Journal of Computational Physics 229 (2010) 2866–2897

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

Fully anisotropic goal-oriented mesh adaptation for 3D steady Euler equations

A. Loseille [a,c,*], A. Dervieux [b], F. Alauzet [a]

[a] INRIA, Projet Gamma, Domaine de Voluceau, Rocquencourt, BP 105, 78153 Le Chesnay Cedex, France
[b] INRIA, Projet Tropics, 2004 route des lucioles, BP 93, 06902 Sophia Antipolis Cedex, France
[c] George Mason University, Computational Fluid Dynamics Center, 4400 university drive, MS6A2 Fairfax, VA, USA

ARTICLE INFO

Article history:
Received 5 June 2009
Received in revised form 11 December 2009
Accepted 17 December 2009
Available online 4 January 2010

Keywords:
Anisotropic unstructured mesh adaptation
Goal-oriented mesh adaptation
Metric-based mesh adaptation
Steady compressible Euler equations
A priori error estimate
Adjoint

ABSTRACT

This paper studies the coupling between anisotropic mesh adaptation and goal-oriented error estimate. The former is very well suited to the control of the interpolation error. It is generally interpreted as a local geometric error estimate. On the contrary, the latter is preferred when studying approximation errors for PDEs. It generally involves non local error contributions. Consequently, a full and strong coupling between both is hard to achieve due to this apparent incompatibility. This paper shows how to achieve this coupling in three steps.

First, a new *a priori* error estimate is proved in a formal framework adapted to goal-oriented mesh adaptation for output functionals. This estimate is based on a careful analysis of the contributions of the implicit error and of the interpolation error. Second, the error estimate is applied to the set of steady compressible Euler equations which are solved by a stabilized Galerkin finite element discretization. A goal-oriented error estimation is derived. It involves the interpolation error of the Euler fluxes weighted by the gradient of the adjoint state associated with the observed functional. Third, rewritten in the continuous mesh framework, the previous estimate is minimized on the set of continuous meshes thanks to a calculus of variations. The optimal continuous mesh is then derived analytically. Thus, it can be used as a metric tensor field to drive the mesh adaptation. From a numerical point of view, this method is completely automatic, intrinsically anisotropic, and does not depend on any *a priori* choice of variables to perform the adaptation.

3D examples of steady flows around supersonic and transsonic jets are presented to validate the current approach and to demonstrate its efficiency.

1. Introduction

When dealing with real-life CFD problems, mesh adaptation is recognized as a complementary approach to high-order schemes classically used to solve the problem at hand. This interest for mesh adaptation relies on its ability to approach the asymptotic convergence and, therefore, to obtain more easily an accurate prediction for complex flows. Among mesh adaptation methods, anisotropic unstructured adaptation is of paramount influence on the accuracy of many CFD predictions. This technique allows (i) to substantially reduce the number of degrees of freedom, thus impacting favorably the cpu time, (ii) to reduce (optimize) the numerical scheme dissipation by automatically taking into account the anisotropy of the physical phenomena inside the mesh [43], and (iii) to access to high order asymptotic convergence, see for instance

* Corresponding author. Address: George Mason University, Computational Fluid Dynamics Center, 4400 university drive, MS6A2 Fairfax, VA, USA. Tel.: +1 703 505 1602.
E-mail address: aloseill@gmu.edu (A. Loseille).

0021-9991/$ - see front matter © 2009 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2009.12.021

[15,43]. So far, anisotropic features are mainly deduced from an interpolation error estimate. Consequently, a Hessian-like anisotropic approach does not directly apply to the goal-oriented mesh adaptation methods that take into account both the solution and the PDE in the error estimation. As the objective of this paper is to couple both approaches, we briefly recall their inherent formulations and their intrinsic properties.

1.1. State of the art

A recent family of methods, often referred to as *metric analysis methods*, or *Hessian-based methods* have shown a very fertile development, from the pioneering works in [12,21]. Thanks to recent formalisms, see for instance [42], these ideas turned into a clean set of functional analysis problematics relying on an ideal representation of the interpolation error and of a mesh. Getting rid of error iso-distribution and prefering $\mathbf{L}^p$ error minimization allow to take into account discontinuities with higher-order convergence [16,43]. This theory combines perfectly with unstructured mesh generation [25,27]. Addressed applications are either steady or unsteady [3,23,5]. Metric-based mesh adaptation efficiency and genericity have been proved by many successful applications for 3D complex problems [3,11,18,19,23,45,50,51]. However, these methods are limited to the minimization of some interpolation errors for some solution fields. If for many applications, this simplifying standpoint is an advantage, there are also many applications where Hessian-based adaptation is far from optimal regarding the way the degrees of freedom are distributed in the computational domain. Indeed, metric-based methods aim at controlling the interpolation error but this goal is not often so close to the objective that consists in obtaining the best solution of a PDE. This is particularly true in many engineering applications where a specific functional needs to be accurately evaluated: lift, drag, heat flux, pressure field, etc.

In contrast to the previous set of methods, the formulation of **goal-oriented** mesh adaptation, [28,29,47,52,53], has brought many improvements in the formulation and the resolution of mesh adaptation for PDE approximations. Let us write the continuous PDE as:

$$\Psi(w) = 0, \tag{1}$$

and the discrete one as:

$$\Psi_h(w_h) = 0. \tag{2}$$

The objective of goal-oriented mesh adaptation is different from the one of deriving the optimal mesh to control the global approximation error $\|w - w_h\|$, see [9,54] for *a posteriori* error estimate devoted to that latter task. Indeed, we focus on deriving the best mesh to observe a given functional j depending of the solution w. To this end, we examine how to control the approximation error of the functional: $j(w) - j(w_h)$. Assuming that the functional j is enough regular to be observed through its Jacobian g we simplify it as follows:

$$j(w) = (g, w).$$

We also assume that there is no discrete error evaluation on j, this means that $j_h(w_h) = j(w_h)$. On this basis, we seek for the mesh $\mathcal{H}$ which gives the smallest error for the evaluation of j from the solution field w_h:

$$\min_{\mathcal{H}} |(g, w_h) - (g, w)|, \tag{3}$$

where w and w_h verify (constraint) state Eqs. (1) and (2), respectively. The initial mesh adaptation problem is recast with Eq. (3) as an optimization problem. In order to go a step forward in the analysis, we need to implicitly take into account Constraints (1) and (2) in Eq. (3). The initial approximation error on the cost functional $|(g, w_h) - (g, w)|$ can be simplified as a local error thanks to the introduction of the adjoint state:

$$(g, w_h - w) \approx \left(g, \left(\frac{\partial \Psi}{\partial w}\right)^{-1} \Psi(w_h)\right) = (w^*, \Psi(w_h)), \tag{4}$$

where the adjoint state w^* is solution of:

$$\left(\left(\frac{\partial \Psi}{\partial w}\right)^{*} w^*, \psi\right) = (g, \psi).$$

In practice, the exact adjoint w^* is not available. By introducing an approximate adjoint w_h^*, we get:

$$(g, w_h - w) \approx \left(w_h^*, \Psi(w_h)\right). \tag{5}$$

The right-hand side is a spatial integral the integrand of which can be used to decide where to refine the mesh. The iso-distribution of the error can be approximated by refining the mesh according to a tolerance, as in [9]. In [31], it is proposed to use this right-hand side as a correction that importantly improves the quality (in particular the convergence order) of the approximation of j by setting:

$$j^{corrected} = (g, w_h) + \left(w_h^*, \Psi(w_h)\right).$$

However, by substituting w^* by w_h^*, we introduce an error in $O(w_h^* - w^*)$, which results in being the main error term when we use $j^{corrected}$. In [52,53], it is proposed to keep the corrector and to adapt the mesh to this higher-order error term, i.e.,:

$$j^{corrected} - j \approx \left(w_h^* - w^*, \Psi(w_h)\right),$$

or equivalently:

$$j^{corrected} - j \approx \left(w - w_h, g - \left(\frac{\partial \Psi}{\partial w}\right)^* w_h^*\right).$$

In order to evaluate numerically these terms, the authors chose to approach these approximation errors by interpolation errors, by computing differences between the linear representation $L_{h/2}^h$ and a quadratic representation $Q_{h/2}^h$ reconstructed on a finer mesh:

$$w_h^* - w^* \approx L_{h/2}^h w_h^* - Q_{h/2}^h w_h^*$$

$$w_h - w \approx L_{h/2}^h w_h - Q_{h/2}^h w_h.$$

1.2. Our approach

In our point of view, metric analysis and goal-oriented analysis are complementary. Indeed, a metric-based method specifies the object of our search through an accurate description of the ideal mesh while a goal-oriented method specifies precisely the purpose of the search in terms of which error will be reduced. It is then very motivating to seek for a combination of both methods, with the hope of obtaining a metric-based specification of the best mesh for reducing the error committed on a target functional. A few works address this purpose. In [53], an anisotropic step relying on the Hessian of the Mach number is introduced into the *a posteriori* estimate. In [48], an ad hoc formula gives a better impact to the anisotropic component.

This paper presents a different contribution to the combination of both methods. This will be achieved from some considerations: (i) a reliable continuous mesh model, (ii) an *a priori* estimate and (iii) specific numerical schemes allowing approximations of $\Psi_h(w) - \Psi(w)$. These three items differ from classical approaches of goal-oriented mesh adaptation.

The first key point of this work is to use a metric-based parameterization of meshes. This means to work in a continuous (non-discrete) formulation. We shall follow the continuous interpolation analysis proposed in [42]. Metric-based methods usually use an interpolation error, the deviation between the exact solution and its linear interpolation on the mesh. This assumes the knowledge of the solution, i.e., this is an *a priori* standpoint. Further, metric-based methods express an optimal metric as a function of this exact solution. This reinforce the *a priori* feature of the approach.

In contrast, goal-oriented methods are generally envisaged from an *a posteriori* standpoint, we refer to [6,9,19,31,46,54]. Following this option, we would start on the basis of errors committed on an existing mesh. It seems then not feasible to transform this information into the specification of the optimal ideal mesh. Moreover, mesh refinement scheme based on *a posteriori* estimations depends on an equi-distribution principle by refining elements where the error is greater than a given threshold. Such process is thus isotropic by nature. Consequently, it does not provide an optimal distribution of the degrees of freedom with respect to the accuracy. Indeed, this process is equivalent to a local optimization so that a global minimizer is hard to achieve. Fortunately, goal-oriented methods do not need to be systematically associated with an *a posteriori* analysis.

Now, according to, for instance, [8], *a priori* analysis can bring many useful informations. Anisotropy is often one of these informations [20]. Further, the goal-oriented error can also be easily expressed by an *a priori* analysis, as we shall demonstrate in the sequel, and this is second key point of this work.

The third key point results from working with a numerical scheme that expresses the difference $\Psi_h(w) - \Psi(w)$ in term of interpolation errors. This can be done in a straightforward way by considering finite element variational formulations.

1.3. Outline

The theoretical abstract framework is introduced in Section 2. Within this framework, a first *a priori* goal-oriented error estimate, Eq. (11), is derived. Its application to the compressible Euler equations is then studied in Section 3 for a class of specific Galerkin-equivalent numerical schemes. From this study, a generic anisotropic error estimate, Eq. (16), is expressed. The estimate is then minimized globally on the abstract space of continuous meshes, Section 4. Finally, the numerical part in Section 6 gives some details on the main modifications of the adaptive loop as compared to classical Hessian-based mesh adaptation. The pratical optimal metric field minimizing the goal-oriented error estimate is then exhibited, Eq. (31). Several 3D detailed examples conclude this last section by providing a numerical validation of the theory.

2. A more accurate non-linear error analysis

2.1. Assumptions and definitions

Let V be a space of functions (at least a Banach space). We write the state equation under a variational statement:

$$w \in V, \quad \forall \varphi \in V, \qquad (\Psi(w), \varphi) = 0, \tag{6}$$

where the operator $(,)$ holds for a $V\prime \times V$ product, V' is the topological dual of V and w is the solution of this equation. Symbol Ψ holds for a functional that is linear with respect to test function φ but *a priori* non linear with respect to w. The continuous adjoint w^* is solution of:

$$w^* \in V, \quad \forall \psi \in V, \qquad \left(\frac{\partial \Psi}{\partial w}(w)\psi, w^*\right) = (g, \psi), \tag{7}$$

where g is the Jacobian of a given functional j. Let V_h be a subspace of $\mathcal{V} = V \cap \mathcal{C}^0$ of finite dimention N, we write the discrete state equation as follows:

$$w_h \in V_h, \quad \forall \varphi_h \in V_h, \quad (\Psi_h(w_h), \varphi_h) = 0.$$

Then, we can write:

$$(\Psi_h(w), \varphi_h) - (\Psi_h(w_h), \varphi_h) = (\Psi_h(w), \varphi_h) - (\Psi(w), \varphi_h) = ((\Psi_h - \Psi)(w), \varphi_h). \tag{8}$$

For the *a priori* analysis, we assume that the solutions $w$and w^* are sufficiently regular:

$$w \in V \cap \mathcal{C}^0, \quad w^* \in V \cap \mathcal{C}^0,$$

and that we have an interpolation operator:

$$\Pi_h : \ V \cap \mathcal{C}^0 \to V_h.$$

2.2. A priori estimation

We start from a functional defined as:

$$j(w) = (g, w),$$

where g is a function of V. Our objective is to estimate the following approximation error on the functional:

$$\delta j = j(w) - j(w_h),$$

as a function of continuous solutions, of continuous residuals and of discrete residuals. The error δj is split as follows:

$$\delta j = j(w) - j(w_h) = (g, w - \Pi_h w) + (g, \Pi_h w - w_h).$$

δj is now composed of an *interpolation error* and of an *implicit error* which involves only discrete terms. Let us introduce the discrete adjoint system:

$$w_h^* \in V_h, \forall \psi_h \in V_h, \qquad \left(\frac{\partial \Psi_h}{\partial w}(\Pi_h w)\psi_h, w_h^*\right) = (g, \psi_h).$$

We can derive the following extension of δj with the choice $\psi_h = \Pi_h w - w_h$:

$$\delta j = (g, w - \Pi_h w) + \left(\frac{\partial \Psi_h}{\partial w}(\Pi_h w)(\Pi_h w - w_h), w_h^*\right).$$

This new right-hand side second term is simplified by using an extension of Ψ_h. According to (8), we have:

$$\left(\Psi_h(\Pi_h w), w_h^*\right) - \left(\Psi_h(w_h), w_h^*\right) = \left(\Psi_h(\Pi_h w), w_h^*\right) - \left(\Psi_h(w), w_h^*\right) + \left((\Psi_h - \Psi)(w), w_h^*\right),$$

which gives by using a Taylor extension:

$$\left(\frac{\partial \Psi_h}{\partial w}(\Pi_h w)(\Pi_h w - w_h), w_h^*\right) = \left(\Psi_h(\Pi_h w), w_h^*\right) - \left(\Psi_h(w), w_h^*\right) + \left((\Psi_h - \Psi)(w), w_h^*\right) + R_1,$$

where the remainder R_1 is:

$$R_1 = \left(\frac{\partial \Psi_h}{\partial w}(\Pi_h w)(\Pi_h w - w_h), w_h^*\right) - \left(\Psi_h(\Pi_h w), w_h^*\right) + \left(\Psi_h(w_h), w_h^*\right). \tag{9}$$

Thus, we get the following expression of δj:

$$\delta j = (g, w - \Pi_h w) + \left(\Psi_h(\Pi_h w), w_h^*\right) - \left(\Psi_h(w), w_h^*\right) + \left((\Psi_h - \Psi)(w), w_h^*\right) + R_1,$$

We now apply a second Taylor extension to get:

$$\left(\Psi_h(\Pi_h w), w_h^*\right) - \left(\Psi_h(w), w_h^*\right) = \left(\frac{\partial \Psi_h}{\partial w}(w)(\Pi_h w - w), w_h^*\right) + R_2,$$

with remainder term

$$R_2 = \left(\Psi_h(\Pi_h w), w_h^*\right) - \left(\Psi_h(w), w_h^*\right) - \left(\frac{\partial \Psi_h}{\partial w}(w)(\Pi_h w - w), w_h^*\right). \tag{10}$$

This implies:

$$\delta j = (g, w - \Pi_h w) + \left(\frac{\partial \Psi_h}{\partial w}(w)(\Pi_h w - w), w_h^*\right) + \left((\Psi_h - \Psi)(w), w_h^*\right) + R_1 + R_2.$$

In contrast to an *a posteriori* analysis, this analysis starts with a discrete adjoint w_h^*. However, our purpose is to derive a continuous description of the main error term. Thus, we get rid of the discrete solutions in the dominating terms. To this end, we re-write δj as follows:

$$\delta j = (g, w - \Pi_h w) + \left(\frac{\partial \Psi}{\partial w}(w)(\Pi_h w - w), w^*\right) + ((\Psi_h - \Psi)(w), w^*) + R_1 + R_2 + D_1 + D_2 + D_3,$$

where R_1 and R_2 are given by Relations (9) and (10) and we set:

$$D_1 = \left(\left(\frac{\partial \Psi_h}{\partial w} - \frac{\partial \Psi}{\partial w}\right)(w)(\Pi_h w - w), w_h^*\right),$$
$$D_2 = \left(\frac{\partial \Psi}{\partial w}(w)(\Pi_h w - w), w_h^* - w^*\right),$$
$$D_3 = \left((\Psi_h - \Psi)(w), w_h^* - w^*\right).$$

The latter expression of δj can be even more simplified thanks to the continuous adjoint of Eq. (7), leading to:

$$\delta j = ((\Psi_h - \Psi)(w), w^*) + R_1 + R_2 + D_1 + D_2 + D_3. \tag{11}$$

At least formally, the R_i and the D_k are higher order terms, and the first term in the right-hand side of (11) is the dominating one. It remains to give the studied context and to exhibit from (11) a formulation specifying the optimal mesh.

It is interesting to compare the output of this *a priori* analysis[1]

$$\delta j \approx ((\Psi_h - \Psi)(w), w^*)$$

with the output of the *a posteriori* analysis:

$$\delta j \approx -((\Psi_h - \Psi)(w_h), w^*).$$

Thus,

$$((\Psi_h - \Psi)(w), w^*) \approx -((\Psi_h - \Psi)(w_h), w^*), \tag{12}$$

which means that both terms can be transformed into correctors, by approximating continuous ingredients by discrete ones. The validity of these correctors will hold as far as all the neglected terms are small. This also gives asymptotic estimates of the error which is also valid when other terms are small. A practical way to verify that these presumed high-order terms are small is to check by mesh refinement that the numerical convergence order is very close to the asymptotic/theoretical one. Now, a way to get more easily asymptotic convergence order is to apply a good mesh adaptation refinement. Therefore, we intend to build such a good mesh adapted refinement by minimizing the above *a priori* error estimate and to check it numerically. If this goal is achieved, then:

- Eq. (12) holds accurately,
- since other terms are small, both correctors are efficient,
- as the mesh is optimally adapted to minimize the error estimate, the error bound is as small as possible.

To sum up, we get, at the same time, an accurate prediction of the corrector with a rather small and secure incertainty interval and an efficient corrector.

[1] Notice that the standard *a priori* analysis will provide $\delta j \approx \left((\Psi_h - \Psi)(w), w_h^*\right)$.

3. The case of the steady Euler equations

In this section, we study how Eq. (11) can be applied in the context of the steady Euler equations. To this end, we restrict to a particular discretization of these equations and we consider a variational analysis.

3.1. Weak-formulation

We write the steady Euler equations as follows in the computational domain $\Omega \subset \mathbb{R}^3$:

$$\Psi(W) = \nabla \cdot \mathcal{F}(W) = 0 \text{ in } \Omega,$$

where $W = {}^t(\rho, \rho u, \rho v, \rho w, \rho E)$ is the vector of conservative variables. $\mathcal{F}$ is the convection operator $\mathcal{F}(W) = (\mathcal{F}_1(W), \mathcal{F}_2(W), \mathcal{F}_3(W))$ with:

$$\mathcal{F}_1(W) = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ (\rho E + p)u \end{pmatrix}, \quad \mathcal{F}_2(W) = \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ \rho vw \\ (\rho E + p)v \end{pmatrix}, \quad \mathcal{F}_3(W) = \begin{pmatrix} \rho w \\ \rho uw \\ \rho vw \\ \rho w^2 + p \\ (\rho E + p)w \end{pmatrix},$$

so that the state equation becomes:

$$\frac{\partial \mathcal{F}_1(W)}{\partial x} + \frac{\partial \mathcal{F}_2(W)}{\partial y} + \frac{\partial \mathcal{F}_3(W)}{\partial z} = 0,$$

ρ, p and E hold respectively for the density, the thermodynamical pressure and the total energy per mass unit. Symbols u, v and w stand for the Cartesian components of velocity vector $\mathbf{u} = (u, v, w)$. For a calorically perfect gas, we have

$$p = (\gamma - 1)\left(\rho E - \frac{1}{2}\rho \|\mathbf{u}\|_2^2\right),$$

where γ is constant. A weak formulation of this system writes for $W \in V = [H^1(\Omega)]^5$ as follows:

$$\forall \phi \in V, \quad (\Psi(W), \phi) = \int_\Omega \phi \nabla \cdot \mathcal{F}(W)\, d\Omega - \int_\Gamma \phi\, \widehat{\mathcal{F}}(W).\mathbf{n}\, d\Gamma = 0, \tag{13}$$

where Γ is the boundary of the computational domain Ω, $\mathbf{n}$ the outward normal to Γ and the boundary flux $\widehat{\mathcal{F}}$ contains the boundary conditions. Eq. (13) will play the role of Eq. (6) of the abstract analysis of the previous section.

Eq. (13) is discretized into a vertex-centered formulation apply to unstructured meshes. Let $\mathcal{H}$ be a tetrahedral mesh of Ω. We denote by Ω_h and Γ_h the linear approximate of Ω and Γ defined by $\mathcal{H}$. Let us introduce the following approximation space:

$$V_h = \{\phi_h \in V \cap \mathcal{C}^0 | \phi_{h|K} \quad \text{is affine } \forall K \in \mathcal{H}\}.$$

The interpolation operator of the previous section is chosen as the usual $\mathcal{P}^1$ operator:

$$\Pi_h : \ V \cap \mathcal{C}^0 \to V_h \quad \text{such that } \Pi_h \varphi(\mathbf{x}_i) = \varphi(\mathbf{x}_i),$$

for all vertices $\mathbf{x}_i$ of $\mathcal{H}$. The weak discrete formulation writes:

$$\forall \phi_h \in V_h, \quad (\Psi_h(W_h), \phi_h) = \int_{\Omega_h} \phi_h \nabla \cdot \mathcal{F}_h(W_h)\, d\Omega_h - \int_{\Gamma_h} \phi_h \widehat{\mathcal{F}_h}(W_h) \cdot \mathbf{n}\, d\Gamma_h = 0, \tag{14}$$

with $\mathcal{F}_h = \Pi_h \mathcal{F}$ and $\widehat{\mathcal{F}_h} = \Pi_h \widehat{\mathcal{F}}$. Taking as in (14) the $\mathcal{P}^1$-interpolation of the fluxes $\mathcal{F}_($ as discretisation principle, produce a finite-element scheme which is identical to the central-differenced finite-volume scheme built on the so-called median dual cells, built around vertices by limiting them by plans through mid-edges, face centroids, element centroids. In practice, this family of mixed-element-volume schemes cannot be used in a non-dissipative purely centered version. In [14,44], MUSCL versions are described and analysed. We shall recall in Section 6 some features of these schemes which will be used in the numerical experiments. A proof that the abstract analysis of Section 2 applies to these schemes with neglectible remainders has not yet been obtained. However, we shall apply it anyway. The scheme under study is then enriched with artificial stabilisation terms, we denote this as follows:

$$\forall \phi_h \in V_h, \quad \int_{\Omega_h} \phi_h \nabla \cdot \mathcal{F}_h(W_h)\, d\Omega_h - \int_{\Gamma_h} \phi_h \widehat{\mathcal{F}_h}(W_h) \cdot \mathbf{n}\, d\Gamma_h = -\int_{\Omega_h} \phi_h D_h(W_h) d\Omega_h.$$

According to [44], the diffusion term is of higher order as soon as it is applied to the interpolation of a smooth enough field W on a sufficiently regular mesh:

$$\left| \int_{\Omega_h} \phi_h D_h(W_h) \mathrm{d}\Omega_h \right| \leqslant h^3 K(W) |\phi_h|_{L^2}.$$

As a result, the dissipation term will be neglected in the same way we neglect the remainders R_i and D_k of Relation (11). In the case of a flow with shocks, we have chosen to follow the strategy of the Hessian-based study in [43] which consists in avoiding to introduce the error term from artificial dissipation.

3.2. Approximation error estimation

Returning to the output functional $j(W) = (g, W)$ and according to Estimate (11), the main term of the *a priori* error estimation of δj becomes:

$$\delta j = (g, W - W_h) \approx ((\Psi_h - \Psi)(W), W^*),$$

where W^* is the continuous adjoint state, solution of:

$$\frac{\partial \Psi}{\partial W} W^* = g.$$

Using the exact solution W in Eqs. (13) and (14) while neglecting the dissipation D_h leads to:

$$(g, W - W_h) \approx \int_{\Omega_h} W^* \left(\nabla \cdot \mathcal{F}_h(W) - \nabla \cdot \mathcal{F}(W) \right) \mathrm{d}\Omega_h - \int_{\Gamma_h} W^* \left(\widehat{\mathcal{F}_h}(W) - \widehat{\mathcal{F}}(W) \right) \cdot \mathbf{n} \, \mathrm{d}\Gamma_h.$$

By integrating by parts the previous estimate, it comes:

$$(g, W - W_h) \approx \int_{\Omega_h} \nabla W^* \left(\mathcal{F}(W) - \mathcal{F}_{\langle}(W) \right) \mathrm{d}\Omega_h - \int_{\Gamma_h} W^* \left(\overline{\mathcal{F}}(W) - \overline{\mathcal{F}_h}(W) \right) \cdot \mathbf{n} \, \mathrm{d}\Gamma_h,$$

where fluxes $\overline{\mathcal{F}}$ are defined by:

$$\overline{\mathcal{F}}(W) \cdot \mathbf{n} = \mathcal{F}(W) \cdot \mathbf{n} - \widehat{\mathcal{F}}(W) \cdot \mathbf{n}.$$

Using the definition of $\mathcal{F}_h$ and $\widehat{\mathcal{F}}_h$, we get:

$$\delta j \approx \int_{\Omega_h} \nabla W^* \left(\mathcal{F}(W) - \Pi_h \mathcal{F}(W) \right) \mathrm{d}\Omega_h - \int_{\Gamma_h} W^* \left(\overline{\mathcal{F}}(W) - \Pi_h \overline{\mathcal{F}}(W) \right) \right) \cdot \mathbf{n} \, \mathrm{d}\Gamma_h. \tag{15}$$

We observe that this estimate of δj is expressed in terms of interpolation errors for the fluxes and in terms of the continuous functions W and W^*.

3.3. Error bound with a safety principle

The integrands in (15) contain positive and negative parts which can compensate for some particular meshes. In our strategy, we prefer to avoid these parasitic effects. To this end, all integrands are bounded by their absolute values:

$$(g, W_h - W) \leqslant \int_{\Omega_h} |\nabla W^*| |\mathcal{F}(W) - \Pi_h \mathcal{F}(W)| d\Omega_h + \int_{\Gamma_h} |W^*| |(\overline{\mathcal{F}}(W) - \Pi_h \overline{\mathcal{F}}(W)).\mathbf{n}| d\Gamma_h. \tag{16}$$

In other words, we prefer to locally over-estimate the error.

4. Error model minimization

Starting from Bound (16), several options are possible to derive an optimal mesh for the observed functional. A first one consists in estimating the right-hand side of (16) for each element then comparing it to a threshold value to conduct refinement. As already stated in the introduction, this method is by nature isotropic. In addition, this process is equivalent to the equi-distribution principle which is similar to a $\mathbf{L}^\infty$ norm error control. A weakness of a $\mathbf{L}^\infty$ norm error control is that small scales variations of (16) are ignored while strong variations regions are highly refined. This leads to a non optimal distribution of the degrees of freedom to capture all the scales of the solution [3,43]. A second method is to derive local mesh operators (smoothing, swaps, collapse,...) that are tried and applied only when they decrease the value of the right-hand side of (16), see [38]. However, as in the previous choice, this method consists in a local refinement and can only converge to a local minimum. Moreover, it corresponds to a steepest descent method which has a very poor convergence property. The common point of both approaches is to work on the discrete mesh. It seems particularly tedious to derive well-posed numerical operators directly on discrete meshes to decrease the error on a functional. This is even more complicated when trying to generate anisotropic meshes.

To avoid this, we propose to work in the continuous mesh framework by adopting a completely continuous standpoint, which is made easier thanks to the *a priori* estimate. It allows us to define proper differentiable optimization [1,7] or to use

the calculus of variations that is undefined on the class of discrete meshes. This framework lies in the class of metric-based methods. Consequently, every mesh generator which is able to deal with a metric field can be utilized whatever the meshing technique it uses: Delaunay, local Delaunay, local refinements,... Note that a lot of adaptive mesh generators are now able to interpret this metric concept. Let us mention [22] for discrete surface mesh adaptation and [11,13,17,26,37,39,45,51] in 3D. Working in this framework enables us to write Estimate (16) in a continuous form:

$$(g, W_h - W) \approx E(\mathbf{M}) = \int_\Omega |\nabla W^*|\,|\mathcal{F}(W) - \pi_\mathcal{M}\mathcal{F}(W)|\,d\Omega + \int_\Gamma |W^*|\,|(\overline{\mathcal{F}}(W) - \pi_\mathcal{M}\overline{\mathcal{F}}(W)).\mathbf{n}|\,d\Gamma, \tag{17}$$

where $\mathbf{M} = (\mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\Omega}$ is a continuous mesh defined by a Riemannian metric space and $\pi_\mathcal{M}$ is the continuous linear interpolate defined hereafter. We are now focusing on the following (continuous) mesh optimization problem:

$$\text{Find}\mathbf{M}_{opt} = \text{Argmin}_\mathbf{M} E(\mathbf{M}). \tag{18}$$

A constraint is added to the previous problem in order to bound mesh fineness. In this continuous framework, we impose the total number of nodes to be equal to a specified positive integer N. We now detail the continuous mesh and continuous interpolation framework.

4.1. Continuous mesh model

A continuous mesh $\mathbf{M} = (\mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\Omega}$ of Ω is a Riemannian metric field [10]. For all $\mathbf{x}$ of Ω, $\mathcal{M}(\mathbf{x})$ is a symmetric tensor having $(\lambda_i(\mathbf{x}))_{i=1,3}$ as eigenvalues along the principal directions $\mathcal{R}(\mathbf{x}) = (\mathbf{v}_i(\mathbf{x}))_{i=1,3}$. Sizes along these directions are denoted $(h_i(\mathbf{x}))_{i=1,3} = (\lambda_i^{-2}(\mathbf{x}))_{i=1,3}$. With this definition, $\mathbf{M}$ admits the more practical local decomposition:

$$\mathcal{M}(\mathbf{x}) = d^{\frac{2}{3}}(\mathbf{x})\,\mathcal{R}(\mathbf{x}) \begin{pmatrix} r_1^{-\frac{2}{3}}(\mathbf{x}) & & \\ & r_2^{-\frac{2}{3}}(\mathbf{x}) & \\ & & r_3^{-\frac{2}{3}}(\mathbf{x}) \end{pmatrix} {}^t\mathcal{R}(\mathbf{x}),$$

where

- the node density d is equal to: $d = (h_1 h_2 h_3)^{-1} = (\lambda_1\lambda_2\lambda_3)^{\frac{1}{2}} = \sqrt{\det(\mathcal{M})}$,
- the three anisotropic quotients r_i are equal to: $r_i = h_i^3 (h_1 h_2 h_3)^{-1}$.

The anisotropic quotients represent the overall anisotropic ratio of a tetrahedron taking into account all the possible directions. It is a complementary measure to anisotropic ratio given by $\max_i(h_i)/\min_i(h_i)$. By integrating the node density, we define the complexity $\mathcal{C}$ of a continuous mesh which is the continuous counterpart of the total number of vertices:

$$\mathcal{C}(\mathbf{M}) = \int_\Omega d(\mathbf{x})\,d\mathbf{x} = \int_\Omega \sqrt{\det(\mathcal{M}(\mathbf{x}))}\,d\mathbf{x}.$$

This real-value parameter is useful to quantify the global level of accuracy of the continuous mesh $\mathbf{M} = (\mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\Omega}$.

It has been shown in [42] that $\mathbf{M}$ defines a class of equivalence of discrete meshes. The equivalence relation is based on the notion of *unit mesh* with respect to $\mathbf{M}$. A mesh $\mathcal{H}$ is unit wih respect to $\mathbf{M}$ when each tetrahedron $K \in \mathcal{H}$ defined by its list of edges $(\mathbf{e}_i)_{i=1\ldots6}$ verifies:

$$\forall i \in [1,6], \quad \ell_\mathcal{M}(\mathbf{e}_i) \in \left[\frac{1}{\sqrt{2}}, \sqrt{2}\right] \quad \text{and} \quad Q_\mathcal{M}(K) \in [\alpha, 1] \text{ with } \alpha > 0.$$

A classical and admissible value of α is 0.8. The length of an edge $\ell_\mathcal{M}(\mathbf{e}_i)$ and the quality of an element $\mathcal{Q}_\mathcal{M}(K)$ are integrated to take into account the variations of $\mathbf{M}$ in Ω:

$$Q_\mathcal{M}(K) = \frac{36}{3^{\frac{1}{3}}} \frac{|K|_\mathcal{M}^{\frac{2}{3}}}{\sum_{i=1}^6 \ell_\mathcal{M}^2(\mathbf{e}_i)} \in [0,1], \quad \text{with } |K|_\mathcal{M} = \int_K \sqrt{\det(\mathcal{M}(\mathbf{x}))}\,d\mathbf{x},$$

$$\text{and} \quad \ell_\mathcal{M}(\mathbf{e}_i) = \int_0^1 \sqrt{{}^t\mathbf{ab}\,\mathcal{M}(\mathbf{a}+t\,\mathbf{ab})\,\mathbf{ab}}\,dt, \text{with } \mathbf{e}_i = \mathbf{ab}.$$

This model is also particularly well suited to the study of the interpolation error. Indeed, there exists a unique continuous interpolation error that models the (infinite) set of interpolation errors computed on the class of unit meshes. See [42] for the proof along with equivalence between discrete and continuous formulations. For a smooth function u, the continuous linear interpolate $\pi_\mathcal{M} u$ is a function of the Hessian H_u of u and verifies:

$$(u - \pi_\mathcal{M} u)(\mathbf{x}) = \frac{1}{10}\text{trace}\left(\mathcal{M}^{-\frac{1}{2}}(\mathbf{x})\,|H_u(\mathbf{x})|\,\mathcal{M}^{-\frac{1}{2}}(\mathbf{x})\right) = \frac{1}{10} d(\mathbf{x})^{-\frac{2}{3}} \sum_{i=1}^3 r_i(\mathbf{x})^{\frac{2}{3}} {}^t\mathbf{v_i}(\mathbf{x})\,|H_u(\mathbf{x})|\,\mathbf{v_i}(\mathbf{x}), \tag{19}$$

where $|H_u|$ is deduced from H_u by taking the absolute values of its eigenvalues. $\pi_{\mathcal{M}}$ replaces the discrete operator Π_h in this continuous framework. Note that (19) does not require any hypothesis linking u and $\mathbf{M}$ as, for instance, any alignment condition.

Relation (19) provides a point-wise estimation of the continuous interpolation error. Now, it is then possible to set the global optimization problem of finding the optimal continuous mesh minimizing the $\mathbf{L}^p$ norm of the continuous interpolation error:

$$\text{Find } \mathbf{M}_{opt} = \text{Argmin}_{\mathbf{M}} \left(\int_\Omega \|u - \pi_{\mathcal{M}} u\|^p \, d\Omega \right)^{\frac{1}{p}},$$

under the equality constraint $\mathcal{C}(\mathbf{M}) = N$. This problem can be solved analytically by a calculus of variation and an analytical expression of the optimal continuous mesh can be exhibited [43]. The solution is unique. The same reasoning as [43] can be applied for the three following particular cases.

4.2. Weighted interpolation error

Let u be a twice continuously differentiable function and g be a strictly positive function. We consider the following optimization problem in the continuous mesh framework:

$$\text{Find } \mathbf{M}_{wgt} = \text{Argmin}_{\mathbf{M}} E_{wgt}(\mathbf{M}) \quad \text{with } E_{wgt}(\mathbf{M}) = \int_\Omega \mathbf{g} |\mathbf{u} - \pi_{\mathcal{M}} \mathbf{u}| \mathbf{d} \Omega,$$

under the equality constraint $\mathcal{C}(\mathbf{M}) = N$. The continuous interpolation error related to continuous mesh $\mathbf{M}$, can be expressed (up to a constant negligible thanks to the constraint $\mathcal{C}(\mathbf{M}) = N$) in terms of the Hessian H_u of function u, see Eq. (19):

$$E_{wgt}(\mathbf{M}) = \int_\Omega \mathbf{g} \left(\sum_{\mathbf{i}=\mathbf{1}}^{\mathbf{3}} \mathbf{h_i^2}^{\mathbf{t}} \mathbf{v_i} |\mathbf{H_u}| \mathbf{v_i} \right) \mathbf{d} \Omega.$$

This functional can be reformulated by using the anisotropic quotients r_i and the density d of $\mathbf{M}$:

$$E_{wgt}(\mathbf{M}) = \int_\Omega \mathbf{g} \mathbf{d}^{-\frac{2}{3}} \left(\sum_{\mathbf{i}=\mathbf{1}}^{\mathbf{3}} \mathbf{r_i^{\frac{2}{3}}}^{\mathbf{t}} \mathbf{v_i} |\mathbf{H_u}| \mathbf{v_i} \right) \mathbf{d} \Omega. \tag{20}$$

Note that the weight g does not affect the anisotropy contained in the functional

$$\left(\sum_{i=1}^{3} r_i^{\frac{2}{3}t} \mathbf{v}_i |H_u| \mathbf{v}_i \right),$$

which can be minimized locally in an independent manner. Consequently, following the reasoning of [43] used for Hessian-based mesh adaptation, $\mathbf{M}$ must be aligned locally with the eigenvectors of H_u and the anisotropic quotients are equal to the anisotropic quotients of H_u. At optimum, it simplifies to:

$$\left(\sum_{i=1}^{3} r_i^{\frac{2}{3}t} \mathbf{v}_i |H_u| \mathbf{v}_i \right) = 3 \det(|H_u|)^{\frac{1}{3}}. \tag{21}$$

As a consequence, the weighting influences only the density of the optimal metric. For a variation δd such that $\int_\Omega \delta d = 0$, we get

$$\int_\Omega g \, d^{-\frac{5}{3}} \det(|H_u|)^{\frac{1}{3}} \delta d = \int_\Omega d^{-\frac{5}{3}} \det(g |H_u|)^{\frac{1}{3}} \delta d = 0.$$

This means that the above integrand is equal to a constant C. The value of C is deduced from the constraint $\mathcal{C}(\mathcal{M}) = N$. Finally we get the following point-wise expression of the optimal continuous mesh $\mathbf{M}_{wgt} = (\mathcal{M}_{wgt}(\mathbf{x}))_{\mathbf{x} \in \Omega}$:

$$\mathcal{M}_{wgt}(g, u) = D(g, u) \det(g |H_u|)^{-\frac{1}{5}} g |H_u|,$$

where $D(g, u)$ is given by:

$$D(g, u) = N^{\frac{2}{3}} \left(\int_\Omega (\det(g |H_u|))^{\frac{2}{5}} \right)^{-\frac{2}{3}}.$$

4.3. Sum of interpolation errors

We now prove that the previous variational calculus extends to a linear combination of interpolation errors. In Hessian-based mesh adaptation, taking into account several interpolation errors in an anisotropic way is difficult. Practically, this is

done by intersecting several metric fields as in [3]. However, in many cases the two criteria are not enough compatible and this results in poor anisotropy. In addition, the intersection procedure often relies on simultaneous reduction which is not well-posed as soon as more than two metric fields are intersected and tends to result in an isotropic metric field. We specify in details how the same issue is solved in the goal-oriented case under study.

Let $u, v, \alpha > 0$ and $\beta > 0$ be four twice continuously differentiable functions. We aim at finding the metric which optimizes the $\mathbf{L}^1$ norm of the weighted sum of interpolation errors:

$$\text{Find } \mathbf{M}_{sum} = \text{Argmin}_{\mathbf{M}} E_{sum}(\mathbf{M}),$$

with

$$E_{sum}(\mathbf{M}) = \int_\Omega \alpha |\mathbf{u} - \pi_{\mathcal{M}}\mathbf{u}| \, \mathbf{d}\Omega + \int_\Omega \beta |\mathbf{v} - \pi_{\mathcal{M}}\mathbf{v}| \, \mathbf{d}\Omega,$$

under the constraint $\mathcal{C}(\mathbf{M}) = \mathbf{N}$.

It can be shown that minimizing E_{sum} is equivalent to minimizing a single interpolation error of a function having as Hessian the linear combination $\alpha|H_u| + \beta|H_v|$, where $|H_u|$ and $|H_v|$ are the absolute value of the Hessians of u and v. Indeed, using the definition of the continuous interpolate, we get:

$$\alpha|u - \pi_{\mathcal{M}}u| + \beta|v - \pi_{\mathcal{M}}v| = \alpha \sum_{i=1}^{3} h_i^2 \, {}^t\mathbf{v}_i |H_u| \mathbf{v}_i + \beta \sum_{i=1}^{3} h_i^2 \, {}^t\mathbf{v}_i |H_v| \mathbf{v}_i = \sum_{i=1}^{3} h_i^2 \, {}^t\mathbf{v}_i (\alpha|H_u| + \beta|H_v|) \mathbf{v}_i.$$

It remains to apply the same type of calculation as in the previous section. The optimal continuous mesh $\mathbf{M}_{sum} = (\mathcal{M}_{sum}(\mathbf{x}))_{\mathbf{x}\in\Omega}$ is given by the following point-wise optimal metric:

$$\mathcal{M}_{sum}(\alpha, u, \beta, v) = D(\alpha, u, \beta, v) \det(\alpha|H_u| + \beta|H_v|)^{-\frac{1}{5}} (\alpha|H_u| + \beta|H_v|),$$

where $D(\alpha, u, \beta, v)$ is given by:

$$D(\alpha, u, \beta, v) = N^{\frac{2}{3}} \left(\int_\Omega \det(\alpha|H_u| + \beta|H_v|)^{\frac{2}{5}} \right)^{-\frac{2}{3}}.$$

4.4. Mixing boundary and volume error contributions

We consider now an optimization problem involving at the same time a volume interpolation error term and a surface interpolation error term. Let u and $\bar{u}$ be two functions that are defined on Ω and $\Gamma = \partial\Omega$, respectively. The function $\bar{u}$ is simply the trace of u on the boundary. The need of considering a surface error term is that the trace of the optimal volume metric is not optimal on the surface. In other words, even if $\bar{u}$ is the trace of u on Γ, the optimal metric deduced from $\bar{u}$ is not the trace of the optimal metric deduced from u.

We recall that the projection of a 3D Hessian H_u of u on a surface element with an orthonormal basis $(\mathbf{s}_1, \mathbf{s}_2)$ writes:

$$\overline{H}_{\bar{u}} = \begin{pmatrix} {}^t\mathbf{s}_1 H_u \mathbf{s}_1 & {}^t\mathbf{s}_1 H_u \mathbf{s}_2 \\ {}^t\mathbf{s}_1 H_u \mathbf{s}_2 & {}^t\mathbf{s}_2 H_u \mathbf{s}_2 \end{pmatrix}.$$

As 3D and 2D Hessians are not necessarily correlated, we choose to find two independant optimal metrics for the volume and the surface, respectively.

The optimal solution is then sought as a couple of two 3D metric fields: $\mathbf{M}_{vol}$ defined in the whole domain Ω and $\mathbf{M}_{surf}$ defined only on the boundary Γ. $\mathbf{M}_{surf}$ is intended to control the interpolation error on the boundary with respect to $\bar{u}$, consequently, only two principal directions of $\mathbf{M}_{surf}$ are of interest. Indeed, along the normal to the boundary, no interpolation error is committed. Therefore, the following point-wise decomposition of $\mathbf{M}_{surf} = (\mathcal{M}_{surf}(\mathbf{x}))_{\mathbf{x}\in\Omega}$ is used:

$$\mathcal{M}_{surf} = (\mathbf{n}, \mathbf{s}) \begin{pmatrix} h_{max}^{-2} & \\ & \overline{\mathcal{M}}_{surf} \end{pmatrix} {}^t(\mathbf{n}, \mathbf{s}), \tag{22}$$

where $\mathbf{n}$ is the unit normal to the boundary, $\mathbf{s} = (\mathbf{s}_1, \mathbf{s}_2)$ stands for two orthonormal vectors lying in the orthogonal plane to $\mathbf{n}$ and $\overline{\mathcal{M}}_{surf}$ is a 2D metric. The initial unknown $\mathcal{M}_{surf}$ is replaced by the reduced one $\overline{\mathcal{M}}_{surf}$ in the analysis. The parameter h_{max} is a sufficiently large user-defined real value. This choice has no particular consequence on the final solution as a regularization process will be applied between the volume and the surface metrics, as explained hereafter. We can now state the optimization problem in its reduced form.

We consider g and $\bar{g}$ two positive functions. The problem reads:

$$\text{Find } (\mathbf{M}_{vol}, \overline{\mathbf{M}}_{surf}) = \text{Argmin}_{\mathbf{M},\overline{\mathbf{M}}} E_{surf}(\mathbf{M}, \overline{\mathbf{M}}),$$

with

$$E_{surf}(\mathbf{M},\overline{\mathbf{M}}) = \int_{\Omega} \mathbf{g}|\mathbf{u} - \pi_{\mathcal{M}}\mathbf{u}|\,\mathbf{d}\Omega + \int_{\Gamma} \bar{\mathbf{g}}|\bar{\mathbf{u}} - \pi_{\overline{\mathcal{M}}}\bar{\mathbf{u}}|\,\mathbf{d}\Gamma,$$

under the constraint

$$\mathcal{C}(\mathbf{M}) + \mathcal{C}(\overline{\mathbf{M}}) = N.$$

We recall that the interpolation error in the volume (resp. on the surface) for a function u (resp. $\bar{u}$) of Hessian H_u (resp. $\overline{H}_{\bar{u}}$) with respect to a continuous mesh $\mathbf{M}$ (resp. $\overline{\mathbf{M}}$) is given by:

$$|u - \pi_{\mathcal{M}}u| = \sum_{i=1}^{3} h_i^2\, {}^t\mathbf{v}_i |H_u| \mathbf{v}_i,$$

and respectively:

$$|\bar{u} - \pi_{\overline{\mathcal{M}}}\bar{u}| = \sum_{i=1}^{2} \bar{h}_i^2\, {}^t\bar{\mathbf{v}}_i |\overline{H}_{\bar{u}}| \bar{\mathbf{v}}_i,$$

where $\mathbf{v}_i$ and $\bar{\mathbf{v}}_i$ are the principal directions of volume metric $\mathcal{M}$ and of surface metric $\overline{\mathcal{M}}$ and, h_i and $\bar{h}_i$ are the mesh size in these directions.

E_{surf} is then expressed in function of Hessians H_u and $\overline{H}_{\bar{u}}$:

$$E_{surf}(\mathbf{M},\overline{\mathbf{M}}) = 3\int_{\Omega} g\left(d^{-\frac{2}{3}} \det(|H_u|)^{\frac{1}{3}}\right)\mathrm{d}\Omega + 2\int_{\Gamma} \bar{g}\left(\bar{d}^{-1} \det(|\overline{H}_{\bar{u}}|)^{\frac{1}{2}}\right)\mathrm{d}\Gamma, \tag{23}$$

under the constraint

$$\int_{\Omega} d + \int_{\Gamma} \bar{d} = N.$$

The Euler–Lagrange optimality condition associated with (23) writes:

$$\int_{\Omega} g \det(|H_u|)^{\frac{1}{3}} d^{-\frac{5}{3}}\delta d + \int_{\Gamma} \bar{g} \det(|\overline{H}_{\bar{u}}|)^{\frac{1}{2}} \bar{d}^{-2}\delta\bar{d} = 0, \tag{24}$$

where functions $(\delta d, \delta\bar{d})$ satisfy:

$$\int_{\Omega} \delta d + \int_{\Gamma} \delta\bar{d} = 0.$$

This still holds if the variation δd is perturbed by any $\delta d'$ of zero integral on Ω. Thus:

$$\int_{\Omega} g \det(|H_u|)^{\frac{1}{3}} d^{-\frac{5}{3}}\delta d' + \int_{\Omega} g \det(|H_u|)^{\frac{1}{3}} d^{-\frac{5}{3}}\delta d + \int_{\Gamma} \bar{g} \det(|\overline{H}_{\bar{u}}|)^{\frac{1}{2}} \bar{d}^{-2}\delta\bar{d} = 0,$$

it results:

$$\int_{\Omega} g \det(|H_u|)^{\frac{1}{3}} d^{-\frac{5}{3}}\delta d' = 0,$$

with the constraint

$$\int_{\Omega} \delta d' = 0.$$

Then, there exists a constant C such that:

$$g \det(|H_u|)^{\frac{1}{3}} d^{-\frac{5}{3}} = C \quad \Rightarrow \quad d = C^{-\frac{3}{5}} g^{\frac{3}{5}} \det(|H_u|)^{\frac{1}{5}}. \tag{25}$$

We apply the same argument to the surface term. Then, there exists a constant $\overline{C}$ such that:

$$\bar{g} \det(|\overline{H}_{\bar{u}}|)^{\frac{1}{2}} \bar{d}^{-2} = \overline{C} \quad \Rightarrow \quad \bar{d} = \overline{C}^{-\frac{1}{2}} \bar{g}^{\frac{1}{2}} \det(|\overline{H}_{\bar{u}}|)^{\frac{1}{4}}. \tag{26}$$

It remains to satisfy the constraint requiring the total number of vertices to be equal to N. This gives a relation between C and $\overline{C}$:

$$C^{-\frac{3}{5}} \int_{\Omega} g^{\frac{3}{5}} \det(|H_u|)^{\frac{1}{5}} + \overline{C}^{-\frac{1}{2}} \int_{\Gamma} \bar{g}^{\frac{1}{2}} \det(|\overline{H}_{\bar{u}}|)^{\frac{1}{4}} = N.$$

To complete the identification of the optimal solution, we observe that the following couple $(\delta d, \delta\bar{d})$ is admissible:

$$(\delta d, \delta\bar{d}) = \left(\frac{1}{|\Omega|}, -\frac{1}{|\Gamma|}\right),$$

since

$$\int_\Omega \delta d + \int_\Gamma \delta \bar{d} = 1 - 1 = 0.$$

We introduce it in Relation (24). Using (25) and (26), we get a second relation between C and $\overline{C}$:

$$C - \overline{C} = 0.$$

Therefore, C and $\overline{C}$ are the solutions of the nonlinear algebraic problem.

$$\begin{aligned} & aC^{-\frac{3}{5}} + bC^{-\frac{1}{2}} = N \\ & \overline{C} = C \\ & \text{with } a = \int_\Omega g^{\frac{3}{5}} \det(|H_u|)^{\frac{1}{5}}, \\ & \text{and } b = \int_\Gamma \bar{g}^{\frac{1}{2}} \det(|\overline{H}_{\bar{u}}|)^{\frac{1}{4}}. \end{aligned} \tag{27}$$

This ends the definition of the optimal volume and surface metrics. Finally, the global optimal metric $\mathbf{M}_{opt} = (\mathcal{M}_{opt}(\mathbf{x}))_{\mathbf{x}\in\Omega}$ is defined by:

$$\mathcal{M}_{opt}(\mathbf{x}) = \begin{cases} \mathcal{M}_{vol}(\mathbf{x}) & \text{for } \mathbf{x} \in \Omega \\ \mathcal{M}_{vol}(\mathbf{x}) \cap \mathcal{M}_{surf}(\mathbf{x}) & \text{for } \mathbf{x} \in \Gamma \end{cases}$$

where $\mathcal{M}_{surf}$ is defined by Relation (22). The metric intersection $\mathcal{M}_{vol} \cap \mathcal{M}_{surf}$ is to be understood as in [3].

The metrics $\mathcal{M}_{vol}$ and $\mathcal{M}_{surf}$ may not be compatible, and practically, a smooth matching is applied between both in the neighboring of the adapted surface. Note that the curvature of the surface is automatically taken into account by the used adaptive surface mesh generator [22]. If it is not the case, it is advised to also take into account the surface curvature in the surface metric definition. Practically, this is done by intersecting a pure geometric surface metric to the current computational surface metric. We refer to [24] for a geometric error estimate dedicated to surfaces from which a surface metric field is exhibited.

4.5. Optimal goal-oriented metric

The optimal metric is composed of a volume tensor field $\mathcal{M}_{go}$ defined in Ω and a surface one $\bar{\mathcal{M}}_{go}$ defined on Γ. We have:

- for each vertex $\mathbf{x}$ of Ω, a 3×3 matrix arising from the volume contribution of the sum of the Hessian of each component of the Euler fluxes weighted by the gradient of the adjoint state:

$$\mathbf{H}(\mathbf{x}) = \sum_{j=1}^{5} ([\Delta x]_j(\mathbf{x}) + [\Delta y]_j(\mathbf{x}) + [\Delta z]_j(\mathbf{x})), \tag{28}$$

  where

$$[\Delta x]_j = \left|\frac{\partial W_j^*}{\partial x}\right| \cdot |H(\mathcal{F}_1(W_j))|, \quad [\Delta y]_j = \left|\frac{\partial W_j^*}{\partial y}\right| \cdot |H(\mathcal{F}_2(W_j))|, \quad [\Delta z]_j = \left|\frac{\partial W_j^*}{\partial z}\right| \cdot |H(\mathcal{F}_3(W_j))|,$$

  with W_j^* denoting the j^{th} component of the adjoint vector W^* and $H(\mathcal{F}_i(W_j))$ the Hessian of the j^{th} component of the vector $\mathcal{F}_i(W)$,
- for each vertex $\mathbf{x}$ of Γ, a 2×2 matrix arising from the surface contribution:

$$\overline{\mathbf{H}}(\mathbf{x}) = \sum_{j=1}^{5} |W^*| \cdot \left|\overline{H}\left(\sum_{i=1}^{3} \overline{\mathcal{F}}_i(W) \cdot n_i\right)\right|, \tag{29}$$

  where $\mathbf{n} = (n_1, n_2, n_3)$ is the outward normal of Γ.

The standard $\mathbf{L}^1$ norm normalization is then applied independently on each metric goal oriented ("*go*") tensor field:

$$\mathcal{M}_{go}(\mathbf{x}) = C \det(|\mathbf{H}(\mathbf{x})|)^{-\frac{1}{5}} |\mathbf{H}(\mathbf{x})| \quad \text{and} \quad \bar{\mathcal{M}}_{go}(\mathbf{x}) = \overline{C} \det(|\overline{\mathbf{H}}(\mathbf{x})|)^{-\frac{1}{4}} |\overline{\mathbf{H}}(\mathbf{x})|. \tag{30}$$

Constants C et $\overline{C}$ depends on the desired complexity N. Their explicit values are given by the solution of non-linear Problem (27). Note that the metric $\overline{\mathcal{M}}_{go}(\mathbf{x})$ is 2D and needs to be transformed onto a 3D metric prior to the intersection with $\mathcal{M}_{go}(\mathbf{x})$ to get the final goal-oriented metric. This is done by setting an $h_{\max}$ size along the normal direction to the surface in $\bar{\mathcal{M}}_{go}(\mathbf{x})$ leading to $\mathcal{M}_{gosurf}$. Finally, the global optimal continuous mesh $\mathbf{M}_{opt} = (\mathcal{M}_{opt}(\mathbf{x}))_{\mathbf{x}\in\Omega}$ is defined by:

$$\mathcal{M}_{opt}(\mathbf{x}) = \begin{cases} \mathcal{M}_{go}(\mathbf{x}) & \text{for } \mathbf{x} \in \Omega \\ \mathcal{M}_{go}(\mathbf{x}) \cap \mathcal{M}_{gosurf}(\mathbf{x}) & \text{for } \mathbf{x} \in \Gamma \end{cases} \tag{31}$$

The continuous problem (18) has been solved from an explicit optimality condition producing the optimal metric field as a function of state and adjoint. In practice, it remains to approximatively solve the coupled system of state, adjoint, and mesh-optimality. This is done with a goal-oriented adaptive loop which we describe in Section 5.

4.6. About mesh anisotropy

In three dimensions, mesh anisotropy can be quantified by two notions: the anisotropic ratios and the anisotropic quotients. We first recall both notions and how they are evaluated. 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 $\mathcal{M}_K$:

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

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 zero. Once $\mathcal{M}_K$ is computed, the anisotropic ratio and the anisotropic quotient associated with element K are simply given by

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

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. 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. Adaptive strategy

The adaptive strategy for the proposed goal-oriented mesh adaptation is quite similar to any anisotropic metric-based mesh adaptation. As both the solution and the mesh are changing during the computation, a non-linear loop is set up in order to converge toward a fixed point for the couple mesh-solution. A sketch of the algorithm for steady problems is depicted in Fig. 1. From an initial couple mesh-solution $(\mathcal{H}_0, \mathcal{S}_0)$, it is composed of the following sequences. At step i, the flow is first converged on the current mesh $\mathcal{H}_i$ to get the solution $\mathcal{S}_i$. Then, a metric tensor field $\mathcal{M}_i$ is deduced from $(\mathcal{H}_i, \mathcal{S}_i)$ thanks to an anisotropic error estimate. The latter is used by the adaptive mesh generator which generates a unit mesh with respect to $\mathcal{M}_i$. The previous solution is then linearly interpolated on the new mesh. This procedure is repeated until convergence of the

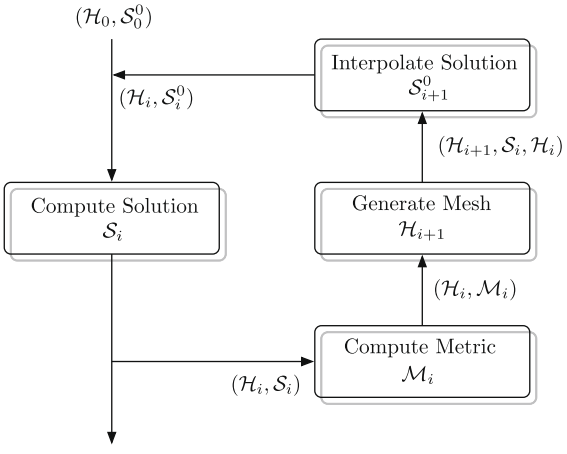


Fig. 1. Adaptive loop for steady flow simulations. $\mathcal{H}_i$ is the ith mesh, $\mathcal{S}_i$ the i*th* solution and $\mathcal{M}_i$ the metric computed from the couple $(\mathcal{H}_i, \mathcal{S}_i)$.

couple mesh-solution. We refer to [43] for more details. We now investigate the differences when dealing with the adjoint-based anisotropic error estimate. The main modifications concern the flow solver and the remeshing stage.

In this section, the following notations are used. $\mathcal{H}$ denotes the mesh of the domain Ω_h, $\partial\mathcal{H}$ the mesh of the boundary Γ_h of Ω_h, W_h is the state provided by the flow solver and $j(W_h)$ the observed functional defined on $\gamma \subset \Omega_h$.

5.1. Computing the goal-oriented metric

The optimal metric found in Section 4 defined by Relations (28)–(31) is function of the adjoint state, the gradient of the adjoint state and the Hessian of the Euler fluxes. In practice, these continuous states are approximated by the discrete states and derivative recovery from discrete states at each vertex of the mesh $\mathcal{H}$. The discrete adjoint state W_h^* is taken to represent the adjoint state W^*. The gradient of the adjoint state ∇W^* is replaced by $\nabla_R W_h^*$ and the Hessian of each component the flux vector $H(\mathcal{F}_i(W))$ is obtained from $H_R(\mathcal{F}_i(W_h))$. H_R stands for the operator that recovers numerically the second order derivatives of an initial piecewise linear by element solution field. In this paper, the recovery method is based on the Green formula. Its formulation along with some comparisons to other methods is available in [4].

5.1.1. Flow solver and adjoint state

As compared to Hessian-based mesh adaptation, the new step in the solver is the resolution of the linear system providing the adjoint state:

$$A_h^* W_h^* = g_h,$$

where g_h is the approximated Jacobian of $j(W_h)$ with respect to the conservative variables vector W_h and W_h^* is the adjoint state. A_h^* is the adjoint matrix of spatial order one deduced by linearizing the numerical scheme. A_h^* has been developed using Automatic Differentation tools as in [32,34]. A_h^* is a sparse matrix composed of 5×5 blocks. If we assume that the 3D mesh is composed of N vertices, the number of non zero block entries is given by $N + 2N_e$, where Ne is the number of edges of $\mathcal{H}$. In 3D, N_e can be approximated by a safe upper bound $8N$ (6 or $7N$ is usually observed practically), thus the size of the matrix is approximately $425N$. Storing this matrix in double precision for a mesh composed of one million vertices requires about 3.3 GB of memory.[2] The memory cost is even worse depending on the preconditioning technique. In our case, an incomplete $BILU(0)$ preconditioner is used [49]. This choice doubles the memory cost. This preconditioner is jointly used with an iterative solver as GMRES [49]. With this strategy, the adjoint state W_h^* is converged within 10 iterations with a residual of 10^{-6} for a 1 million vertices mesh. The CPU spent to solve the adjoint system is less than 2 min on a Intel Core 2 at 2.8 GHz. We mention that other techniques to solve this system exist and seem to be as efficient while requiring less memory, see [40].

Once W_h^* is computed, its point-wise gradient $\nabla_R W_h^*$ is recovered by using a $\mathbf{L}^2$ projection from the neighboring element-wise constant gradients [4]. We summarize the final couples of variables made available by the flow solver:

- the gradients of the discrete adjoint state $\nabla_R W_h^* = \left(\frac{\partial W_h^*}{\partial x}, \frac{\partial W_h^*}{\partial y}, \frac{\partial W_h^*}{\partial z}\right)$ associated with the discrete fluxes vector $(\mathcal{F}_1(W_h), \mathcal{F}_2(W_h), \mathcal{F}_3(W_h))$. Each entity is of size $N \times 5$,
- the discrete adjoint state W_h^* associated with the discrete boundary fluxes $\overline{\mathcal{F}}(W_h) \cdot \mathbf{n}$ on γ.

For a one million vertices mesh, storing all these couples of fields requires less than 30 MB which is negligible with respect to the adjoint matrix size.

As regards the approximation scheme implemented in our in-house flow solver `Wolf`, we refer to [4] for a complete description. To give a brief overview, the Euler system is discretised by means of a vertex-centered finite volume scheme on unstructured tetrahedral meshes. In this article, we use the HLLC approximate Riemann solver to compute the numerical flux. The high-order scheme is derived according to a MUSCL type method using downstream and upstream tetrahedra. An high-order scheme is deduced by using upwind and downwind gradients leading to a numerical dissipation of 4th or 6th order. As it, the previous MUSCL scheme is not monotone and so it needs to be limited to guarantee the TVD property of the scheme. The considered limiter is a generalization of the Superbee limiter with three entries. The time integration is an explicit algorithm using a 5-stage, 2-order strong-stability-preserving (SSP) Runge–Kutta scheme that allows us to consider a *CFL* coefficient up to 4. With these features, the solver is conservative, positivity preserving and monotone.

5.1.2. Mesh adaptation

Goal-oriented mesh adaptation requires to adapt the surface mesh of the surface γ on which the functional is observed. This standpoint is needed in order to ensure a valid coupling between the volume mesh and the surface mesh. This constraint implies numerous complications for the re-meshing phase. In our case, a global re-meshing is carried out after re-meshing the surface γ. We use `Yams` [22] for the adaptation of the surface and an anisotropic extension of `Gamhic` [26] for the volume mesh. When the surface is not adapted, we use `Mmg3d` [17]. In 2D, the mesh adaptation is also done by using `Yams`.

[2] The matrix memory requirement can be halved by storing it in simple precision.

6. Numerical examples

Numerical experiments are carried out in 2D and 3D on several observation functionals. The efficiency of the proposed approach is proved by confronting numerical results to wind-tunnel experimentations on a simple axisymmetric and on a wing-body geometries. Then, drawbacks and advantages of the adjoint-based and the Hessian-based mesh adaptations are discussed on several aeronautics flows: the high-fidelity prediction of pressure signature arising during supersonic flights, a challenging computation of the flow around a F15 fighter with the Quiet Spike concept attached [35] and the vorticity prediction in the wake of a transonic flight for a Falcon business jet.

6.1. 2D example: a half-scramjet

In this 2D example, we emphasize the ability of the adjoint state to predict flow areas that directly impact the observed functional. We consider an internal flow on a scramjet geometry at Mach 3. The density is observed on the bottom half of the output of the scramjet as illustrated in Fig. 2(left). The functional is given by:

$$j(W) = \frac{1}{2}\int_{\gamma}\left(\frac{\rho-\rho_{\infty}}{\rho_{\infty}}\right)^2 \mathrm{d}\gamma.$$

This choice is particularly interesting as the trailing edges (just before the output) create contact discontinuities that are generally difficult to capture automatically. To illustrate this complexity, the flow has been computed on an uniform mesh of about 150,000 vertices. In the results depicted in Fig. 2(right), we show that contact discontinuities are not clearly identified. Moreover, in Hessian based mesh adaptation, capturing a contact discontinuity depends on the specific choice of a variable. For instance, an adaptation on the pressure variable will ignore contact discontinuities. In the present approach, all the variables are implicitly taken into account as the adaptation depends on all the flux functions.

In this example, 20 mesh adaptation iterations have been done divided into five steps at a constant complexity:

$$[1000, 2000, 4000, 8000, 16,000].$$

For each step at a constant complexity, four sub-iterations are performed to converge. The final adapted mesh composed of 31,337 vertices and 61,630 triangles, and the final solution are depicted in Fig. 3. We note that areas that do not impact the functional are coarsely meshed as the shock in the upper part of the scramjet. This is clearly illustrated on the contact discontinuities emitted by the trailing edges before the exhaust: if the one issued from the lower trailing edge is accurately meshed, the upper one is completely diffused since it has no more impact on the functional.

The evolution of the adapted meshes sequence and the associated solutions are shown in Fig. 4. We can see that the chosen areas for refinement by the goal oriented error estimate evolve during the computation depending on the information currently available. For steps 1,2 and 3, the estimate mainly prescribes a refinement in the upper part of the scramjet. Indeed, the shock issued from the upper trailing edge is the main contribution to the error in the observed functional. When the contact discontinuity issued from the lower trailing edge becomes well captured (at step 3), the adaptive process specifies the lower part of the scramjet to be refined for step 4. During step 4, lower and upper parts of the scramjet are similarly refined. During the last step, the lower part is even more refined in order to increase the accuracy in the observed area, see Fig. 3.

6.2. Supersonic flows for several geometries

The presented goal-oriented mesh adaptation process is first validated by comparing numerical results to wind-tunnel experiments for a simple axisymmetric body and on a lifting wing-body configuration proposed by NASA to point out that the method can accurately predict full 3D configurations. Finally, the Hessian-based and the adjoint-based adaptations are compared for a flow around a supersonic business jet.

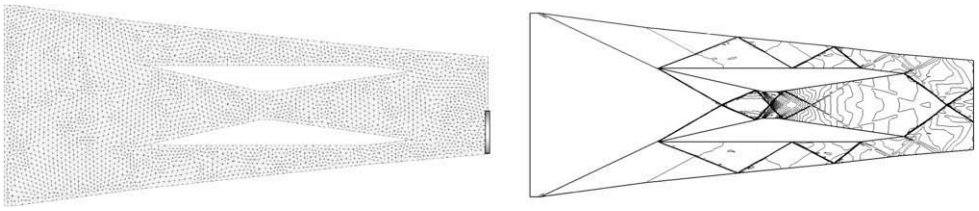


Fig. 2. Left, scramjet initial mesh where the grey area represent the surface of observation. Right, density iso-lines for a uniform mesh composed of 150,000 vertices.

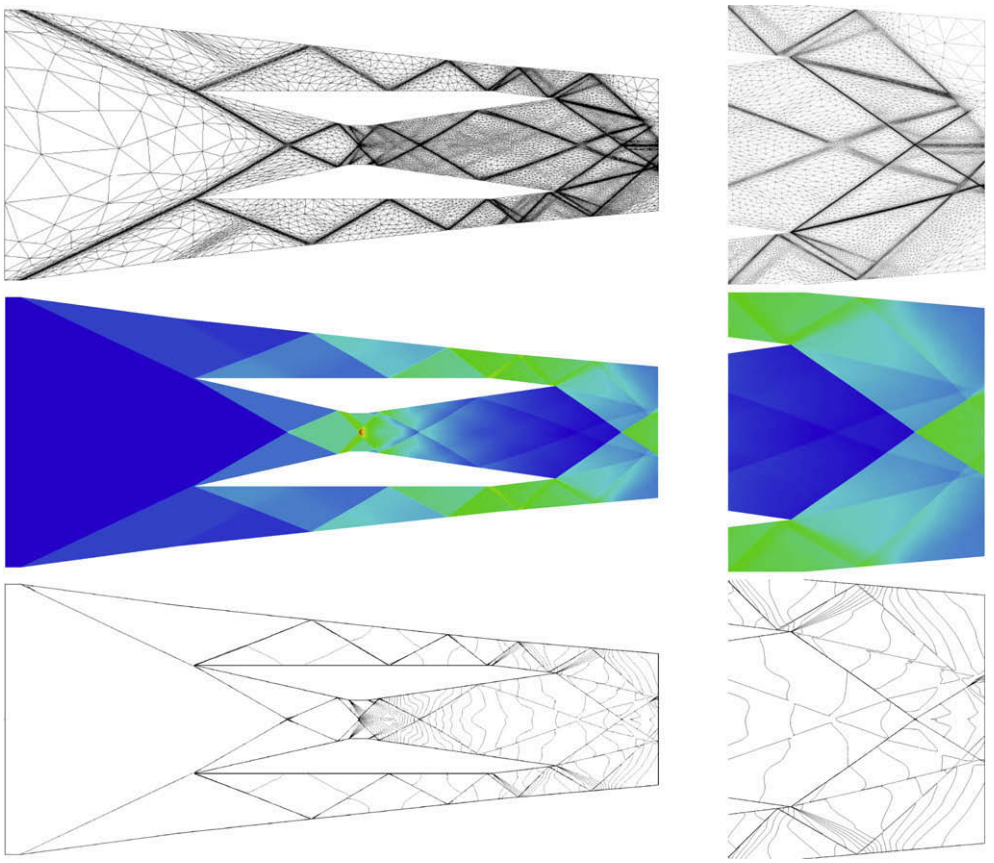


Fig. 3. Global view and close-up of the observation region view of the final adapted mesh and the final density iso-lines. Note that areas impacting the observation region are accurately meshed resulting in a non-symmetric mesh and solution.

6.3. Comparisons with experiments: an axisymmetric geometry

This section is a validation of the complete adaptive loop: solver and error estimate. To this end, we compare mid-field signatures obtain by our adaptive method with experimental wind tunnel results. The geometry of the problem corresponds to the 8th model of [33]. It represents a spike composed of a double cone and is given in inches by the following set of equations:

$$r = x\sqrt{\frac{0.08}{\pi}} \quad \text{if} \quad 0 \leqslant x \leqslant 0.25\,l$$
$$r = \sqrt{\frac{0.02}{\pi}} \quad \text{if} \quad 0.25\,l \leqslant x \leqslant 0.75\,l$$
$$r = \frac{2}{2+\sqrt{2}}\sqrt{\frac{0.04}{\pi}}\left(x - \frac{2-\sqrt{2}}{2}\right) \quad \text{if} \quad 0.75\,l \leqslant x \leqslant l,$$

where l is the reference length equal to 2 in. (5.08 cm). The geometry is depicted in Fig. 5(left). To simulate the sting support of the initial wind-tunnel configuration, another cylinder has been added of length 2l, see Fig. 5(middle). The computational domain is a cylinder of diameter 5 m and of length 6 m, see Fig. 5(right). Flow conditions are an angle of attack of 0° and a Mach number of 1.41. In this example, the flow is computed in the whole domain, i.e., the axis-symmetry of the geometry is not take into account. Indeed, our wish is to validate the 3D code that is going to be used on 3D fully non axis-symmetric complex configurations. This test case is also studied in the following references [4,37,55].

The pressure field is observed in the purple area shown in Fig. 5:

$$j(W) = \frac{1}{2}\int_{\gamma}\left(\frac{p-p_\infty}{p_\infty}\right)^2 \mathrm{d}\gamma.$$

Results are analyzed by extracting the mid-field pressure signatures along lines at various distances under the geometry from 5 to 20 body lengths. More precisely, we plot:

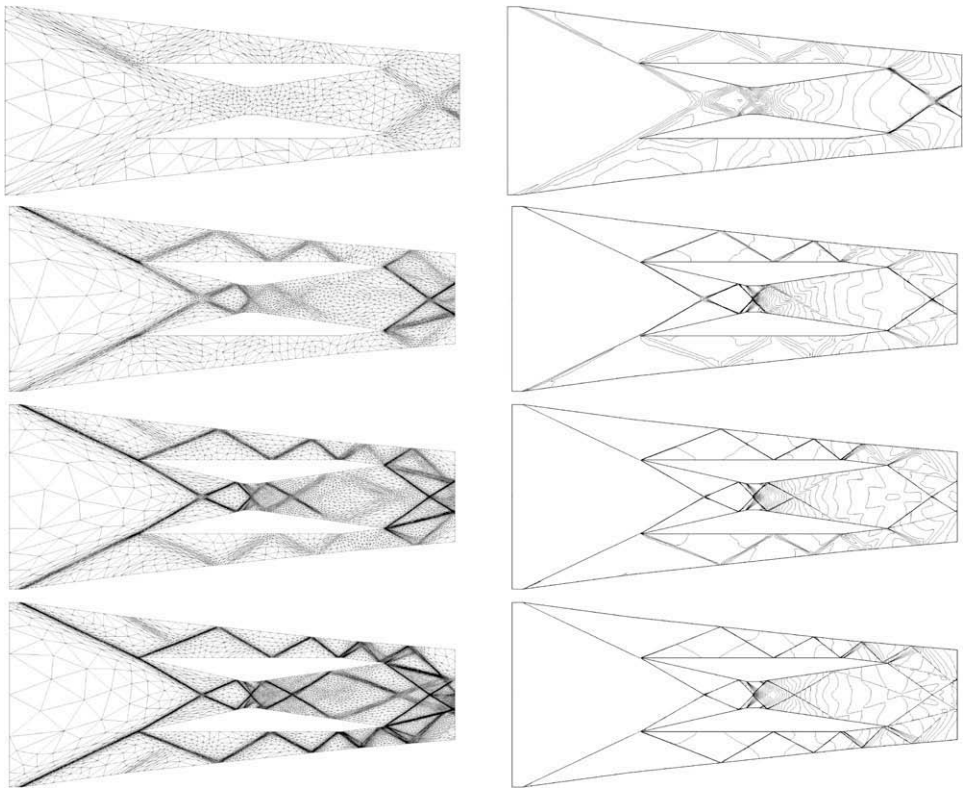


Fig. 4. Final adapted meshes (left) and associated density iso-lines (right) for the first four steps of the adaptive loop. From top to bottom, a complexity from 1000 to 8000 has been specified.

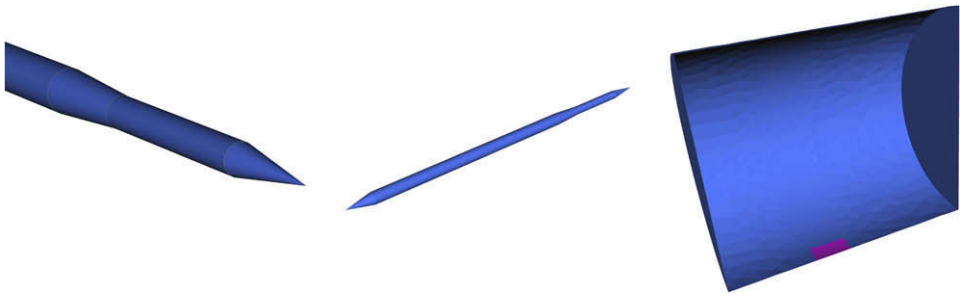
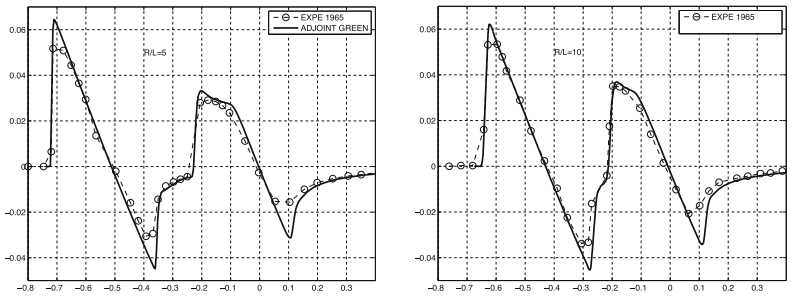


Fig. 5. Spike geometry (left), final spike geometry with a body added to stimulate the sting (middle), computational domain with the functional observation area in purple (right).

$$\Delta p = \left(\frac{R}{l}\right)^{\frac{3}{4}} \frac{p - p_\infty}{p_\infty} \text{ as a function of } \Delta x = \frac{x}{l}\left(\frac{R}{l}\right)^{-\frac{1}{4}},$$

where R the distance to the body. The final signatures are given in Fig. 6. These results correspond to a final adapted mesh composed of 2 million vertices. Notice that in [33], it is specified that the rounding of the measured pressure peaks in experimental data is believed to be due in part to the wind-tunnel vibrations and the boundary layer effects. Therefore, sharp shock wave peaks obtained in our simulations, see Fig. 6, are the good answer for our model. For both cases, the agreement with experimental data is thus excellent.

We can observe that, whatever the accuracy of the mesh, the sharpness and the intensity of the shock waves at $R/l = 20$ is preserved (or increased). The shock waves are not diffused at all through their propagation pointing out the drastic reduction of the flow solver numerical dissipation. This result agrees with the linear supersonic aerodynamic theory where asymptotically it is predicted that the pressure front δp is decreasing proportionally to $(R/l)^{-1/2}$.



6.3.1. Comparisons with experiments: a lifting wing-body configuration

The aim of this section is to validate the adaptive CFD model on a lifting configuration proposed by NASA and to point out that the method can accurately predict full 3D configurations. A lifting wing-body produces a flow field disturbance which is proportional to the degree of lift generated. The typical run time of the goal-oriented adaptive loop is also given.

Model 4 geometry of wind tunnel study [36] is selected. This configuration has also been studied in [55]. This lifting wing-body geometry consists of an axisymmetric fuselage and a sweep delta wing. The body is a cylinder with a parabolic nose of length 17.52 cm and diameter 1.08 cm. The fitness ratio (l/d) is thus 16.22. The thickness r of the parabolic nose, of length 7.01, is defined by

$$r = 0.54 - 0.011(x - 7.01)^2.$$

The wing is a 69° leading edge sweep delta wing and double-wedge 5%-thick diamond sections with the ridge line located at mid-chord. The wings are mounted on the cylindrical portion of the fuselage at the longitudinal plane of symmetry at $x = 8.21$ cm. The dimensions of the wing tunnel sting are not provided in [36]. Based on this limited data, the sting is represented by a simple body of revolution extending four body-lengths behind the aircraft. The sting geometry is exactly the same as the one in Ref. [55]. The geometry is immersed in a cylindrical domain aligned with the x-axis. The cylinder has a length of 2.75 m and a radius of 80 cm. This geometry is illustrated in Fig. 7.

The flow conditions are Mach 1.68 and a lift coefficient (C_L) of 0.08. The angle of attack was set to match the desired lift using the reference area equal to 33.106 cm^2.

The goal-oriented mesh adaptation considers again the observation of the pressure field but this time on a line γ located on the cylindrical domain in the symmetry plane:

$$j(W) = \frac{1}{2}\int_{\gamma}\left(\frac{p - p_\infty}{p_\infty}\right)^2 \mathrm{d}\gamma.$$

A mesh gradation of 3.5 has been set with an increase law coefficient of 1.1 [2]. A total of 10 adaptations have been performed split into two steps of five adaptations. At each step, the pair mesh-solution is algorithmically converged at a fixed complexity. We have fixed a complexity of 15,000 for the first step and 20,000 for the second one. It results in a final adapted mesh the size of which is almost 76,999 vertices and 378,592 tetrahedra.

The final mesh and its associated solution, the C_p iso-values, are shown in Fig. 8. The flow field is accurately computed under the lifting geometry in the sense that the solution has not been diffused throughout its propagation in the mid-field whereas the wake and the upper part of the flow have been poorly capture due to the coarse mesh. With only 76,999 vertices, the shock waves have been propagated down to the domain boundaries at a distance greater than 3-body lengths. This result points out that the numerical dissipation of the flow solver has been drastically reduced thanks to the anisotropic mesh refinement. Notice, Fig. 8(right), the strong refinement of the surface mesh on the domain boundary where the observation line lies.

The accuracy of the obtained solution is deeper analyzed by a comparison to the experimental data of [36]. The mid-field pressure signature is extracted at 3.6 body lengths, $R/L = 3.6$, under the geometry at the end of the simulation. The pressure signal plot, Fig. 9(middle-left), shows very good agreement with the wind tunnel data. As in [55], we observe a discrepancy between the numerical solution and the experiment for $x \in [1; 1.1]$ probably due to the inaccurate representation of the body sting.

This accurate solution has been obtained on a coarse mesh "only" composed of 76,999 vertices. This complexity reduction is achieved by the refinement of only the region of interest and the mesh anisotropy. The mesh anisotropy is quantified by the anisotropic ratios and the anisotropic quotients, as explained in Section 4.6. The mean anisotropic ratio is 47 and the mean anisotropic quotient is 1410. The last quantity signifies that the anisotropy leads to a mesh complexity reduction of three order of magnitudes as compared to an isotropic adapted mesh. Detailed histograms of the anisotropic ratio and quotient are given in Table 1.

This simulation is run on a eight-processors 64-bits Mac Pro with an Intel Core 2 chipsets with a clock speed of 2.8 GHz with 16 Gb of RAM. The CPU time for the whole computation is 23 min. The flow solver consumes 73% of the CPU time, the adjoint computation is included in this CPU. However, the total CPU time for solving all the adjoint equations is 55 s that corresponds to about 5% of the total CPU time of the flow solver. Detailed wall-clock time for each step of the adaptation loop are reported in Table 2. Note that 3 min of CPU times have to be added to the flow solver to compute the initial state on the initial mesh.

6.3.2. Additional validation

The two additional lift conditions for Mach 1.68 presented in [36] are considered. Exactly the same data are set for each simulation. The pressure signals predicted for $C_L = 0$ and $C_L = 0.15$ at each step of the adaptation loop are shown in Fig. 9(left). In both cases, agreement with the wind tunnel data is very good. The CFD solutions are obtained with relatively coarse meshes composed of almost 77,000 vertices and the CPU time is between 23 and 24 min.

Finally, the three additional cases at higher Mach, $M = 2.7$, for C_L equal to 0,0.08 and 0.15 of [36] are run. The comparison to the wind tunnel data is done by observing the pressure signal at $R/L = 3.1$. Again, agreement with the experimental data is

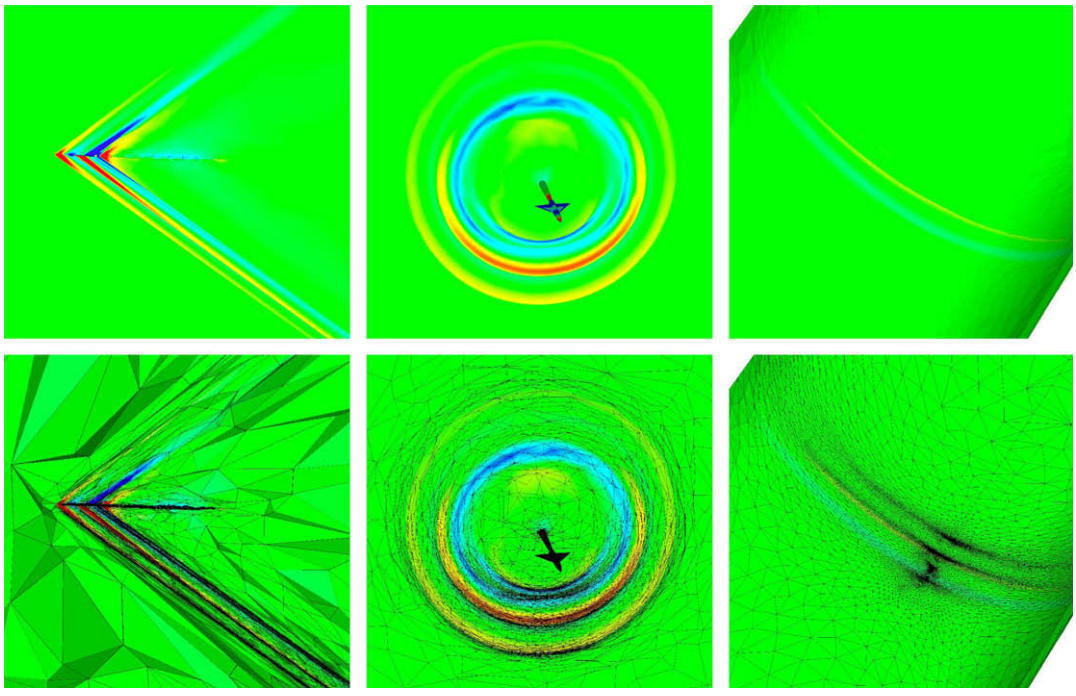


Fig. 8. Lifting wing-body. C_p iso-values (top) and final adapted mesh (bottom) for two cut planes and the domain surface. Left, the symmetry plane $y = 0$, middle, the plane $x = 0.35$ and, right, bottom surface of the domain where the observation line is located.

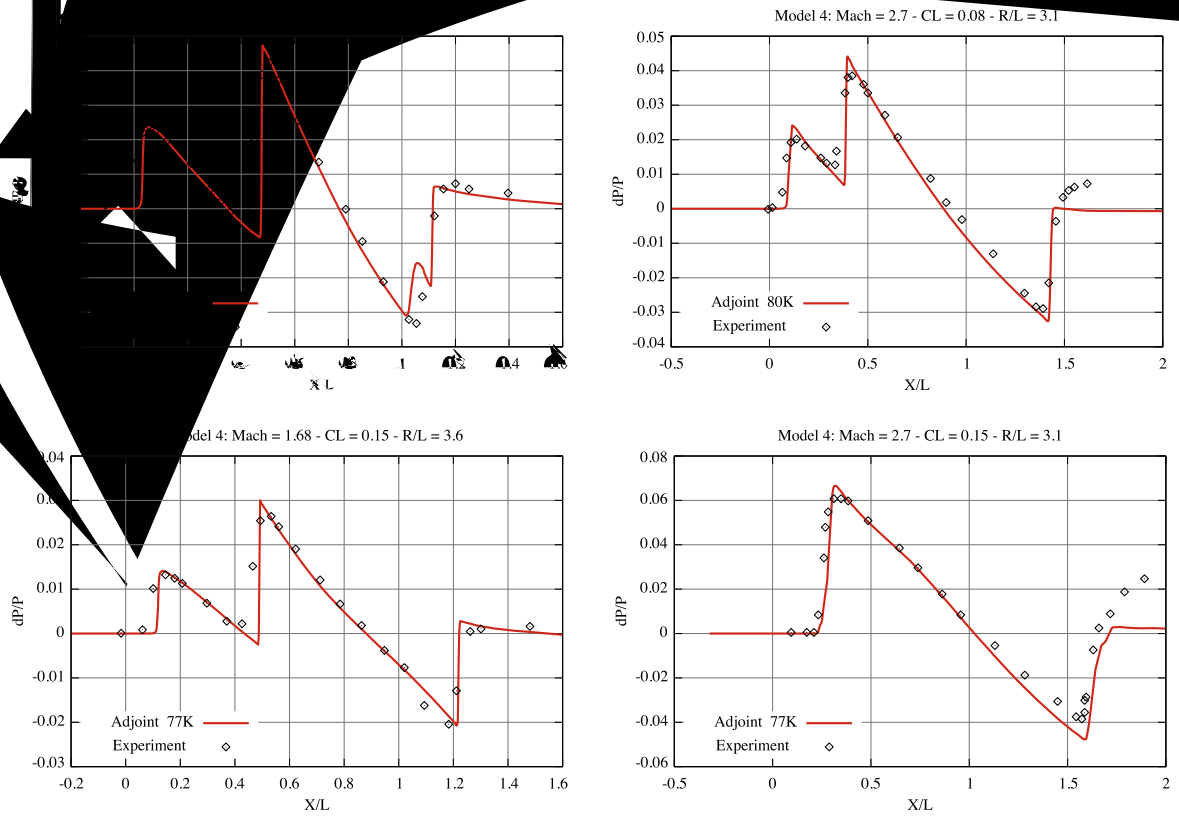


Fig. 9. Lifting wing-body. Left, pressure signals at $R/L = 3.6$ for $M = 1.68$. Right, pressure signals at $R/L = 3.1$ for $M = 2.7$. From top to bottom, lift coefficient equal to 0, 0.08 and 0.15.

Table 1
Anisotropic ratio (left) and quotient (right) histograms for the lifting wing-body final adapted mesh composed of 378 592 tetrahedra. For each interval, the number of tetrahedra is given with the corresponding percentage.

Anisotropic ratio (%)	Lifting wing-body (%)	
$1 < \text{ratio} \leqslant 2$	2015	0.53
$2 < \text{ratio} \leqslant 3$	10,228	2.70
$3 < \text{ratio} \leqslant 4$	15,290	4.04
$4 < \text{ratio} \leqslant 5$	16,452	4.35
$5 < \text{ratio} \leqslant 10$	65,243	17.23
$10 < \text{ratio} \leqslant 50$	159,536	42.14
$50 < \text{ratio} \leqslant 100$	60,030	15.86
$100 < \text{ratio} \leqslant 10^3$	49,758	13.14
$10^3 < \text{ratio} < 10^4$	40	0.0
Mean ratio	47	
Mean quotient	1 410	
$1 < \text{quo} \leqslant 2$	546	0.14
$2 < \text{quo} \leqslant 3$	2909	0.77
$3 < \text{quo} \leqslant 4$	4770	1.26
$4 < \text{quo} \leqslant 5$	5515	1.46
$5 < \text{quo} \leqslant 10$	27,305	7.21
$10 < \text{quo} \leqslant 50$	88,124	23.28
$50 < \text{quo} \leqslant 100$	38,026	10.04
$100 < \text{quo} \leqslant 10^3$	120,849	31.92
$10^3 < \text{quo} \leqslant 10^4$	80,058	21.15
$10^4 < \text{quo} \leqslant 10^5$	10,389	2.74
$10^5 < \text{quo} \leqslant 10^6$	98	0.03
$10^6 < \text{quo}$	3	0.00

Table 2
CPU times for each step of the mesh adaptation loop for the lifting wing-body test case at $M = 1.68$ and $C_L = 0.08$.

Step	Solver	Metric	Surf. mesh	Vol. mesh	Interpolation	Global
Total CPU time	16m50s	19s	49s	4m43s	21s	23m02s
Percentage	73.13	1.36	3.53	20.48	1.50	100

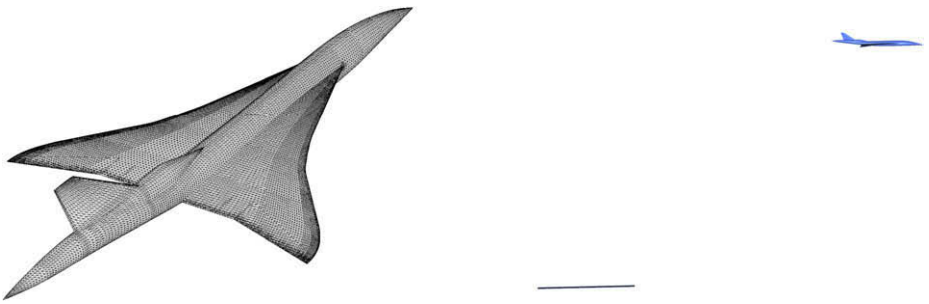


Fig. 10. Surface mesh of the geometry of the supersonic business jet SSBJ (left). Location of the observation plane below the aircraft (right).

width of 2 m whereas the wing span is about 17 m. The scope of this test case is to evaluate the ability of the adjoint to prescribe refinements only in areas that impact the observation region. The functional is given by:

$$j(W) = \frac{1}{2}\int_{\gamma}\left(\frac{p - p_\infty}{p_\infty}\right)^2 \mathrm{d}\gamma,$$

with $\gamma = \{(x,y,z) \in \mathbb{R}^3 \,|\, 100 \leqslant x \leqslant 140,\ -1 \leqslant y \leqslant 1,\ z = -100\}$. Observation area γ and its position with respect to the aircraft is shown in Fig. 10(right).

In order to exemplify how adjoint-based mesh adaptation gives an optimal distribution of the degrees of freedom to evaluate the functional, this adaptation is compared to a Hessian-based mesh adaptation. The adaptation is done on the local

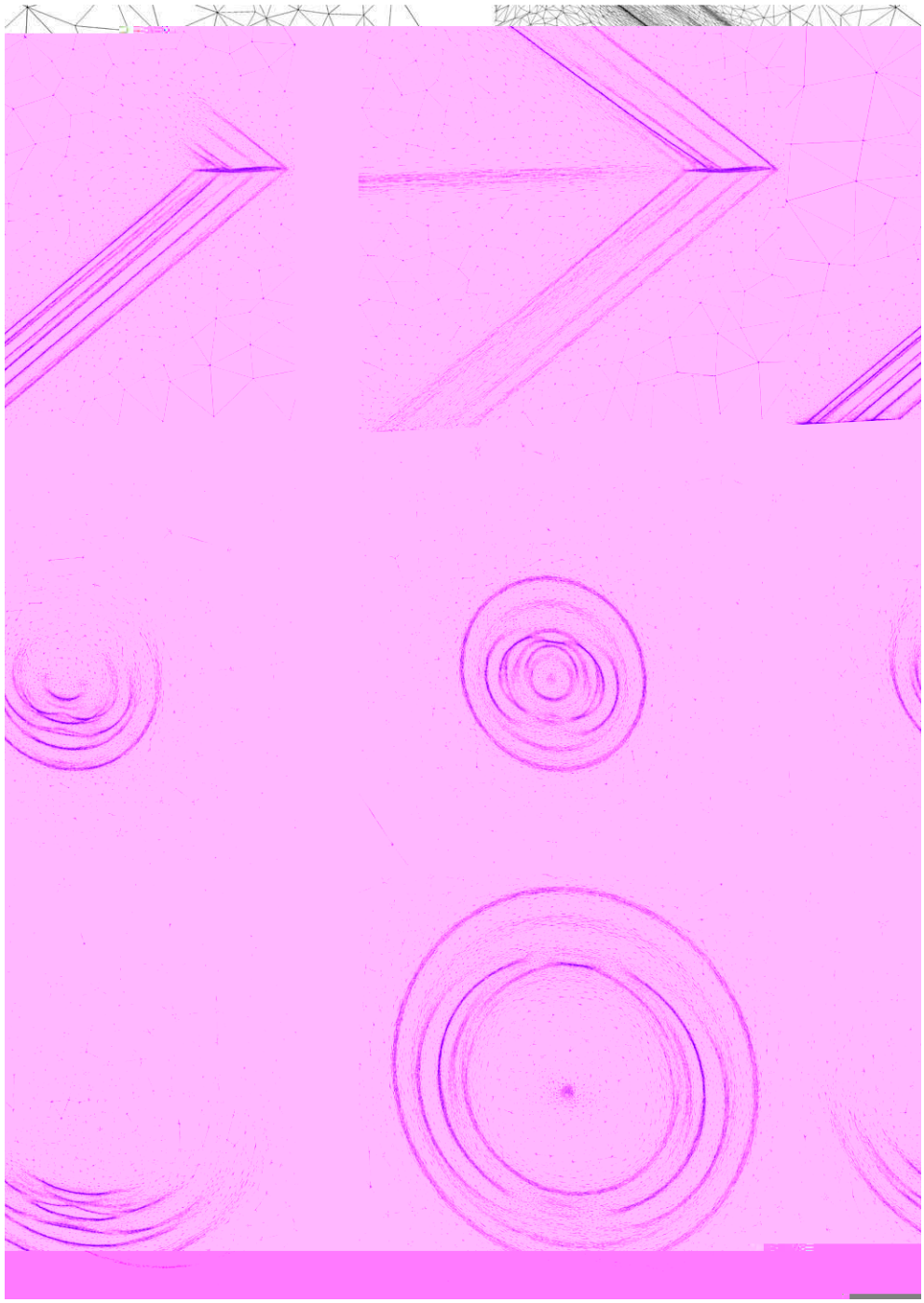


Fig. 12. Cut planes through the final adapted meshes for the adjoint-based (left) and Hessian-based (right) methods. Top, a cut in the symmetry plane and, middle and bottom, two cuts with an increasing distance behind the aircraft orthogonal to its path.

the adjoint-based mesh adaptation. It also demonstrates how the adjoint defines an optimal distribution of the degrees of freedom for the specific target. However, it is important to note that the mesh obtained with the Hessian-based strategy is optimal to evaluate globally the local Mach number.

It is worth mentioning that this method is completely automatic and gives an optimal result. It seems quite difficult to find a manual adaptation strategy to obtain an accurate evaluation of the functional while reducing the number of degrees of freedom. For instance, one may consider an approach that would consist in ignoring the upper part of the flow, however such

a mesh would not be optimal as there exists at each distance a specific angle for refinement depending on the width of the observation plane γ. Moreover, considering the amplitude on the physical phenomenon to deduce a size prescription is not optimal as it gives weights that are not optimal to each physical phenomena involved in the flow.

To conclude with this example, we compare the anisotropic quotients and the anisotropic ratios for the goal-oriented and the Hessian-based mesh adaptation. We aim at showing that performing the sum of interpolation errors involved in error Estimate (18) does not destroy the strength of the prescribed mesh anisotropy. Indeed, at first thought, one may think that, similarly to the metric intersection process, summing the interpolation errors on the fluxes may result in meshes with a poor anisotropy level. These quantities for the two final meshes with both approaches are reported in Tables 3 and 4. We note that there is no significant difference and no loss of anisotropy. This fact is confirmed by the mean values. An anisotropic mean ratio of 11.721 is obtained for the Hessian-based strategy versus 11.404 for the adjoint-based one. And, an anisotropic mean quotient of 109.74 is observed for the Hessian-based adaptation versus 117.24 for the adjoint-based method.

Table 3
SSBJ example: anisotropic ratio histograms for the SSBJ final adapted mesh with the adjoint-based and the Hessian-based methods. For each ratio interval, the number of tetrahedra is given with the corresponding percentage.

Anisotropic ratio			Adjoint-based		Hessian-based	
1	$<$ ratio $\leqslant$	2	87,152	1.81%	63,900	1.34%
2	$<$ ratio $\leqslant$	3	344,171	7.15%	254,689	5.33%
3	$<$ ratio $\leqslant$	4	408,150	8.48%	326,727	6.84%
4	$<$ ratio $\leqslant$	5	383,587	7.97%	333,693	6.99%
5	$<$ ratio $\leqslant$	10	1 417,279	29.43%	1,464,200	30.67%
10	$<$ ratio $\leqslant$	50	2 160,709	44.87%	2 318,963	48.57%
50	$<$ ratio $\leqslant$	100	14,589	0.30%	11,748	0.25%

Table 4
SSBJ example: anisotropic quotient histograms for the SSBJ final adapted mesh with the adjoint-based and the Hessian-based methods. For each quotient interval, the number of tetrahedra is given with the corresponding percentage.

Anisotropic ratio			Adjoint-based		Hessian-based	
1	$<$ quo $\leqslant$	2	20,670	0.43%	15,391	0.32%
2	$<$ quo $\leqslant$	3	98,030	2.04%	71,910	1.51%
3	$<$ quo $\leqslant$	4	135,076	2.80%	99,694	2.09%
4	$<$ quo $\leqslant$	5	140,389	2.92%	105,367	2.21%
5	$<$ quo $\leqslant$	10	570,124	11.84%	459,995	9.64%
10	$<$ quo $\leqslant$	50	1,635,197	33.96%	1,635,882	34.27%
50	$<$ quo $\leqslant$	100	731,548	15.19%	855,954	17.93%
100	$<$ quo $\leqslant$	1000	1,435,724	29.81%	1,502,571	31.47%
10^3	$<$ quo $\leqslant$	10^4	48,955	1.02%		
10^4	$<$ quo $\leqslant$	10^5	4	0.00%		
10^5	$<$ quo		1	0.00%		

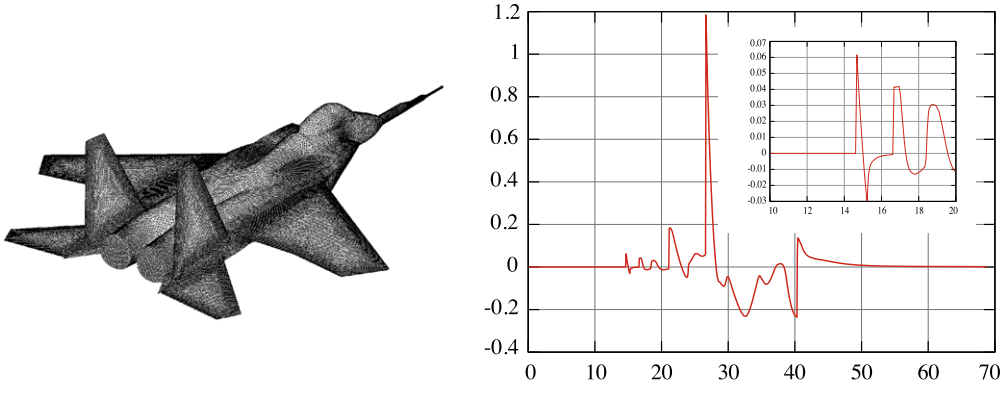


Fig. 13. F15 fighter geometry equipped with the Quiet Spike concept (left), pressure signature 2 m below the aircraft (right).

6.4. A challenging computation: studying a F15 fighter with the Quiet Spike

We consider in this example the accurate prediction of the mid-field pressure signature all around the F15 fighter equipped with the Quiet Spike concept [35] during a supersonic flight. The aircraft is flying at Mach 1.8 with an angle of attack of 0 degree. This complex geometry is shown in Fig. 13(left). This concept was devised to soften the sonic boom by splitting the initial strong bow shock in several shocks of smaller amplitude. The Quiet Spike is an evolution of the simple double cone geometry studied previously: it is composed of three cones linked by cylinders of increasing radius. The smaller cylinder has a radius of 5 cm while the greater one has a radius of 20 cm. These sizes must be compared to the aircraft length 19.3 m and wing-span 13 m. The scale variations of the geometry give a first idea of the complexity of this simulation.

The challenge in this computation is to capture automatically and accurately all the shock waves of small amplitude emitted by the spike. This is crucial as they modify the downstream flow. The disparity in physical scales is illustrated in Fig. 13(right) where the pressure signature 2 m below the aircraft along the aircraft path is drawn. The amplitude of the first spike shock is almost 20 times smaller than the main shock due to the engine inlet.

In the literature, this simulation is currently envisaged in a 2-stage process by coupling a structured solver with an unstructured one [35]. The unstructured solver is considered to predict the near field pressure signature (until 6 m) to take into account the geometry complexity and to capture all the 3D details. Since unstructured solvers are too dissipative, it is difficult to capture the pressure signal far from the aircraft with unstructured meshes. Consequently, the near field solution obtained on the unstructured mesh is used as an input for the structured flow solver that gives an accurate pressure field far below the aircraft (at 70 m). We propose to show that the present fully anisotropic adjoint-based strategy automatically provides an accurate signature 100 m all around the aircraft while using only unstructured meshes.

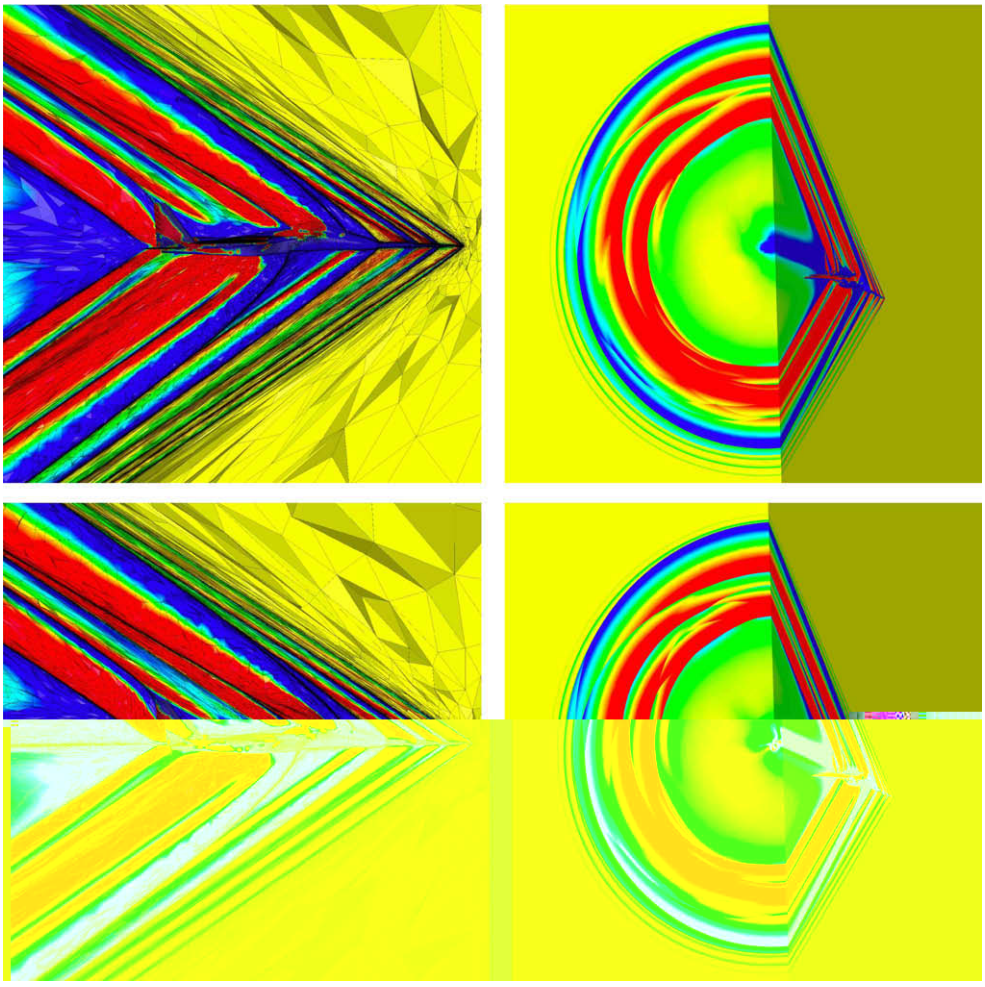


Fig. 14. Comparison between the meshes and solutions obtained with the goal-oriented (top) and the hessian-based (bottom) strategies. Left, final mesh and solution near the aircraft in the symmetry plane $y = 0$ m. Right, Mach number iso-values plotted in the plane $y = 0$ m and in the plane $x = 50$ m.

Once more, the Hessian-based mesh adaptation on the local Mach number is compared to adjoint-based method where the pressure field is observed 100 m all around the aircraft. The computational domain is a cylinder of radius 150 m having a length of 400 m. The final Hessian-based adapted meshes is composed of 10 050 445 vertices and 60 280 606 tetrahedra, and the final adjoint-based adapted mesh of 4 505 060 vertices and 26 864 368 tetrahedra. These anisotropic meshes and the associated solutions are shown in Fig. 14(left) through the cut plane $y = 0$. The goal-oriented method captures accurately the spike shocks independently of their amplitudes. Indeed, Fig. 14, three shocks emitted by the Quiet Spike are distinguishable inside the mesh and in the solution whereas only one shock clearly appears with the Hessian-based method. This fact is emphasized with the Mach number iso-values in Fig. 14(right) or with pressure signatures in Fig. 15. The zoom on the front of the pressure signal shows that the stiffness of the two first shock is lost after 50 m of propagation with the Hessian-based method, Fig. 15(top right). And, it is even worse at 100 m below the aircraft, Fig. 15(bottom right). In this case, an anisotropic adapted mesh with 10 million vertices is not sufficient to accurately predict the pressure signature. In comparison, the adjoint-based adaptation compute accurately these shocks without any dissipation, and this, with an adapted mesh of 4 million vertices. Tables 5 and 6 gives anisotropic ratios and quotients for the adjoint-based and Hessian-based simulations. In this example, the anisotropy level reached is even better for the adjoint-based case than for the Hessian-based case.

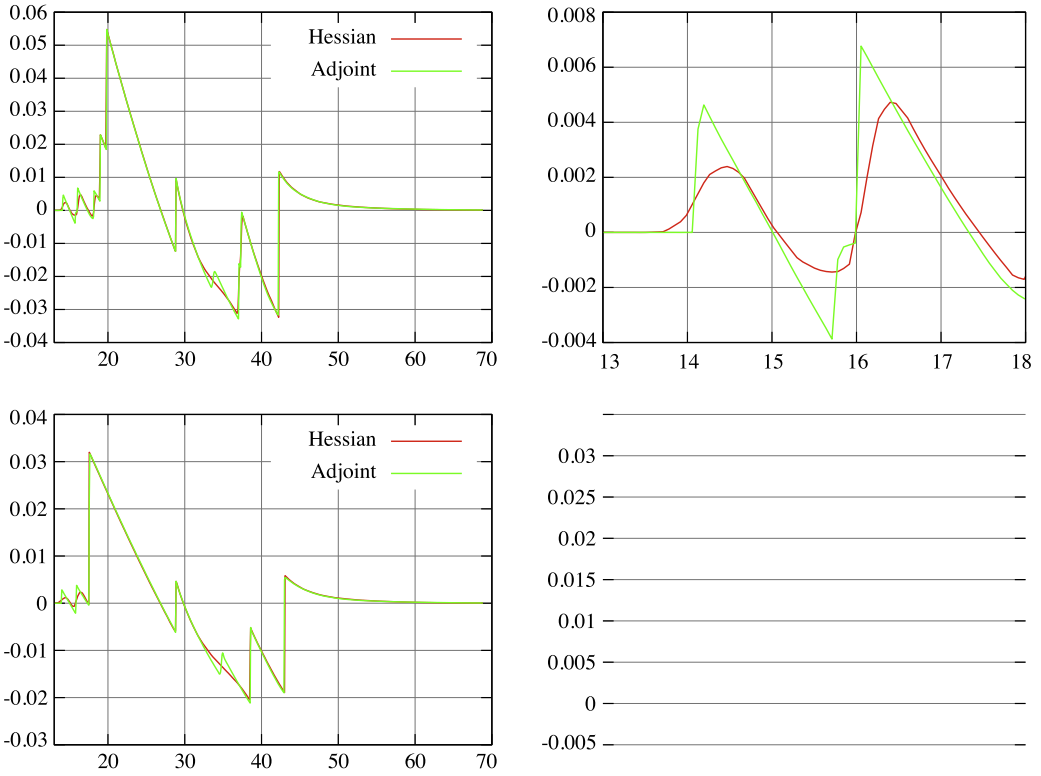


Table 6
F15 example: anisotropic quotient histograms for the F15 final adapted mesh with the adjoint-based and the Hessian-based methods. For each quotient interval, the number of tetrahedra is given with the corresponding percentage.

Anisotropic quotient			Adjoint-based		Hessian-based	
1	$\leqslant$ quo $\leqslant$	2	12,459	0.04%	13,096	0.02%
2	$\leqslant$ quo $\leqslant$	3	66,635	0.20%	64,548	0.11%
3	$\leqslant$ quo $\leqslant$	4	105,626	0.31%	102,693	0.17%
4	$\leqslant$ quo $\leqslant$	5	119,120	0.35%	118,128	0.20%
5	$\leqslant$ quo $\leqslant$	10	545,105	1.62%	598,219	0.99%
10	$\leqslant$ quo $\leqslant$	50	2,190,527	6.50%	3,176,207	5.27%
50	$\leqslant$ quo $\leqslant$	10^2	1,420,864	4.22%	2,676,990	4.44%
10^2	$\leqslant$ quo $\leqslant$	10^3	7,092,096	21.04%	17,153,343	28.46%
10^3	$\leqslant$ quo $\leqslant$	10^4	10,660,556	31.63%	26,329,992	43.68%
10^4	$\leqslant$ quo $\leqslant$	10^5	9,262,559	27.48%	9,808,547	16.27%
10^6	$\leqslant$ quo		2,230,723	6.62%	238,843	0.40%

This example demonstrates the advantage of this full coupling between the adjoint-based error estimate and the anisotropic mesh adaptation. Even if we choose to adapt everywhere in the domain, we obtain a more accurate solution as compared to the hessian-based method with two times less vertices. Indeed, the adjoint-based estimate takes into account the spike shocks thanks to adequate weights and the use of anisotropic adapted meshes reduce drastically the numerical dissipation when propagating the shocks in the domain. The CPU time for the whole simulation is one week on 8 Intel Core 2 processors at 2.8 GHz.

6.5. Vortical wake emission of a Falcon business jet

In this example, we study the accurate prediction of wing tip vortices at large distance in the wake for transsonic flow conditions. We consider the Falcon business jet geometry, see Fig. 16. The jet is flying at transonic cruise speed with Mach number 0.8 and an angle of attack of 3 degrees. The computational domain is a cylinder of radius 250 m and of length 700 m. The Hessian-based adaptation on the local Mach number is compared to the adjoint-based adaptation on the vorticity functional:

$$j(W) = \frac{1}{2}\int_{\gamma} \|\nabla \times (\mathbf{u} - \mathbf{u}_\infty)\|_2^2 \, \mathrm{d}\gamma,$$

where γ is a plane located 400 m behind the aircraft orthogonal to the aircraft path, $\mathbf{u}$ the velocity field and $\mathbf{u}_\infty$ the far field velocity.

As the aircraft is flying at a transonic speed, the flows is composed of both shocks and smooth vortices. These phenomena have different magnitudes and mathematical properties. Across a shock, most variables become discontinuous whereas a vortex corresponds to a smooth variation of the variables while having a very small amplitude. These features are exemplified in Fig. 16(right). An extraction of the pressure across the wing extrados where a shock occurs (green curve)

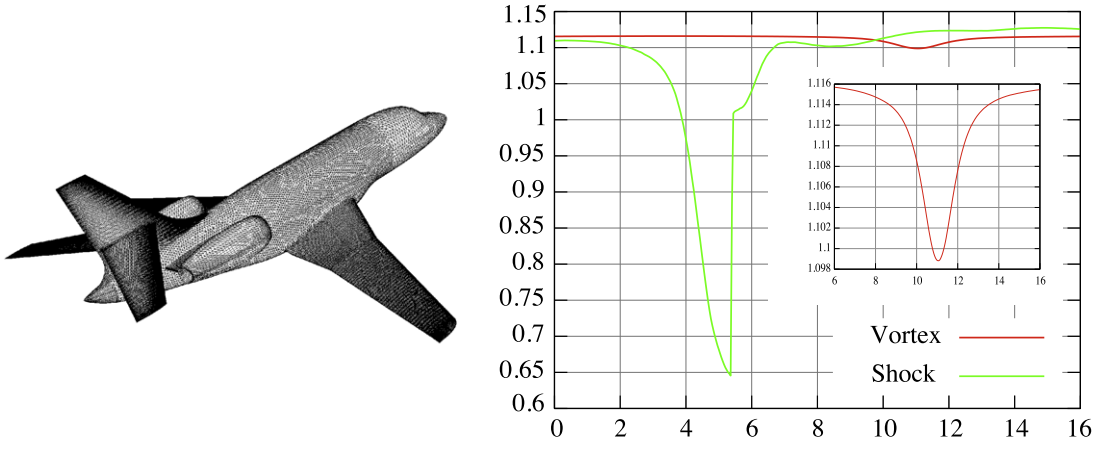


Fig. 16. Left, surface mesh of the Falcon business jet geometry. Right, pressure variation through a section across the wing (green curve) and across the wake 400 m behind the aircraft (red curve). For the green curve, the jump represents the shock on the wing extrados. For the red curve, the small gaussian represents the vortex amplitude. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

is superposed to the pressure variation in the wake across a vortex located 400 m behind the aircraft (red curve). The amplitude of the vortex is less than 2% of the amplitude of the shock. Moreover, the smoothness property of the vortex is a supplementary difficulty as its derivatives involved in our estimate are also smooth. Consequently, vortices are difficult to detect and to not diffuse. Detecting and preserving these vortices are still a challenge in the field of CFD.

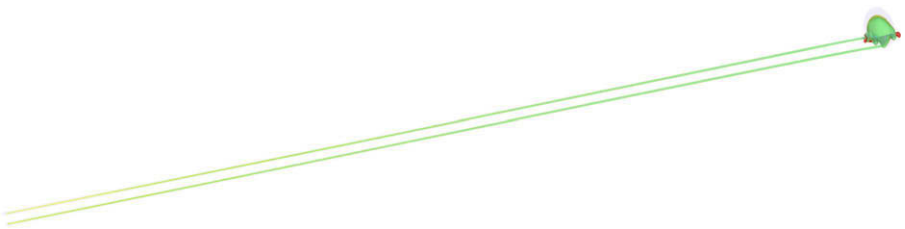


Fig. 17. Pressure iso-surfaces along a 400 m path behind the Falcon obtained with the adjoint-based mesh adaptation.

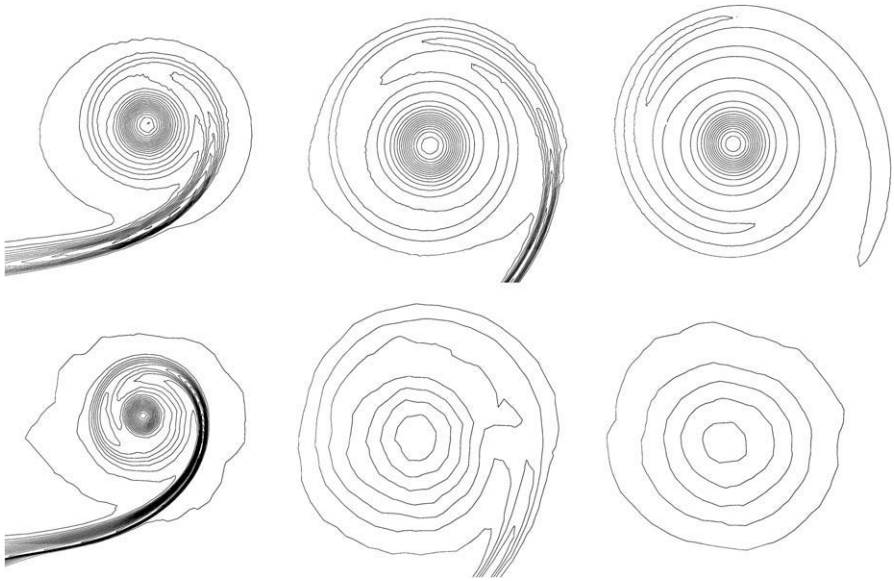


Fig. 18. Comparison between the solutions obtained with the goal-oriented (top) and the hessian-based (bottom) strategies in planes 100 m, 200 m and 400 m (from left to right) behind the Falcon and orthogonal to the aircraft path.

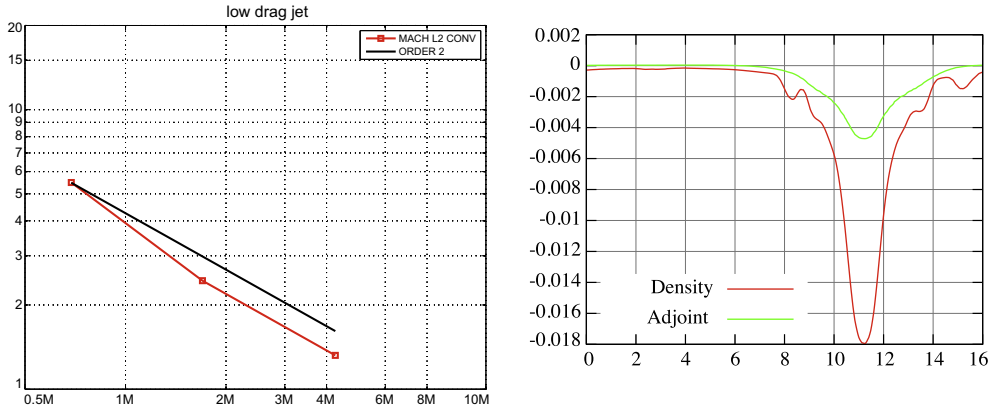


Fig. 19. Left, mesh convergence order estimate on the SSBJ case obtained with Hessian-based mesh adaptation. Right, Adjoint state of the density and density along a cut line through a vortex in the wake of the Falcon.

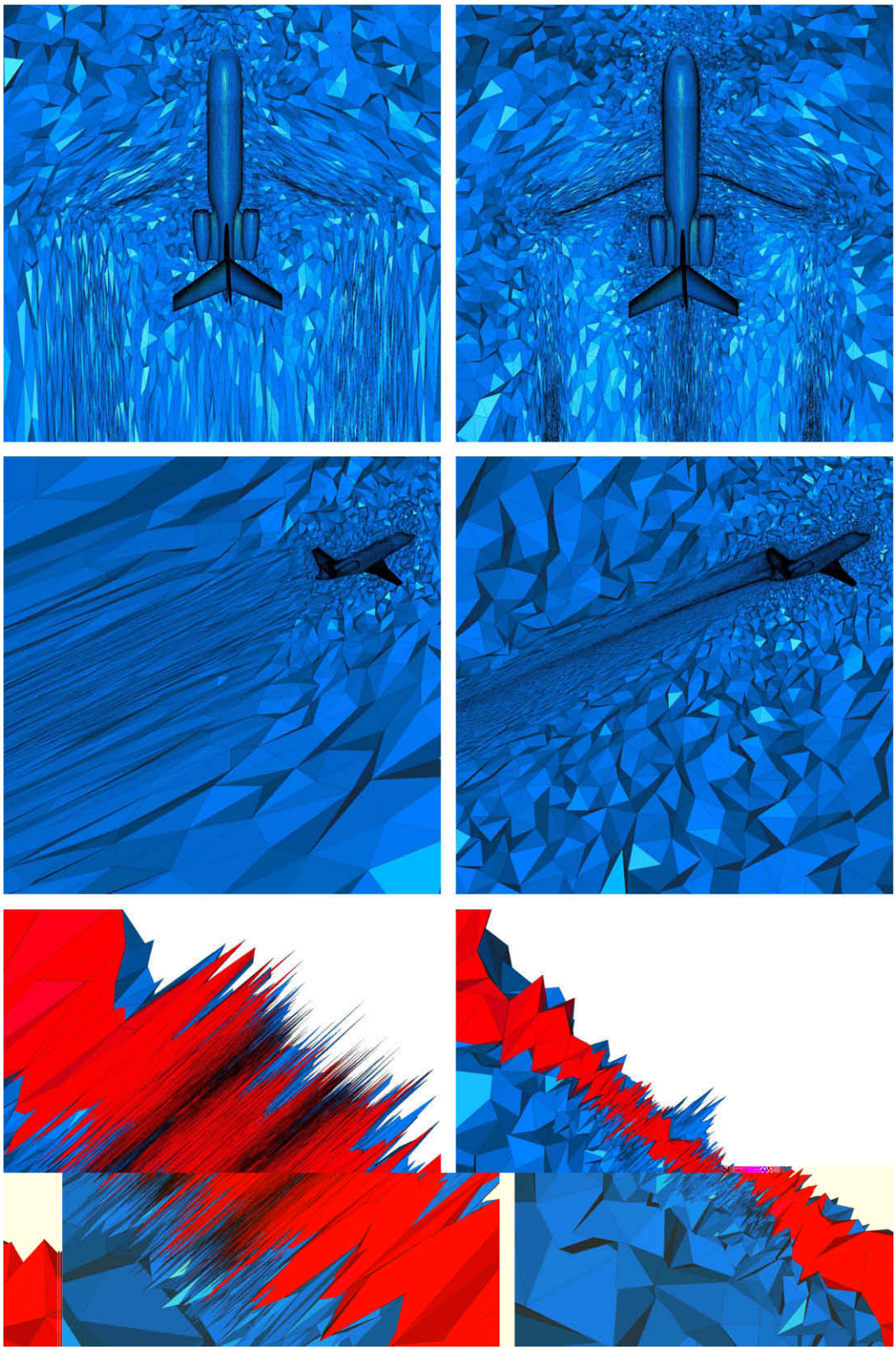


Fig. 20. Comparison between the adapted meshes obtained with the goal-oriented (left) and the hessian-based (right) strategies. Cuts through the volume along the planes $z = 0$ m (top), $y = 0$ m (middle) and $x = 400$ m (bottom).

Final adapted meshes are composed of almost 1.5 million vertices for both methods. Vorticity iso-values are visualized in Fig. 18 for planes $x = 100$ m,$x = 200$ m and $x = 400$ m, that are located behind the aircraft and orthogonal to the aircraft path. As regards Hessian-based adaptation, the vortex is accurately captures up to 100 m behind the aircraft and then it is diffused with the distance to the Falcon. On the contrary, with adjoint-based method, the vortex keeps a constant size and its core is not diffused when increasing the distance to the aircraft, see Fig. 17. When looking at the cuts in $x = cte$ planes, the meshes are almost isotropic. Indeed, as the vortex iso-values are circular the ideal mesh is isotropic. In fact, the anisotropic gains are along the x-axis as illustrated in Fig. 20. We also observe that the wing shock is merely refined in the adjoint-based mesh contrary to the Hessian-based mesh. In the observation plane 400 m behind the aircraft, the adjoint-mesh is strongly anisotropic whereas Hessian-based mesh has already lost the vortex and is poorly anisotropic, see Fig. 20(bottom). A similar conclusion as previous example also applied here: the utilization of anisotropic meshes prevents the numerical diffusion of the physical phenomena while an appropriate weighting of the physical variables captures the small scale phenomena.

6.6. About mesh convergence

Mesh convergence is a delicate issue when many small discontinuous structures are present in the flow field. The theoretical convergence order is generally not so easy to obtain even with smooth flows. In our case, this theoretical order is 2 when dealing with classical Hessian-based mesh adaptation in $\mathbf{L}^p$ norm. We concentrate here on the more difficult case of non-regular shocked flows [43]. In Fig. 19 mesh convergence curve for the SSBJ case of Section 6.2 is depicted. For this geometry, we obtain numerical convergence order of about 2 for meshes varying from 1.5 to 4.5 million vertices. On the contrary, the prediction of convergence order on functional outputs deserves a complete theory as the kernel given by Relation (17) may be composed of both regular and none regular terms. Indeed, it involves both the classical interpolation error term weighted by the gradient of the adjoint state. To give an idea of the theoretical complexity, we refer to the 1D studies on the adjoint regularity [30]. Preliminary convergence estimates in 2D are available in [41]. They show that the order depends highly on the observed functional contrary to the Hessian-based where even with flows with shocks a near optimal convergence order can be reached, see examples in [4].

7. Conclusion

We have proposed a new method providing the anisotropic adapted mesh optimizing the first error term in the approximation of a functional depending on the solution of a flow problem. This method is based on a new formal *a priori* estimation of the functional approximation error and its resolution in an abstract continuous framework. It has been applied successfully to the compressible Euler equations. This new method exploits two advanced technology and their good synergy:

- up-to-date anisotropic mesh generators that contribute to build optimal anisotropic adapted meshes
- the flow solver which is basically a P^1 Galerkin one relying on a continuous piecewise linear representation of the solution. It satisfies P^1-exactness property allowing a P^1-interpolation based error estimate to be used. Moreover, it is stabilized with a modern shock capturing method enabling the accurate computation of thin numerical shocks in strongly anisotropic adapted meshes.

In addition to be fully automatic, the method has the following features:

- it produces an optimal anisotropic metric uniquely specified as the optimum of a functional and explicitly given by variational calculus from the continuous state and the adjoint state. The coupled system of the metric and of the two states is the object of the discretisation. This should be put in contrast with the usual process of starting from a (discrete) mesh and then improving it
- to apply it, there is no need to choose in a more or less arbitrary way any local refinement “criterion” and no need to fix any parameter except the total number of vertices which represents the error threshold
- mesh convergence is performed in a natural way by increasing the total number of prescribed vertices at each stage of the mesh adaptation process.

The new method has been applied to several challenging 3D problems. Numerical experiments show that the new method enjoys at best level the advantages of Hessian-based anisotropic methods and of goal-oriented methods. As compared to the Hessian-based method, the anisotropic stretching of the meshes is not lost but even more strengthened and better distributed along shocks. As compared with goal-oriented methods, the new method behaves like a goal-oriented method, but also naturally takes the anisotropy related to functional into account.

Some issues have not been addressed in this work such as the asymptotic convergence order of the error on the functional. The authors are addressing some of them, together with the issues of extending the above method to viscous and unsteady flows.

References

[1] P.-A. Absil, R. Mahony, R. Sepulchre, Optimization Algorithms on Matrix Manifolds, Princeton University Press, Princeton, NJ, 2008.
[2] F. Alauzet, Size gradation control of anisotropic meshes, Finite Elem. Anal. Des. 46 (2010) 181–202.
[3] F. Alauzet, P .J. Frey, P.-L. George, B. Mohammadi, 3D transient fixed point mesh adaptation for time-dependent problems: application to CFD simulations, J. Comput. Phys. 222 (2007) 592–623.
[4] F. Alauzet, A. Loseille, High order sonic boom modeling by adaptive methods, J. Comput. Phys. 229 (3) (2010) 561–593.
[5] O. Allain, D. Guégan, F. Alauzet, Studying the impact of unstructured mesh adaptation on free surface flow simulations, in: Proceedings of the ASME 28th International Conference on Ocean, Offshore and Arctic Engineering, 2009.
[6] T. Apel, Anisotropic Finite Elements: Local Estimates and Applications, Book Series: Advances in Numerical Mathematics, Teubner, Stuttgart, 1999.
[7] V. Arsigny, P. Fillard, X. Pennec, N. Ayache, Log-Euclidean metrics for fast and simple calculus on diffusion tensors, Magn. Reson. Med. 56 (2) (2006) 411–421. August.
[8] I. Babuška, T. Strouboulis, The Finite Element Method and its Reliability, Oxford Scientific Publications, New York, 2001.
[9] R. Becker, R. Rannacher, A feed-back approach to error control in finite element methods: basic analysis and examples, East-West J. Numer. Math. 4 (1996) 237–264.
[10] M. Berger, A Panoramic View of Riemannian Geometry, Springer Verlag, Berlin, 2003.
[11] C.L. Bottasso, Anisotropic mesh adaption by metric-driven optimization, Int. J. Numer. Meth. Eng. 60 (2004) 597–639.
[12] M.J. Castro-Díaz, F. Hecht, B. Mohammadi, O. Pironneau, Anisotropic unstructured mesh adaptation for flow simulations, Int. J. Numer. Meth. Fluids 25 (1997) 475–491.
[13] T. Coupez, Génération de maillages et adaptation de maillage par optimisation locale, Revue Européenne des Éléments Finis 9 (2000) 403–423.
[14] P.-H. Cournède, B. Koobus, A. Dervieux, Positivity statements for a mixed-element-volume scheme on fixed and moving grids, Eur. J. Comput. Mech. 15 (7-8) (2006) 767–798.
[15] A. Dervieux, D. Leservoisier, P.-L. George, Y. Coudiere, About theoretical and practical impact of mesh adaptations on approximation of functions and of solution of PDE, Int. J. Numer. Meth. Fluids 43 (2003) 507–516.
[16] A. Dervieux, A. Loseille, F. Alauzet, High-order adaptive method applied to high-speed flows, in: West-East High Speed Flow Field Conference, November 19–22, Moscou, Russia, 2007.
[17] C. Dobrzynski, P.J. Frey, Anisotropic delaunay mesh adaptation for unsteady simulations, in: Proceedings of the 17th International Meshing Rountable, Springer, 2008, pp. 177–194.
[18] J. Dompierre, M.G. Vallet, M. Fortin, Y. Bourgault, W.G. Habashi, Anisotropic Mesh Adaptation: Towards a Solver and User Independent cfd, AIAA Paper 97-0861, 1997.
[19] L. Formaggia, S. Micheletti, S. Perotto, Anisotropic mesh adaptation in computational fluid dynamics: application to the advection-diffusion-reaction and the Stokes problems, Appl. Numer. Math. 51 (2004) 511–533.
[20] L. Formaggia, S. Perotto, New anisotropic a prioiri error estimate, Numer. Math 89 (2001) 641–667.
[21] M. Fortin, M.-G. Vallet, J. Dompierre, Y. Bourgault, and W.G. Habashi. Anisotropic mesh adaptation : theory, validation and applications, in: Proceedings of the ECCOMAS CFD, 1996.
[22] P.J. Frey. `Yams`, A Fully Automatic Adaptive Isotropic Surface Remeshing Procedure, RT-0252, INRIA, 2001.
[23] P.J. Frey, F. Alauzet, Anisotropic mesh adaptation for CFD computations, Comput. Meth. Appl. Mech. Eng. 194 (48–49) (2005) 5068–5082.
[24] P.J. Frey, H. Borouchaki, Surface meshing using a geometric error estimate, Int. J. Numer. Methods Eng. 58 (2) (2003) 227–245.
[25] P.J. Frey, P.-L. George, Mesh Generation, Application to Finite Elements, Hermès Science, Paris, Oxford, 2000.
[26] P.-L. George, Tet meshing: construction, optimization and adaptation, in: Proceedings of the 8th Meshing Rountable, South Lake Tao, CA, USA, 1999.
[27] P.-L. George, H. Borouchaki, Delaunay Triangulation and Meshing: Application to Finite Elements, Hermès Science, Paris, Oxford, 1998.
[28] M.B. Giles, On Adjoint Equations for Error Analysis and Optimal Grid Adaptation in CFD, Technical Report NA-97/11, Oxford, 1997.
[29] M.B. Giles, N. Pierce, Improved Lift and Drag Estimates Using Adjoint Euler Equations, AIAA Paper, 99-3293, 1999.
[30] M.B. Giles, N.A. Pierce, Analytic adjoint solutions for the quasi-one-dimensional euler equations, J. Fluids Mech. 426 (2001) 327–345.
[31] M.B. Giles, A. Suli, Adjoint methods for PDEs: a posteriori error analysis and postprocessing by duality, Acta Numer. (2002) 145–236.
[32] A. Griewank, On Automatic Differentiation, in: M. Iri, K. Tanabe (Eds.), Mathematical Programming: Recent Developments and Applications, Kluwer Academic Publishers, 1989, pp. 83–108.
[33] H.W. Carlson, R.J. Mack, O.A. Morris, A Wind-tunnel Investigation of the Effect of Body-shape on Sonic-boom Pressure Distributions, TN. D-3106, Nasa, 1965.
[34] L. Hascoët, V. Pascual, TAPENADE 2.1 User's Guide, Rapport technique 300, NRIA, Sophia Antipolis, 2004.
[35] D.C. Howe, K.A. Waithe, E.A. Haering, Quiet Spike Near Field Flight Test Pressure Measurement with Computational Fluid Dynamics Comparisons, AIAA Paper, 2008-128, 2008.
[36] L.W. Hunton, R.M. Hicks, J.P. Mendoza, Some Effects of Wing Planform on Sonic Boom, TN. D-7160, Nasa, 1973.
[37] W.T. Jones, E.J. Nielsen, M.A. Park, Validation of 3D Adjoint Based Error Estimation and Mesh Adaptation for Sonic Boom Prediction, AIAA Paper, 2006-1150, 2006.
[38] J.-F. Lagüe, F. Hecht, Optimal mesh for P_1 interpolation in H^1 semi-norm, in: Proceedings of 15th Meshing Rountable, Springer, 2006, pp. 259–270.
[39] X.L. Li, M.S. Shephard, M.W. Beall, 3D anisotropic mesh adaptation by mesh modification, Comput. Meth. Appl. Mech. Eng. 194 (48–49) (2005) 4915–4950.
[40] R. Löhner, Applied CFD Techniques, Wiley, New-York, 2001.
[41] A. Loseille, Adaptation de maillage 3D anisotrope multi-échelles et ciblé à une fonctionnelle. Application à la prédiction haute-fidélité du bang sonique. PhD thesis, Université Pierre et Marie Curie, Paris VI, Paris, France, 2008. <http://tel.archives-ouvertes.fr/tel-00361961/fr/>.
[42] A. Loseille, F. Alauzet. Continuous Mesh Model and Well-posed Continuous Interpolation Error Estimation, RR-6846, INRIA, 2009. <http://hal.inria.fr/inria-00370235/fr/>.
[43] A. Loseille, F. Alauzet, A. Dervieux, P.J. Frey, Achievement of Second Order Mesh Convergence for Discontinuous Flows with Adapted Unstructured Mesh Adaptation, AIAA Paper, 07-4186, 2007.
[44] K. Mer, Variational analysis of a mixed element/volume scheme with fourth-order viscosity on general triangulations, Comput. Meth. Appl. Mech. Eng. 153 (1998) 45–62.
[45] C. Pain, A.P. Umpleby, C.R.E. de Oliveira, A.J.H. Goddard, Tetrahedral mesh optimisation and adaptivity for steady-state and transient finite element calculations, Comput. Meth. Appl. Mech. Eng. 190 (2001) 3771–3796.
[46] M. Picasso, An anisotropic error indicator based on Zienkiewicz–Zhu error estimator: application to elliptic and parabolic problems, SIAM J. Sci. Comput. 24 (4) (2003) 1328–1355.
[47] N.A. Pierce, M.B. Giles, Adjoint recovery of superconvergent functionals from PDE approximations, SIAM Rev. 42 (2) (2000) 247–264.
[48] G. Rogé, L. Martin, Goal-oriented anisotropic grid adaptation de maillage anisotrope orientTe objectif, Comptes Rendus MathTmatique 346 (19–20) (2008) 1109–1112.
[49] Y. Saad, Iterative Methods for Sparse Linear Systems, second ed., SIAM, Philadelpha, PA, 2003.
[50] E. Schall, D. Leservoisier, A. Dervieux, B. Koobus, Mesh adaptation as a tool for certified computational aerodynamics, Int. J. Numer. Meth. Fluids 45 (2004) 179–196.
[51] A. Tam, D. Ait-Ali-Yahia, M.P. Robichaud, M. Moore, V. Kozel, W.G. Habashi, Anisotropic mesh adaptation for 3D flows on structured and unstructured grids, Comput. Meth. Appl. Mech. Eng. 189 (2000) 1205–1230.

[52] D.A. Venditti, D.L. Darmofal, Grid adaptation for functional outputs: application to two-dimensional inviscid flows, J. Comput. Phys. 176 (1) (2002) 40–69.
[53] D.A. Venditti, D.L. Darmofal, Anisotropic grid adaptation for functional outputs: application to two-dimensional viscous flows, J. Comput. Phys. 187 (1) (2003) 22–46.
[54] R. Verfürth, A review of A Posteriori Error Estimation and Adaptative Mesh-Refinement techniques, Wiley Teubner Mathematics, New York, 1996.
[55] M. Wintzer, M. Nemec, M.J. Aftosmis, Adjoint-based Adaptive Mesh Refinement for Sonic Boom Prediction, AIAA Paper, 2008-6593, 2008.