Additive multilevel optimization and its application to sonic boom reduction.

A. Dervieux*, F. Courty*, M. Vázquez* and B. Koobus**

*INRIA - Sophia Antipolis, 2004 Route des Lucioles, BP. 93, 06902 Sophia Antipolis
**Univ Montpellier II, Dept Mathématiques, CC.051, 34095 MONTPELLIER Cedex 5

Abstract

New theory, methods, and tools for optimization in a finite dimension space has known much evolution in the last 20 years. Quadratic convergence is the rule, thanks, for example, to Sequential Quadratic Programming algorithms equipped with quasi-Newton mechanisms such as the BFGS one. However, some new applications, generally coming from the discretisation of PDE's, involve a very large number of parameters. It is then necessary to design some PDE-inspired preconditioners. Several theories can help us in this direction, such as the Bramble-Pasciak-Xu one, or its wavelets analogs. We discuss the building of a multi-level preconditioner on an unstructured mesh connected with these ideas. As an application, the optimal shape design of a supersonic jet is considered. We have started a prospective investigation of the shapes that would produce a very low level of sonic boom. Instead of parameterizing the shape with a small macroscopical quantities, we consider as parameters the whole set of wing and body skin mesh (up to 12000 parameters for this particular case).

Key words : Computational fluid dynamics - Compressible flows - Shape optimal design - Sonic boom optimization

1 Functional optimization

The progress of optimization in the last two decades has completely renewed the panoply of optimisers. To restrict to differentiable optimization, SQP algorithms have replaced gradient ones, thanks to the integration of very efficient quasi-Newton strategies. But as the whole new theory is built in $\mathbb{R}^n$, it will miss important informations when applied to large scale problems coming from Partial Differential Equations.

To illustrate our affirmation, we consider the best-seller problem of the calculus of variation:

$$\min_u \quad 1/2 \int |\nabla u|^2 dx - \int u \ f \ dx \ . \tag{1}$$

In most quasi-Newton optimization methods, the basic brick for the corrections applied to the unknown will be the gradient of the functional:

$$g = \Delta u - f.$$

This correction has the important disadvantage to be a function much less regular that the previous iterate u. Indeed, if u has continuous derivatives up to k-th order, it is true only up to $k-2$-th for g. To pursue the functional analysis, the iterative process of an algorithm relying on g would produce a new u with only $k-2$-th continuous derivatives, then $k-4$-th ones and so on. Ultimately, it will be impossible to compute the functional to minimize.

It is well known that, after discretization, using g for building corrections is equivalent to apply a Jacobi iteration. This will amplify many high frequency modes, unless the step length is reduced in order to satisfy a Courant-like stability condition. In this case, if discretization is fine and high frequency mode very numerous, the quasi-Newton process cannot compensate the difficulties in converging. This is why quasi-Newton solvers for PDE are generally associated with preconditioners.

We note in passing the strong relation between preconditioners and smoothers in this context. In the $\mathbb{R}^n$ theory, a preconditioner is used basically to improve the condition number of a linear system, that is to reduce the ratio between the largest and the smallest eigenvalue. In the discrete PDE case, ill-conditioning comes from mesh dependent high frequency eigenvalues, that increases with mesh fineness and with the degree of differential operator. Then two ways to reduce this is either to use a coarse mesh, not an acceptable solution, or to apply special devices for filtering high frequencies. In the functional context, it seems necessary (and somewhat equivalent to the above strategies) to work with an equation in which regularity is not lost, for example by multiplying it by an inverse Laplace operator, when the order of derivatives should be recover.

The functional context can then be iterated in a smooth manner, and will converge with its own convergence speed. An important consequence is that a good discretization of this iteration should ideally have a convergence rate close to the functional one, that is a mesh-size independent rate. Then building a preconditioner that renders the functional iteration regular increases the potential qualities of the discretized iteration.

Further, in the case of quasi-Newton iterations, an additional functional mechanism has to be taken into account. By recovering all necessary smoothness, we can obtain that the underlying operator is compact, which means that its spectrum has only zero as an accumulation point. In the discrete case, highest eigenvalues correspond to low frequencies and are solved by the quasi-Newton process, which results in a condition number approaching progressively unity (superlinear convergence).

To sum up, the principle we shall follow is to introduce a preconditioner or smoother inspired by functional properties in order to improve the optimization iteration of our optimal shape design problem.

The paper is organized as follows: a first section presents the additive multi-level method. A second section introduces the features of the flow optimization problem under study. The next section presents numerical examples and the last one is the conclusion.

2 Additive multilevel preconditioner

2.1 Theoretical background

We focus first on the solution of a generic elliptic problem written:

$$(Au, v) = (f, v) \forall u, v \in V_k \ , \ f given in V'. \tag{2}$$

According to the work of Bramble, Pasciak and Xu (BPX from now on), good smoothers or preconditioners can be derived from a sequence of discretization spaces:
Let $(V_k)_{1 \leq k \leq n}$ be a hierarchy of subspaces of V :

$$V_1 \subset \cdots \subset V_k \subset \cdots \subset V_n \subset V$$

In the BPX theory, $V_n = V$ but it will be useful for us not to assume this and to consider that V is the continuous functional space. For each $k = 1, .., n$ we introduce the operators $Q_k : V \longrightarrow V_k$ defined for all $u \in V, v \in V_k$ by

$$(Q_k u, v) = (u, v)$$

and the operators $A_k : V_k \longrightarrow V_k$ defined by

$$(A_k u, v) = A(u, v) \forall u, v \in V_k$$

The BPX preconditioner writes:

$$C \ = \ A_1^{-1} Q_1 \ + \ \sum_{k=2}^{n} \lambda_k^{-1} Q_k$$

where $\lambda_k = \rho(A_k)$, is the spectral radius of A_k.

The knowledge of A or even of A_1^{-1} is in fact not necessary and the above preconditioner can be replaced by a wavelet-type preconditioner:

$$C_w \ = \sum_{k=1}^{n} \mu_k^{-1} (Q_k - Q_{k-1}) \tag{3}$$

with $Q_0 = 0$. The coefficients μ_k are then choosen in order to damp high frequencies.

In [11] , the above method is reconsidered from the standpoint of a general non-quadratic optimization problem to be solved with a gradient method in an unstructured fine mesh. A crucial relation between the coarse and fine discrete

problems and the continuous one was expressed and presented as an "approximation condition" (according to the multi-grid theory). It can be expressed in short as follows:
"Minimum problems on coarse subspace are consistent approximations of the continous minimum problem".
For this, assuming that the functional J is continuous from V to $\mathbb{R}$, the main assumption is that for any k the union of $V_{k,h}$ when the fine mesh step h is converged to zero are dense into V.

This condition takes into account the continuity property of the cost functional, but not its differentiability ones, which can be experssed in a different functional space in the case of complex application such as variations of a flow variable when the geometric domain of it is modified.

We shall then go further by analysis of the functional properties of the preconditioner (3). To do this we need first to replace it by its functional limit, obtained by taking the sum of the infinite series.

$$P = \sum_{k=1}^{\infty} \mu_k^{-1}(Q_k - Q_{k-1}) \tag{4}$$

In the case where V is the Sobolev space H^1 and under adequate design assumptions, our preconditioner will be a functional smoother, satisfying typically the following:

Lemma: P maps H^{-1} into $H^{-1+\alpha}$.

Since P will multiply the gradient of the cost functional, we can imagine that,as soon as α is chosen large enough, it will allow a smooth enough correction of the control variable and favourize a better convergence of the optimization process. We refer to [4] for the details of this analysis.

2.2 The multidimensional case

We consider the minimization problem :

$$\text{Find} \quad \bar{u} \in V \quad \text{such that} \quad \bar{u} = \arg\min_{u \in V} j(u) \tag{5}$$

where V is a Hilbert space (think of a Sobolev space) and where the cost functional j is continuously differentiable in the Hilbert space V. We investigate the application of the multilevel optimization to Problem (5) in order to put in evidence the basic options that will allow to obtain a multilevel solution in an efficient way. Let us write a *coarse level correction* as follows:

$$u_{2h}^{n+1} = u_{2h}^{n} - \rho_{opt}\ \mathcal{L}\mathcal{P}\mathcal{P}^*\mathcal{L}^*\ j'(u_{2h}^{n}). \tag{6}$$

In the above expression, the linear operators $\mathcal{L}$ and $\mathcal{P}$ are projections to a smaller space that are defined below; u_{2h} is the discretized value of u on the coarse mesh. A key condition for efficiency is that the fixed point u_{2h}^* of (6) do satisfy $\mathcal{L}\mathcal{P}\mathcal{P}^*\mathcal{L}^*\ j'(u_{2h}^*) = 0$, i.e. it is a *convergent approximation* of "arg min j" when the mesh size is increased; in other words:

$$u_{2h}^* \quad \longrightarrow \quad \bar{u} \text{ in } V \quad \text{as} \quad h \to 0 \tag{7}$$

In (6), $\mathcal{P}$ is a prolongation operator from coarse level to fine level and its transpose $\mathcal{P}^*$ is a restriction operator from fine level to coarse level. $\mathcal{L}$ is an average smoothing operator ($\mathcal{L}^*$ is its transpose) defined by :

$$(\mathcal{L}u)_i = (1-\theta)u_i + \theta\ \frac{\displaystyle\sum_{j\in\mathcal{V}(i)\cup\{i\}} Area(j)\ u_j}{\displaystyle\sum_{j\in\mathcal{V}(i)\cup\{i\}} Area(j)} \tag{8}$$

where j runs over the $\mathcal{V}(i)$ cells neighboring i and $Area(j)$ is the area of each of them built around vertices with the triangle medians.

2.3 Parametrization of 3D surfaces

We now consider the parametrization for optimizing an aircraft in a 3D Euler flow. The parametrized shape is then a 3D surface and flow calculations are performed on an unstructured 3D mesh. The building of a multilevel parametrization [1] of this shape will rely on a node-agglomeration principle (see [10]). The surface is assimilated to a manifold Σ, that is smooth enough. A deformation produced on its discretization Σ_h, is noted $\delta\Sigma_h$; the new manifold $\Sigma_h + \mathcal{L}\mathcal{P}\mathcal{P}^*\mathcal{L}^*\delta\Sigma_h$ is built by a projection $\mathcal{P}^*$ to a coarser level, a prolongation $\mathcal{P}$, transpose of $\mathcal{P}^*$, to the initial level, combined with an operator $\mathcal{L}$ (details are given in [10]). The smoothing operator $\mathcal{L}$ is an average weighted by a scalar product of normals :

$$(\mathcal{L}\ \vec{x})_i = (1-\theta)\vec{x}_i + \theta\ \frac{\displaystyle\sum_{j\in\mathcal{V}(i)\cup\{i\}} w_{ij}\vec{x}_j}{\displaystyle\sum_{j\in\mathcal{V}(i)\cup\{i\}} w_{ij}} \tag{9}$$

where w_{ij} are the weights defined by :

$$w_{ij} = \max\ \left(Area(i)\cdot Area(j)\cdot \vec{n}_i\cdot\vec{n}_j\ ,\ 0\right) \quad \|\vec{n}_i\| = 1\ \ \forall\, i \tag{10}$$

where θ is the smoothing parameter. Again $\mathcal{V}(i)$ represents the neighbors of cell i.

The above geometry is the surfacic boundary of a 3D unstructured tetrahedrization.

With the above transfer and smoothing operators from any level $m-1$ to level m as elementary bricks, we can derive a projection operator related to level k:

$$P_k \quad = \quad \prod_{1\leq m\leq k} \mathcal{L}_m \mathcal{P}_m \mathcal{P}_m^* \mathcal{L}_m^* \tag{11}$$

2.4 Multilevel preconditioner and optimization

In a way analog to [2], we introduce the multilevel preconditioner as follows:

$$P^* g \quad = \quad P_n g \quad - \quad \sum_{k}^{n-1} \frac{1}{4^{n-k}} \left(P_{k+1} g \quad - \quad P_k g\right) \tag{12}$$

where n is the coarsest level. The multilevel gradient approaches considered here rely on the following algorithm:

Multilevel Preconditioned Algorithm:

`Do` nc

- `Compute state` W `and adjoint` Π `and compute the gradient` $g(\gamma_{nc}, W, \Pi)$
- `Compute the preconditioner` P^*
- `Compute` ρ `(internal cycle)`
- `Update the shape correction:`

$$\gamma^{nc} \quad = \quad \gamma^{nc-1} - \rho \quad P^* \quad g(\gamma_{nc}, W, \Pi)$$

`Next` nc

Here, $g(\gamma_{nc}, W, \Pi)$ is a function of variables γ,W and Π , that is identical to $j'(\gamma)$ only if $W = W(\gamma)$ (solution of state equation) and $\Pi = \Pi(\gamma)$ (solution of adjoint state equation). The parameter ρ is either fixed or defined by a 1D search (steepest version). This algorithm results in a *gradient method* when G is exactly $j'(\gamma)$, which is a descent one in a weak sense since the preconditioner is symmetric. Conversely, when W and Π are obtained by applying only a few iterations of state equation and adjoint state equation iterative solution, the G is not the gradient of j, but aims to converge towards $j'(\gamma)$ when the whole loop is converging; we refer to this algorithm as a **one-shot method** (according to [8]) for solving the optimality system of the optimization problem. The performances of this approach for 2D applications are discussed in [5]

We have seen that an additional smoothing, applied on $P^*\ g(\gamma_{nc}, W, \Pi)$ can indeed improve the solution, specially when the skin mesh that covers the shape to discretize is rather coarse or have acute edges. It is enough a single iteration of a Least Squares Smoothing [7] or simply the average smoothing operator defined in (8) with $\theta = 1$.

3 Adaptation to shape design: sonic boom reduction

The shape optimization strategy described above can be applied to sonic boom reduction [14]. Briefly, the idea is to reduce as much as possible the shock signature on the ground produced by a supersonic aircraft on cruise flight. The optimization strategy comprises then two main nested blocks: the inner flow solver and the outer optimization loop. The former is based on the numerical solution of the Euler compressible flow equations. The latter follows the lines described above. Both of them are addressed in the sections above.

3.1 The model problem

The flow problem is physically modeled by the compressible Euler equations, for under the present conditions, viscous effects are negligeable. In its conservation form, the stationary Euler equations can be written as follows:

$$\frac{\partial}{\partial x_i}(F_i(W)) = 0, \tag{13}$$

where the system state is described by the solution $W = (\rho, \rho u_1, \rho u_2, \rho u_3, \rho e)$ and the flows are $F_i(W) = (\rho u_i, \rho u_i u_1 - p\delta_{i1}, \rho u_i u_2 - p\delta_{i2}, \rho u_i u_3 - p\delta_{i3}, u_i(\rho e + p))$. Volume forces and thermal sources are absent. The equations are written in terms of density ρ, velocity u_i and total energy $e = C_v T + \frac{1}{2} u_i u_i$. Pressure p is related to temperature T and density through the ideal gas state equation. Summation convention on repeated indices is here used and δ_{ij} is the identity tensor.

The weak form of (13) is: for all $\phi = (\phi_1, \phi_2, \phi_3, \phi_4, \phi_5)$ belonging to an appropriate test function space, find W such as

$$-\int_\Omega \frac{\partial \phi}{\partial x_i}\, F_i(W)\; dV \;+\; \int_{\partial\Omega} \phi\; F_i(W)\; n_i\; dS = 0, \tag{14}$$

whith some boundary conditions. Ω is the domain where the problem is defined and its boundary is $\partial\Omega$.

In order to find an approximate flow solution, (14) is numerically solved using a mixed finite volume/elements method, whith a second order Van Leer flux vector splitting scheme. In order to speed up flow convergence, the stationary solution is obtained through an iterative transient-like process, which uses implicit time advancing combined with local time steps.

The boundary $\partial\Omega$ is formed by inflows and outflows, non-slip condition contours and *transpiration* condition contours. This last kind of boundary corresponds to the part of $\partial\Omega$ that is to be optimized. The next two sections describes this concept deeper.

3.2 Aplication to shape design

Unlike many previous attempts in 3D (for instance [3]), where a **parametric optimization** is envisaged, we follow a so-called **CAD-Free** [12] approach. In a parametric optimization, the cost functional is minimized by changing a set of shape parameters (a set usually determined by the aircraft designer) like wing sweep or dihedral angles, wing position and so on. In this case, the appendages' shapes remain unchanged. On the other hand, in CAD-free approaches, the parameters space is given by the discretized geometry itself, through the nodes placed in the skin to be optimized, leading to a much more flexible optimization process (very flexible indeed: in [14] this is applied not only to wings but also to a generic airplane nose). Another important difference is that in [3] the minimization process takes into account the far-field signature, which relies in the Witham's linearized theory (see [9] for a complete review of the problem). On the other hand, we minimize what we call the **sonic boom downwards emission**, represented by the pressure gradient in the near field below the aircraft (see 1).

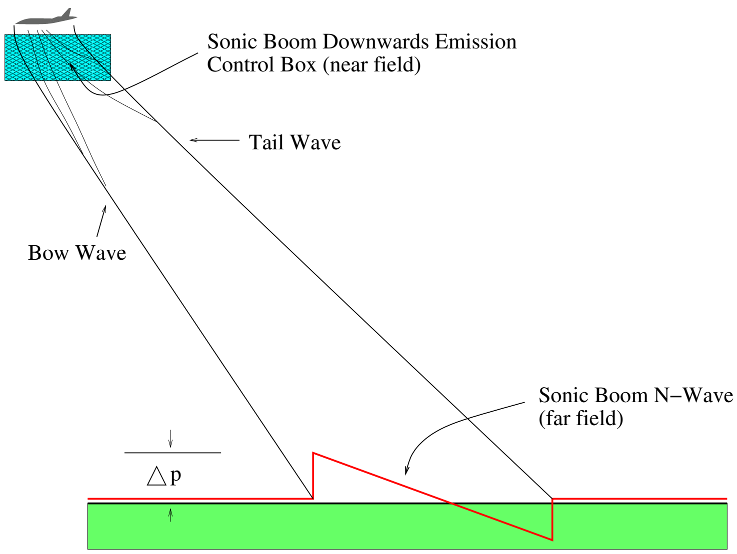


Figure 1: The sonic boom. Sketch of near and far field shock wave patterns of a supersonic aircraft.

Our objective is to seek for new wing forms, independently of the presence of additional appendages (like canards). These forms must have the following properties:

- **"Silent"** regarding sonic boom emission,
- **Aerodynamically performant**, and
- **Feasible**, from the constructive point of view.

3.3 Cost functional specification: transpiration and sonic boom emission

The application of a shape design loop should involve the repeated rezoning of the mesh to account for the modification of the shape of the aircraft. In this work, inspired by the approach used by Young *et al.* ([6]), we are considering in a first phase the option of representing the shape modification by applying a transpiration condition; this means that the current shape is defined with respect to the mesh skin as a perturbation simulated by transpiration (see for example [13]), refered in the sequel as the "*transpired perturbation*". Let us denote by γ the perturbation function; it is the algebraic length of the displacement of the boundary along its normal. We recall the transpiration condition for Euler flows: Let us denote by *shell* the shape to be emulated by transpiration and by $\vec{n}_{shell}$ the normal of the shell. The slip boundary term of the flux $\Psi(W)$ is defined as follows: for each component of the Euler flow equations set,

$$\Psi(W)_{slip\ boundary} = \big(\, 0\, ,\, p(W)\ n_x^\gamma\, ,\, p(W)\ n_y^\gamma\, ,\, p(W)\ n_z^\gamma\, ,\, p(W)\ q\, \big) \tag{15}$$

where

$$q = \vec{V}\ .\ (\vec{n}^\gamma - \vec{n}_{shell}).$$

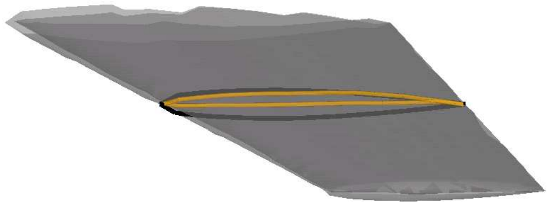
Here $\vec{V}$ is the velocity of the fluid and $\vec{n}^\gamma$ is the (fixed) normal defined on the surface to optimize. This approximation has proved to be enough accurate for rather large perturbations of the boundary and very robust. The sensitivity analysis has been exactly derived, but only for the first-order accurate upwind scheme.

For each step of the optimization process, a cost functional $j(\gamma)$ is evaluated. As said above, this functional depends both on the aerodynamic properties and the sonic boom downwards emission. The former is typically represented by lift and/or drag coefficients and their differences respect to target values. The latter is measured by the pressure gradient in the near field volume control in figure 1. Then, a suitable cost functional is

$$j(\gamma) = \alpha_1 (C_D - C_D^{target})^2 + \alpha_2 (C_L - C_L^{target})^2 + \alpha_3 \int_{\Omega^B} |\nabla p|^2\ dV \tag{16}$$

where α_1, α_2 and α_3 are constants that allow to vary the relative weight between the three constraints in $j(\gamma)$ that we want to consider.

Figure 2 shows the kind of result we are looking for. It corresponds to an M6 ONERA wing inmersed in a Mach 1.8 flow, where the incidence angle is 3°. The skin mesh is very coarse, for this is only a motivation example. On top, 2 shows both the original and the optimized shapes. After the optimization process, the wing's downards face becomes flattened, with a small "flap-like" trailing edge. The sonic boom emission reduction can be clearly seen in along a line just below the wing. It is worth to mention that it has changed not *only* the peak pressure value but the *form* of the pressure distribution. The typical Witham's F-function of the original wing, that produces a N-like ground signature has become a plateau-like distribution, with no shock at all. Additionally to the dramatic boom reduction, the optimized aerodynamic properties of the wing are good: the lift has diminished only 3% from the target value (that of the original).



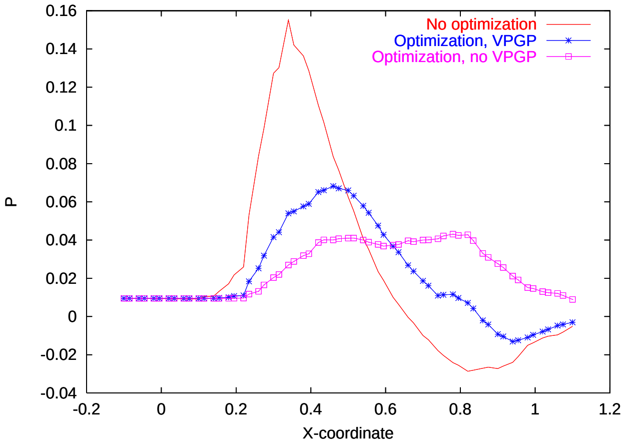


Figure 2: ONERA M6 wing. Optimized wing and original shape (top) and pressure along a line below the wing (bottom).

4 Numerical example: Dassault's Supersonic Business Jet

The proposed optimization procedure is applied to a projected supersonic business jet. The geometry (provided by Dassault Aviation) corresponds to half of the aircraft, as seen in fig. 3. The spatial grid has 173526 nodes in 981822 tetrahedra and corresponds to half of the aircraft, with a vertical symetry plane The inflow Mach number is 1.8 and the incidence angle is 3°. Figure 4 shows the Mach number and pressure distribution over the surface of the aircraft, symetrized now for postprocessing purposes.

As described in [14], the tuning of the scheme derived from simpler examples, allows us to set $\alpha_1 = 0$ and $\alpha_2 = 1$, being $\alpha_3 = 10$. The aircraft wings are the

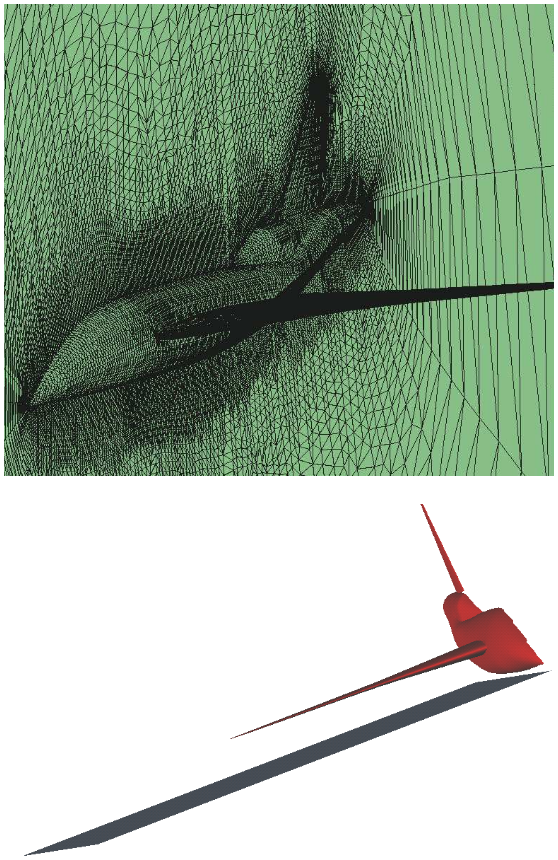


Figure 3: Dassault's Supersonic Business Jet. Top, spatial grid close-up. Bottom, aircraft and plane below.

targets of the optimization. The simplified wings provided by the constructor for this generic geometry are horizontally symetrical, with two different sweep angles of 17° and 38° respectively, and a rather smooth transition between them. The Mach angle for $M = 1.8$ is around 34°. Therefore, while the first part of the wings falls within the Mach cone with no shock wave ahead and a lower wave drag, the second part cut through the Mach cone. As a consequence, the shocks will be produced ahead of the 38° sweep angle portion of the wing. This can be clearly seen in fig. 5, top, where the pressure distribution below the aircraft for the original shape is shown. This fact renders the problem of optimizing a double-sweep angle wing more complex than a single-sweep angle one, like the M6. The reason that not all of the wing surface will produce a shock will result in an optimized shape that is different whether it is into or out of the Mach cone. This will also motivate the necessity of further refinement of the cost functional that is discussed in the conclusions section.

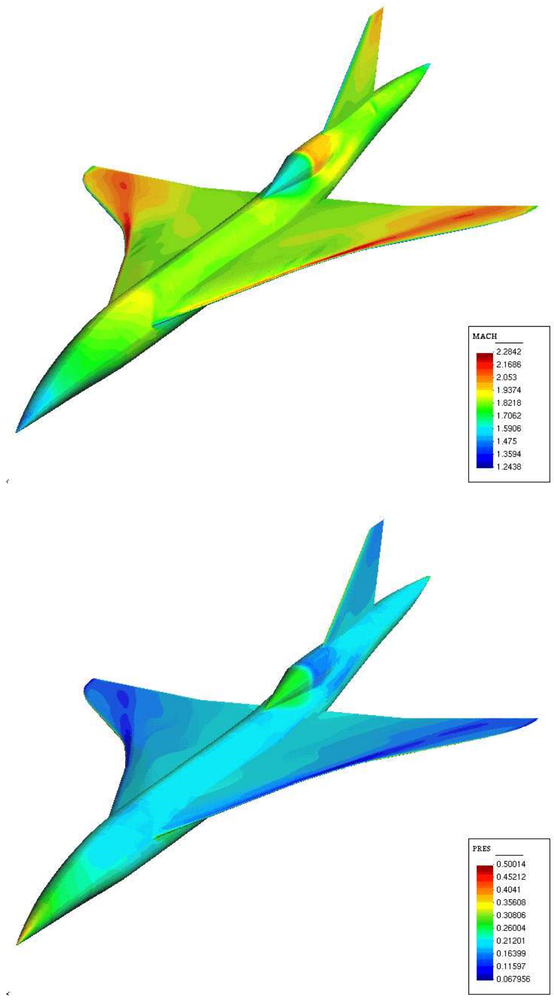


Figure 4: Dassault's Supersonic Business Jet. Contour levels. Top, Mach number. Bottom, pressure.

The outcome of 10 optimization cycles is shown in 5, right. In this figure it can be seen that the main peak has indeed diminished as in the M6 wing example.

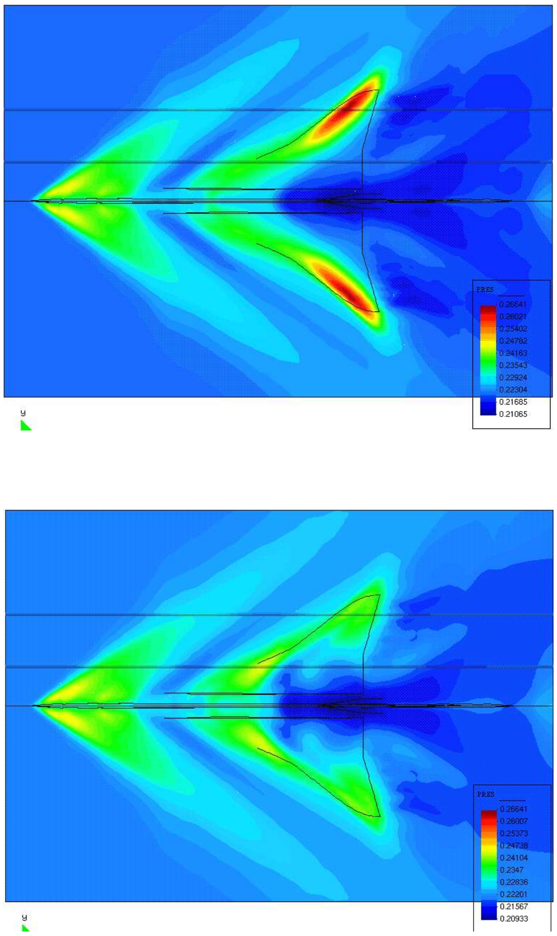


Figure 5: SBJ. Pressure distribution in a plane below the aircraft. Top, original geometry. Bottom, optimized geometry.

5 Conclusion and future lines

Acknowledgements : We thank Dassault Aviation for yielding the Supersonic Business Jet mesh. We are also grateful to CINES (Centre Informatique National de l'Enseignement Supérieur) for providing the computational facilities where the largest examples where ran.

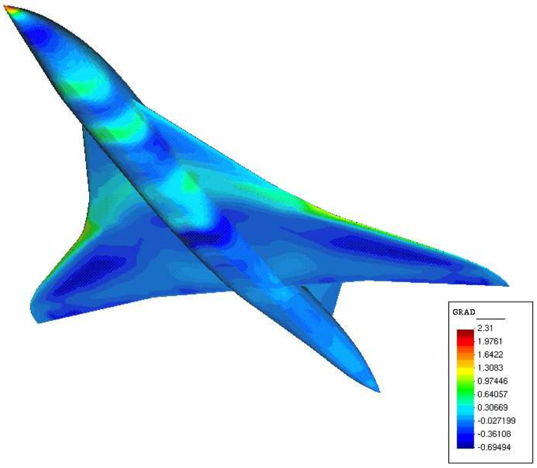


Figure 6: Dassault's Supersonic Business Jet. Cost functional gradient distribution for the complete aircraft's surface.

References

[1] F. BEUX and A. DERVIEUX. A Hierarchical Approach for Shape Optimization. *Engineering Computations*, 11(1):25–48, February 1994.

[2] J.H. BRAMBLE, J.E. PASCIAK, and J. XU. Parallel multilevel preconditionners. *Math. Comp.*, 55:1–22, 1990.

[3] B. ARGROW C. FARHAT, K. MAUTE and M. NIKBAY. A shape optimization methodology for reducing the sonic boom initial pressure rise. *AIAA Paper*, 2002-0145:1–11, 2002.

[4] F. COURTY. Functional properties of an multilevel preconditioner. Technical report, INRIA - Sophia Antipolis, 2002.

[5] H. GUILLARD and N. MARCO. Some aspects of Multigrid Methods on Non-Structured meshes. In *Proceedings of the Conference of Copper Mountain on Multigrid Methods*, April 1995.

[6] W. P. HUFFMAN, R. G. MELVIN, D. P. YOUNG, F. T. JOHNSON, J. E. BUSSOLETTI, M. B. BIETERMAN, and CRAIG L. HILMES. Practical Design and Optimization in Computational Fluid Dynamics. *AIAA Paper 93-3111*, 1993.

[7] T.J.R. HUGHES. *The Finite Element Method.* Prentice-Hall, 1987.

[8] G. KURUVILA, S. TA'ASAN, and M.D. SALAS. Airfoil Optimization by the One-Shot method. Aiaa paper, 1994.

[9] D.J. MAGLIERI and K.J. PLOTKIN. *Aeroacoustics of flight vehicles: theory and practice.* Acoustical Society of America, Publications, 1991.

[10] N. MARCO and A. DERVIEUX. Agglomeration method applied to the Hierarchical Parametrization of a Skin Mesh in 3D Aerodynamics, 1994. Contributions to 12th month of European Project ECARP.

[11] N. MARCO, B. KOOBUS, and A. DERVIEUX. An Additive Multilevel Preconditioning Method. Research Report 2310, INRIA Sophia-Antipolis, Août 1994.

[12] B. MOHAMMADI and O. PIRONNEAU. *Applied shape optimization for fluids*. Clarendon Press - Oxford, 2001.

[13] G.D. MORTCHELEWICZ. Résolution des équations d'Euler tridimensionnelles instationnaires en maillages non structurés. *La Recherche Aérospatiale*, (6):17–25, Novembre-Décembre 1991.

[14] M. VAZQUEZ, B. KOOBUS, and A. DERVIEUX. Aerodynamical and sonic boom optimization of a supersonic aircraft. Technical report, INRIA - Sophia Antipolis, 2002.