag{2}$$

under the constraint

$$\mathcal{C}(\mathbf{M}) = \int_\Omega \sqrt{\det \mathcal{M}(\mathbf{x})}\mathrm{d}\mathbf{x} = N. \tag{3}$$

which models the total number of nodes. According to [11], if $\mathcal{H}$ is a unit mesh with respect to $\mathbf{M}$ and $u$ is a smooth function, then the following bound holds:

$$\|u - \Pi_\mathcal{H} u\|_{\mathbf{L}^p(\Omega_h)} \le \|u - \pi_\mathcal{M} u\|_{\mathbf{L}^p(\Omega)} = \left( \int_\Omega \left( \operatorname{trace}(\mathcal{M}^{-\frac{1}{2}}(\mathbf{x}) |H_u(\mathbf{x})| \mathcal{M}^{-\frac{1}{2}}(\mathbf{x})) \right)^p \mathrm{d}\mathbf{x} \right)^{\frac{1}{p}}, \tag{4}$$

where $H_u$ is the Hessian of $u$ and $|H_u|$ the matrix deduced from $H_u$ by taking the absolute value of its eigenvalues. Writing the optimality conditions provides the unique (by convexity) optimal continuous mesh $\mathbf{M}_{\mathbf{L}^p} = (\mathcal{M}_{\mathbf{L}^p}(\mathbf{x}))_{\mathbf{x}\in\Omega}$ solution of Problem (2) under Constraint (3):

$$\mathcal{M}_{\mathbf{L}^p} = D_{\mathbf{L}^p} \left(\det |H_u|\right)^{-\frac{1}{2p+d}} |H_u| \quad \text{with} \quad D_{\mathbf{L}^p} = N^{\frac{2}{d}} \left(\int_\Omega (\det |H_u(\mathbf{x})|)^{\frac{p}{2p+d}} \mathrm{d}\mathbf{x}\right)^{-\frac{2}{d}}. \tag{5}$$

In Relation (5), we find three ingredients. First, the constant $D_{\mathbf{L}^p}$ defines the global mesh accuracy imposed by Constraint (3). Second, the matrix term $|H_u|$ specifies the local mesh orientation and anisotropy (or stretching). Third, the scalar term $(\det |H_u|)^{\frac{-1}{2p+d}}$ modifies the local mesh density to take into account the sensitivity of the $\mathbf{L}^p$ norm used in the error estimate. Therefore, the $\mathbf{L}^p$ norm choice is essential in a mesh adaptation process depending on the problem at sake. First, in our context, it is mandatory that $p$ is a bounded number. Indeed, with the usual option $p = \infty$, which means an equi-distribution of the local error, the interpolation error does not converge when the solution is discontinuous. Secondly, CFD fields usually involve different scales. The metrics constructed with a lower $p$ norm are more sensitive to weaker variations of the solution whereas the $\mathbf{L}^\infty$ norm mainly concentrates on the strongest singularities.

Now, to obtain an optimal discrete mesh $\mathcal{H}_{\mathbf{L}^p}$, it is sufficient to generate a unit mesh with respect to $\mathcal{M}_{\mathbf{L}^p}$ thanks to Relation (4). Finally, Bound (4) can be rewritten for $\mathbf{M}_{\mathbf{L}^p}$, the following bound follows up for a unit mesh $\mathcal{H}_{\mathbf{L}^p}$ with respect to $\mathbf{M}_{\mathbf{L}^p}$:

$$E(\mathcal{H}_{\mathbf{L}^p}) = \|u - \Pi_{\mathcal{H}_{\mathbf{L}^p}} u\|_{\mathbf{L}^p(\Omega_h)} \leq \mathcal{E}(\mathbf{M}_{\mathbf{L}^p}) = d N^{-\frac{2}{d}} \left(\int_\Omega (\det |H_u(\mathbf{x})|)^{\frac{p}{2p+d}} \mathrm{d}\mathbf{x}\right)^{\frac{2p+d}{dp}} \leq \frac{Cst}{N^{2/d}}. \tag{6}$$

A main result arises from the previous bound: a global second-order asymptotic mesh convergence is expected for the considered variable $u$. Indeed, a simple analogy with regular grids leads to consider that $N = O\left(h^{-d}\right)$ so that the previous estimate becomes:

$$\|u - \Pi_{\mathcal{H}_{\mathbf{L}^p}} u\|_{\mathbf{L}^p(\Omega_h)} \leq Cst' h^2.$$

For all $p \in [1, \infty[$, it is possible to analyze in which conditions the last integral in (6) is bounded. Following this idea, it is observed in [13, 12], that the second-order convergence property still holds even when some singularities are present in the flow field.

It can be useful to prescribe a target error $\varepsilon$ instead of a target number of vertices $N$. The expression of the optimal metric respecting a given error threshold $\varepsilon$ is deduced from Relations (5) and (6):

$$\mathcal{M}_{\mathbf{L}^p} = D_{\mathbf{L}^p} \left(\det |H_u|\right)^{-\frac{1}{2p+d}} |H_u| \quad \text{with} \quad D_{\mathbf{L}^p} = \frac{d}{\varepsilon} \left(\int_\Omega (\det |H_u(\mathbf{x})|)^{\frac{p}{2p+d}} \mathrm{d}\mathbf{x}\right)^{\frac{1}{p}}. \tag{7}$$

# 3 EXTENSION FOR TIME-DEPENDENT PROBLEMS

## 3.1 The transient fixed-point mesh adaptation scheme

We recall some features of the transient fixed-point mesh adaptation scheme introduced in [2] and explain how it applies to unsteady flows. The simulation time frame $[0, T]$ is split into several adaptation sub-intervals:

$$[0, T] \; = \; [0 = t_0, t_1] \cup ... \cup [t_i, t_{i+1}] \cup ... \cup [t_{n_{adap}-1}, t_{n_{adap}}].$$

The idea is to build, for each sub-interval $[t_i, t_{i+1}]$, a mesh which is adapted to the chosen time evolving features of the solution occurring during the sub-interval. This is done by means of a fixed-point iteration procedure which enables to enhance the capture of the phenomena and the mesh adaptation at these features at each new iterations. The mesh-solution couple is improved in this way until convergence. Then, the next sub-interval is treated. This algorithm reads:

```
Set initial state $(\mathcal{H}^0, \mathcal{S}_0^0)$

//--- Adaptive loop dealing with all sub-intervals successively
For i=1,nadap

 • $\mathcal{M}_{i-1}^{nptfx} =$ ComputeLpMetric$(\mathcal{H}_{i-1}^{nptfx}, \mathcal{S}_{i-1}^{nptfx})$

 • $\mathcal{H}_i^0 =$ GenerateAdaptedMesh$(\mathcal{H}_{i-1}^{nptfx}, \mathcal{M}_{i-1}^{nptfx})$

 • $\mathcal{S}_{0,i}^0 =$ InterpolateSolution$(\mathcal{H}_{i-1}^{nptfx}, \mathcal{S}_{i-1}^{nptfx}, \mathcal{H}_i^0)$

 • $\mathcal{S}_i^0 =$ SolveState$(\mathcal{S}_{0,i}^0, \mathcal{H}_i^0)$

 //--- Fixed-point loop to converge the mesh adaptation on $[t_i, t_{i+1}]$
 For j=1,nptfx

   • $\mathcal{M}_i^{j-1} =$ ComputeLpMetric$(\mathcal{H}_i^{j-1}, \{\mathcal{S}_i^{j-1}(k)\}_{k=1,nk})$

   • $\mathcal{H}_i^j =$ GenerateAdaptedMesh$(\mathcal{H}_i^{j-1}, \mathcal{M}_i^{j-1})$

   • $\mathcal{S}_{0,i}^j =$ InterpolateSolution$(\mathcal{H}_{i-1}^{nptfx}, \mathcal{S}_{i-1}^{nptfx}, \mathcal{H}_i^j)$

   • $\mathcal{S}_i^j =$ SolveState$(\mathcal{S}_{0,i}^j, \mathcal{H}_i^j)$

 End for
End for
```

Now, let us detail how the adaptation process is driven in each sub-interval. At each new solver stage, the solution is advanced from initial time $t_i$ to final time $t_{i+1}$. At each solver time step, the solution field is saved. It provides a sample of the solution on the time sub-interval $[t_i, t_{i+1}]$. If $nk-1$ solver time steps have been performed, we get the solution sample $\{\mathcal{S}_i(k)\}_{k=1,nk}$ with $\mathcal{S}_i(1) = \mathcal{S}(t_i)$ and $\mathcal{S}_i(nk) = \mathcal{S}(t_{i+1})$. A metric field $\mathcal{M}_i(k)$ is computed thanks to Relation (7) from each of these samples. Then, all these metrics are intersected into a single one: $\mathcal{M}_i = \cap_{k=1}^{m} \mathcal{M}_i(k)$. We come up with metric $\mathcal{M}_i$ which specifies a suitable mesh adaptation for the whole sub-interval $[t_i, t_{i+1}]$. In practice, not all metrics are intersected, but a few tens of them. The intersection procedure is explained in [2]. Finally, this metric is provided to the adaptive mesh generator and the initial solution of the sub-interval at time $t_i$ is interpolated to restart the computation. This internal loop is stopped, as for steady case, when the deviation between two successive solutions at $t_{i+1}$ is sufficiently small. The next action is to start the fixed point adaptation loop for $[t_{i+1}, t_{i+2}]$.

### 3.2 An $\mathbf{L}^\infty$-$\mathbf{L}^p$ anisotropic mesh adaptation

In this work, we consider the following framework: ”as an explicit scheme is used for time advancing, then the error in time is controlled by the error in space under CFL condition”, see [2]. As far as that is true, the spatial interpolation error is a good measure of the total space-time error of the discretized unsteady system. Indeed, our model is convection dominated and time accuracy is less or equal two, while space accuracy is of order two. Then, with a Courant-type time-step condition, the spatial error analysis is sufficient to represent the space-time error.

Therefore, we do not consider here a space-time error analysis and we may want to immediately apply the previous error estimate at each time step, providing at each time step to the optimal mesh given by Relation (5) or (7). The multi-scale mesh adaptation theory allows us to control the $\mathbf{L}^p$ norm of the spatial interpolation error at time $t$, *i.e.,* the instantaneous error $\varepsilon(t) = \|u(t) - \Pi_{\mathcal{H}} u(t)\|_{\mathbf{L}^p(\Omega_h)}$. Now, we have to discuss how to optimally control the space-time error for each adaptation sub-interval $[t_i, t_{i+1}]$ as a unique adapted mesh is required by the transient fixed-point mesh adaptation scheme presented in the previous section. We consider a $\mathbf{L}^\infty(t_i, t_{i+1}; \mathbf{L}^p(\Omega))$ space-time adaptation strategy which consists in requiring that the adapted mesh fulfills a specified $\varepsilon$-error criterion for the whole sub-interval $[t_i, t_{i+1}]$. In this context the following pseudo-optimal metric in $\mathbf{L}^p$ norm for the whole sub-interval $[t_i, t_{i+1}]$ reads:

$$\mathcal{M}_{\mathbf{L}^\infty\mathbf{L}^p} = D_{\mathbf{L}^\infty\mathbf{L}^p} \left(\det |H_{\max}|\right)^{-\frac{1}{2p+d}} |H_{\max}| \tag{8}$$

$$\text{with} \quad D_{\mathbf{L}^\infty\mathbf{L}^p} = \frac{d}{\varepsilon} \left( \int_\Omega (\det |H_{\max}(\mathbf{x})|)^{\frac{p}{2p+d}} \mathrm{d}\mathbf{x} \right)^{\frac{1}{p}} \quad \text{and} \quad |H_{\max}(\mathbf{x})| = \max_{t\in[t_i,t_{i+1}]} |H_u(\mathbf{x},t)| \,.$$

To compute the max on the definite positive Hessians (which are metrics), we use a solution sample $\{\mathcal{S}(k)\}_{k=1,nk}$ with $\mathcal{S}(1) = \mathcal{S}(t_i)$ and $\mathcal{S}(nk) = \mathcal{S}(t_{i+1})$, from which a sample of definite positive Hessians is computed $\{|H_u(k)|\}_{k=1,nk}$. The max is evaluated by means of the metric intersection procedure:

$$|H_{\max}| = \bigcap_{k=1}^{nk} |H_u(k)|$$

### 3.3 A space-time mesh complexity

However, when dealing with unsteady flow the previous optimal mesh prescription has the following weakness. If the nondimensionned continuous mesh complexity

$$\int_\Omega (\det |H_{\max}(\mathbf{x})|)^{\frac{p}{2p+d}} \mathrm{d}\mathbf{x}$$

varies between each sub-interval, *i.e.,* in time, then the global normalization constant

$$N^{\frac{2}{d}} \left( \int_\Omega (\det |H_{\max}(\mathbf{x})|)^{\frac{p}{2p+d}} \mathrm{d}\mathbf{x} \right)^{-\frac{2}{d}} \quad \text{or} \quad \frac{d}{\varepsilon} \left( \int_\Omega (\det |H_{\max}(\mathbf{x})|)^{\frac{p}{2p+d}} \mathrm{d}\mathbf{x} \right)^{\frac{1}{p}}$$

varies also in time. Consequently, if new physical phenomena develop in the flow during the simulation this integral grows and thus the global normalization constant diminishes. This means that a physical phenomenon which does not vary in time, such as a moving shock defined by a step function, will be more or less refined depending on the physic of the whole flow.

To solve this issue, we propose to set $N$ as the space-time mesh complexity and the integral becomes the nondimensionned continuous mesh complexity associated with the space-time mesh:

$$\begin{aligned} \mathcal{C}(\mathbf{M}_{\mathbf{L}^\infty \mathbf{L}^p}) &= \int_0^T \left( \int_\Omega \sqrt{\det \mathcal{M}_{\mathbf{L}^\infty \mathbf{L}^p}(\mathbf{x},t)} \mathrm{d}\mathbf{x} \right) \tau^{-1}(t) \mathrm{d}t \\ &= \int_0^T \left( \int_\Omega (\det |H_{\max}(\mathbf{x},t)|)^{\frac{p}{2p+d}} \mathrm{d}\mathbf{x} \right) \tau^{-1}(t) \mathrm{d}t \end{aligned}$$

where $\tau(t)$ is the solver time step at time $t$. As the metric is constant for each sub-interval $[t_i, t_{i+1}]$ and that $m(i) = \frac{t_{i+1}-t_i}{\tau(t)}$ is the number of iterations of the flow solver during the sub-interval, we get:

$$\mathcal{C}(\mathbf{M}_{\mathbf{L}^\infty \mathbf{L}^p}) = \sum_{i=1}^{n_{adap}} m(i) \left( \int_\Omega (\det |H_{\max}(\mathbf{x})|_i\,)^{\frac{p}{2p+d}} \mathrm{d}\mathbf{x} \right)$$

The global normalization constant becomes:

$$\begin{aligned} & N^{\frac{2}{d}} \left( \int_0^T \left( \int_\Omega (\det |H_{\max}(\mathbf{x},t)|)^{\frac{p}{2p+d}} \mathrm{d}\mathbf{x} \right) \tau^{-1}(t) \mathrm{d}t \right)^{-\frac{2}{d}} \qquad (9) \\ \text{or} \quad & \frac{d}{\varepsilon} \left( \int_0^T \left( \int_\Omega (\det |H_{\max}(\mathbf{x},t)|)^{\frac{p}{2p+d}} \mathrm{d}\mathbf{x} \right) \tau^{-1}(t) \mathrm{d}t \right)^{\frac{1}{p}} . \end{aligned}$$

In this case, the mesh size prescription is coherent along the computation.

### 3.4 A global fixed-point mesh adaptation algorithm

In this context, the transient fixed-point mesh adaptation strategy recalled in Section 3.1 cannot be applied anymore as the evaluation of the normalization constant (9) requires to perform the whole simulation. Thus, we propose a new global fixed-point strategy. The new fixed-point unsteady mesh adaptation scheme is:

```
//--- A first guess on the initial mesh
Set initial state $(\mathcal{H}^0, \mathcal{S}_0^0)$
For i=1,nadap
    • $\mathcal{S}_i^0 = \texttt{SolveState}(\mathcal{H}^0, \mathcal{S}_{i-1}^0)$
    • $|H_{\max}|_i^0 = \texttt{ComputeHessianMetric}(\mathcal{H}^0, \{\mathcal{S}_i^0(k)\}_{k=1,nk})$
End for
$\mathcal{C}^0 = \texttt{ComputeSpaceTimeComplexity}(\{|H_{\max}|_i^0\}_{i=1,nadap})$

//--- Fixed-point loop to converge the global space-time mesh adaptation
For j=1,nptfx
 For i=1,nadap
    • $\mathcal{M}_i^{j-1} = \texttt{ComputeUnsteadyLpMetric}(\mathcal{C}^{j-1}, |H_{\max}|_i^{j-1})$
    • $\mathcal{H}_i^j = \texttt{GenerateAdaptedMesh}(\mathcal{H}_i^{j-1}, \mathcal{M}_i^{j-1})$
    • $\mathcal{S}_{0,i}^j = \texttt{InterpolateSolution}(\mathcal{H}_{i-1}^j, \mathcal{S}_{i-1}^j, \mathcal{H}_i^j)$
    • $\mathcal{S}_i^j = \texttt{SolveState}(\mathcal{S}_{0,i}^j, \mathcal{H}_i^j)$
    • $|H_{\max}|_i^j = \texttt{ComputeHessianMetric}(\mathcal{H}_i^j, \{\mathcal{S}_i^j(k)\}_{k=1,nk})$
 End for
 $\mathcal{C}^j = \texttt{ComputeSpaceTimeComplexity}(\{|H_{\max}|_i^j\}_{i=1,nadap})$
End for
```

### 3.5 Conservative solution interpolation

After each adaptive remeshing, the solution need to be transferred from the previous mesh to the new one in order to pursue the computation. This is the solution interpolation stage. This stage becomes crucial in the context of unsteady problems and even more if a large number of interpolation is performed, as the error introduced by the interpolation can spoil the solution accuracy. In the context of the resolution of a PDE system of conservation laws, as the compressible Euler system, by a second order numerical scheme, it is mandatory for the interpolation method to satisfy the following properties:

- mass conservation
- $\mathbf{P}^1$ exactness implying an order 2 for the method
- maximum principle

in order to obtain a consistent mesh adaptation scheme. Indeed, this stage must preserve the second order convergence of the mesh adaptation process and conserve the mass which is crucial for the simulation accuracy. In the following, we recall the conservative interpolation proposed in [4].

The mass conservation property of the interpolation operator is achieved by local mesh intersection, *i.e.,* intersections are performed at the element level. The use of mesh intersection for conservative interpolation seems natural for unconnected meshes. The locality is primordial for efficiency and robustness. The idea is, for each element of the new mesh, to compute its geometric intersection with all elements of the previous mesh that it overlaps and to mesh with simplices this geometric intersection. We are then able to use Gauss quadrature formulae to exactly compute the transfered mass.

The high-order accuracy is obtained by a solution gradient reconstruction from the discrete data and the use of Taylor formulae. This high-order interpolation can lead to loss of monotonicity. The maximum principle is then enforced by correcting conservatively the interpolated solution. Finally, vertex value solutions are reconstructed from this piecewise linear by element discontinuous representation of the solution.

The algorithm can be summarized as follow:

1. we have a piecewise linear (continuous or discontinuous) representation of the solution on the background mesh $\mathcal{H}_{back}$
2. $\forall K_{back} \in \mathcal{H}_{back}$, compute the solution mass $m_{K_{back}}$ and the solution (constant) gradient $\nabla_{K_{back}}$
3. $\forall K_{new} \in \mathcal{H}_{new}$, recover the solution mass $m_{K_{new}}$ and the solution gradient $\nabla_{K_{new}}$ :
    (a) compute the intersection of $K_{new}$ with all $K_i \in \mathcal{H}_{back}$ its overlaps
    (b) mesh the intersection polygon of each couple of elements
    (c) compute $m_{K_{new}}$ and $\nabla_{K_{new}}$ using Gauss quadrature formulae

    $\Longrightarrow$ we obtain a piecewise linear discontinuous representation of the mass on $\mathcal{H}_{new}$
4. correction to verify the maximum principle
5. set the solution values to vertices by averaging

## 4 A BLAST MODEL PROBLEM

We consider a pure three-dimensional blast problem proposed in [10]. In this simulation, shocks interact with variable density regions. One practical application of such problems is the study of how vorticity produced by these interactions mixes two different gases. Initially, the gas is at rest. The pressure and the density equal unity everywhere, except for two cylindrical regions perpendicular to each other. Both cylinders contain constant state gas. For the vertical cylinder the density is equal to 1 and the pressure is 10, thus cylindrically shaped shock waves will emanate due to the overpressure. The horizontal cylinder represents a low density region with a density of 0.1 and a pressure equal to 1. The resulting contact discontinuty is stationary until it is disturbed by the shock waves. As the shock wave propagates through the low-density region, the latter winds up into two rotating regions (or rolls). The contact discontinuity resulting from the high-pressure region will also be rolled up in this vortical motion. The upper roll rotates counter-clockwise, while the one below rotates in the clockwise direction. These two regions will after some time interact. More details on the initialization of the problem and on the physic of the flow are given in [10].

For the adaptation strategy we consider the density and the pressure as sensor variables. We control the $\mathbf{L}^2$ norm of their space-time interpolation error. We choose to split the time period into 20 adaptation sub-intervals and to perform 6 fixed-point iterations to converge the non-linear mesh adaptation problem. The desired accuracy is a space-time complexity of 20 million.

Figures 1 and 2 shows solutions and the adapted meshes at several dimensionless times. The mesh adaptation for the whole sub-interval is clearly illustrated. Indeed, refinement along band-shaped regions are clearly visible, they represent regions where physical phenomena evolves during the sub-interval. The size of the depicted meshes varies between 28 000 and 55 000 vertices. Thanks to the multi-scale mesh adaptation, the adaptive strategy captures the rolls even if strong shocks are moving in the flow field. It results in an accurate solution with very few vertices. In the strong moving shock region, the mesh accuracy is around $1.7\,e^{-3}$ at dimensionless time 0.175 and $4\,e^{-3}$ at dimensionless time 0.5 for a domain size of $[0, 1.5] \times [0, 1] \times [0, 0.5]$. For the last fixed-point iteration, the CPU time to perform the whole adaptive computation (*i.e.,* the 20 iterations of flow solver, error estimate, mesh adaptation and solution interpolation) is almost 20 minutes, the flow solver being run on a eight-processors 64-bits MacPro with an Intel Core2 chipsets.

We study now the amount of anisotropy obtained for this simulation. In three dimensions, mesh anisotropy can be quantified by two notions: the anisotropic ratios and the anisotropic quotients. Deriving these quantities for an element relies on the fact that there always exists a unique metric tensor for which this element is unit. If $\mathcal{M}_K$ denotes the metric tensor associated with element $K$, solving the following linear system provides

$\mathcal{M}_K$:

$$(S) \left\{ \begin{array}{l} \ell^2_{\mathcal{M}_K}(\mathbf{e}_1) = 1 \\ \vdots \\ \ell^2_{\mathcal{M}_K}(\mathbf{e}_6) = 1 \,, \end{array} \right.$$

where $(\mathbf{e}_i)_{i=1,6}$ is the edges list of $K$ and $\ell^2_{\mathcal{M}_K}(\mathbf{e}_i) = {}^t\mathbf{e}_i \, \mathcal{M}_K \, \mathbf{e}_i$. $(S)$ admits a unique solution as soon as the volume of $K$ is not null. Once $\mathcal{M}_K$ is computed, the anisotropic

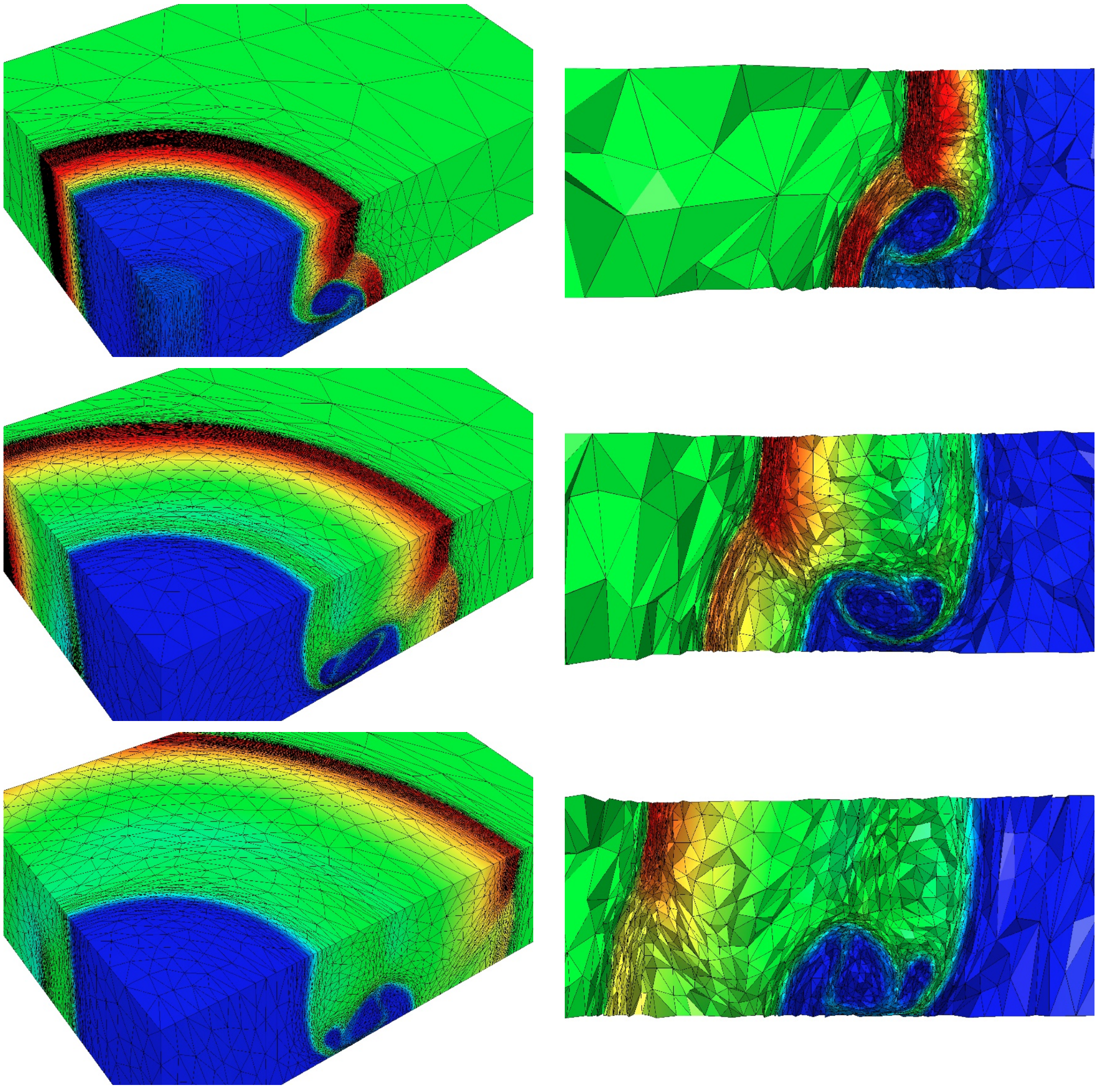

Figure 1: Adapted meshes and solutions (density) at dimensionless times 0.175, 0.35 and 0.5. The adapted meshes contain respectively 54 218, 38 580 and 28 907 vertices. Left, surface adapted meshes. Right, volume adapted meshes in a plane orthogonal to $y$.

ratio and the anisotropic quotient associated with element $K$ are simply given by

$$\text{ratio} = \sqrt{\frac{\min_i \lambda_i}{\max_i \lambda_i}} = \frac{\max_i h_i}{\min_i h_i}, \quad \text{and} \quad \text{quo} = \frac{\max_i h_i^3}{h_1 h_2 h_3},$$

where $(\lambda_i)_{i=1,3}$ are the eigenvalues of $\mathcal{M}_K$ and $(h_i)_{i=1,3}$ are the corresponding sizes. The anisotropic ratio stands for the maximum elongation of a tetrahedron by comparing two principal directions. The anisotropic quotient represents the overall anisotropic ratio of a tetrahedron taking into account all the possible directions.

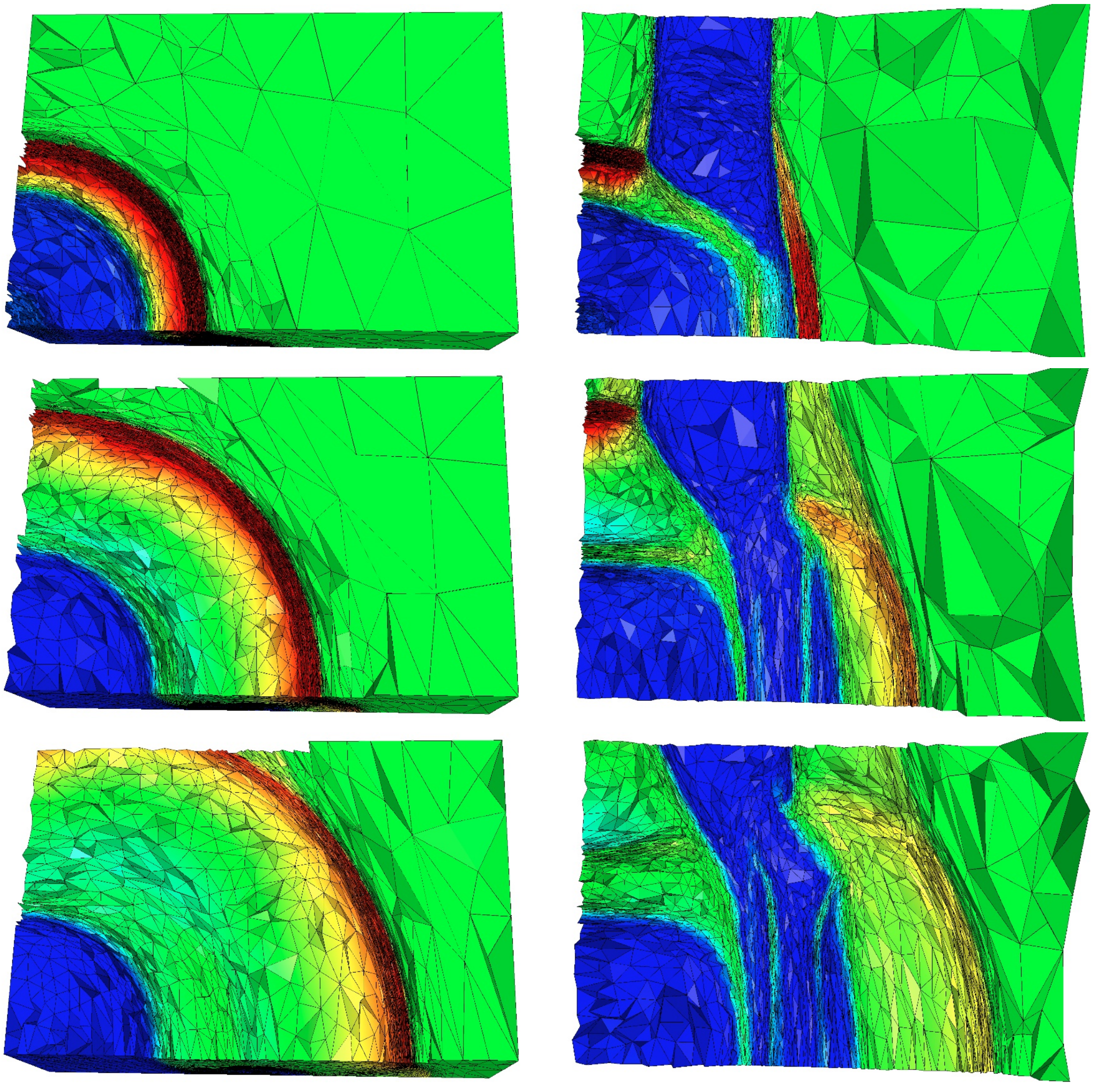

Figure 2: Adapted meshes and solutions (density) at dimensionless times 0.175, 0.35 and 0.5. The adapted meshes contain respectively 54 218, 38 580 and 28 907 vertices. Left and right, volume adapted meshes in two planes orthogonal to $z$.

For this simulation, an average anisotropic ratio of almost 12 is obtained while the maximal stretching can be of the order of 200. The mean anisotropic quotient is almost 109. It corresponds to the overall gain in three dimensions of *an anisotropic adapted mesh* as compared to *an isotropic adapted mesh*. The gain is of course even much greater when compared to a uniform mesh.

## 5 ENHANCING THE SPACE-TIME CONVERGENCE

This section aims at illustrating how the multi-scale mesh adaptation process automatically acts on the mesh throughout a convergence study to enjoy the Early Capturing property in the context of flows with discontinuities: *"a mesh adaptive method enjoys the Early Capturing property if it is able to converge to discontinuous solutions at the same rate as for smooth solutions".* We say that a convergence rate equal to $\alpha$ is obtained if we have:

$$||u - u_h|| \leq const.\ N^{-\alpha/d}\,, \tag{10}$$

where $u$ denotes the exact solution, $u_h$ the approximate (*i.e.,* discretized) one, $N$ stands for the number of vertices and $d$ stands for the dimension of the computational domain.

In this study, we consider that the numerical scheme behaves at second-order accuracy in smooth regions and at first-order accuracy in discontinuities vicinities. We first shortly describe the process in the steady case. Then, the unsteady case is analyzed.

### 5.1 The steady case

For smooth flows, it has been demonstrated, that a sequence of meshes of increasing complexity defined by Relation (5) is second-order convergent thanks to Bound (6). Moreover, it is observed in [13, 12], that the second-order convergence property still holds even when some singularities are present in the flow field. We now give an idea on how the adaptive process acts to satisfy the EC property for discontinuous flows.

We consider a 3D discontinuous solution. For the convergence analysis, we build a sequence of meshes where from one mesh to the finer one, the mesh step is divided by two in the smooth subregion excluding a progressively smaller vicinity of the discontinuity. In this part of the mesh, the number of vertices is $2^d = 8$ times larger at every finer mesh. In the vicinity of the discontinuity, the local mesh size of the finer mesh is taken four times smaller than the size of the previous one. Since the region of singularity is of zero measure, it is possible to maintain a number of vertices in the discontinuity vicinity which does not increase faster than the number of vertices in the rest of the domain. This can be done by reducing the discontinuity vicinity region while increasing the mesh accuracy. Therefore, after $k$ steps, the error is made $4^k$ times smaller for a mesh that involves $8^k$ more vertices. According to (3), this characterizes the second-order convergence of this mesh adaptive strategy on a discontinuous function.

In the next section, we shall extend this idea to discontinuous unsteady flows in 3D.

### 5.2 The unsteady case

The proposed unsteady mesh adaptation strategy generates an adapted mesh for each adaptation sub-interval $[t_i, t_{i+1}]$. Therefore, in the context of several computations where the mesh convergence to the continuous solution is analyzed, the choice of these adaptation sub-intervals impacts the convergence rate. The question to answer is how these adaptation time sub-intervals or this global time discretization should be chosen to enhance the space-time convergence order and, if possible, to preserve the *second-order space-time accuracy*.

A hint is given by the example of the motion of a step function representing a traveling discontinuity. We assume that the numerical method is second-order accurate in smooth regions and first-order accurate at the discontinuity vicinity. We have a reference mesh, denoted Mesh 1, for which the space-time interpolation error is $\varepsilon$. To simplify the analysis, we assume that all sub-intervals are homogeneous:

- the simulation time frame is split into M sub-intervals
- each interval $[t_i, t_{i+1}]$ has the same length $\Delta t$ and for each interval $m$ time steps are performed to advance the solution from $t_i$ to $t_{i+1}$
- for each interval, the generated adapted spatial mesh is of size $n$.

Therefore, the size of the space-time mesh is: $N = M\,(n \times m)$. In order to get a space-time second order convergence in 3D, we wish to divide the error by a factor 4 for a new adapted space-time mesh, denoted Mesh 2, having an number of vertices at each time step equal to $16\,N$.

Now, we assume that the simulation time frame is still split into $M$ sub-intervals whatever the chosen error threshold, in other words, we assume that the sub-interval $[t_i, t_{i+1}]$ are kept with the same length $\Delta t$. In consequence, the vicinity of the discontinuity is defined by the area swept by the discontinuity between $t_i$ and $t_{i+1}$ and it thus cover the same part of the computational domain for Mesh 1 and Mesh 2.

For anisotropic mesh adaptation in three dimensions, the number of vertices in the region covered by the discontinuity will pass from $n_d$ vertices for Mesh 1 to $(2^2 \times 4)\,n_d = 16\,n_d$ vertices for Mesh 2 as the local mesh size is divided by 4 in the direction normal to the discontinuity (first-order accuracy) and by 2 in the two other directions (second-order accuracy). As this region is not negligible with respect to smooth region, the spatial mesh size become $16\,n$ for each sub-interval. As regards the time discretization, the time step is defined by the CFL condition and is proportional to the minimal size of the mesh. In the worst case, the minimal size is attained in the discontinuity region which is generally true. Therefore, the time step is divided by a factor 4 like the mesh size in the direction normal to the discontinuity leading to a time mesh size of $4m$. The space-time mesh size becomes $M\,(16\,n \times 4\,m) = 64\,N$. The mesh size is too large by a factor 4 to reach the second-order space-time convergence.

Nevertheless, it is possible to cancel this factor 4 and to theoretically obtain a second-order space-time convergence by changing the size of the sub-intervals. Indeed, if we now consider that the simulation time frame is split into $4\,M$ sub-intervals, for each sub-interval we obtain a spatial mesh of size $\frac{16\,n}{4} = 4\,n$ and $\frac{4\,m}{4} = m$ iterations are performed to advance the solution on the sub-interval. Thus, the space-time mesh of size becomes: $4\,M\,(4\,n \times m) = 16\,N$. The order two is recovered. This correction of the algorithm allows us to optimize the space-time mesh during a convergence study.

It is important to note that this coefficient 4 is admissible as the number of sub-intervals does not grow faster that the number of solver iterations (the time mesh size). Moreover, it is coherent with the (impracticable) strategy where each sub-interval correspond to a solver time step, *i.e.,* the mesh is adapted at each solver iteration. Indeed, in that specific case, if $M$ sub-intervals are considered for simulation 1, then $4\,M$ sub-intervals will be considered for simulation 2.

This study leads to the following result:

**Lemma 1** *For an unsteady anisotropic mesh adaptation using the global fixed-point mesh adaptation scheme coupled with a solver which is second-order accurate in smooth region and first-order accurate in discontinuity vicinity, a necessary condition for second order space-time convergence is that each sub-interval* $[t_i, t_{i+1}]$ *is taken four times smaller when the error threshold is four times smaller.*

### 5.3 The blast model problem

This strategy is applied to the blast model problem. The adaptive strategy has been applied for space-time complexity 5, 10, 20, 30, 40 and 60 million with constant sub-intervals or adapted sub-intervals. For each method, the adaptive solution at complexity 60 million is considered as the reference solution. For each simulation, the space-time error has been approximated by:

$$\mathcal{E} \approx \sum_{i=1}^{n_{adap}} (t_i - t_{i-1})\, \varepsilon(t_i)\,,$$

where $\varepsilon(t_i)$ is the instantaneous gap at $t_i$ between the solution and the reference one. As regards the space-time mesh complexity, we have been confronted to the problematic of large variation of the time discretization. Indeed, the time discretization is given by the solver time step which is generally defined by the smallest altitude of the mesh. The smallest altitude may vary a lot during the adaptive process. In consequence, the number of time steps is not representative of the time discretization as it is strongly affected by the quality of the spatial mesh. In consequence, to compare both methods we consider the average spatial complexity per sub-interval:

$$\mathcal{C} = \frac{\sum_{i=1}^{n_{adap}} n_i}{n_{adap}}$$

where $n_i$ is the spatial mesh size for sub-interval $[t_{i-1}, t_i]$. The convergence of both methods is depicted in Figure 3. As expected, we observed a better convergence order for the method with adapted sub-intervals.

In Figure 4, we compare the adapted mesh and solution for the mesh adaptation for a space-time complexity of 20 million and 20 adaptation sub-intervals (mesh 1), and for a space-time complexity of 60 million and 35 adaptation sub-intervals (mesh 2). Mesh 1 and mesh 2 contains 39 665 and 59 604 vertices, respectively . Despite the large increase in accuracy (a factor 3), the mesh size at each sub-interval has been increase by a factor less than two thanks to the adaptation of the number of sub-intervals. With this strategy, the global space-time mesh is optimized.

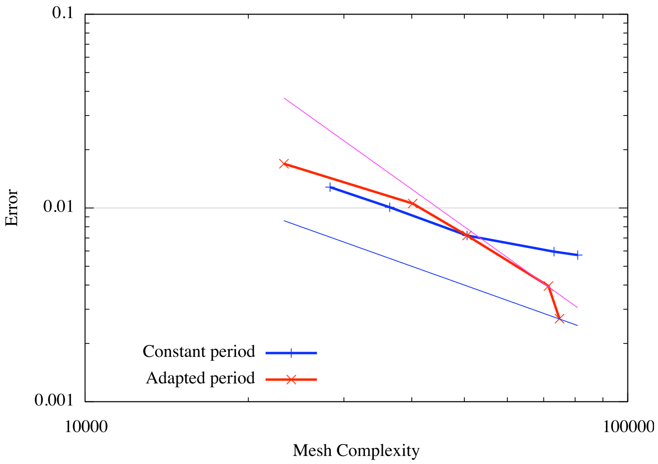

Figure 3: Convergence of the $\mathbf{L}^\infty$-$\mathbf{L}^p$ mesh adaptation strategy. In blue, convergence with a constant number of adaptation sub-intervals. In red, convergence with a varying number of adaptation sub-intervals depending on the desired error threshold.

## 6 CONCLUSIONS

In this work, we have proposed a first extension of the multi-scale mesh adaptation to unsteady flows. It involves a $\mathbf{L}^\infty$-$\mathbf{L}^p$ mesh adaptation strategy and a global fixed-point mesh adaptation scheme. Moreover, we have proposed a strategy to improve the space-time mesh size during a convergence study, improving thus the order of convergence of the method. This unsteady mesh adaptation strategy has been successfully applied to a blast model case.

However, this study is just a first step. It points out that a space-time error estimation seems mandatory at least to have a theoretical estimation of the optimal time discretization. This will allow convergence study to be performed without relying on the solver time step which is not a reliable.

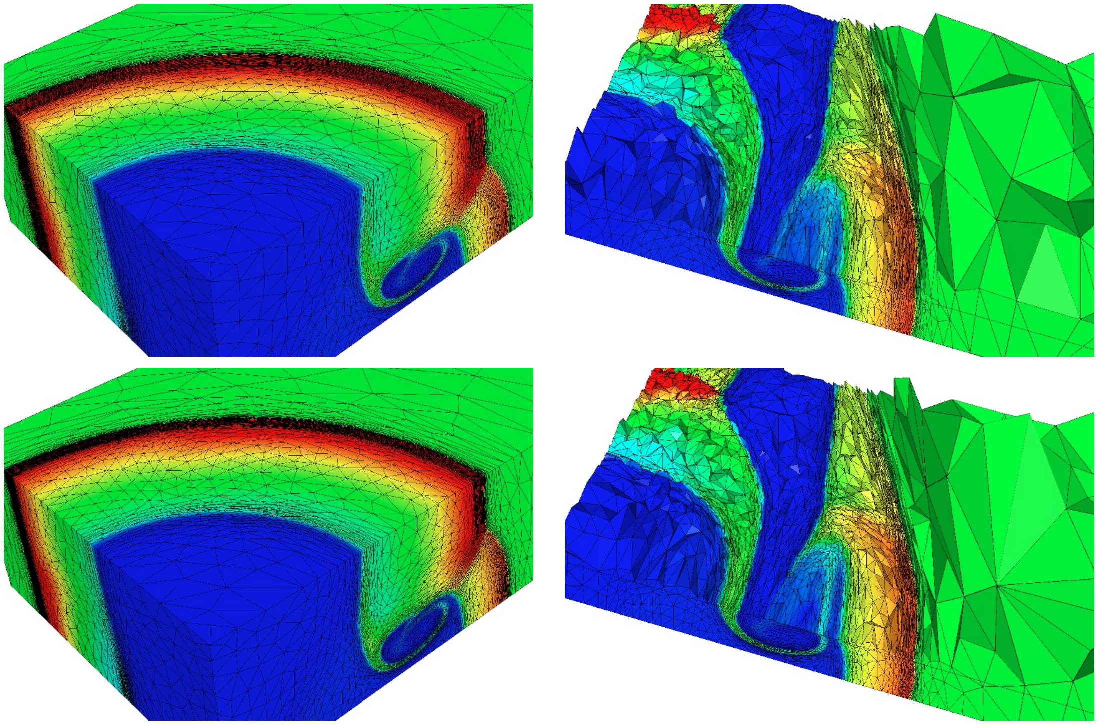

Figure 4: Adapted meshes and solutions (density) at dimensionless time 0.3. Top, mesh adaptation for a space-time complexity of 20 million and 20 adaptation sub-intervals. Bottom, mesh adaptation for a space-time complexity of 60 million and 35 adaptation sub-intervals.

## REFERENCES

[1] A. Agouzal, K. Lipnikov, and Y.V. Vassilevski. Adaptive generation of quasi-optimal tetrahedral meshes. *East-West J.*, 7(4):879–882, 1999.

[2] F. Alauzet, P.J. Frey, P.L. George, and B. Mohammadi. 3D transient fixed point mesh adaptation for time-dependent problems: Application to CFD simulations. *J. Comp. Phys.*, 222:592–623, 2007.

[3] F. Alauzet and A. Loseille. High order sonic boom modeling by adaptive methods. *J. Comp. Phys.*, 229:561–593, 2010.

[4] F. Alauzet and M. Mehrenberger. P1-conservative solution interpolation on unstructured triangular meshes. RR-6804, INRIA, January 2009.

[5] F. Courty, D. Leservoisier, P.L. George, and A. Dervieux. Continuous metrics and mesh adaptation. *Appl. Numer. Math.*, 56(2):117–145, 2006.

[6] P.J. Frey and F. Alauzet. Anisotropic mesh adaptation for CFD computations. *Comput. Methods Appl. Mech. Engrg.*, 194(48-49):5068–5082, 2005.

[7] P.J. Frey and P.L. George. *Mesh generation. Application to finite elements.* ISTE Ltd and John Wiley & Sons, 2nd edition, 2008.

[8] P.L. George, F. Hecht, and M.G. Vallet. Creation of internal points in Voronoi's type method. Control and adaptation. *Adv. Eng. Software*, 13(5-6):303–312, 1991.

[9] W. Huang. Metric tensors for anisotropic mesh generation. *J. Comp. Phys.*, 204(2):633–665, 2005.

[10] J.O. Langseth and R. J. LeVeque. A wave propagation method for three-dimensional hyperbolic conservation laws. *J. Comp. Phys.*, 165:126–166, 2000.

[11] A. Loseille and F. Alauzet. Continuous mesh model and well-posed continuous interpolation error estimation. RR-6846, INRIA, March 2009.

[12] A. Loseille and F. Alauzet. Optimal 3D highly anisotropic mesh adaptation based on the continuous mesh framework. In *Proceedings of the 18th International Meshing Roundtable*, pages 575–594. Springer, 2009.

[13] A. Loseille, A. Dervieux, P.J. Frey, and F. Alauzet. Achievement of global second-order mesh convergence for discontinuous flows with adapted unstructured meshes. In *37th AIAA Fluid Dynamics Conference and Exhibit*, AIAA-2007-4186, Miami, FL, USA, Jun 2007.

[14] A. Loseille and R. Löhner. Adaptive anisotropic simulations in aerodynamics. In *48th AIAA Aerospace Sciences Meeting and Exhibit*, AIAA-2010-169, Orlando, FL, USA, Jan 2010.