**8th World Congress on Structural and Multidisciplinary Optimization**
June 1-5, 2009, Lisbon, Portugal

# Flow Control of Curved Air Ducts using Topological Derivatives

**Jonathan Chetboun**[1,2], **Grégoire Allaire**[1]

[1]CMAP - École Polytechnique
91128 Palaiseau, France

[2]Dassault-Aviation DGT/DTIAE/AERAV
78, quai Marcel Dassault, 92214 Saint Cloud, France

e-mail : `Jonathan.Chetboun@dassault-aviation.com`

## 1. Abstract

A method for flow control of curved air ducts using mechanical vortex generators (VG) is described. This method includes the creation of vortex generators in a reference flow using the notion of topological derivative. A classical procedure of parametric optimization of the created VG by an adjoint state approach is also presented. Three-dimensional numerical simulations are performed illustrating the control of an s-shaped duct by VG.

**2. Keywords:** Topological derivative, Parametric optimization, Flow control, Vortex generators.

## 3. Introduction

The control of aerodynamic flows is a crucial challenge in the aircraft and automotive industries. On military aircraft, the engine is supplied in air by a duct ; its length and shape are dictated by the global design of the airplane. The stealth constraints yield very curved ducts in order to hide the engine rotating blades, largely responsible for the radar signature of the aircraft. These levels of curvature cause flow separation harmful to the quality of the air feeding the engine, therefore imposing the use of control devices. Figure 1 shows the project of a European unmanned combat air vehicle (UCAV) called nEUROn and a generic u-shaped air duct.

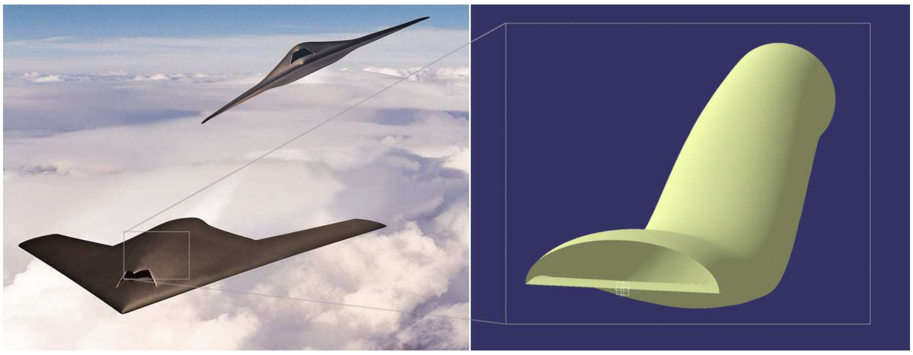

Figure 1: nEUROn and u-shaped air duct

We are interested here in controling the flow inside a curved air duct by mechanical vortex generators. These are typically small aerodynamic surfaces attached to the duct wall and causing a swirl in their wake, thus delaying the appearance of separations. Rather than finely mesh the VG in numerical simulations, we prefer to use a model for their action on the fluid through the addition of a source term to the Navier-Stokes equations. Our goal is to determine whether it is relevant to introduce an additional VG into a reference flow in order to minimize a given cost function J. Note that one or several VG may already be present in this reference flow field. We use the idea of topological derivative, first introduced in the framework of structural optimum design [1], [2], [3], [4], [5], and subsequently applied, among other areas, to the incompressible Navier-Stokes equations [6]. In section 4, emphasis is made on the computation of the topological derivative for the creation of a source term in the case of a simple model problem.
As a basis for our numerical simulations, we use *Aether*, a finite element industrial code developed at Dassault-Aviation, which solves the turbulent compressible Navier-Stokes equations [7]. Section 5

briefly describes the numerical method used in the code and its adjoint version obtained by automatic differentiation. We also deal with the modeling of vortex generators by source term.
Section 6 is devoted to numerical simulations of the flow control of an s-shaped air duct using topological derivatives. In particular, we show the reduction of a given cost function consecutive to the placement of the VG according to the topological derivative map.
Finally, in section 7, we apply a classical parametric optimization procedure, using the same adjoint state as in the topological derivative, in order to achieve a complete optimization of the created vortex generator.

## 4. Topological derivative of a source term

As a vortex generator is modeled by a source term in numerical simulations, we need to adapt the concept of topological derivative developed in shape optimization to the case of the creation of a source term. The aim of this section is thus to compute the topological derivative of a source term in the simple case of the Laplace equation. The (formal) computation of this topological derivative in the more interesting case of the compressible Navier-Stokes system will be found in [8]. Let then $\Omega$ be an open subset of $\mathbb{R}^N$ ($N \geq 2$) and let us consider the following problem :

$$\left\{\begin{array}{rcll} -\Delta u & = & \chi f & \text{in} \quad \Omega \\ u & = & 0 & \text{on} \quad \partial\Omega, \end{array}\right. \tag{1}$$

with $\chi$ the characteristic function of the source term $f$. We define the cost function $J$ :

$$J(\chi) = \int_\Omega j(u)\, dx. \tag{2}$$

Our goal is to evaluate the sensibility of the cost function $J$ when a small perturbation $\delta\chi$ is added to the characteristic function $\chi$ of the source term $f$. We thus define the following perturbed problem :

$$\left\{\begin{array}{rcll} -\Delta u^\delta & = & (\chi + \delta\chi)\, f & \text{in} \quad \Omega \\ u^\delta & = & 0 & \text{on} \quad \partial\Omega. \end{array}\right. \tag{3}$$

The inclusion $\delta\chi$ is defined from an elementary inclusion $B$ (for example $B = \left\{x \in \mathbb{R}^N : \|x\| \leq 1\right\}$) as follows :

$$\delta\chi(x) = \chi_B \left(\frac{x - x_0}{\delta}\right). \tag{4}$$

Moreover, we assume that the supports of the original source term and of the inclusion do not overlap :

$$\chi(x)\delta\chi(x) = 0. \tag{5}$$

In the sequel, we need to evaluate the difference between the initial solution $u$ and the perturbed one $u^\delta$. Thus, we define :

$$v^\delta = u^\delta - u, \tag{6}$$

which is solution of :

$$\left\{\begin{array}{rcll} -\Delta v^\delta & = & \delta\chi f & \text{in} \quad \Omega \\ v^\delta & = & 0 & \text{on} \quad \partial\Omega. \end{array}\right. \tag{7}$$

We introduce the microscopic space variable $y = \frac{x-x_0}{\delta}$ in order to rescale the previous problem. This rescaling, centered on the inclusion, transforms the problem posed on $\Omega$ into a problem posed on $\mathbb{R}^N$, in the limit as $\delta$ tends to zero. Consequently, let us define $w$ the solution of the canonical problem :

$$\left\{\begin{array}{rcll} -\Delta w & = & \chi_B f(x_0) & \text{on} \quad \mathbb{R}^N \\ \nabla w & \rightarrow & 0 & \text{at} \quad \infty, \end{array}\right. \tag{8}$$

and the rescaled solution :

$$w^\delta(x) = \delta^2 w \left(\frac{x - x_0}{\delta}\right). \tag{9}$$

Then we have :

$$\left\{\begin{array}{rcll} -\Delta v^\delta & = & \delta\chi\left(f(x) - f(x_0)\right) - \Delta w^\delta & \text{in} \quad \Omega \\ v^\delta & = & 0 & \text{on} \quad \partial\Omega. \end{array}\right. \tag{10}$$

Using the decay properties at infinity of $w$, we can show that :

$$\left\| v^{\delta} \right\|_{L^2(\Omega)}^{2} \leq \mathcal{O}\left(\delta^{N+1}\right). \tag{11}$$

This estimate is not optimal, but sufficient for our purpose. One can refer to [8] for the detailed calculus. Introducing the adjoint state $p$ solution of :

$$\left\{ \begin{array}{rcll} -\Delta p & = & j'(u) & \text{in} \quad \Omega \\ p & = & 0 & \text{on} \quad \partial\Omega, \end{array} \right. \tag{12}$$

and assuming that $j''(u)$ is bounded (for example by taking $j$ quadratic), we can prove that :

$$J(\chi + \delta\chi) = J(\chi) + \delta^N |B| f(x_0) p(x_0) + \mathcal{O}\left(\delta^{N+1}\right). \tag{13}$$

In conclusion, the topological derivative $G$ consecutive to the addition of a source term $f$ at a point $x_0$ simply reads :

$$G(x_0) = f(x_0)p(x_0). \tag{14}$$

Consequently, in order to decrease the cost function $J$, and for a small $\delta$, it is relevant to create a new source term $f$ at the points where $G$ is the more negative.

The same kind of result can be obtained in the case of a source term $f$ depending on the state $u$ and when the perturbation lies at a distance $\delta$ of the boundary (which is the case for the source term modeling a vortex generator attached to the wall). We can also consider the cases of the Stokes problem and the incompressible Navier-Stokes equations. In the case of the compressible Navier-Stokes equations, only a formal calculus has been made.

## 5. Implementation of the topological derivative in a CFD industrial software

### 5.1. The Navier-Stokes solver

Our numerical simulations are made using the software *Aether*, a compressible Navier-Stokes finite element solver developed at Dassault-Aviation. Let us first recall the expression of the compressible Navier-Stokes equations. Let $\rho$, $u$ and $e$ be respectively the density, the velocity and the total energy per unit mass of fluid. The Navier-Stokes equations read :

$$\left\{ \begin{array}{rcl} \dfrac{\partial \rho}{\partial t} + \nabla \cdot (\rho u) & = & 0 \\ \dfrac{\partial \rho u}{\partial t} + \nabla \cdot (\rho u \otimes u) & = & \nabla \cdot \sigma + \rho f \\ \dfrac{\partial \rho e}{\partial t} + \nabla \cdot (\rho e u) & = & \nabla \cdot (\sigma u - q) + \rho f \cdot u, \end{array} \right. \tag{15}$$

where $\sigma$ is the Cauchy stress tensor, $q$ the heat-flux vector and $f$ the source term. The numerical method used in the *Aether* solver is a Galerkin / least-squares formulation of the system written in terms of entropy variables and includes a dicontinuity capturing operator. The code also includes turbulent flow modeling through the Reynolds Averaged Navier-Stokes equations (RANS). The closure of the Reynolds stress tensor and heat flux is obtained using a classical Boussinesq hypothesis and the concept of eddy-viscosity. The steady numerical simulations presented in the sequel are performed using a two-layer $k - \varepsilon$ turbulence model.

### 5.2. Modeling vortex generators

In order to control the fluid inside the duct, we use passive devices called mechanical vortex generators. These are small blades attached to the duct wall and causing a swirl in their wake. This swirl prevents flow separation by giving energy to the boundary layer.

As said in the introduction, vortex generators are not meshed in numerical simulations. Indeed, meshing an actual VG would require a too high level of mesh refinement. Moreover, one wants to test several configurations of vortex generators without the obligation of performing a different mesh. That is why we rather use a source term model in the code. The source term added to the Navier-Stokes equations Eq.(15) is defined as follows [9] :

$$f = c\, S_{VG}\, \rho\, (n \cdot u)\, (b \times u), \tag{16}$$

where $c$ is a modeling constant, $S_{VG}$ is the surface of the vortex generator and vectors $b$ and $n$ are deduced from the orientation $t$ of the vortex generator according to Figure 2. Therefore, the force $f$ is

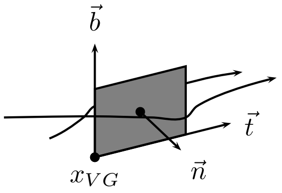

Figure 2: Geometrical description of a vortex generator

perpendicular to the flow and tends to deviate it into a direction parallel to the vortex generator. In the code, the force is multiplied by an exponential decreasing with the distance from the vortex generator, in such a way that its action is distributed over a small volume surrounding the VG. Note that, as $u$, $f$ is equal to zero at the wall.

### 5.3. Computation of the topological derivative

We formally rewrite the compressible Navier-Stokes equations as :

$$E(U, f(U, l)) = 0 \tag{17}$$

where $U$ is the vector of entropy variables and $l$ is the vector of the vortex generator parameters (its size and orientation). Then, we can write the adjoint system as follows [10], [11] :

$$\left(\frac{\partial E}{\partial U}\right)^T P = \left(\frac{\partial J}{\partial U}\right)^T \tag{18}$$

This system is obtained by automatic differentiation of the *Aether* solver using the software *Tapenade* developed at INRIA [12]. It is clearly a discrete approach to the computation of the adjoint state rather than a discretization of the continuous equations. Note that only the Navier-Stokes part and not the turbulence part of the code has been differentiated. Numerical simulations show that this is sufficient to compute accurate sensitivities. The adjoint version of the code is validated by comparison with finite differences. As the source term modeling a vortex generator lies on a small volume around the VG, we rather use in our code an averaged version of the expression Eq. (14) :

$$G(x_0) = \int_{\delta_{VG}} f \cdot P \, dx. \tag{19}$$

The volume $\delta_{VG}$ is taken as the support of the small source term we want to add.

## 6. Numerical simulations

### 6.1. The test case

Our numerical simulations are performed on the generic s-shaped air duct with circular section presented on Figure 3. The duct is 1020 mms long. In the sequel, we make comparisons between two meshes : the coarse one includes approximately 300000 nodes and the fine one approximately 800000 nodes. First, we compute a steady reference flow using the *Aether* solver and a $k - \varepsilon$ turbulence model on half a mesh with symmetry conditions. The cost function considered here is the swirl at the duct output. It is the ratio between the velocity on the output plane and the velocity perpendicular to this plane.

### 6.2. Creation of a first vortex generator

Figure 4 and Figure 5 show the topological derivative map on the two meshes for the creation of a 5-mm-long and 2-mm-high vortex generator. The created vortex generators are oriented at 45 degrees apart from flow direction. Circles represent results of numerical simulations with actual VG and show the decrease of cost function. Each circle comes with its corresponding value of cost function (Swirl) and topological derivative (TD).

We can notice that, in the case of the coarse mesh, the localization VG2 presents a swirl smaller than the localization VG1 corresponding to the minimum of topological derivative. Nevertheless, this default is no longer visible in the case of the fine mesh. Besides, the swirl of localizations VG2 and VG3 escapes the classification predicted by the topological derivative. However, the topological derivative gives us a good idea of where to create a vortex generator (areas in blue and green on the two images, the cost function

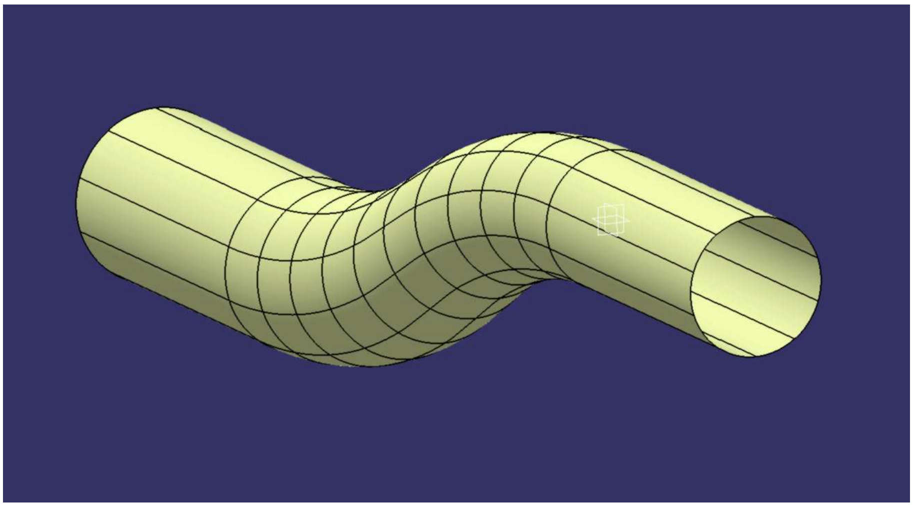

Figure 3: Geometry of the test case

decreases) and where not to create a vortex generator (areas in red, the cost function increases). Nevertheless, in order to keep decreasing the cost function, it is useful to implement a method of parametric optimization to adjust the size and orientation of the created vortex generator.

6.3. Addition of a second vortex generator

Figure 6 and Figure 7 show the topological derivative map on the two meshes for the addtition of a second vortex generator. The first vortex generator has been put on the localization VG1 of Figure 4 and Figure 5.

In this case, the topological derivative map is different between the two meshes. Unfortunately, the swirl resulting from numerical simulations with actual VG does not match the value of the topological derivative. The cost function increases wherever we put the second VG. However, in the case of the fine mesh, we can notice that this increase is more significant at the points where the topological derivative is positive, whereas it is very small where the topological derivative is negative. In this test case, where the flow separation is not massive at all, we can assume that a single vortex generator is sufficient to control the flow. It seems that the addition of a second VG is not suitable.

## 7. Parametric optimization

In the previous section, we saw that the topological derivative allows us to know where it is relevant to create a vortex generator. Rather than a precise localization of its minimum, the topological derivative gives us an area where the cost function can be decreased. In order to find this minimum, it is useful to perform a classical method of parametric optimization, involving the same adjoint state that appeared in the computation of the topological derivative. Therefore, we optimize parameters such as the size or the orientation of the VG, which were fixed in the computation of the topological derivative. Let us first recall that, formally, the *Aether* software solves the following system of equations :

$$E(U, f(U, l)) = 0, \tag{20}$$

and that the adjoint state equations can be written as :

$$\left(\frac{\partial E}{\partial U}\right)^T P = \left(\frac{\partial J}{\partial U}\right)^T. \tag{21}$$

Then, it is well-known that the variation of cost function $\delta J$ resulting from the variation $\delta l$ of the parameters is given by :

$$\delta J = -P^T \frac{\partial E}{\partial l} \cdot \delta l. \tag{22}$$

The derivative in Eq. (22) is also computed using automatic differentiation of the code. We performed a parametric optimization of the vortex generator named VG1 given by the topological derivative on the coarse mesh using an algorithm of sequential quadratic programming. The five parameters we want

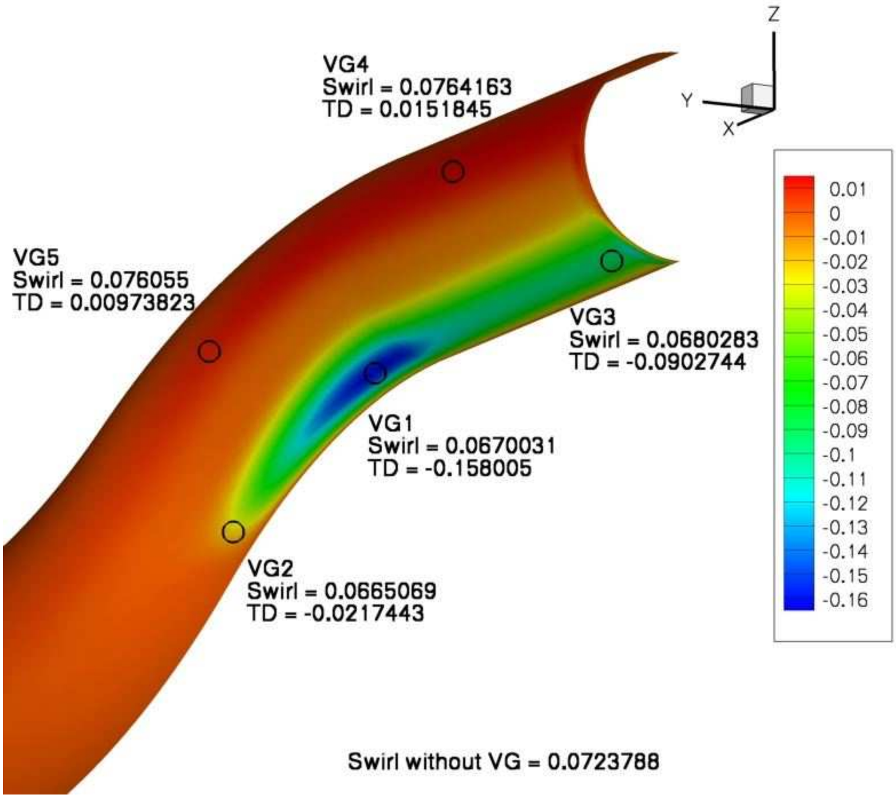

Figure 4: Topological derivative for the creation of a first VG - Coarse mesh

to optimize are the two variables of the VG position on the surface of the duct, its angle (orientation compared with flow direction) and its size (height and length). Now, the cost function considered is $J = 500 \times Swirl^2$. Figure 8 shows the cost function and parameters evolution during the optimization process. In this picture, we present each step of the optimization process, which includes gradient computation and line search. That is why we can notice a brief increase in the cost function as the algorithm searches for the optimal step. In term of actual swirl, the creation of the first VG using topological derivative allows to reduce the cost function of 7 % whereas the parametric optimization produces a decrease of 3 % more. At the end of the optimization process, the position is not very different from the one given by the topological derivative. The orientation changes from 45 to approximately 30 degrees. The size of the VG is reduced from 5 x 2 mms to approximately 4 x 1.8 mms.

## 8. Conclusion

An efficient method for the flow control inside a curved air duct using mechanical vortex generators has been presented. The idea of topological derivative has been used in order to decide where it is relevant to create a new VG in a reference flow. A classical method of parametric optimization by adjoint state has been applied successfully to the created vortex generator.

Further developments are under way concerning the use of this method in the case of unsteady flows. Indeed, in order to achieve more complex simulations with shorter or more curved air ducts, it will be essential to take into account the unsteady nature of the flow with methods such as Detached Eddy Simulation. In particular, the computation of the adjoint state in this framework will be a crucial challenge, both from a mathematical and a numerical point of view.

## 9. References

**[1]** H.A. Eschenauer, V.V. Kobelev and A. Schumacher, Bubble method for topology and shape optimization of structures, *Structural Optimization*, 8, 42-51, 1994.

**[2]** J. Sokolowski and A. Zochowski, On the topological derivative in shape optimization, *SIAM J. Control*

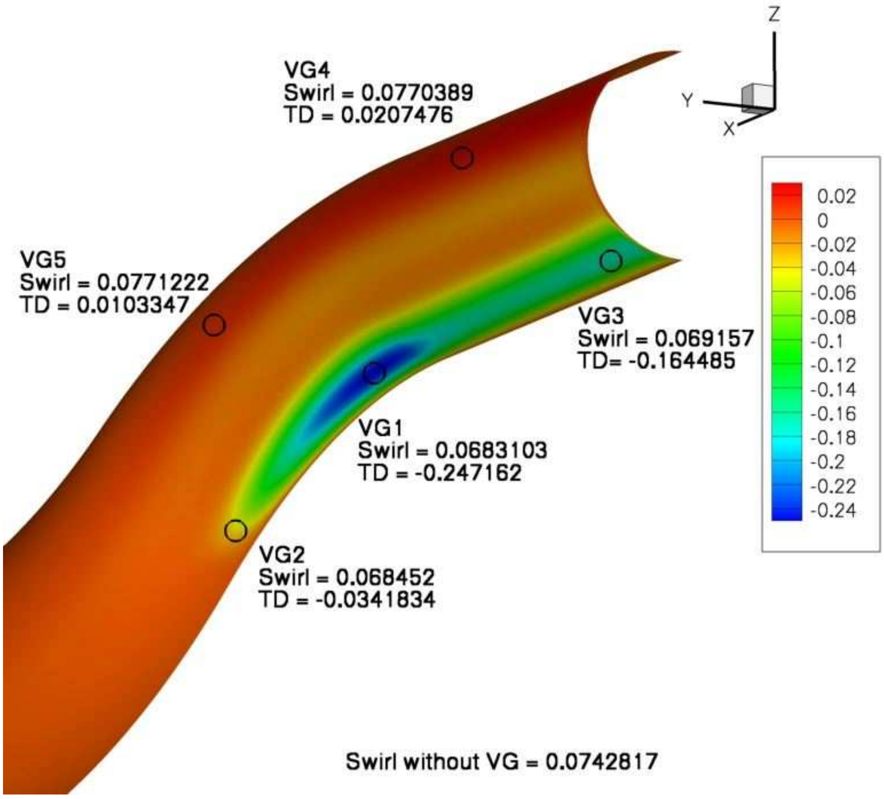

Figure 5: Topological derivative for the creation of a first VG - Fine mesh

*Optim.*, 37, 1251-1272, 1999.

**[3]** J. Céa, S. Garreau, P. Guillaume and M. Masmoudi, The shape and topological optimizations connection, *Comput. Methods Appl. Mech. Engrg.*, 188, 713-726, 2000.

**[4]** S. Garreau, P. Guillaume and M. Masmoudi, The topological asymptotic for PDE systems : the elasticity case, *SIAM J. Control Optim.*, 39, 1756-1778, 2001.

**[5]** P. Guillaume and K. Sid Idris, The topological asymptotic expansion for the Dirichlet problem, *SIAM J. Control Optim.*, 41, 1042-1072, 2002.

**[6]** S. Amstutz, The topological asymptotic for Navier-Stokes equations, *ESAIM Control Optim. Calc. Var.*, 11, 401-425, 2005.

**[7]** F. Chalot, M. Mallet and M. Ravachol, A comprehensive finite element Navier-Stokes solver for low- and high-speed aircraft design, *AIAA*, 814, 1994.

**[8]** J. Chetboun, Design of aerodynamic shapes in presence of flow separations : control and optimization, *Phd thesis*, in preparation.

**[9]** K. Waithe, Source term model for vortex generator vanes in a Navier-Stokes computer code, *AIAA*, 1236, 2004.

**[10]** Q.V. Dinh, G. Rogé, C. Sevin and B. Stoufflet, Shape optimization in computational fluid dynamics, *European Journal of Finite Elements*, 5, 569-594, 1996.

**[11]** L. Daumas, Q.V. Dinh, S. Kleinveld and G. Rogé, A CAD-based supersonic business jet optimization, *ECCOMAS*, 2004.

**[12]** L. Hascoët, TAPENADE : a tool for Automatic Differentiation of programs, *ECCOMAS*, 2004.

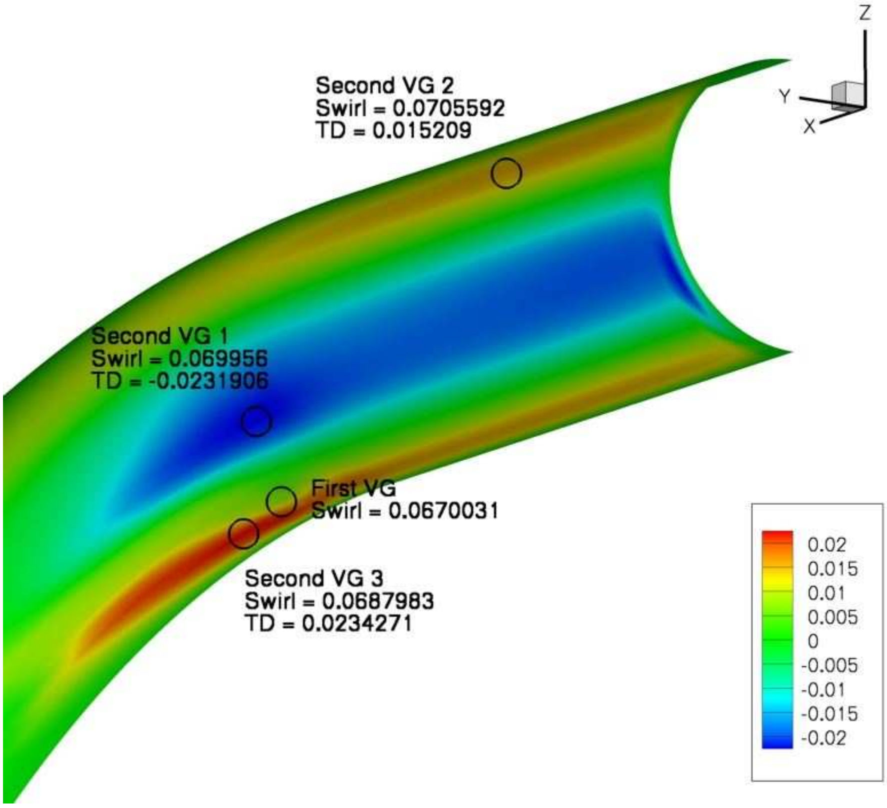

Figure 6: Topological derivative for the addition of a second VG - Coarse mesh

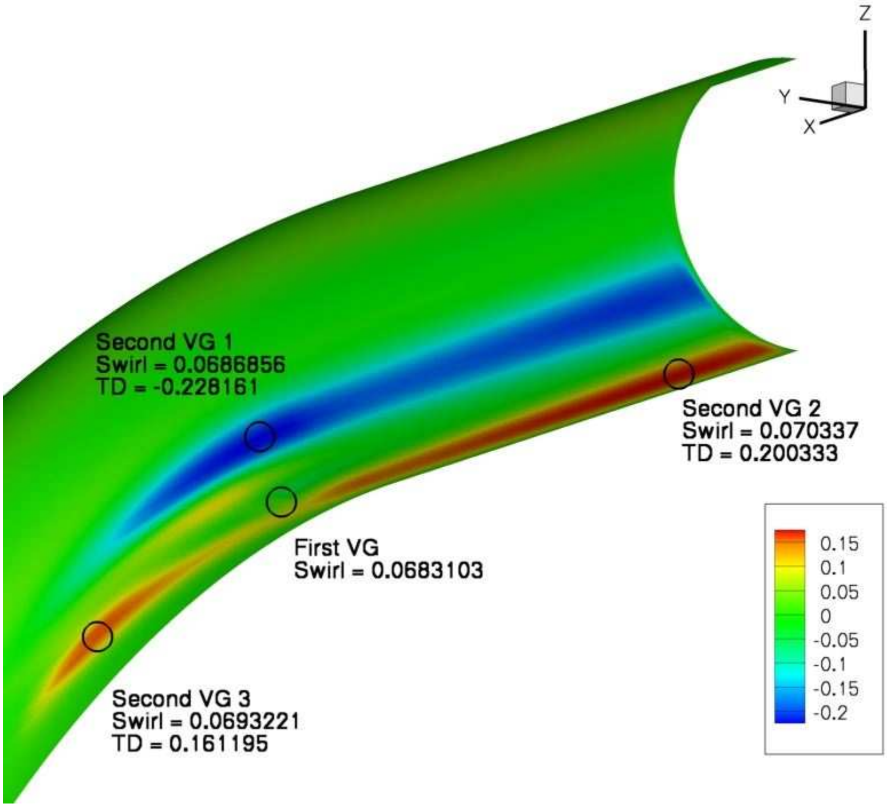

Figure 7: Topological derivative for the addition of a second VG - Fine mesh

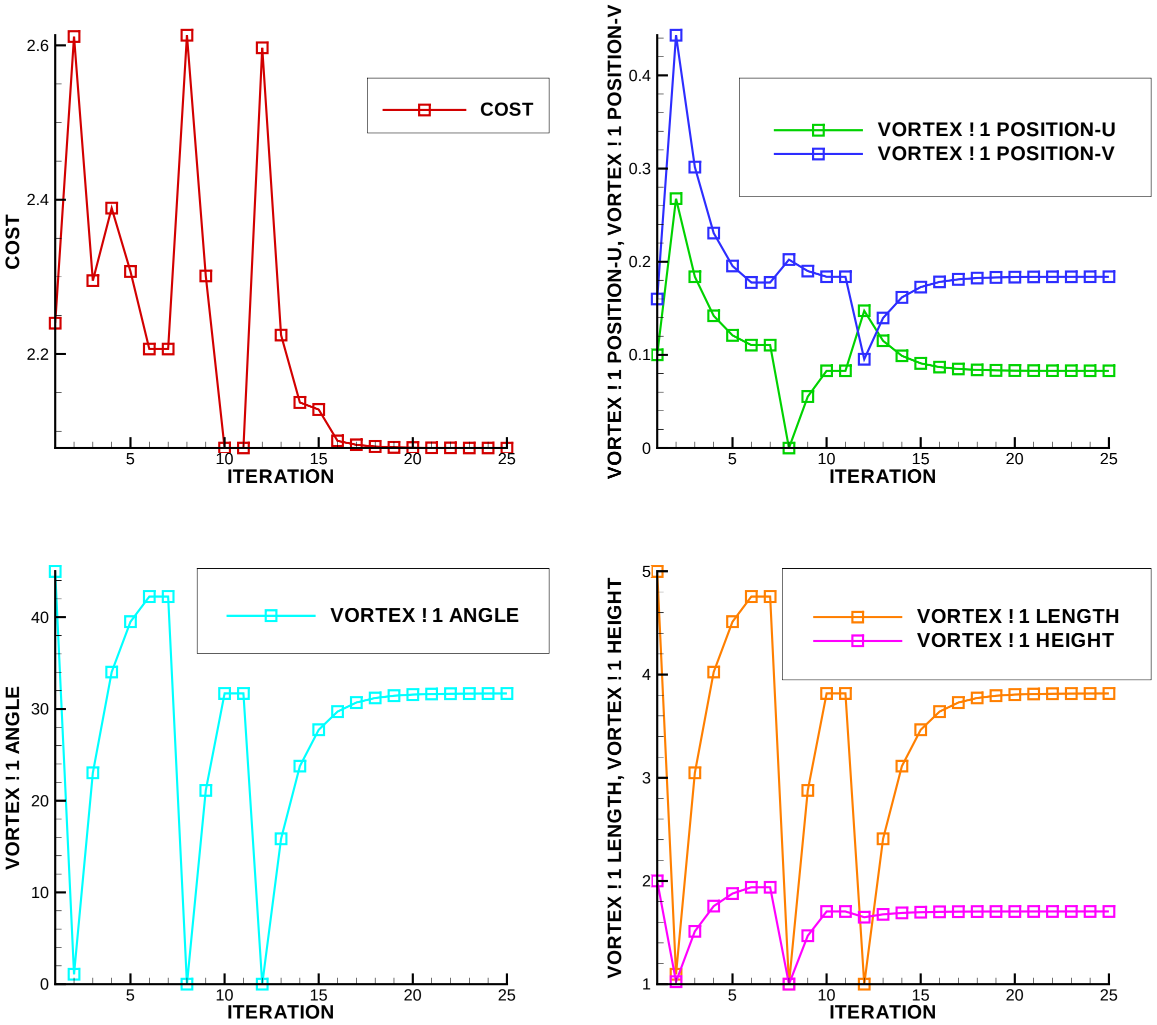

Figure 8: Evolution of cost function and parameters during optimization process