

# A fixed-point mesh-adaptive shape design algorithm applied to sonic boom reduction

Youssef Mesri\*, Frédéric Alauzet \*\* and Alain Dervieux \*

(\*) INRIA, Smash project, BP 93, 06902 Sophia-Antipolis Cedex, France.

(\*\*) INRIA, Gamma project, Domaine de Voluceau Rocquencourt, BP 105, 78153 Le Chesnay Cedex, France

November 6, 2008

## Abstract

This paper presents a combination of mesh adaptation and shape design optimization. The optimization loop is based on an Euler model and an adjoint-based gradient descent algorithm. Mesh adaptation provides here, a control of accuracy of the numerical solution by modifying the domain discretization according to size and stretching directions. A novel algorithm is designed for coupling mesh adaptation and shape optimization. Application of this algorithm to sonic boom mitigation is presented and compared to a more natural approach.

**Keywords:** Computational Fluid Dynamics, Mesh adaptation, shape design.

## 1 Introduction

Shape Design based on Optimal Control and adjoint state is becoming a frequent practice in industry (see [12, 8, 13] and [6]). Using it assumes some confidence in the High-Fidelity simulation tool involved in the optimization platform. In CFD, this confidence relies on the increasing robustness and accuracy of CFD solvers. However in some particular cases, the accuracy strongly depends on the ability of the solver to capture small scales. Because of these, second-order convergence to exact solution is difficult and anisotropic mesh adaptation is understood as a mean for obtaining a higher order of convergence.

The particular mesh adaptative method which we want to use is described in details in [1]. It is generally referred as a Hessian-based anisotropic unstructured mesh adaptation and applied a fixed point iteration in order to converge the coupling between approximate solution and its adapted mesh. The fast and reliable mesh reconstruction involved in this algorithm does not introduce an important computer cost, but, at the contrary, contributes to a strong acceleration to attaining an adapted solution, enjoying the benefit of mesh refinement inside iterative flow convergence. The mesh adaptation algorithm chosen here has been observed as producing second-order convergence to exact solutions for a rather small number of discrete unknowns.

Mesh-adaptation for Optimal Control is a topic addressed by several authors, in particular for the choice of error estimators. Indeed, when the optimization problem is solved up to the complete satisfaction of the optimality system, it is natural to consider the optimum parameter, here the shape, as the target unknown and to build an adaptation strategy dedicated to get the lowest error on this unknown. See for example [5]. In some other cases, the optimum system is not enough solved and a strategy looking for a good evaluation of the functional is reasonably preferred. This is the option considered here.

A typical example of phenomenon which is better solved with mesh adaptation is the sonic boom emission. Sonic boom basic modelisation is nothing else but the compressible Navier-Stokes equations. Unfortunately, for sonic boom this system would need to be solved in such level of accuracy that shock structure be accurately described and propagated from the aircraft to the ground at a distance of more than 15,000 meters. This, in practice, is out of reach with today's computers. Simplified models are generally applied, or, for a higher accuracy, a near field Euler or Navier-Stokes High Fidelity model is coupled with a simplified, and much less computer demanding, "boom propagation" model.

Shape design in sonic boom mitigation is the next step. In case of an evaluation of sonic boom using mesh adaptation, the design will need combining shape and mesh deformation, see [17][4]. In the HISAC project, see for example [16], several strategies have been investigated. Among them, mesh adaptation has been introduced in design in order to accurately evaluate the near field production and propagation of the complex shock structure producing the boom. This idea is not new, see for example [3].

When applied to sonic boom applications, the ratio of error between an poorly adapted mesh and the mesh adaptation produced by this method can be several order of magnitude. As a consequence, starting from the couple formed by an adapted mesh and the corresponding numerical flow solution, applying a slight variation of the design parameter (here the shape) will change the flow in such a way that the former adapted mesh cannot be used for obtaining an accurate evaluation of the new flow. This can be explained as follows: the shocks of new flow are outside previous mesh refinements, and error on boom evaluation functional is larger than the effect of shape perturbation. This point makes delicate the use of the mesh-adaptive solver inside an optimization loop.

In this paper, we consider the research by a descent method of an optimum in combination with this mesh adaptation algorithm. In other words, we want to reach by a progressive descent a shape that is optimal when the objective function is evaluated on a mesh that is strongly adapted to the optimal flow. We emphasize that the final mesh is not known in advance, but, instead, built at the same time we optimize. Further, since the adaptation mechanism involves topology changes (face swapping for example), we cannot define a fixed discrete optimization problem. Instead, we propose to approximatively solve the continuous optimality condition with a mesh adaptive method involving a descent step on a frozen mesh that is adapted to the whole descent..

Our standpoint is inspired by the unsteady fixed point adaptation algorithm studied in [2]. This latter algorithm determines a mesh adapted to the whole set of transient states of an adaptation sub-interval  $[T_i, T_{i+1}]$  of time interval  $[T_0, T_n]$ . We advance in time by subcycling with a time step  $\Delta t$  much smaller than the adaptation sub-interval  $T_{i+1} - T_i$ . Inside  $[T_i, T_{i+1}]$ , the mesh is implicitly adapted with the whole set of instantaneous solutions. For this purpose, a fixed point iteration is applied several times over the time levels of sub-interval  $[T_i, T_{i+1}]$  until convergence.

Let us examine the extension of this idea to a gradient loop. A gradient loop consists of to embedded loops. In an external loop, a state, an adjoint and a gradient are evaluated. Then starts a research of a better control in the direction of gradient. In a genuine descent algorithm (and also in more modern Sequential Quadratic Programming), a central condition is the verification that to the better control corresponds a smaller functional. This holds only if mesh is adapted to the different context of the descent step. What we propose in this paper is a fixed point iteration applied over the steps of each directional descent phase inside a gradient loop. After defining the new composite adaptation/optimization algorithm, we verify by a few numerical optimizations that the proposed concept is indeed useful to solve the addressed issue.

The plan of the paper is as follows. In next section, we formulate the minimization problem to solve. Section 3 presents the two numerical techniques to couple, viz. optimization and mesh adaptation, and propose a way to couple them. Section 4 presents the numerical application.

## 2 Optimization/Adaptation model

### 2.1 Continuous model

Given a set of admissible shapes  $\Gamma_{ad}$ , a continuous optimal shape design problem writes:

$$\min_{\gamma \in \Gamma_{ad}} j(\gamma) \tag{1}$$

where  $j(\gamma) = J(\gamma, W(\gamma))$  and  $W(\gamma)$  is the solution of a state equation, a PDE posed on a domain  $\Omega_\gamma$  with a shape parametrized by the control variable  $\gamma$ :

$$\Psi(\gamma, W(\gamma)) = 0 . \tag{2}$$

The solution  $W(\gamma)$  of (2) is computed through an approximation  $\bar{W}(\mathcal{M}, \gamma)$  with some error depending of a field  $\mathcal{M}$  defined on  $\Omega_\gamma$ :

$$\mathcal{E}(\gamma, \mathcal{M}) = \|\bar{W}(\mathcal{M}, \gamma) - W(\gamma)\| \tag{3}$$

Then we minimize the approximate functional:

$$\bar{j}(\mathcal{M}, \gamma) = J(\gamma, \bar{W}(\mathcal{M}, \gamma)) \tag{4}$$

under conditions on approximation error. We would prefer to choose once for all a particular  $\mathcal{M}$  such that  $\mathcal{E}(\mathcal{M}) = 0$  to avoid the approximation error. But this choice is not possible, because it would take an infinite time on a computer. We consider having a maximum CPU effort, measured by the fact that the complexity of the approximation  $c(\mathcal{M})$  (typically the number of mesh nodes) is specified to a fixed number  $N$ . Then two options can be considered. A

natural option ([5]) is to look for the couple  $(\mathcal{M}^+, \gamma^+)$  which offers the best approximation of the continuous optimal shape:

$$(\mathcal{M}^+, \gamma^+) \text{ such that } |\gamma^+ - \gamma_{opt}| = Min .$$

In the industrial practice, the option minimizing the error in flow variable evaluation can be preferred. We express it in terms of the error estimate (3):

$$\begin{aligned} (\mathcal{M}^*, \gamma^*) \text{ such that} \\ \mathcal{M}^* &= ArgMin \mathcal{E}(\gamma^*, \mathcal{M}) \\ \bar{j}(\mathcal{M}, \gamma^*) &\leq \bar{j}(\mathcal{M}, \gamma) . \end{aligned}$$

Where the *ArgMin* is taken for a complexity  $N$ .

The research of a minimum of  $j$  will be based on a descent algorithm. Descent algorithms may take the historical form of the steepest gradient or the form of Sequential Quadratic Programming (SQP). In both cases, a first part of the algorithm is devoted to build a correction, and a second part is devoted to adapt the correction to make it match a simplified quadratic model. A central condition for the success of second part is that we have a reliable descent direction. The algorithm discussed in this paper is designed in order to satisfy this condition by using an exact gradient approach.

## 2.2 Shape design numerical model

The state equation under study is the 3D steady Euler system, expressed in terms of conservative variables  $W = (\rho, \rho \mathbf{U}, E)$ , where  $\rho$  is the fluid density,  $\mathbf{U}$  is velocity and  $E$  its total energy per volume unit. Triplet  $(F, G, H)$  denotes the usual conservative Euler fluxes. In order to optimize the shape without handling a variable computational domain, the variation of the shape is taken into account by a **transpiration condition**. The state equation then writes:

$$\begin{aligned} \forall \phi, \quad (\Psi(\gamma, W), \phi) = \\ - \int_{\Omega_0} (F(W) \frac{\partial \phi}{\partial x} + G(W) \frac{\partial \phi}{\partial y} + H(W) \frac{\partial \phi}{\partial z}) d\Omega_0 \\ + \int_{\partial\Omega_0} p (\mathbf{n}_x^0 \phi_2 + \mathbf{n}_y^0 \phi_3 + \mathbf{n}_z^0 \phi_4) d\partial\Omega_0 \\ + \int_{\partial\Omega_0} q(\gamma) (W_1 \phi_1 + W_2 \phi_2 + W_3 \phi_3 + \\ W_4 \phi_4 + (W_5 + p) \phi_5) d\partial\Omega_0 = 0 . \end{aligned} \quad (5)$$

Where  $p = p(W)$  is the pressure of fluid (perfect gas law is assumed), and, according to the transpiration condition formulation, the slip boundary term of the flux  $\Psi(W)$  is defined on a **fixed reference geometry**  $\Omega_0$  of normal exterior  $\mathbf{n}^0$  by:

$$q(\gamma) = \mathbf{U} \cdot (\mathbf{n}^0 - \mathbf{n}^\gamma)$$

in which  $\mathbf{n}^\gamma$  is the exterior normal to varying domain  $\Omega_\gamma$ .

This **discrete Euler system**, denoted

$$\Psi_h(\gamma, W_h) = 0 , \quad (6)$$

is built by means of a vertex-centered Mixed-Element-Volume approximation on unstructured tetrahedrisations. The consistent part is a Galerkin formulation. The stabilizing part relies on a Roe Riemann solver combined with a MUSCL reconstruction with limiters defined as in [7]. This produces a space accuracy of order two. Let us mention that for solution of the steady system, an explicit multi-stage pseudo-time integration which does not influence the spatial accuracy is applied. For a given  $\gamma$ , we denote by  $W_h(\gamma)$  the solution of (6).

We evaluate the cost function at a plane  $z = -R$ , where  $R$  is a multiple of the aircraft length (i.e.  $R = L, R = 2L, R = 3L$  etc.). This plane is discretised as the intersection between tetrahedra and the plane represented by the equation  $z = -R$ . The cost function is computed over intersection points between mesh and a plane. The cost function is defined by:

$$j(\gamma) = J(\gamma, W_h(\gamma)) = \left( \sum_{ifac} |ifac| (P_{ifac}(W_h(\gamma)) - P_{ifac}^{target})^2 \right) / n_{ifac}$$

where  $|ifac|$  is the area of the face  $ifac$ , this face is obtained by intersection of a mesh element and the plane, then the resulting face is or a triangle or a quadrilateral.  $n_{fac}$  is the number of intersected faces. Denoting by  $P(W_h)$  the pressure field corresponding to  $W_h(\gamma)$ ,  $P_{ifac}(W_h)$  is obtained by  $P_1$  interpolation of  $P(W_h)$  on intersection face  $ifac$ .  $P_{ifac}^{target}$  is the desired or target pressure value specified at the face  $ifac$ . One way to control the initial pressure rise is to define on an interval  $[x_a, x_b]$  a target pressure chosen equal to farfield pressure  $p_\infty$ .

The functional minimum we are looking for is the solution of the following *Karush-Kuhn-Tucker* (KKT) system:

$$\left\{ \begin{array}{rcl} \Psi_h(\gamma, W_h) & = & 0 \\ & & \text{(State)} \\ \frac{\partial J}{\partial W}(\gamma, W_h) - \left( \frac{\partial \Psi_h}{\partial W_h}(\gamma, W_h) \right)^* \cdot \Pi & = & 0 \\ & & \text{(Adjoint state)} \\ \frac{\partial J}{\partial \gamma}(\gamma, W_h) - \left( \frac{\partial \Psi_h}{\partial \gamma}(\gamma, W_h) \right)^* \cdot \Pi & = & j'(\gamma) = 0 \\ & & \text{(Optimality)} \end{array} \right. \quad (7)$$

The demonstration platform used in this paper takes into account several methodological contributions. Shape parameterization and multilevel preconditioning methods were elaborated in [15],[13],[18]. The residual for adjoint state  $\Pi$  and the functional gradient software are developed with the help of reverse Automatic Differentiation, see [10]. Efficient strategies for adjoint state solving are proposed in [19]. Comparisons with another platform and some mutual method enrichment are discussed in [16].

### 2.3 Mesh adaptation model

With the introduction of mesh adaptation in a design loop, a first question is risen: what do we want to compute accurately? Let us discuss the three following standard options:

- (a)- we adapt for a better flow evaluation,
- (b)- we adapt for a better functional evaluation,
- (c)- we adapt for a better evaluation of optimal design.

Option (c) seems the most theoretically convenient, but in practice, the optimality system is not well solved, and the optimal design is not well known. Further, with our particular application, the optimisation is difficult with cost evaluation that do not use an accurate mesh adaptation. Then an external loop as in option (a) is very difficult to apply. Then options (b) and (c) seem more realistic. Option (b) is of high interest. It is still in infancy for anisotropic mesh adaptation and we shall discuss it in a forthcoming paper. Then option (a) is considered here.

The mesh adaptation method relies on a continuous theory and a fixed point discrete algorithm.

**Continuous metric theory** ([1]): the mesh is represented by a continuous field, the metric  $\mathcal{M}$ , with a density  $d_{\mathcal{M}}$ . The integral  $N = \int_{\Omega} d_{\mathcal{M}} d\Omega$  of density over the computational domain represents the total number of nodes. The adapted metric  $\mathcal{M}_{\mathbf{L}^2}$  is the metric of number of nodes  $N$  which minimises a continuous model for the  $\mathbf{L}^2$   $P_1$ -interpolation error for a flow variable, typically the Mach number.

XXX

The mesh adaptation algorithm is built in order to compute a mesh and a numerical flow. Both are implicitly coupled since flow is computed on the mesh and mesh is specifically adapted to flow. This coupling is attained by iterating a fixed-point algorithm. At each stage, a numerical solution is computed on the current mesh with the Euler flow solver and has to be analyzed by means of an error estimate. The considered error estimate aims at minimizing the interpolation error in norm  $\mathbf{L}^2$ , independently of the problem at hand. From the continuous metric theory in [1], an analytical expression of the  $\mathbf{L}^2$ -optimal metric  $\mathcal{M}_{\mathbf{L}^2}$  is exhibited that minimizes the interpolation error in norm  $\mathbf{L}^2$ . Let  $u$  be an analytic solution defined on computational domain  $\Omega$  and let  $N$  denotes the desired number of vertices for the mesh, the aims is to create the "best" mesh  $\mathcal{H}$ , *i.e.*, the optimal continuous metric  $\mathcal{M}$ , to minimize the interpolation error  $\|u - \Pi_h u\|_2$  in  $\mathbf{L}^2$  norm.  $\Pi_h u$  denotes the linear interpolate of  $u$  on  $\mathcal{H}$ . The local interpolation error in the neighborhood of the vertex  $a$  is written:

$$e_{\mathcal{M}}(a) = \sum_{i=1}^n h_i^2 \left| \frac{\partial^2 u}{\partial \alpha_i^2} \right|, \quad (8)$$

where  $h_i$  stand for metric sizes and  $\frac{\partial^2 u}{\partial \alpha_i^2}$  represents the eigenvalues of the Hessian of the variable  $u$ . Now, we are looking for a function  $\mathcal{M}$  that minimizes, for a given number  $N$  of vertices, the  $\mathbf{L}^p$  norm of this error. To this end,

we have to solve the following problem:

$$\begin{aligned} \min_{\mathcal{M}} \mathcal{E}(\mathcal{M}) &= \min_{\mathcal{M}} \int_{\Omega} (e_{\mathcal{M}}(\mathbf{x}))^2 \, d\mathbf{x} \\ &= \min_{h_i} \int_{\Omega} \left( \sum_{i=1}^n h_i^2(\mathbf{x}) \left| \frac{\partial^2 u}{\partial \alpha_i^2}(\mathbf{x}) \right| \right)^2 \, d\mathbf{x}, \end{aligned} \tag{9}$$

under the constraint:

$$\mathcal{C}(\mathcal{M}) = \int_{\Omega} \prod_{i=1}^n h_i^{-1}(\mathbf{x}) \, d\mathbf{x} = \int_{\Omega} d(\mathbf{x}) \, d\mathbf{x} = N. \tag{10}$$

The resulting optimal metric solution of the Problem (9) and (10) for the  $L^2$  norm in three dimensions reads:

$$\begin{aligned} \mathcal{M}_{L^2} &= D_{L^2} (\det |H_u|)^{-\frac{1}{7}} \mathcal{R}_u^{-1} |\Lambda| \mathcal{R}_u \\ \text{with } D_{L^2} &= N^{\frac{2}{3}} \left( \int_{\Omega} \left| \prod_{i=1}^3 \frac{\partial^2 u}{\partial \alpha_i^2} \right|^{\frac{2}{7}} \right)^{-\frac{3}{2}}. \end{aligned} \tag{11}$$

**Mesh adaptive algorithm:** The sensor  $u$  will be for us the Mach number. The Hessian of it is reconstructed from the numerical solution by a double  $\mathbf{L}^2$  projection ([1]) and we derive an approximation of the metric. Next, an adapted mesh is generated with respect to this metric where the aim is to generate a mesh such that all edges have a length equal (or close to) unity in the prescribed metric and such that all elements are almost regular. The volume mesh is adapted by local mesh modifications of the previous mesh (the mesh is not regenerated) using mesh operations: vertex insertion, edge and face swap, collapse and node displacement. The vertex insertion procedure uses an anisotropic generalization of the Delaunay kernel. Finally, the solution is linearly interpolated on the new mesh. In a *static adaptation*, this procedure is repeated up to convergence. Process is considered as converged in Step 7 when the difference between two metrics is small. In practice, this fixed point iterates about 5 times. Computing expenses can be reduced by saving and transferring flow arrays between remeshings. It is shown in [11] that for flow calculation, this mesh adaptive loop produces second-order accurate Mach number outputs.

XXX

### 3 Coupling in discrete case

We discuss now the strategy for combining a high-level mesh adaptation and the solution of the KKT system (7). We first describe the two main ingredients of this combination.

#### 3.1 Minimizing for a fixed mesh

Assuming we are applying a steepest descent algorithm, we use an exact gradient of the discrete functional in order to keep a reliable descent direction. The following sequence can be applied to shape design without changing the computational mesh, thanks to the transpiration condition:

*Gradient and line search:*

- compute the flow (state equation),
- compute the adjoint state,
- compute the (exact) gradient of functional,
- descent: update shape control variable.

#### 3.2 Mesh adaptation for a fixed control

#### 3.3 Coupled iteration

The flow changes when the shape is changed and adapted mesh needs be changed too . We shall refer to static adaptation when only one shape and one flow are concerned, and to dynamic adaptation else. We distinguish two level of coupling between adaptation and optimization:

### 3.3.1 Weak coupling between optimization and adaptation:

It relies on a static anisotropic adaptation loop applied at beginning of every gradient step.

#### Algorithm 1: static adaptation/gradient step:

*Input:*  $\mathcal{M}_0, \gamma_0$

*Output:*  $\mathcal{M}, \gamma_{opt}$ , the converged mesh and optimal shape

1- Do

2- Do

2.1- compute on current mesh the flow (state equation),

2.2- compute the metrics for flow in steps 2,  
build a new mesh specified by the new metric  
and by a fixed number  $N$  of nodes,

While adaptation is not converged.

3- compute on current mesh the adjoint state,

4- compute on current mesh the (exact) gradient of functional,

5- perform descent: update shape control variable the descent direction,

While control is not optimized.

### 3.3.2 Strong coupling:

A mesh that is accurately adapted to a flow will be much less accurate when used for computing an -even slightly-different flow. Then starting a line search with a mesh adapted to the first flow may result in poor evaluation of the other flows and a poor evaluation of the descent step length. To avoid this, we adapt the transient fixed point adaptation method introduced in [2]. In the fixed-point adaptation/gradient loop, the mesh is adapted to the  $k$ -th gradient+search step by adapting it to all the flows of this step:

- to each flow correspond an optimal metric,
- the *intersection* of all these metrics is computed,
- the adapted mesh is built from this intersection metric.

#### Algorithm 2: dynamic adaptation/gradient step:

*Input:*  $\mathcal{M}_0, \gamma_0$

*Output:*  $\mathcal{M}, \gamma_{opt}$ , the converged mesh and optimal shape

1- Do

2- Do

2.1- compute on current mesh the flow (state equation),

2.2- compute on current mesh the adjoint state,

2.3- compute on current mesh the (exact)gradient of functional,

2.4- perform descent: update shape control variable

2.5- compute the intersection of metrics for each intermediate flow in steps 2.1-2.4,

2.6- build a new mesh specified by the intersected metric and by a fixed  
number  $N$  of nodes,

While adaptation is not converged.

While control is not optimized.

The fixed point adaptation/gradient step is then itself included in the gradient loop.

It is necessary to fix either the number of nodes or a certain level of accuracy in order to have, in the fixed-point process a well-posed problem with respect to the metric. It is possible to imagine to vary this number of nodes from one gradient step to the other. A uniform accuracy between time steps is a sound option in the case for an unsteady process, this is not necessary in the case of optimisation (we could try to increase progressively accuracy. In the experiments presented in the sequel, we simply have fixed  $N$ .

### 3.3.3 Improved shape representation:

The use of a transpiration condition is useful for the application of the descent with a fixed mesh. But it may introduce inaccuracies in the numerical description of the shape being computed. In particular, transpiration approximation is known as producing an inaccurate flow when the amplitude of the shape perturbation to account for in terms of surface displacement and orientation is not small. Now, the progression towards optimal shape is generally combined with a non-negligible increasing of the amplitude. In [14], it is proposed to recover flow accuracy by updating the mesh at the end of every optimization step. The new mesh is fitted to the shape. This is obtained by a mesh deformation system relying on a spring analogy introduced in [9]. This combines two advantages, first the flow at the beginning of each optimization iteration is evaluated without transpiration, and, second, this also holds for the optimum context when shape update is converging to zero. In that case, in the set of three optimality conditions, state and adjoint are not affected by the transpiration simplification. Only the stationarity of functional (“gradient is zero” in case of non-constraint) is influenced by the transpiration approximation. This strategy is summarized in Algorithm 3:

**Algorithm 3: dynamic adaptation/gradient step with mesh deformation:**

*Input:*  $\mathcal{M}_0, \gamma_0$

*Output:*  $\mathcal{M}, \gamma_{opt}$ , the converged mesh and optimal shape

1- Do

    2- Do

        2.1- compute on current mesh the flow (state equation),

        2.2- compute on current mesh the adjoint state,

        2.3- compute on current mesh the (exact)gradient of functional,

        2.4- perform descent: update shape control variable

        2.5- compute the intersection of metrics for each intermediate flow in steps 2.1-2.3,

        2.6- build a new mesh specified by the intersected metric and by a fixed number  $N$  of nodes,

    While adaptation is not converged.

Compute a deformed mesh conforming the new shape

    While control is not optimized.

Finally, the optimization/adaptation platform can be represented as in figure 1.

## 4 Applications to problem under study

Preliminary optimization computations have been applied to an HISAC test case (figure ??). We describe in Table 1, the initial configuration used to perform the pseudo-strong and the strong coupling computations. Both strategies use mesh deformation introduced in Algorithm 3. Further details can be found in [20].

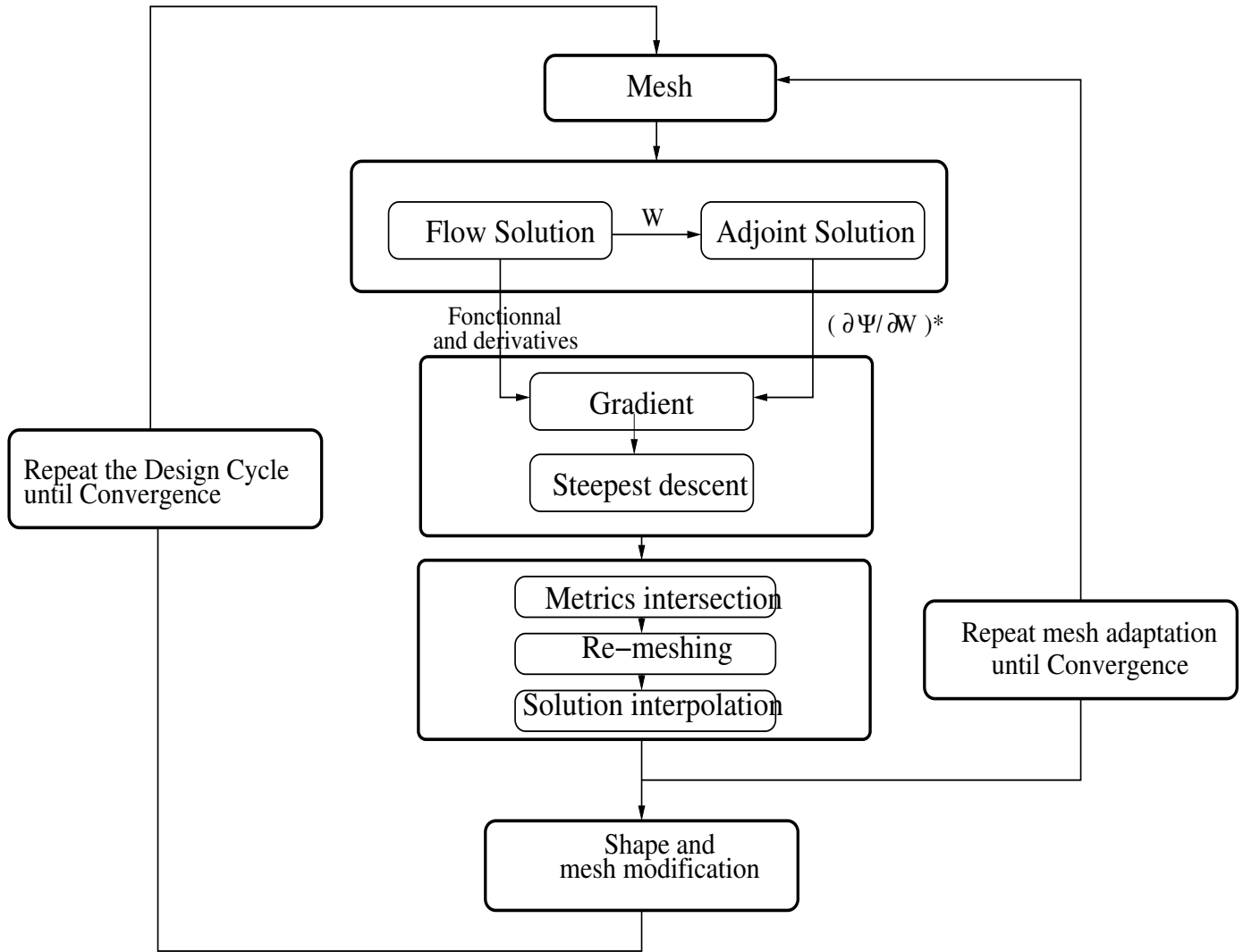


Figure 1: Strong coupling platform

Initial mesh vertices size	42120
Initial mesh elements size	223657
MACH number	1.6
Angle of attack	3
Aircraft length	30m
Observation plane	$z = -R = -30m$
Number of optimization iterations	10
Number of mesh adaptation iterations by one optimization iteration	5

Table 1: simulation parameters

Applications are based on optimizing for minimum initial shock pressure rise (ISPR) using transpiration conditions and torsional springs deformation. Minimizing the ISPR leads not only to the lowest ISPR but also to the lowest maximum overpressure.

#### 4.1 Application of pseudo-strongly coupled scheme to coarse meshes:

A pseudo-strongly coupled scheme consists in using slightly the mesh adaptation process in the optimization platform, which means that we apply just a single time the mesh adaptation process by iteration of optimization. The aim of this exercise is to measure the impact of mesh adaptation in shape optimization platforms. Figures below, represent some comparisons between the original and the optimized configurations.



Figure ?? shows the horizontal cuts of pressure value at  $z = -30m$  in order to compare the initial flow and the final flow after optimization. We observe that the first shock localization of initial flow is slightly reduced.

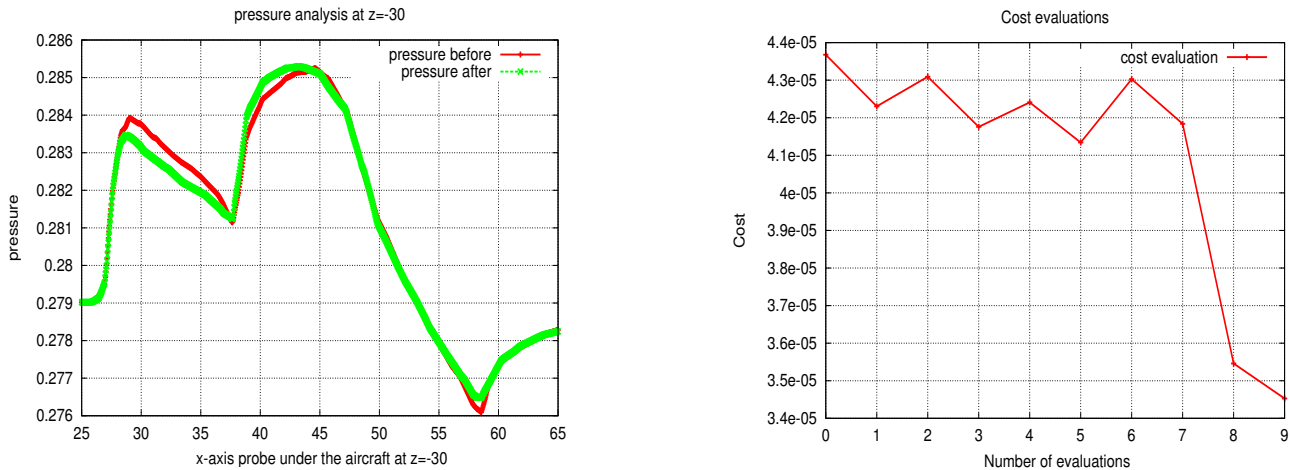
Figure 2 shows on the right the evaluation of functional during the coupled loop. The oscillations observed in the functional curve are associated to the re-meshing stage. On the other hand one notes that with the last iteration, a brutal descent of the function. This behavior lets us to raise questions about the certainty of the functional computation.

On the left of the figure 2 we depict the progress obtained on the near-field pressure after optimization. The red line (green line) corresponds to the initial (final) near-field pressure respectively.

Figure 3 shows the reduction obtained after propagation of the near-field pressure to the ground.

Figure ?? shows the final pressure distribution and adapted mesh respectively.

Figure 4 shows from the left to the right a comparison between the original and the optimized shapes at the nose tip respectively.



Pressure comparison

Cost evaluation

test-case 3:  $l_{front} = 25$  and  $l_{back} = \mathbf{35}$ .

Figure 2: Pseudo-strong coupling. Left: Comparison of the near-field signatures for the original and the optimized configurations, right: cost function evaluation around coupled mesh adaptation and optimization loop.

## 4.2 Application of strongly coupled scheme to coarse meshes:

A strongly coupled scheme consists in using a sufficiently converged fixed-point mesh adaptation process in the optimization loop.

Figure ?? shows the horizontal cuts of pressure value at  $z = -30m$  in order to compare the initial flow and the final flow after optimization. We observe that the first shock localization of initial flow is well reduced.

Figure 5 shows on the right the evaluation of functional during the coupled loop. The oscillations observed in the functional curve are associated to the mesh adaptation phase which is devoted to find the best adapted mesh and then ensure the good evaluation of the functional. This mesh adaptation influence over global optimization loop gives us a computation certainty along optimization cycle.

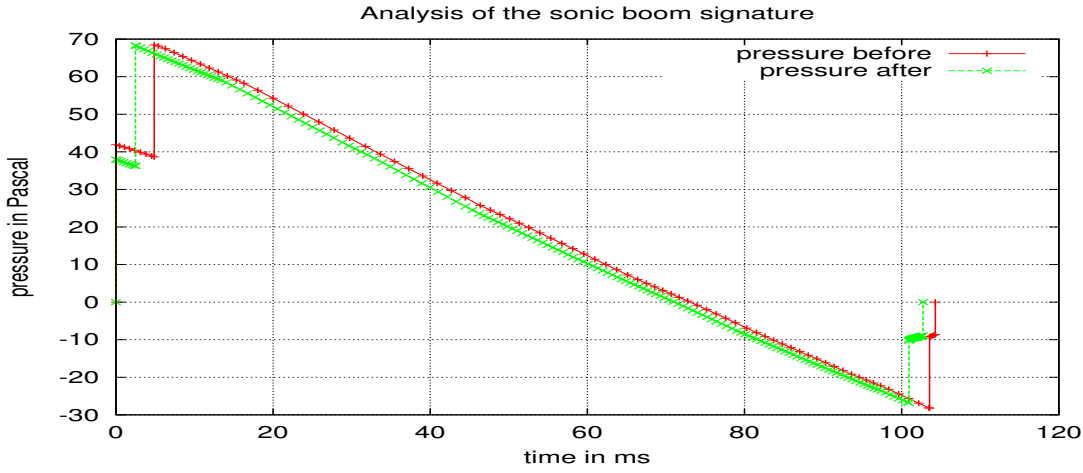
On the left of the figure 5 we depict the progress obtained on the near-field pressure after optimization. The red line (green line) corresponds to the initial (final) near-field pressure respectively.

Figure 6 shows the reduction obtained after propagation of the near-field pressure to the ground.

Figure ?? shows the final pressure distribution and adapted mesh respectively.

Figure 7 shows from the left to the right a comparison between the original and the optimized shapes at the nose tip respectively.

We summarize the ground signatures obtained with the pseudo-strong coupling and the strong coupling strategies in the following table:



Test case 3:  $H = 45000\text{ft}$

Figure 3: Pseudo-strong coupling: comparison of the ground signatures for the original and the optimized configurations.

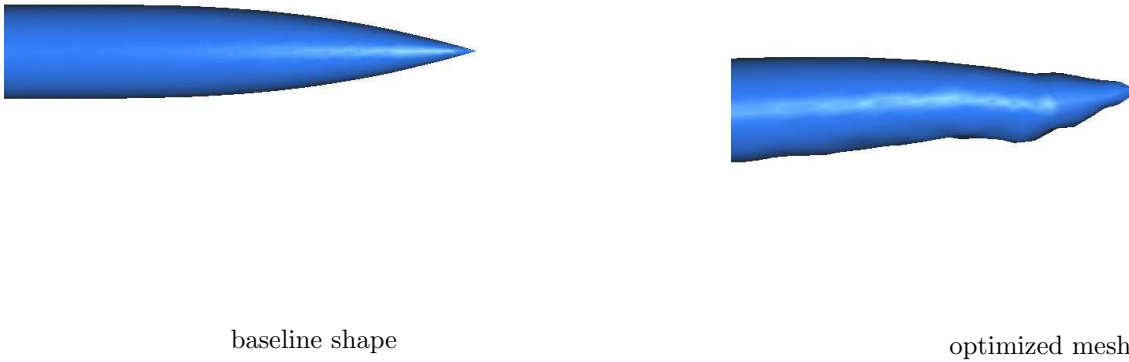
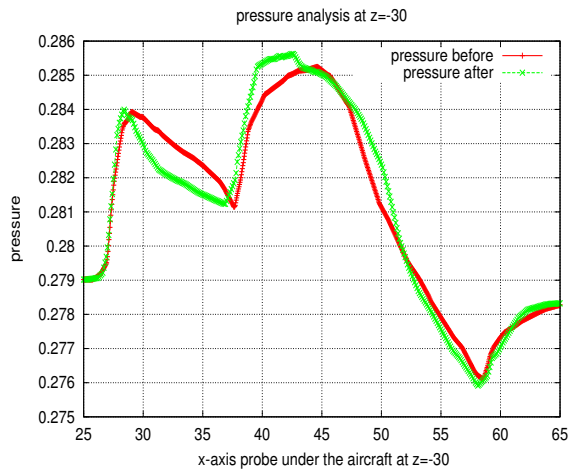


Figure 4: Pseudo-strong coupling: comparison of the original and optimized shapes at the nose tip for the HISAC case

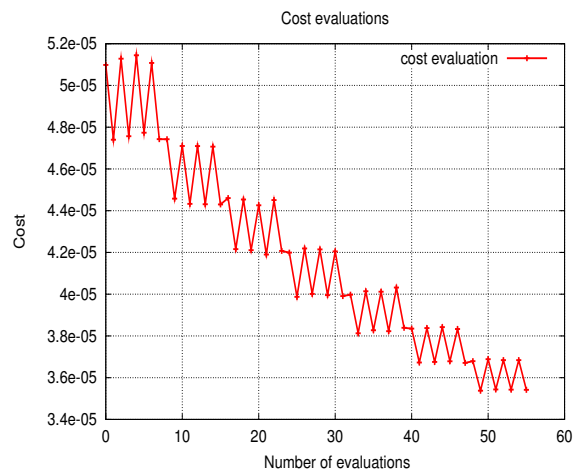
case	$ISPR(Pa)$
Original shape	42
Optimized shape by pseudo-strong coupling	37.5
Optimized shape by strong coupling	35

Table 2: Initial shock pressure rise in Pascal for different strategies.

These results suggest that coupling strategies are suited for minimizing the ISPR. The sonic boom is reduced in both cases by creating a blunt nose (figs. 4 and 7). In particular the strong coupling strategy is more efficient than the pseudo-strong coupling strategy. Indeed, the gain reached by **SC** (strong coupling) is about 20% and the gain reached by **PSC** (pseudo-strong coupling) does not exceed the 12%.



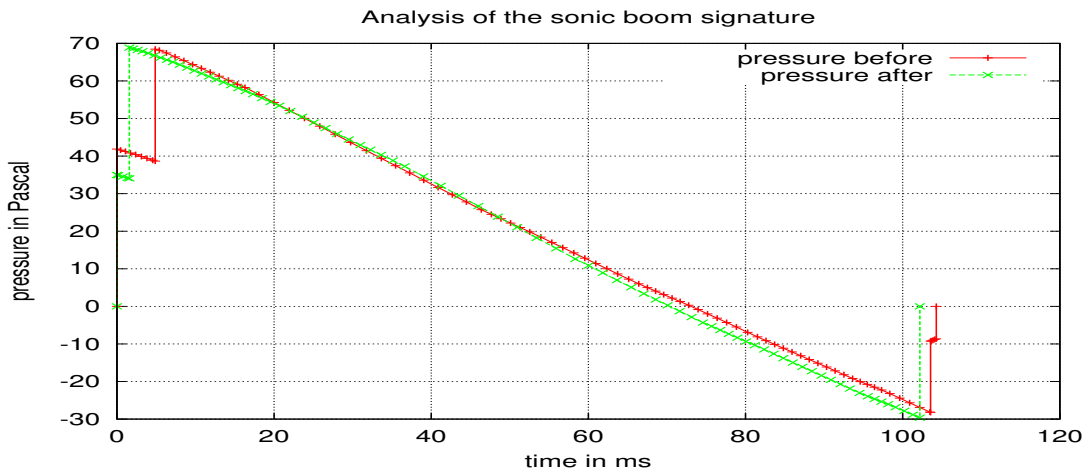
Pressure comparison



Cost evaluation

test-case 3:  $l_{front} = 25$  and  $l_{back} = 35$ .

Figure 5: Strong coupling: Left: Comparison of the near-field signatures for the original and the optimized configurations, right: cost function evaluation around coupled mesh adaptation and optimization loop.

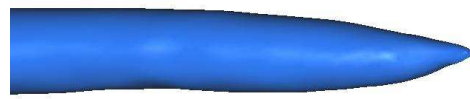


Test case 3:  $H = 45000 ft$

Figure 6: Strong coupling: Left: comparison of the ground signatures for the original and the optimized configurations, right: comparison of the near-field signatures for the original and the optimized configurations



baseline shape

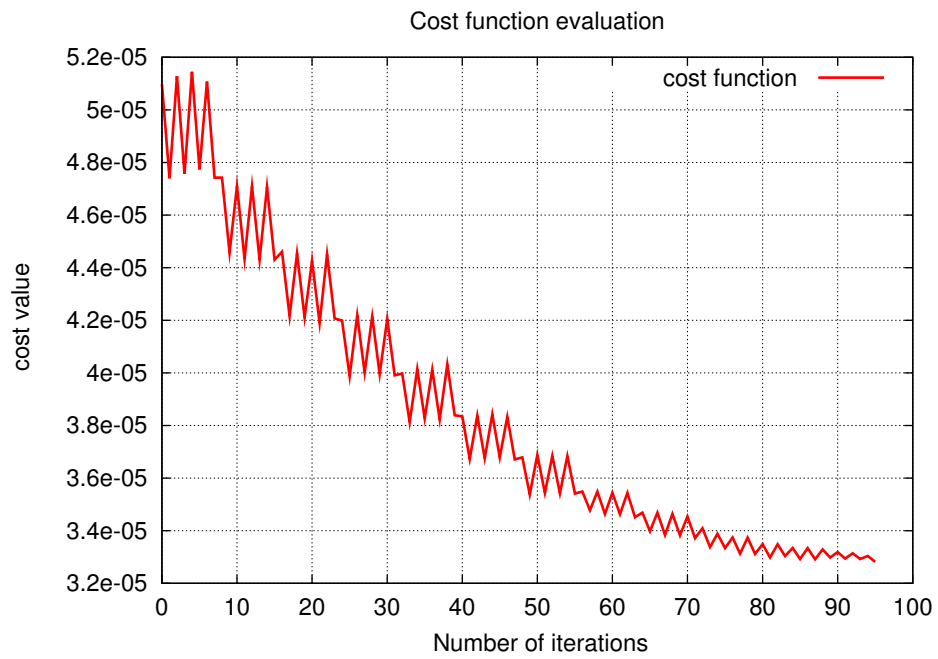


optimized mesh

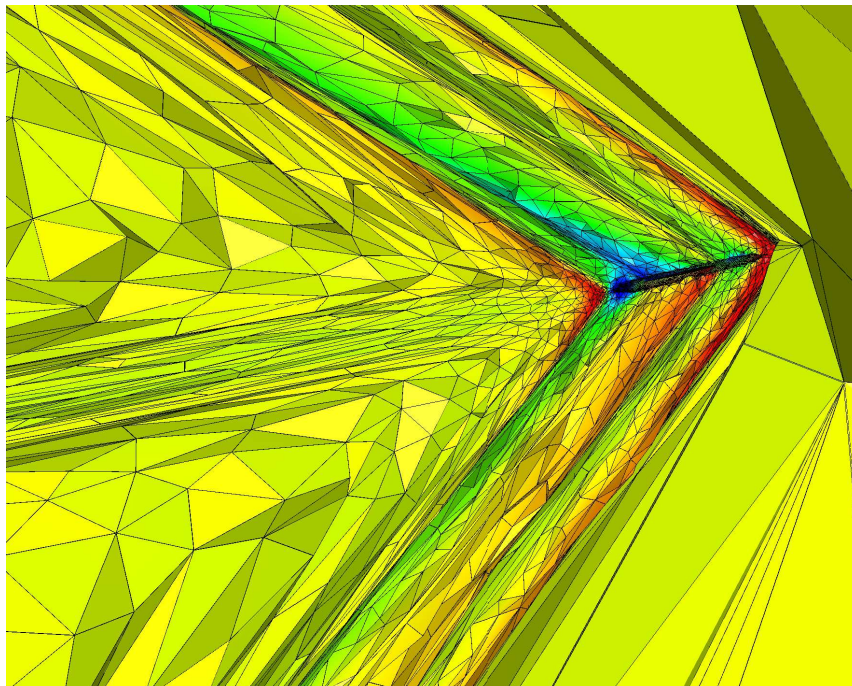
Figure 7: Strong coupling: Comparison of the original and optimized shapes at the nose tip for the HISAC case

### 4.3 Application of strong coupling with finer meshes

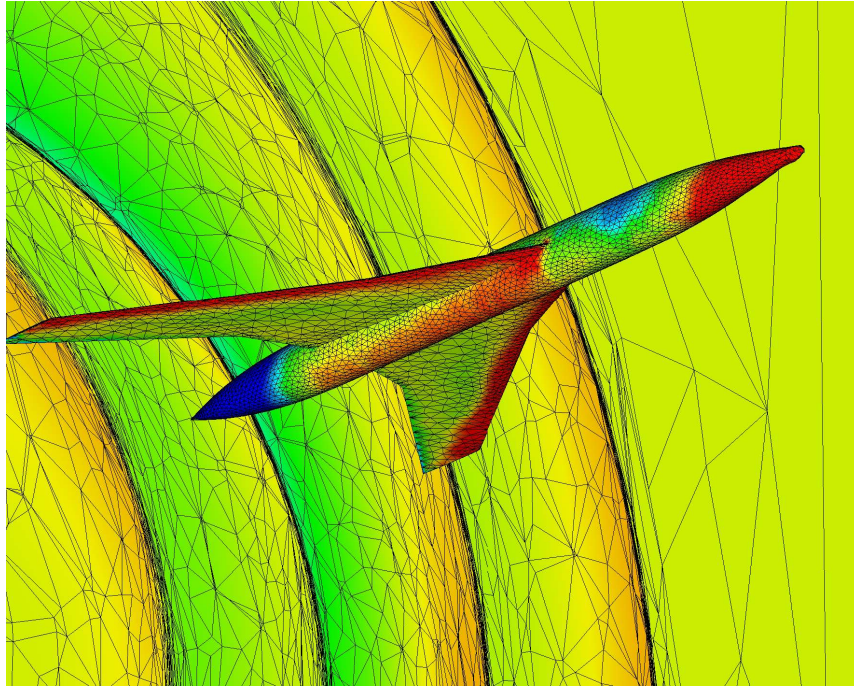
We recompute the above optimum with a finer specification of the number of nodes. This number is now fixed to 100,000 nodes.



Strong coupling, fine mesh: Optimization of the HISAC shape with a mesh size of 100Kcells



Strong coupling, fine mesh: Optimized shape: adapted mesh



Optimized shape

## 5 Concluding remarks

We have addressed a design problem in which mesh adaptation is mandatory. It can be considered as an equality constraint, in complement of the state system. Observing that this constraint is strongly nonlinear, we studied fixed point strategies in order to couple more or less strongly the adaptation process and the optimisation loop.

This has been applied to a sonic boom mitigation study. Some of our numerical experiments show that when choosing strong coupling, the optimization can be successfully performed.

In the presented application, mesh adapted optimisation brings information of higher quality than before. We are expecting an extra improvement by introducing a goal-oriented adaptation criterion.

## 6 Acknowledgements

We acknowledge for the support of this work by FP6-HISAC European project.

## References

- [1] ALAUZET F., LOSELLE A., DERVIEUX A. AND FREY P., “Multi-dimensional continuous metric for mesh adaptation”, *15th Int. Meshing Roundtable*, Birmingham, USA, Springer Verlag, pp. 191-214, 2006.
- [2] ALAUZET F., GEORGE P.L., MOHAMMADI B., FREY P., BOROUCHAKI H., “Transient fixed point based unstructured mesh adaptation”, *Int. J. Numer. Methods in Fluids*, 43:6-7, 729-745, 2003.
- [3] CHOI, S., ALONSO, J. J., AND VAN DER WEIDE, E., “Numerical and Mesh Resolution Requirements for Accurate Sonic Boom Prediction of Complete Aircraft Configurations”, *AIAA Paper 2004-1060*, 2004.
- [4] NADARAJAH S.K., JAMESON A., ALONSO J., “Adjoint-Based Sonic Boom Reduction for Wing-Body Configurations in Supersonic Flow”, *The Canadian Aeronautic and Space Journal*, 51:4, 187-199, 2005.
- [5] BECKER R., “Adaptive Finite Elements for Optimal Control problems”, Habilitation thesis, Universität Heidelberg (2001)



- [6] BEUX F., DERVIEUX A. (EDS), "Shape Design in Aerodynamics, Parameterization and sensitivity", *European Journal of Computational Mechanics*, 17:1-2 (2008)
- [7] COURNÈDE P.-H., KOOBUS B., DERVIEUX A., "Positivity statements for a Mixed-Element-Volume scheme on fixed and moving grids", *European Journal of Computational Mechanics*, 15:7-8, 767-799, 2006.
- [8] FARHAT C., MAUTE K., ARGROW B. AND NIKBAY M., "A shape optimization for reducing the sonic boom initial pressure rise", *AIAA Journal*, 10:10, 2091-2093, 2007.
- [9] DEGAND C. AND FARHAT C., "A three-dimensional torsional spring analogy method for unstructured dynamic meshes", *Comput. Struct.*, 80 (2002) 305.
- [10] HASCOËT L., VÀZQUEZ M., DERVIEUX A., "Automatic Differentiation for Optimum Design, applied to Sonic Boom reduction", *Proceedings of the International Conference on Computational Science and its Applications, ICCSA'03*, Montreal, Canada, V.Kumar et al., editors, LNCS 2668, pp 85-94, Springer, 2003
- [11] LOSEILLE A., DERVIEUX A., FREY P. AND ALAUZET F., "Achievement of global second order mesh convergence for discontinuous flows with adapted unstructured meshes", *18th AIAA Computational Fluid Dynamics Conference*, Miami (FL), 25 - 28, 2007
- [12] NADARAJAH S. K. AND JAMESON A., "Optimum shape design for unsteady three- dimensional viscous flows using a non-linear domain method", *24th Applied Aerodynamics Conference*, June 2006, San Francisco, California.
- [13] VÀZQUEZ M., KOOBUS B. AND DERVIEUX A., "Multilevel Optimization of a supersonic aircraft", *Finite Element in Analysis and Design*, 40, 2101-2124, 2004.
- [14] VÀZQUEZ M., DERVIEUX A. AND KOOBUS B., "A methodology for the shape optimization of flexible wings", *Engineering Computations*, 23:4, 344-367, 2006.
- [15] MARCO N., KOOBUS B. AND DERVIEUX A. "An additive multilevel preconditioning method and its application to unstructured meshes", INRIA research report 2310, 1994 and *Journal of Scientific Computing*, 12, no 3, 233-251, 1997
- [16] ALAUZET F., BOREL-SANDOU S, DAUMAS L., DERVIEUX A., DINH Q., KLEINFELD S., LOSEILLE A., MESRI Y. AND ROGÉ G., "Multi-model and multi-scale optimization strategies", *European Journal of Computational Mechanics*, 17:1-2 247-273 (2008)
- [17] MAUTE K., FARHAT C., ARGROW B. AND NIKBAY M., "Sonic boom mitigation vis shape optimization using an adjoint method and application to a supersonic fighter aircraft", *European Journal of Computational Mechanics*, 17:1-2 217-243 (2008)
- [18] COURTY F. AND DERVIEUX A. "Multilevel functional Preconditioning for shape optimisation", *I.J.Comput. Fluid Dynamics*, 20:7,481-490,2006
- [19] MARTINELLI M., "Sensitivity Evaluation in Aerodynamic Optimal Design", Ph.D thesis, Scuola Normale di Pisa and université de Nice-Sophia-Antipolis, 2007.
- [20] MESRI Y. "Gestion et Contrôle des maillages non structurés Anisotropes, Applications en Aérodynamique", Ph.D thesis, Université de Nice-Sophia-Antipolis, 2007 (in french).