
Multimodel design strategies applied to sonic boom reduction

Frédéric Alauzet* — **Sophie Borel-Sandou****
Laurent Daumas*** — **Alain Dervieux****** — **Quang Dinh*****
Steven Kleinveld*** — **Adrien Loseille*** — **Youssef Mesri******
Gilbert Rogé***

* *INRIA, Projet Gamma, Domaine de Voluceau, Rocquencourt
BP 105, F-78153 Le Chesnay cedex*

** *Université Pierre et Marie Curie, Laboratoire J.L. Lions
175 rue du Chevaleret, F-75013 Paris*

*** *Dassault Aviation, DGT, 78 quai Marcel Dassault
F-92552 St-Cloud cedex 300*

**** *INRIA, Projet Tropics, 2004 Route des Lucioles, BP 93
F-06902 Sophia-Antipolis cedex*

ABSTRACT. The shape optimization of a supersonic aircraft need a composite model combining a 3D CFD high-fidelity model and a simplified boom propagation model. The management of this complexity is studied in an optimization loop, with exact discrete adjoints of 3D flow and mesh deformation system. The introduction of a mesh adaptation algorithm is also considered.

RÉSUMÉ. L'optimisation de la forme d'un avion supersonique nécessite un modèle composite comportant une composante 3D haute fidélité en mécanique des fluides et un modèle simplifié de propagation du bang. La prise en compte de cette complexité est étudiée dans le cadre d'une boucle d'optimisation, avec des adjoints discrets exacts de l'écoulement 3D et un système de déformation de maillage. L'introduction d'une méthode d'adaptation de maillage est aussi considérée.

KEYWORDS: optimization, shape design, parameterization, automatic differentiation, anisotropic mesh adaptation, sonic boom.

MOTS-CLÉS : optimisation, forme optimale, paramétrisation, différenciation automatique, adaptation de maillage anisotrope, bang sonique.

DOI:10.3166/REMN.17.245-269 © 2008 Lavoisier, Paris

1. Introduction

When passing to real life applications, optimal design algorithms have their complexity increased from many standpoints. The modeled physics is not always easily put in a unique PDE model, due to the very large and very small scales involved in the problem. A classical approach in that case consists in using specialized models for the different range of scales and considering the system as a multimodel one. Then the Multi-disciplinary Design Optimization (MDO) methods can be applied. These methods are able to couple different physics and also different level of fidelity for the same physics, see for example (Sobieszczanski-Sobieski *et al.*, 1997) and (Alexandrov, 1997).

Our particular application, the sonic boom reduction, needs to handle the difficult problem of predicting sonic boom emission and propagation. The system of Euler equations is recognized as a satisfying model for the whole physical process. But when passing to discrete model for numerical simulations, we have to discretize on a unique mesh small and large scales. Their ratio may be enormous. Indeed, shock width is in microns and the computational domain should extend from the aircraft to the ground, 15 kilometers below. Simpler models than Euler equations may be chosen for the whole physical process but they may lose some accuracy in the sonic boom prediction. To address this issue, two steps are applied. Firstly, the 3D Euler model is applied in the vicinity of the aircraft for predicting nearfield boom emission. Secondly, a propagation model is applied with Euler nearfield outputs as initial conditions and solves pressure perturbation propagation down to ground level sonic boom signature. Turning to optimization of the system, we need determine the place of this second model into the optimization loop. Either the propagation can be directly included into the optimization loop as in (Farhat *et al.*, 2002). Or, the propagation is first used for *a priori* building a nearfield target pressure, and then the optimization loop minimizes an objective functional taking it into account this target, see (Alonso *et al.*, 2002). The latter option, with a *target nearfield pressure*, is considered in the present study.

However, we address a part of the multi-scale issue by applying a *mesh adaptation loop*. The three main ingredients of mesh adaptation are:

- (i) the choice of a criterion governing the local repartition of nodes and the alignments of elements,
- (ii) the type of remeshing,
- (iii) the way the mesh generator is coupled with the rest of the algorithm.

Concerning (i), the user has to know what is the priority between accuracy of flow field evaluation or accuracy of final optimal control. Accuracy of the whole flow can be set in general and is solved by local error estimators, as interpolation estimators, see (Alauzet *et al.*, 2007) and (Zhu *et al.*, 1997). When focussing on functional value accuracy, goal-oriented *a posteriori* estimators are then considered, see (Giles *et al.*, 2001) and (Venditti *et al.*, 2002). Functional error can be also minimized as in (Koobus *et al.*, 2007). The issue of accuracy in optimal control variable itself is addressed for

example in (Polak, 1997) and (Becker *et al.*, 2001). Point (ii), type of remeshing, covers any mesh modification algorithm, from local mesh enrichment with patterns to global mesh regeneration with advanced meshing techniques. In our case, the volume mesh is adapted by local mesh modifications of the previous mesh using mesh an anisotropic generalization of the Delaunay theory. Coupling mesh adaptation with optimization algorithm, (iii), is not a trivial issue. Accuracy of adaptation should not be deteriorated by the optimization and robustness of optimization should not be lost by mesh updates. Our proposed approach is inspired by an analogy with the fixed point mesh adaptation method for unsteady flows proposed in (Alauzet *et al.*, 2007).

An important feature of the study is the option of a gradient-based minimization. It is motivated by the necessity of considering a large number of design variables. An important disadvantage of gradient-based minimization is the need of a differentiable functional. This means that we may need to differentiate the sonic boom propagation model, and the numerical approximation with mesh changes. Two other important ingredients may have to be installed in the optimization loop. CAD shape parameterization belong to the functional to minimize, together with the process generating a new mesh for a new shape. Both have to be handled in the gradient evaluation.

In this paper, we first introduce the problem under study in terms of the basic high-fidelity model. Secondly, we present the two extensions of the basic model, *i.e.*, the propagation model and mesh adaptive model. Thirdly, we describe a numerical optimization platform and show which kind of results can be obtained in combination with the propagation model. Then, preliminary experiments on an optimization platform coupled with mesh adaptation are described.

2. Mathematical design problem

2.1. Continuous standpoint

The flow around a supersonic aircraft is modeled by the Euler equations. Assuming that the gas is perfect, non viscous and that there is no thermal diffusion, the Euler equations for mass, momentum and energy conservation read:

$$\left\{ \begin{array}{l} \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{U}) = 0, \\ \frac{\partial(\rho \vec{U})}{\partial t} + \nabla \cdot (\rho \vec{U} \otimes \vec{U}) + \nabla p = 0, \\ \frac{\partial(\rho E)}{\partial t} + \nabla \cdot ((\rho E + p)\vec{U}) = 0, \end{array} \right.$$

where ρ denotes the density, \vec{U} the velocity vector, $E = e + \frac{\|\vec{U}\|^2}{2}$ the total energy and $p = (\gamma - 1)\rho e$ the pressure with $\gamma = 1.4$ the ratio of specific heats and e the internal energy. These equations are symbolically rewritten:

$$\frac{\partial W}{\partial t} + \nabla \cdot F(W) = 0, \quad [1]$$

where $W = {}^t(\rho, \rho u, \rho v, \rho w, \rho E)$ is the conservative variables vector and the vector F represents the convective flux. In fact we are interested only in the *steady* solution of this system. It has to be evaluated from the region around the aircraft (near field) to the ground level (far field), so that the computational domain boundary involves the two components, viz. the aircraft wall and the ground. The pressure perturbation created by the supersonic aircraft flight extends down to the ground and is the source of uncomfortable noise for populations, *i.e.*, the sonic boom. During this propagation to ground, the perturbation signal transforms, see Figure 1, left.

The optimization problem under study in this paper is to find an aircraft shape that would reduce the impact of this noise. This can be expressed as the research of a minimum of an objective functional measuring the deviation between the ground pressure signature and a target one. We define it *in short* as:

$$j = \int (p(W) - p_{target})^2 ds \quad [2]$$

where the integral is taken on ground surface and the functional j depends on the aircraft shape through flow state W .

2.2. Numerical issues

Predicting the Euler flow down to the ground needs the resolution by the 3D mesh of wide range of scale sizes. Sharp shocks have to be solved with small numerical width, not necessary as small as micron, but yet of the order of centimeter, and this in a domain of tens of kilometers. As a consequence, three-dimensional Euler modeling of the pressure perturbation is not possible with standard meshes and today's computers.

In the literature, authors starting with an accurate model -as Euler equations- restrict the 3D Euler computation to a near field sub-region of the domain, usually measured with the ratio of the diameter R of the near field domain to the aircraft chord-length L . Then, propagation is handled with a simplified model. The accuracy of a composite model combining 3D Euler at near field and a propagation model depends on:

- the accuracy of the 3D Euler computation,
- the propagation model quality,
- and the adequacy between the matching location and simplification assumptions used in the propagation model.

In this paper, the 3D Euler equations are solved with a Finite Volume technique on unstructured tetrahedral meshes. The considered standard propagation model is the one proposed by Thomas (Thomas, 1972). Typically, matching distances to couple both models are set between $R/L = 0.5$ and $R/L = 5$. In order to address the problem of space-scale stiffness and to have a suitable adequacy between both models, we propose to apply an anisotropic mesh adaptation. Indeed, anisotropic mesh adaptation

allows the solution to be propagated accurately in the near field domain. The *a priori* handicap to handle is that the functional to minimize in the discrete optimization problem is changed as soon as mesh is changed.

Lastly, using the mesh adaptation does not dispense to use also Euler and propagation together, and an important question arises:

How to handle the composite model Euler/Propagation inside the optimization loop?

To address this point, two main strategies can be considered:

- the propagation model is put into the state equation of the Optimal Control problem, see for example (Farhat *et al.*, 2002). The difficulty may come from a lack of differentiability of the propagation model,
- the propagation model is used for building a near field target pressure, see for example (Alonso *et al.*, 2002). Then we need, for searching a particular ground signature, to solve an inverse problem providing the near field target pressure. A difficulty occurs from the lack of invertibility of the propagation model. Nonetheless, we still consider this option.

These two extensions, propagation and mesh adaptation models, from the basic numerical model are now examined in more details.

3. Propagation model

3.1. Atmospheric pressure wave propagation

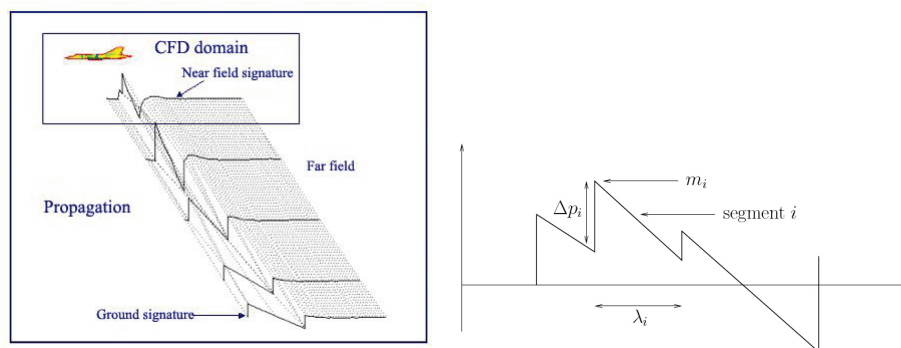


Figure 1. Left, sonic boom problem modeling. Right, illustration of the three parameters m_i , Δp_i and λ_i characterizing the pressure wave

The propagation code uses a ray tracing algorithm based upon the waveform parameter method developed in (Thomas, 1972). It allows us to propagate near field

perturbations to the ground in order to get the sonic boom signature of the aircraft. In this approach, the pressure wave is characterized by three parameters (see Figure 1):

- m_i the slope of pressure waveform segment i ,
- Δp_i the pressure rise across shock at the juncture of pressure waveform segment i and $i - 1$,
- λ_i the time duration of pressure waveform segment i .

A system of three ordinary differential equations, one for each parameter, is solved to propagate the pressure wave in the atmosphere:

$$\begin{cases} \frac{dm_i}{dt} = C_1 m_i^2 + C_2 m_i, \\ \frac{d\Delta p_i}{dt} = \frac{1}{2} C_1 \Delta p_i (m_i + m_{i-1}) + C_2 \Delta p_i, \\ \frac{d\lambda_i}{dt} = -\frac{1}{2} C_1 (\Delta p_i + \Delta p_{i+1}) - C_1 m_i \lambda_i, \end{cases} \quad [3]$$

with notations:

$$C_1 = \frac{\gamma + 1}{2\gamma} \frac{1}{p} \quad \text{and} \quad C_2 = \frac{1}{2} \left(\frac{1}{a} \frac{da}{dt} + \frac{1}{\rho} \frac{d\rho}{dt} - \frac{1}{A} \frac{dA}{dt} \right),$$

where we denote by a the air ambient sound speed, ρ the air ambient density, p the air ambient pressure and A the (acoustic) ray tube area as cut by the waveform. All these quantities are functions of the altitude. Here, the wind velocity is assumed to be zero and we use the ICAO Standard atmosphere (see (ICAO, 1993)). To solve System [3], we just have to advance it in time with a time step sufficiently small for neglecting the variations of C_1 and C_2 .

3.2. Coupling CFD and wave propagation

We denote by L the chord-length of the aircraft. The extraction line Ox , at a distance 0, is the line parallel to the flow direction going through the nose of the aircraft. By a downward vertical translation along the Mach cone at a distance R we get the extraction line at distance L . In practice, the distance from the aircraft is generally characterized by the ratio R/L . We denote by p the acoustical pressure and p_∞ the atmospherical pressure.

The pressure distribution obtained under the aircraft in the near field region is used to set up the initial conditions for the propagation of the acoustic wave to the ground. However, the modeled flow in the near field is three-dimensional and non-linear in nature whereas the propagation is a linear one-dimensional model. Thus, to ensure a valid coupling, the near field solution must be locally axi-symmetric with respect to axis Ox in the vicinity of the line where the pressure distribution is extracted. This is a necessary condition for accurately taking all the elements of the aircraft geometry

(body, wings,...) into account. Using R/L too small results in an error due to insufficient model matching. And using large R/L is difficult because it requires huge three-dimensional meshes. If the numerical solution is propagated with too much numerical damping, then it will reduce the accuracy of the near field signal. We address the latter issues by applying an anisotropic mesh adaptation strategy.

4. Mesh adaptation model

Mesh adaptation provides a way of controlling the accuracy of the numerical solution by modifying the domain discretization according to size and directional constraints. It is well known that mesh adaptation captures accurately shocks issued from the aircraft in the computational domain, see for instance (Frey *et al.*, 2005). Moreover, anisotropic mesh adaptation reduces significantly the flow solver diffusion allowing shock waves to be propagated accurately far from the aircraft.

4.1. Anisotropic mesh adaptation

For stationary problems, the mesh adaptation scheme aims at finding a fixed point for the mesh-solution couple. In other words, the goal is to converge towards the stationary solution of the problem and similarly towards the corresponding invariant adapted mesh.

At each stage, a numerical solution is computed on the current mesh with the Euler flow solver and has to be analyzed with an error estimate. The considered error estimate aims at minimizing the interpolation error in norm L^p over the whole domain. From the continuous metric theory in (Alauzet *et al.*, 2006; Leservoisier *et al.*, 2001), an analytical expression of the optimal metric is exhibited that minimizes the interpolation error in L^p norm. This anisotropic metric is a function of the Hessian of the solution which is recovered from the numerical solution by a double L^2 projection. This metric will replace the Euclidean one to modified the scalar product that underlies the notion of distance used in mesh generation algorithms. 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 of (or close to) one in the prescribed metric and such that all elements are almost regular. Such a mesh is called a *unit mesh*. The tetrahedral volume mesh is adapted by local mesh modifications of the previous mesh (the mesh is not regenerated) using the following operations: vertex insertion, edge and face swap, collapse and node displacement. The vertex insertion procedure uses an anisotropic generalization of the Delaunay kernel (Frey *et al.*, 2000). Finally, the solution is linearly interpolated on the new mesh. This procedure is repeated until the convergence of the couple mesh-solution is achieved.

4.2. Metric construction

The notion of length in a metric space is closely related to the notion of metric and subsequently to the definition of the scalar product in the vector space. A metric is a $n \times n$ symmetric definite positive matrix, where n is the space dimension. When this metric is continuously defined over the whole domain, it is called a *continuous metric*.

Let u be an analytic solution defined on a bounded domain Ω and let N denotes the desired number of vertices for the mesh. We aim at creating the "best" mesh \mathcal{H} , *i.e.*, to find the optimal continuous metric \mathcal{M} , that minimizes the interpolation error ($u - \Pi_h u$) in \mathbf{L}^p norm with N vertices. $\Pi_h u$ denotes the linear interpolate of u on \mathcal{H} . To this end, a model of the interpolation error with respect to a metric \mathcal{M} , denoted $e_{\mathcal{M}}$, is required.

In (Alauzet *et al.*, 2006), a model of the interpolation error for a metric \mathcal{M} is given. It has been proved that locally the optimal metric has for main directions the eigenvectors of the Hessian of u . Therefore, the point-wise local error model for such metric in the neighborhood of a vertex a could be simplified to:

$$e_{\mathcal{M}}(a) = \sum_{i=1}^n h_i^2 \left| \frac{\partial^2 u}{\partial \alpha_i^2} \right|, \quad [4]$$

where $\frac{\partial^2 u}{\partial \alpha_i^2}$ stands for the eigenvalue of the Hessian of u in the direction of the i^{th} eigenvectors of this Hessian, and h_i for the mesh size prescribed by the metric in the same direction. Now, we are looking for the 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:

$$\min_{\mathcal{M}} \mathcal{E}(\mathcal{M}) = \min_{\mathcal{M}} \int_{\Omega} (e_{\mathcal{M}}(\mathbf{x}))^p \, d\mathbf{x} = \min_{h_i} \int_{\Omega} \left(\sum_{i=1}^n h_i^2(\mathbf{x}) \left| \frac{\partial^2 u}{\partial \alpha_i^2}(\mathbf{x}) \right| \right)^p \, d\mathbf{x}, \quad [5]$$

under the constraint:

$$\mathcal{C}(\mathcal{M}) = \int_{\Omega} \prod_{i=1}^n h_i^{-1}(\mathbf{x}) \, d\mathbf{x} = \int_{\Omega} d(\mathbf{x}) \, d\mathbf{x} = N. \quad [6]$$

The resulting optimal metric solution of Problem [5] and [6] for the \mathbf{L}^2 norm in three dimensions reads:

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

For the sonic boom problem, we have considered the continuous metric controlling the \mathbf{L}^2 norm of the error as the choice of an \mathbf{L}^p norm could be essential in mesh

adaptation process regarding the type problem solved. For instance in CFD, physical phenomena can involve large scale variations (e.g. multi-scale phenomena, recirculation, and weak and strong shocks). Capturing weak phenomena is crucial for obtaining an accurate solution by taking into account all phenomena interactions in the main flow area. Intrinsically, metrics constructed with lower p norms are more sensitive to weaker variations of the solution whereas the L^∞ norm mainly concentrates on strong shocks.

4.3. Adaptive sonic boom numerical simulation

We consider a supersonic aircraft flying at a supersonic speed of 1.6 Mach with an angle of attack of 3 degrees at an altitude of 13,716 meters (45,000 feet). The aircraft geometry is the supersonic business jet geometry (SSBJ) of Dassault Aviation (Figure 2a). The length of the SSBJ is 36 meters. The complete aircraft is included in a sphere with a diameter of 1 kilometer, cf. Figure 2b.

As regards mesh adaptation, we choose to control the error on the Mach number, as the Mach number is really representative of the flow even if an accurate near field pressure distribution is required. A total number of 15 iterations of adaptation has been performed, each 150 time steps of the flow solver. We try to obtain the best mesh controlling the error in L^2 norm with 600 000 vertices.

The final adapted anisotropic mesh contains almost 570 000 vertices and 3.3 millions tetrahedra (complete geometry), views of this mesh are shown in Figure 3. We notice that mesh refinement along Mach cones have been propagated in the whole computational domain. Such meshes reduce significantly the numerical diffusion introduced by the flow solver. Consequently, the solution, e.g. Mach cones, is accurately

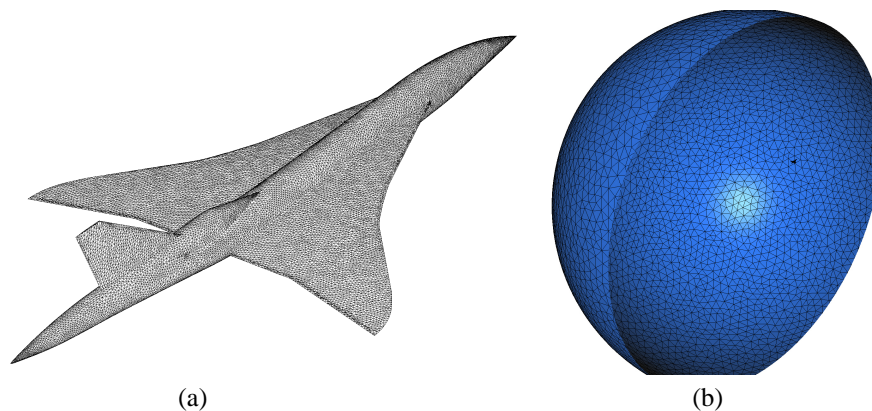


Figure 2. (a) SSBJ's geometry represented by its surface mesh and (b) spherical computational domain with the SSBJ

propagated in the whole domain, Figure 4. A further analysis shows that the near field signal is propagated accurately until $R/L = 6.5$, Figure 5. According to Whitham or linear supersonic theory a signal is accurately transported in the domain and has not been diffused if the quantity $\sqrt{\frac{R}{L} \frac{p(x) - p_\infty}{p_\infty}}$ for the two distances are close to each other. This is the case for signals between $R/L = 4$ and $R/L = 6.5$. After, the signal is slightly diffused but it is still well represented. This is clearly illustrated on sonic boom signatures where almost the same signal is obtained for R/L between 4 and 6, Figure 5.

In conclusion, for this geometry the signal seems to be converged, and thus the coupling is assume to be valid, for an R/L around 5 with less than 600,000 vertices.

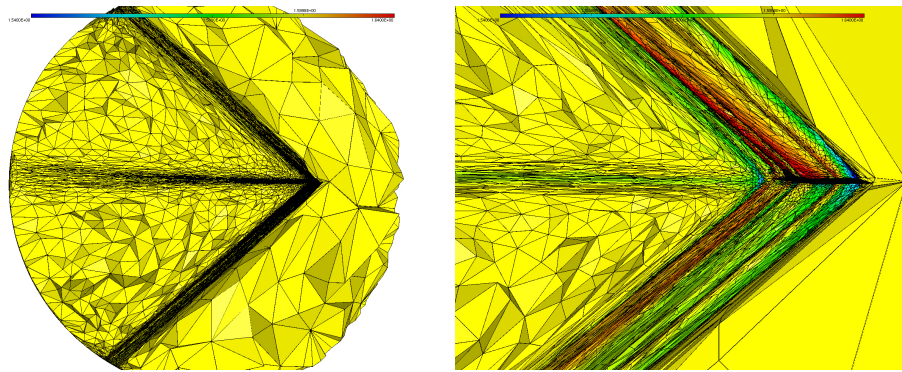


Figure 3. View of the final anisotropic mesh obtained with L^2 norm continuous metric-based mesh adaptation on the Mach number

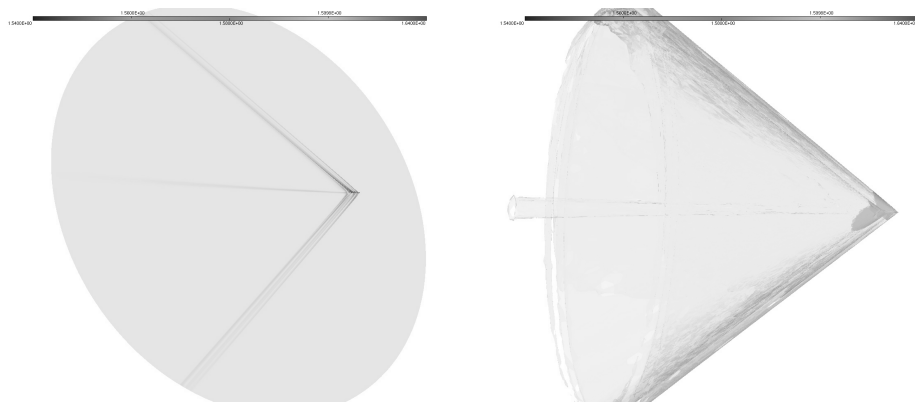


Figure 4. Final solution on the adapted mesh. Left, Mach number iso-value in the symmetry plane. Right, Mach number iso-surface representing Mach cones emitted by the aircraft

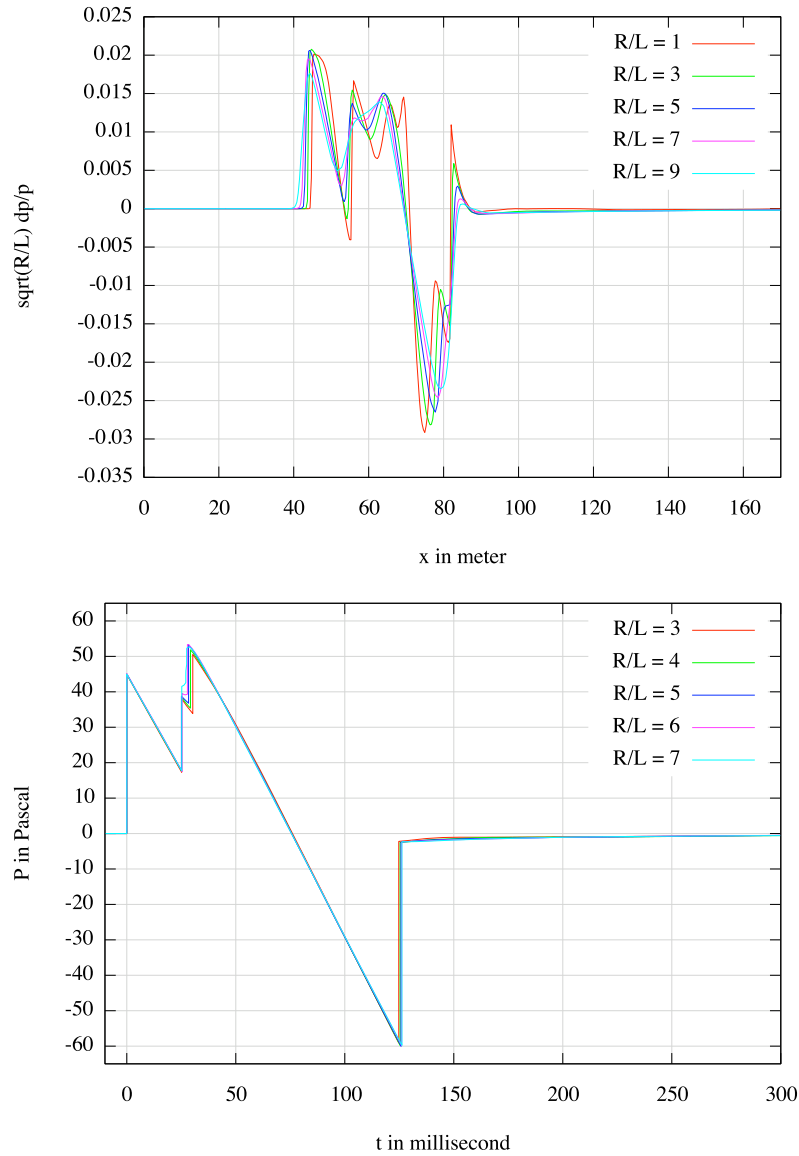


Figure 5. Top, near field signature extracted from several R/L . Bottom, sonic boom signature obtained from different R/L near field initializations. Only choosing R/L more than 5 produces a converged output

5. A numerical platform

Our optimization approach consists in translating first into a globally numerical model the following ingredients:

- a discrete shape definition,
- a CFD flow model,
- a functional.

To this end, we build discrete sensitivities, *i.e.*, we apply the Optimal Control theory directly to the discretized equations. The discrete gradients are computed by generating sensitivities by means of the Automatic Differentiation tool TAPENADE developed at INRIA (Hascoët *et al.*, 2004). It provides efficient and more accurate gradients evaluation than the ones obtained by finite differences methods. This approach always guarantees a descent direction whatever the mesh size compared to continuous gradient techniques where a descent direction is only guaranteed when the mesh size converges toward zero. A feature of the presented optimization process lies in the fact that CAD parametrization has been embedded in the loop. Therefore the total set $\mu = (l, \nu)$ of design variables in the subsequent equations has been split into two parts: l describes aerodynamic design variables such as angle of attack, side slip angle etc., whereas ν stands for geometric design variables handled directly by the CAD modeler. Schematically the CAD modeler can be represented by the following operator: $\nu \mapsto \mathbf{d}(\nu)$ where $\mathbf{d}(\nu)$ denotes the surface mesh displacement. The mathematical optimization problem can be formulated as an optimal control problem (Lions, 1971) as follows (Dinh *et al.*, 1996):

Find a shape μ^* within \mathcal{O} (set of shapes) such that:

$$\begin{cases} \mu^* = \arg \min_{\mu \in \mathcal{O}} j(\mu) \\ \mathbf{g}(\mu) \leq 0 \end{cases}$$

The cost function $j(\mu) = \mathbf{J}(\mu, \mathbf{W}(\mu))$ and constraints $\mathbf{g}(\mu) = \mathbf{G}(\mu, \mathbf{W}(\mu))$ depend on the set of design variables μ and on the solution vector $\mathbf{W}(\mu)$ of the state equations $\mathbf{E}(\mu, \mathbf{W}(\mu)) = \mathbf{0}$. By assembling cost and constraint functions in one vector, we can write: $\mathbf{f}(\mu) = (\mathbf{J}(\mu, \mathbf{W}(\mu)), \mathbf{G}(\mu, \mathbf{W}(\mu)))$. In order to solve the optimization problem at hand, the function $\mathbf{f}(\mu)$ must be evaluated for variations in the set of design variables denoted $\delta\mu$.

The mesh global deformation, resulting from the surface mesh deformation $\mathbf{d}(\nu)$, computed by the geometric modeler is contained within $\mathbf{D}(\nu)$ and is obtained by solving an elliptic problem through the operator $\mathbf{L}(\mathbf{d}(\nu), \mathbf{D}(\nu))$. Then, the vector f can be rewritten as: $\mathbf{f}(\mu) = \mathbf{F}(l, \mathbf{D}(\nu), \mathbf{W}(l, \mathbf{D}(\nu)))$.

The gradient computation consists in expressing the variations of cost and constraints functions with respect to $\delta\mu$:

$$\begin{aligned} \delta\mathbf{f} = & \frac{\partial\mathbf{F}(l, \mathbf{D}(\nu), \mathbf{W}(l, \mathbf{D}(\nu)))}{\partial l} \cdot \delta l + \frac{\partial\mathbf{F}(l, \mathbf{D}(\nu), \mathbf{W}(l, \mathbf{D}(\nu)))}{\partial\mathbf{D}(\nu)} \frac{d\mathbf{D}(\nu)}{d\nu} \cdot \delta\nu \\ & + \frac{\partial\mathbf{F}(l, \mathbf{D}(\nu), \mathbf{W}(l, \mathbf{D}(\nu)))}{\partial\mathbf{W}(l, \mathbf{D}(\nu))} \left(\frac{\partial\mathbf{W}(l, \mathbf{D}(\nu))}{\partial l} \cdot \delta l + \frac{\partial\mathbf{W}(l, \mathbf{D}(\nu))}{\partial\mathbf{D}(\nu)} \frac{d\mathbf{D}(\nu)}{d\nu} \cdot \delta\nu \right), \end{aligned} \quad [8]$$

using the state equations: $\mathbf{E}(l, \mathbf{D}(\nu), \mathbf{W}(l, \mathbf{D}(\nu))) = 0$ and the equation for the mesh deformation: $\mathbf{L}(\mathbf{d}(\nu), \mathbf{D}(\nu)) = \mathbf{0}$.

For gradient computation in direct mode the following linear system needs to be solved:

$$\begin{bmatrix} \frac{\partial\mathbf{E}}{\partial\mathbf{W}} \end{bmatrix} \begin{bmatrix} \frac{\partial\mathbf{W}}{\partial l}, \frac{\partial\mathbf{W}}{\partial\nu} \end{bmatrix} = - \begin{bmatrix} \frac{\partial\mathbf{E}}{\partial l}, \frac{\partial\mathbf{E}}{\partial\mathbf{D}} \frac{d\mathbf{D}}{d\nu} \end{bmatrix}. \quad [9]$$

The operator $\frac{\partial\mathbf{E}}{\partial\mathbf{D}}$ is obtained by differentiating the Euler flux with respect to the mesh coordinates. Writing these coordinates as $\mathbf{X}(\nu)$ and taking $\mathbf{X}_0 = \mathbf{X}(\mathbf{0})$ as the reference mesh coordinates, we have: $\mathbf{X}(\nu) = \mathbf{X}_0 + \mathbf{D}(\nu)$ and thus $\frac{\partial\mathbf{E}}{\partial\mathbf{D}} = \frac{\partial\mathbf{E}}{\partial\mathbf{X}}$.

Differentiation of the mesh deformation operator has been performed using the automatic differentiation tool in direct mode. Subsequently, the following linear system has to be solved:

$$\frac{\partial\mathbf{L}(\mathbf{d}(\nu), \mathbf{D}(\nu))}{\partial\mathbf{D}} \frac{d\mathbf{D}}{d\nu} = - \frac{\partial\mathbf{L}(\mathbf{d}(\nu), \mathbf{D}(\nu))}{\partial\mathbf{d}} \frac{d\mathbf{d}}{d\nu}. \quad [10]$$

The RHS of Equation [10] includes the term $\frac{d\mathbf{d}}{d\nu}$ provided by the geometric CAD modeler.

Referring to Equation [8], the gradients of cost and constraints functions $\mathbf{f}(\mu)$ are then obtained in direct mode for the Euler equations and mesh deformation by:

$$\begin{cases} \frac{d\mathbf{F}}{dl} = \frac{\partial\mathbf{F}}{\partial l} + \frac{\partial\mathbf{F}}{\partial\mathbf{W}} \frac{\partial\mathbf{W}}{\partial l} \\ \frac{d\mathbf{F}}{d\nu} = \frac{\partial\mathbf{F}}{\partial\mathbf{D}} \frac{d\mathbf{D}}{d\nu} + \frac{\partial\mathbf{F}}{\partial\mathbf{W}} \frac{\partial\mathbf{W}}{\partial\nu}. \end{cases} \quad [11]$$

In our problem, the number of design variables is much greater than the number of cost and constraint functions to be evaluated. In that case, regarding the number of linear systems to solve, it is advantageous to work with a system using two adjoint

variables. Let us first introduce an adjoint related to the state equation. One defines Ψ as the solution of:

$$\begin{bmatrix} \frac{\partial \mathbf{E}}{\partial \mathbf{W}} \end{bmatrix}^T \Psi = \begin{bmatrix} \frac{\partial \mathbf{F}}{\partial \mathbf{W}} \end{bmatrix}^T. \quad [12]$$

Next, an adjoint Φ related to the mesh deformation is computed by:

$$\begin{bmatrix} \frac{\partial \mathbf{L}}{\partial \mathbf{D}} \end{bmatrix}^T \Phi = \begin{bmatrix} \frac{\partial \mathbf{F}}{\partial \mathbf{D}} \end{bmatrix}^T - \begin{bmatrix} \frac{\partial \mathbf{E}}{\partial \mathbf{D}} \end{bmatrix}^T \Psi. \quad [13]$$

The variations of cost and constraints functions are then computed using $\delta f = \delta f - \Psi^T \delta \mathbf{E} - \Phi^T \delta \mathbf{L}$:

$$\left\{ \begin{array}{l} \delta f = \frac{\partial \mathbf{F}}{\partial l} \cdot \delta l + \frac{\partial \mathbf{F}}{\partial \mathbf{D}} \frac{d\mathbf{D}}{d\nu} \cdot \delta \nu \\ + \frac{\partial \mathbf{F}}{\partial \mathbf{W}} \frac{\partial \mathbf{W}}{\partial l} \cdot \delta l + \frac{\partial \mathbf{F}}{\partial \mathbf{W}} \frac{\partial \mathbf{W}}{\partial \mathbf{D}} \frac{d\mathbf{D}}{d\nu} \cdot \delta \nu \\ - \Psi^T \frac{\partial \mathbf{E}}{\partial l} \cdot \delta l - \Psi^T \frac{\partial \mathbf{E}}{\partial \mathbf{D}} \frac{d\mathbf{D}}{d\nu} \cdot \delta \nu \\ - \Psi^T \frac{\partial \mathbf{E}}{\partial \mathbf{W}} \frac{\partial \mathbf{W}}{\partial l} \cdot \delta l - \Psi^T \frac{\partial \mathbf{E}}{\partial \mathbf{W}} \frac{\partial \mathbf{W}}{\partial \mathbf{D}} \frac{d\mathbf{D}}{d\nu} \cdot \delta \nu \\ - \Phi^T \frac{\partial \mathbf{L}}{\partial \mathbf{D}} \frac{d\mathbf{D}}{d\nu} \cdot \delta \nu - \Phi^T \frac{\partial \mathbf{L}}{\partial \mathbf{d}} \frac{d\mathbf{d}}{d\nu} \cdot \delta \nu. \end{array} \right. \quad [14]$$

Equation [14] allows us to calculate the gradients in adjoint mode for the state equation and mesh deformation as:

$$\left\{ \begin{array}{l} \frac{d\mathbf{F}}{dl} = \frac{\partial \mathbf{F}}{\partial l} - \Psi^T \frac{\partial \mathbf{E}}{\partial l} \\ \frac{d\mathbf{F}}{d\nu} = -\Phi^T \frac{\partial \mathbf{L}}{\partial \mathbf{d}} \frac{d\mathbf{d}}{d\nu}. \end{array} \right. \quad [15]$$

The gradients of cost and constraints functions $f(\mu)$ have been obtained by solving the systems [12] and [13] where the number of RHS terms does not depend on the number of design variables but is equal to the number of cost and constraints functions. For this reason the adjoint approach is used in the majority of industrial optimization problems.

5.1. Optimization platform

The optimization platform couples the following modules:

- CAD modeler,
- Volume mesh deformation,
- CFD solver (Bastin *et al.*, 1999),
- adjoint of CFD solver (Dinh *et al.*, 1996),
- adjoint of volume-mesh deformation,
- cost and gradient,
- optimizer.

The optimizer has a crucial contribution in the optimization process. Indeed, this tool will drive the whole process by analyzing different values of the cost function and the related constraints and their sensibility with respect to the design parameters. The outputs of the optimizer are a new set of parameter values for which it is necessary to provide: the cost function and the associated constraints along with their derivatives. In the current study, we use the optimizer developed at Dassault Aviation. This code is based on gradient evaluations which should be calculated either by finite differences or by adjoint formulation as explained later. The different available optimizers are Broyden-Fletcher-Goldfarb-Shanno (BFGS) for unconstrained cases and the Method of Feasible Directions, an Interior Point Algorithm (Herskovits *et al.*, 1996), or Sequential Quadratic Programming for constrained cases.

5.2. CAD modeler

The CAD modeler named GANIMEDE (Geometry AND Inherent MESH DEformation) handles both local and global design variables. By local design variables at control point, we mean position, tangent and curvature values. Global design variables redefine several control points enabling to modify characteristics such as thickness, twist and camber of wing sections.

The geometry itself is decomposed by its definition in a hierarchical way. Geometric entities are defined by a set of patches specified by the user. These entities define a set of sections which in turn are defined by a number of control points. Global design variables can act at section or entity level of the hierarchical model whereas local design variables can act at all levels. Therefore, a variable can control a more or less larger part of the aircraft and thereby allowing for a flexible parametrization of the geometry. The decomposition in entities and sections of a CAD model is shown in Figure 6. Surface patches define a polynomial function of degree N_{deg} in the $u - v$ space. For a given set of design variables ν this can be written as:

$$\mathbf{S}(u, v, \nu) = \sum_{(i,j) \in \{0, \dots, N_{deg}\}^2} u^{N_{deg}-i} v^{N_{deg}-j} c_{i,j},$$

where $(u, v) \in [0, 1]^2$ and $(c_{i,j})_{i,j}$ represent the coefficients of the polynomial function. The different geometric entities specified by the user might intersect and the design variables might have an impact on the intersection. The geometric modeler updates the new intersection by computing:

$$\sum_{(i,j) \in \{0, \dots, N_{deg}\}^2} u^{N_{deg}-i} v^{N_{deg}-j} c_{i,j} = \sum_{(i,j) \in \{0, \dots, N_{deg}\}^2} \sigma^{N_{deg}-i} \tau^{N_{deg}-j} \rho_{i,j},$$

where (u, v) and (σ, τ) are the parameters and $(c_{i,j})$ and $(\rho_{i,j})$ the polynomial coefficients of the different patches involved in the intersecting entities.

As the geometry is modified during the optimization process, its corresponding new surface mesh has to be regenerated. The CAD modeler also copes with this aspect as was mentioned previously. To this end, a connectivity is created by projection of the initial surface mesh on the initial geometry. Then, the geometry CAD model is deformed, *i.e.*, a new geometry is created, according to design parameters. Finally, from the initial surface mesh and the new CAD geometry, a new iso-topologic surface mesh is generated.

5.3. Volume deformation

As mentioned previously the CAD modeler can be schematically represented by the operator: $\nu \mapsto \mathbf{d}(\nu)$ where $\mathbf{d}(\nu)$ represents the surface mesh displacement resulting from the geometric design variables ν . This surface mesh displacement is then propagated into the volume mesh by an elliptic operator: $\mathbf{L}(\mathbf{d}(\nu), \mathbf{D}(\nu))$, where $\mathbf{D}(\nu)$ designates the resulting displacement field throughout the volume mesh. The chosen operator is an elliptic operator, more precisely:

$$\begin{cases} -\nabla \cdot (\kappa \nabla \mathbf{D}(\nu)) & = 0 & \text{in } \Omega \\ \mathbf{D}(\nu)|_{\Gamma_c} & = \mathbf{d}(\nu) \\ \mathbf{D}(\nu)|_{\partial\Omega \setminus \Gamma_c} & = 0, \end{cases} \quad [16]$$

where Ω is the computational domain, Γ_c is the shape surface boundary, and κ a local coefficient related to the size (volume) of the local tetrahedral element. As the operator is symmetric and positive semi-definite, the linear system is solved by incomplete Cholesky preconditioning in addition to a preconditioned conjugate gradient (PCG) method. In order to improve robustness of the mesh deformation, the surface mesh displacement can be performed in several steps. At each step, the surface displacement is propagated into the surrounding volume mesh. This process has been parallelized by splitting up the computational domain in several domains and using the MPI library as communication protocol between processors.

5.4. Cost and constraints

In the current optimization process, available cost and constraints functions are based on the six global aerodynamic coefficients (pressure drag, lift, pitching mo-

ment...), on the wing span-wise lift distribution or on the local pressure distribution on the aircraft. Many other functions have been developed to address other problems than sonic boom reduction (like inlet design, engine integration...). Due to the difficulty to differentiate an accurate wave propagation tool, our choice is to develop a functional based on near field deviation of pressure fluctuation to a target one.

Let line $[l_1^\theta, l_2^\theta]$ be the extraction line at distance R characterized by the azimuthal angle θ , see Figure 7. We define the cost function related to angle θ as

$$J_\theta(W) = \frac{1}{2} \int_{l_1^\theta}^{l_2^\theta} [dp_\theta(W) - dp_\theta^{target}(W)]^2 dl, \quad [17]$$

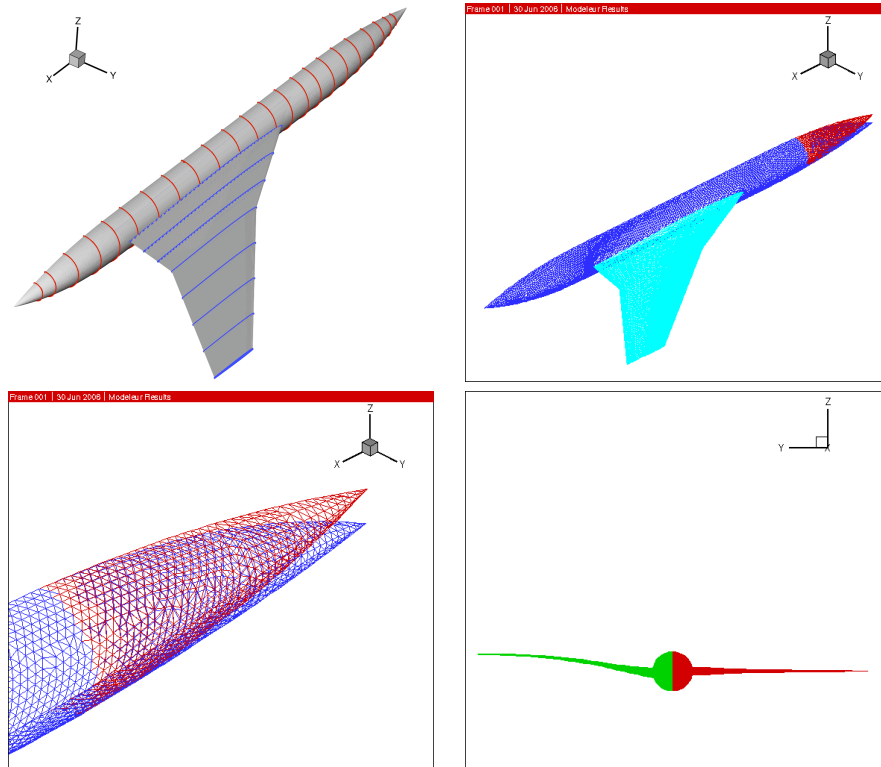


Figure 6. Aircraft simplified geometry for sonic boom reduction. Top left, decomposition of geometry CAD model. Top right and bottom, comparison between baseline and optimized geometries

with $dp(W) = (p(W) - p_\infty) / p_\infty$. If we have to deal with a list of angles $(\theta_i)_{i=1,N}$, we propose to realize a linear combination of cost functions $(f_{\theta_i})_{i=1,N}$. Thus, the global cost functional becomes:

$$J(W) = \sum_{i=1,N} \omega_i J_{\theta_i}(W),$$

with $(\omega_i)_{i=1,N}$ weights associated to each angle θ_i .

5.5. Sensitivity development and optimizer

The TAPENADE automatic differentiation software acts by transformation of the initial code. Having a given set of input variables and a group of programs written in Fortran 77 evaluating the numerical function f , TAPENADE generates sub-programs computing its derivatives with respect to those variables. This software allows us to use direct or reverse mode of automatic differentiation. Reverse mode is applied to get routines computing adjoint residual and functional gradient. See (Hascoët *et al.*, 2004) for further details.

5.6. A numerical example

5.6.1. Shape optimization

Sonic boom reduction is addressed with the near field target pressure method. Then, the platform is applied to the inverse problem of finding the shape for which the flow matches at best the target near field pressure while maintaining the lift coefficient equal to the baseline value and keeping drag coefficient lower or equal to the baseline value. The cost functional is defined as the mean square of deviation between current and target pressure on an horizontal plane at $R/L = 0.5$. The baseline geometry is a wing-body configuration involving a generic fuselage (cone-cylinder-cone) and a high-sweep wing, Figure 6. In addition to flow constraints, the following geometrical constraints are applied: length and wing thickness are kept to baseline values, and cabin has to contain a certain volume for passengers, *i.e.*, its section for a fixed x -interval is larger than the baseline section. The shape parameters involve 55 CAD fuselage parameters and 11 wing parameters. The 67th parameter is the angle of attack. A volume mesh containing 163, 459 vertices (half geometry) is used to compute the flow.

Figure 8 shows the evolution of cost function and constraints as functions of optimizer iterations, each iteration of the interior point algorithm involving one cost evaluation and one gradient evaluation. The cost function is divided by 2 in 40 iterations. The two main shape modifications concern the nose tilting and the wing bending, Figure 6. The effect of shape modifications on shocks emitted from the aircraft are clearly illustrated in Figure 7 where iso-values of pressure are represented in the symmetry

plane. The consequence of the optimization on the sonic boom signature at ground, as given by the propagation post treatment, is a reduction almost 20 % (7 Pascal) of the initial pressure rise, see Figure 8. We observe a smaller shock focalization rather than a weaker shock system. Maximal pressure is lower and minimal pressure is higher.

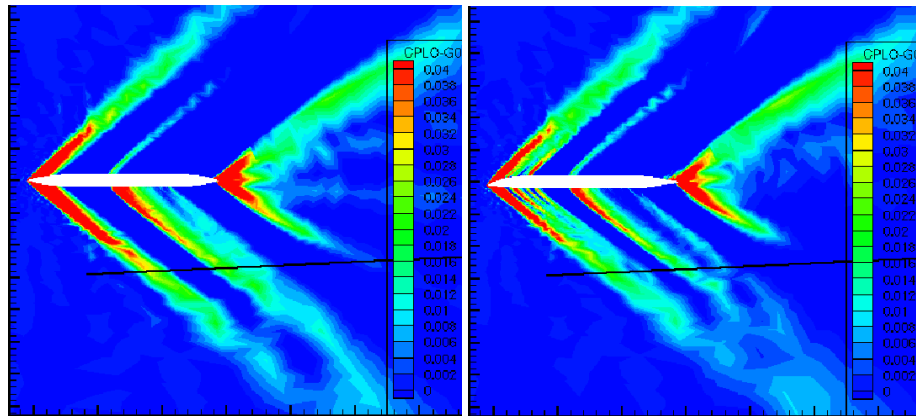


Figure 7. Sonic boom: view of near field pressure. Left, initial shape and right, optimized shape. The line under the aircraft is the pressure extraction line to evaluate the cost function

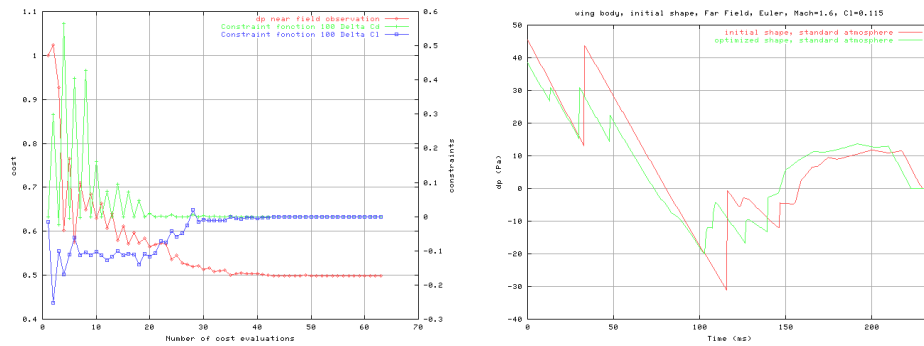


Figure 8. Right, functional and constraints evolution during minimization. Right, sonic boom reduction at ground level

5.6.2. Validation with mesh adaptation

In the previous section, automatic shape optimization has been carried out on a relatively coarse non-adapted mesh. The pressure signal used in the cost function definition has been extracted close to the aircraft at $R/L = 0.5$, this in despite of our remarks of Section 4.3 showing that larger R/L are required to have an accurate coupling between the CFD computation and the propagation. Our choice was motivated by the need to limit the effect the numerical dissipation on such meshes. If the extraction is performed too far from the jet the discrete pressure signal is damped and too weak for being useful. After this necessary compromise, we now want to validate gains obtained with the optimization process on the coarse mesh by studying finely the flow associated with shapes before and after optimization.

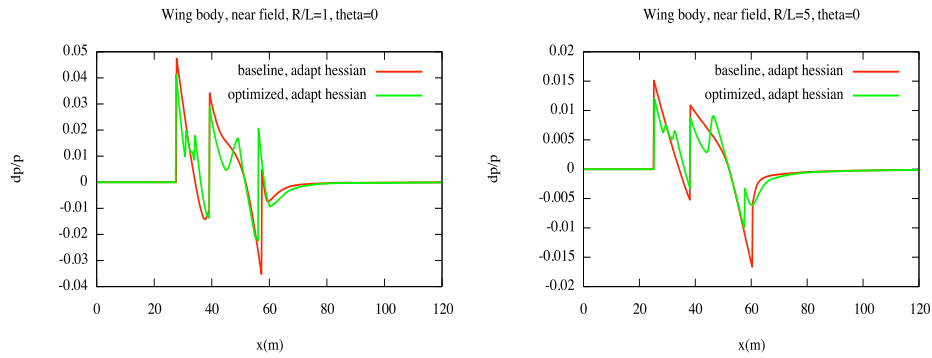


Figure 9. Near field pressure distributions at $R/L = 1$ (left) and $R/L = 5$ (right) for the initial and the optimized shapes obtained with adaptive simulations

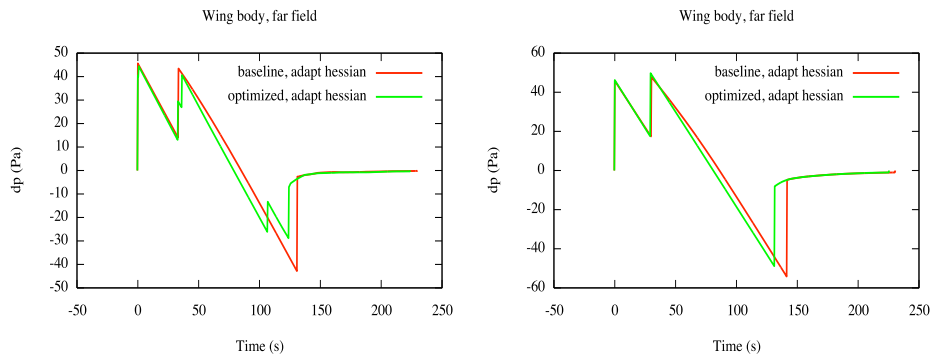


Figure 10. Sonic boom signatures propagated from pressure signals, obtained with adaptive simulations, at $R/L = 1$ (left) and $R/L = 5$ (right) for the initial and the optimized shapes

To this end, accurate adaptive simulations are performed on both shapes to get a reliable pressure signal far from the aircraft, as described in Section 4. The parameters of Section 4.3 have been chosen for the adaptive simulations. Only a slight modification of the angle of attack has been done in order to have the same lift coefficient for both simulations. For each simulation, an accurate signal is obtained until $R/L = 5$. Near field signals at $R/L = 1$ and $R/L = 5$ are presented Figure 9 and the associated sonic boom signatures are depicted in Figure 10. Figure 9 clearly illustrates the impact of the optimized shape on the near field pressure distribution. With this new shape, all shocks are splitted into several ones with a reduced intensity. Nevertheless, if the propagation from the signals at $R/L = 1$ shows the same tendency with a reduced impact as compared to the results obtained at $R/L = 0.5$ on the coarse mesh, the results of the propagation from $R/L = 5$ are more disappointing. In this case, both shapes almost produce the same sonic boom signature.

6. Coupling mesh adaptation and shape optimization

In the previous section, the objective functional has been decreased on a fixed non-adapted mesh. We consider now the research of an optimum in combination with the mesh adaptation algorithm. In other words, we want to get a shape that is optimal when the objective function is evaluated on a mesh that is strongly adapted to the optimal flow. As the optimal shape is not known, the associated adapted mesh cannot be generated in advance. Adapted mesh has to be constructed at the same time we optimize. As a consequence, we cannot define a stand alone *discrete* optimization problem as done in the previous section. Instead of minimizing a discrete functional, we shall approximatively solve the *continuous* optimality conditions with a mesh adaptive algorithm. This will be done by *relaxing* between the two following steps:

- minimizing the functional for a fixed mesh,
- generating an adapted mesh in a fixed domain.

6.1. Minimizing for a fixed mesh

Our option is to use an exact gradient approach in order to keep a reliable descent direction. Then, the following sequence is applied with a fixed mesh:

Gradient and line search

- compute the flow (state equation),
- compute the adjoint state,
- compute the (exact) gradient of functional,
- descent in the direction opposite to gradient.

In order to keep an invariant mesh during that sequence, the shape variation is performed without changing the computational domain, thanks to transpiration boundary conditions.

6.2. Mesh adaptation for a fixed domain

For sonic boom calculations, a mesh that is accurately adapted to a flow maybe inaccurate when used for computing a slightly different flow. Starting a line search with a mesh adapted to the first flow may then result in poor evaluation of the other flows and a erroneous evaluation of the effect of shape update. To avoid this, we propose a specific algorithm based on the fixed point mesh adaptation method introduced in (Alauzet *et al.*, 2007). This algorithm enables us to generate for each step number k of the gradient loop a unique mesh adapted to the different flows computed the descent step. More precisely, in the fixed point mesh adaptation/gradient loop, the mesh is adapted to the k -th gradient as well as to the descent step by taking into account all solutions computed throughout this step:

- to each flow corresponds an optimal metric,
- the *intersection* of all these metrics is computed,
- the adapted mesh is generated from the resulting intersected metric.

Notice that the adapted mesh cannot be generated before the evaluation of the flow variables. It means that an implicit coupling needs to be applied. We address this issue with an iterative process which aims at converging to a fixed point:

Fixed point adaptation/gradient step

- 1) choose an initial mesh,
- 2) compute on current mesh the flow (state equation),
- 3) compute on current mesh the adjoint state,
- 4) compute on current mesh the (exact) gradient of functional,
- 5) perform on current mesh line search in the descent direction,
- 6) compute the intersection of metrics for all flows in steps 2-5, and generate a new mesh specified by the new metric,
- 7) if process is not converged, go to 2.

The process is considered converged in step 7 as soon as the difference between two consecutive metrics is small. In practice, this fixed point loop iterates about five times to converge. Between each remeshing, the computed solution on the previous mesh (step 2) is transferred on the new adapted mesh to reduce computing expenses. The fixed point adaptation/gradient step is then itself included in the gradient loop. It is necessary to initially set the desired accuracy for the solution in the fixed point adaptation process to have a well-posed problem with respect to the metric. The num-

ber of nodes and the solution may change but the prescribe size is constant along the process to keep the accuracy of the solution at a required level.

6.3. Application to problem under study

Preliminary optimization computations have been applied to an HISAC test case. The shape is optimized without aerodynamic constraints and the shape parametrization is CAD-free and shape constraints are not applied. An adapted mesh sample used during the adaptive optimization platform is shown in Figure 11a. Figure 11b depicts 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. Figure 12 points out the reduction obtained with the adaptive optimization platform for the near field pressure signal at $R/L = 1$ and the sonic boom signature after propagation.

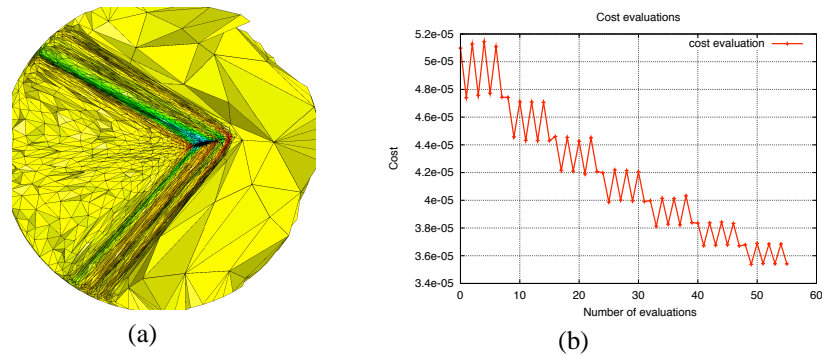


Figure 11. Left, adapted mesh sample used adaptive optimization platform. Right, cost reduction during the coupled mesh adaptation/shape optimization iterations

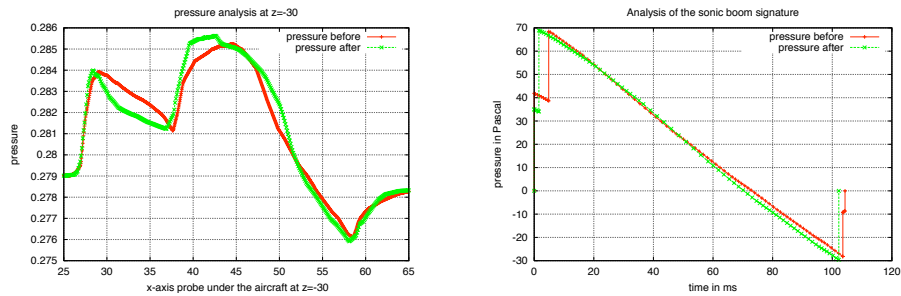


Figure 12. Near field pressure distributions at $R/L = 1$ and associated sonic boom signatures for the initial and the optimized shapes (adaptive optimization platform)

7. Conclusions

In this last chapter, a particular attention is paid to the accurate treatment of sonic boom reduction. Sonic boom *prediction* is subject to accuracy constraints due to the necessary matching between an accurate nearfield 3D calculation and an accurate propagation model. These accuracy constraints imply that (1) the propagation is accurate, which assumes that the matching is far enough from the aircraft, and (2) the nearfield is accurate, that can be obtained by applying a strong anisotropic mesh adaptation. For sonic boom *reduction*, we need to combine the above prediction suite, with its delicate tuning, with the optimal control loop. We have explored different strategies which avoid involving the whole prediction process inside the optimization loop. We describe the adaptation of an industrial platform involving a CAD parametrization and an adjoint-based sensitivity obtained with the help of an Automatic Differentiation tool. Only the nearfield part of the prediction process is introduced in this platform, without mesh adaptation. Then an *a posteriori* verification can be applied. But it can be disappointing in some cases. We have also proposed a method for the installation of the mesh adaptation process inside the optimization loop. The new tool seems more reliable but it is still under study.

Acknowledgments

These works were partly supported by HISAC project (516132) launched by the EC following to the Aero 2nd call of the 6th Framework Programme

8. References

- Alauzet F., Frey P., George P.-L., Mohammadi B., “ 3D transient fixed point mesh adaptation for time dependent problems. Application to CFD simulations ”, *J. Comp. Phys.*, vol. 222, p. 592-623, 2007.
- Alauzet F., Loseille A., Dervieux A., Frey P., “ Multi-dimensional continuous metric for mesh adaptation ”, *Proc. of 15th Int. Meshing Roundtable*, Springer, p. 191-214, 2006.
- Alexandrov N., “ Multilevel methods for MDO ”, *Multidisciplinary Design Optimization: State of the Art*, SIAM Publications, Philadelphia, PA, p. 79-89, 1997.
- Alonso J., Kroo I., Jameson A., “ Advanced Algorithms For Design And Optimization Of Quiet Supersonic Platforms ”, *AIAA Paper*, vol. 2002-0144, p. 1-13, 2002.
- Bastin J., Rogé G., “ A multidimensional fluctuation splitting scheme for three dimensional Euler equations ”, *European Journal of Finite Elements*, vol. 33, n° 6, p. 1241-1259, 1999.
- Becker R., Kapp H., Rannacher R., “ Adaptive Finite Element Methods for Optimal Control of Partial Differential Equations: Basic Concept ”, *SIAM Journal on Control and Optimization*, vol. 39, n° 1, p. 113-132, 2001.
- Dinh Q., Rogé G., Sevin C., Stofflet B., “ Shape Optimisation in Computational Fluid Dynamics ”, *European Journal of Finite Elements*, vol. 5, p. 569-594, 1996.
- Farhat C., Maute K., Argrow B., Nikbay M., “ A Shape Optimization Methodology For Reducing The Sonic Boom Initial Pressure Rise ”, *AIAA Paper*, vol. 2002-0145, p. 1-11, 2002.

- Frey P., Alauzet F., “ Anisotropic mesh adaptation for CFD computations ”, *Comput. Methods Appl. Mech. Engrg.*, vol. 194, n° 48-49, p. 5068-5082, 2005.
- Frey P., George P.-L., *Mesh generation. Application to finite elements*, Hermès Science, Paris, Oxford, 2000.
- Giles M., Pierce N., “ Analytic adjoint solutions for the quasi-one-dimensional euler equations ”, *Journal of Fluid Mechanics*, vol. 3, n° 426, p. 327-345, 2001.
- Hascoët L., Pascual V., TAPENADE 2.1 user’s guide, RT-0300, INRIA, 2004.
- Herskovits J., Lapporte E., Tallec P. L., Santos G., “ A Quasi-Newton Interior Point Algorithm Applied for Constrained Optimum Design in Computational Fluid Dynamics ”, *European Journal of Finite Elements*, vol. 5, p. 595-617, 1996.
- ICAO, *Manual of the ICAO Standard Atmosphere (extended to 80 kilometres)*, Doc 7488, Third edn, 1993.
- Koobus B., Hascoët L., Alauzet F., Loseille A., Mesri Y., Dervieux A., “ Continuous mesh adaptation models for CFD ”, *CFD Journal*, 2007. 12:3.
- Leservoisier D., George P.-L., Dervieux A., Métrique continue et optimisation de maillage, RR-4172, INRIA, April, 2001. (in French).
- Lions J.-L., *Optimal Control of systems governed by Partial Differential Equations*, Springer-Verlag, New York, 1971.
- Polak E., *Optimization: Algorithms and Consistent Approximation*, n° 124 in *Applied Mathematical Sciences*, Springer-Verlag, Berlin-New-York, 1997.
- Sobieszcanski-Sobieski J., Haftka R., “ Multidisciplinary aerospace design optimization: Survey of recent developments ”, *Structural Optimization*, vol. 14, n° 1, p. 1-23, 1997.
- Thomas C., Extrapolation of sonic boom pressure signatures by the waveform parameter method, TN. n° D-6832, NASA, 1972.
- Venditti A., Darmofal D., “ Grid Adaptation for Functional Outputs: Application to Two-Dimensional Inviscid Flows ”, *J. Comput. Physics*, vol. 176, p. 40-69, 2002.
- Zhu J., Zienkiewicz O., “ A Posteriori error estimation and three-dimensional automatic mesh generation ”, *Finite Elements in Analysis and Design*, vol. 25, p. 167 - 184, 1997.