FOI, Swedish Defence Research Agency, is a mainly assignment-funded agency under the Ministry of Defence. The core activities are research, method and technology development, as well as studies conducted in the interests of Swedish defence and the safety and security of society. The organisation employs approximately 1000 personnel of whom about 800 are scientists. This makes FOI Sweden's largest research institute. FOI gives its customers access to leading-edge expertise in a large number of fields such as security policy studies, defence and security related analyses, the assessment of various types of threat, systems for control and management of crises, protection against and management of hazardous substances, IT security and the potential offered by new sensors.

FOI
Defence Research Agency
Defence and Security, Systems and Technology
SE-164 90 Stockholm

Phone: +46 8 555 030 00
Fax: +46 8 555 031 00

www.foi.se

AESOP - A Numerical Platform for Aerodynamic Shape Optimization

OLIVIER AMOIGNON

FOI-R--2538--SE Technical report
ISSN 1650-1942 October 2008

Defence and Security, Systems and Technology

AESOP - A Numerical Platform for Aerodynamic Shape Optimization

FOI-R--2538--SE

Olivier Amoignon

AESOP - A Numerical Platform for Aerodynamic Shape Optimization

Titel	AESOP - En Numerisk Platform för Aerodynamisk Formoptimering
Title	AESOP - A Numerical Platform for Aerodynamic Shape Optimization
Rapportnummer / Report no	FOI-R--2538--SE
Rapporttyp / Report type	Teknisk rapport / Technical report
Utgivningsår / Year	2008
Antal sidor / Pages	44
Kund / Customer	European Commission 6^{6th} Framework Program
Forskningsområde	5. Bekämpning och skydd
Research area	5. Strike and Protection
Delområde	55. Flygteknik
Sub area code	55. Aeronautics
Projektnummer / Project no	B66004
Godkänd av / Approved by	Ardeshir Hanifi
ISSN	ISSN-1650-1942

FOI Swedish Defence Research Agency
Defence and Security, Systems and Technology
SE-164 90 STOCKHOLM

Abstract

Aerodynamic shape optimization based on Computational Fluid Dynamics can automatically improve the design of aircraft components. In order to obtain the best computational efficiency, the adjoint method is applied on the complete mapping, from the parameters of design to the evaluation of the cost function or constraints. The mapping considered here includes the parameterization, the mesh deformation, the primal-to-dual mesh transformation and the flow equations solved by the unstructured flow solver Edge distributed by FOI. The program AESOP integrates as subroutines the flow and adjoint flow solver, mesh deformation schemes, algorithms of shape parameterization and algorithms for gradient-based optimization. The result is a portable and efficient implementation for large scale aerodynamic shape optimization and future applications in multidisciplinary shape optimization. The structure of the program is outlined and examples of applications are presented. The method of shape parameterization using Radial Basis Functions is discussed in more details because it is expected to play a major role in the development of multidisciplinary optimization.

Keywords

Flow equations, Edge-based Finite Volumes, Shape optimization, Adjoint equations, Radial Basis Functions

Sammanfattning

Aerodynamisk formoptimering baserad på datorsimuleringar kan förbättra utformningen av flygplans. Med hjälp av så kallade adjunktsekvationer kan man effektivt beräkna gradienten av funktioner som beror på strömningsekvationernas lösning. Programmet AESOP tillämpar denna metodik på alla de delar av beräkningskedjan som involveras när man optimera aerodynamiska former, från styrning av geometriska förskjutningar till lösning av strömningsekvationerna med hjälp av programmet Edge som utvecklas av FOI. AESOP integrerar alla komponenter som subrutiner, så väl lösaren för strömningsekvationer och adjunktsekvationer som förskjutningsprogram för beräkningsnätet och optimeringsalgoritmer. Resultatet är en flexibel och effektiv programvara som är anpassad till att lösa storskaliga aerodynamiska formoptimerings problem och framtida multidisciplinära formoptimerings tillämpningar. Programmets struktur presenteras tillsammans med exempel på tillämpningar. Parametriseringen av geometriska förskjutningar med radiella basfunktioner (RBF) är särskilt detaljerat eftersom den förväntas spela en viktig roll i utvecklingen av multidisciplinär optimering, i synnerhet när man kopplar strömning och struktur ekvationer.

Nyckelord

Strömningsekvationer, Kantbaserad finita-volymsdiskretisering, Formoptimering, Adjunktsekvationer, Radiella basfunktioner

Contents

1 Introduction

In aeronautic industry, the prospect of aerodynamic shape optimization is to speed up the design of aircraft components. The continuous development of Computational Fluid Dynamics (CFD) plays a major role in the development of this activity because it improves the reliability of the predicted aerodynamic forces and moments that are used in problems of shape optimization. However, the accuracy of the CFD predictions around aircrafts comes at a cost that is incomparably higher than for solving, say, the aircraft structural deformations in the framework of linear elasticity. In aerodynamic optimization this cost is multiplied by the number of flow solutions required for solving the (nonlinear) problem of aerodynamic optimization. This number of CFD simulations is closely related to the algorithm of optimization, for a given problem to be solved. To date, the algorithms based on the derivatives of the cost function and constraints, with respect to the design variables (gradients), are the most efficient, provided that the derivatives can be calculated accurately and efficiently. The cost of the classical computation of gradients by finite differences is proportional to the number of design parameters n, which penalizes this approach when n is larger than the number of constraints. The finite difference method may also be inaccurate because it requires small perturbations of the design parameters whereas shape perturbations smaller than a certain limit may not be resolved by the CFD, thus leading to round-off errors due to the difference scheme. In an *optimal control* approach, the gradient of one function can be calculated by solving once an *adjoint* of the flow equations. Regarding the cost of optimization, the optimal control approach, or adjoint sensitivities analysis, has the advantage on the finite difference method for a number of design parameters larger than the number of constraints. Nonetheless, the accuracy of the adjoint approach does not rely on a perturbation parameter, which is another advantage.

The numerical platform AESOP[1] has been developed for the realization of an *optimal control* approach based on the adjoint flow equations implemented in Edge [1], FOI's software for CFD [8].

Design optimization using an optimal control approach has become popular in the last two decades in all fields of engineering. For a detailed presentation of techniques of optimal shape design for systems governed by elliptic equations, the reader may refer to a book by Pironneau [20]. For an introduction focusing on aerodynamic shape optimization, the reader may refer to Giles & Pierce [12].

Natural Laminar Flow design based on the adjoint approach is one of the first applications that AESOP was developed for [2]. This requires additional solvers, and adjoint solvers, for the laminar boundary layer equations and for the stability equations in the boundary layer. The procedure will not be described here but an example of application is given in §4.2 and the interested reader can find a detailed presentation of the method in reference [2].

The general structure of AESOP is presented in §2. The parameterization of shape deformations based on Radial Basis Functions (RBF) is presented in §3. It has been recently investigated because it can give rise to a general strategy for the optimization of various shapes in both two and three dimensional applications. This method is directly inspired from a previous work on RBF-based mesh deformation [14]. The examples are presented in §4.

[1] Stands here for AErodynamic Shape OPtimization (Aesop is the name of a legendary Greek fabulist).

2 Gradient-based optimization in AESOP

Concerning optimization based on Computational Fluid Dynamics (CFD), the use of adjoint equations has become a common approach in recent years [3, 4, 5, 6, 7, 9, 10, 15, 16, 18, 21, 25, 27].

There are however different views on the application of the theory of optimal control in CFD [13]. In the so-called continuous approach, the adjoint equations and the expression of the gradient are derived from the Partial Differential Equations (PDEs) that model the flow and from the exact cost function, see [4, 10, 15, 25, 27]. The resulting expressions are then discretized. In the so-called discrete approach, the adjoint equations and gradient expression are obtained from the discretized flow equations and cost function [3, 7, 9, 18].

The discrete approach is preferred here because it can provide the exact gradient of the cost function and constraints being optimized. Otherwise, lack of accuracy can cause a failure of the optimization algorithm in finding a descent direction. There are though possible simplifications in the discrete approach, as shown in one example in paragraph §2.5 or as discussed in [26, 17]. To become more familiar with the adjoint approach applied here consider a simpler problem in the spirit of Giles & Pierce [12]. Let a cost function J be linear with respect to the vector of state variables $\mathbf{w}$,

$$J\left(\mathbf{w}\right) = \mathbf{g}^T \mathbf{w}\,, \tag{2.1}$$

with $\mathbf{g} \in \mathbb{R}^m$ given and $\mathbf{w} \in \mathbb{R}^m$ subject to the state equation

$$\mathrm{A}\mathbf{w} = \mathrm{N}\mathbf{a}\,, \tag{2.2}$$

where $\mathbf{a} \in \mathbb{R}^n$ is the vector of design variables, $\mathrm{A} \in \mathbb{R}^{m\times m}$, and $\mathrm{N} \in \mathbb{R}^{m\times n}$.

Assume that A is nonsingular. The reduced gradient of J, that is, the gradient of the mapping $\mathbf{a} \mapsto J\left(\mathbf{w}\left(\mathbf{a}\right)\right)$, denoted ∇J_a, may be obtained by solving the *sensitivity equations* of the state: given a variation of the control variable $\delta\mathbf{a}$, a corresponding variation of the state $\delta\mathbf{w}$ is defined as the solution to the sensitivity equations

$$\mathrm{A}\,\delta\mathbf{w} = \mathrm{N}\,\delta\mathbf{a}\,, \tag{2.3}$$

which enables us to express the variation of the function J

$$\delta J = \mathbf{g}^T \delta\mathbf{w} \equiv \mathbf{g}^T \mathrm{A}^{-1}\mathrm{N}\,\delta\mathbf{a}\,. \tag{2.4}$$

Therefore, solving the sensitivity equations, once for each component of the vector $\mathbf{a}$, yields the gradient ∇J_a, component by component.

However, rewriting (2.4) as

$$\delta J = \left(\mathrm{N}^T \left(\mathrm{A}^T\right)^{-1} \mathbf{g}\right)^T \delta\mathbf{a}\,, \tag{2.5}$$

reveals that replacing $\left(\mathrm{A}^T\right)^{-1}\mathbf{g}$ in (2.5) by the adjoint state $\mathbf{w}^*$, defined as the solution to

$$\mathrm{A}^T\mathbf{w}^* = \mathbf{g}\,, \tag{2.6}$$

gives an expression for ∇J_a,

$$\nabla J_a = \mathrm{N}^T\mathbf{w}^*\,. \tag{2.7}$$

The cost for computing the gradient by expression (2.7) is one costate solution (2.6) and a matrix-vector product (2.7), instead of m solutions of the sensitivity equations (2.3) when expression (2.4) is used.

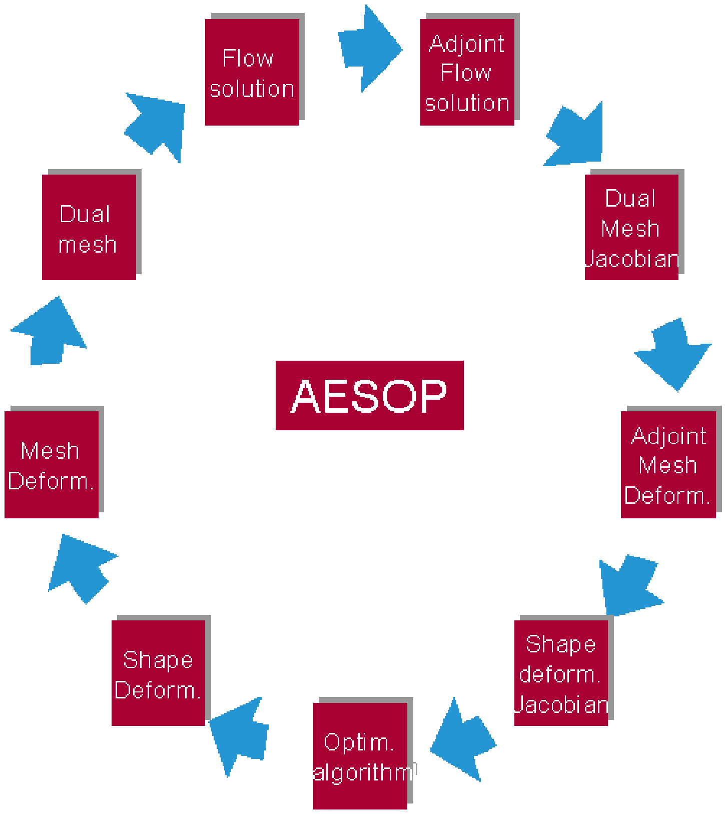

Figure 2.1: Iterative gradient-based aerodynamic shape optimization as implemented in the numerical platform AESOP (“deform.” stands for “deformation”).

The generalization to nonlinear state equations and nonlinear functions J is straightforward:

- $\mathbf{g}$ is the vector of partial derivatives of the function J with respect to the vector of the state variables $\mathbf{w}$.

- A is the Jacobian matrix of the system of discretized state equations with respect to the state variables. The efficient assembly of the products between the transpose of this matrix and costate vectors ($\mathbf{w}^*$) is crucial for solving efficiently equation (2.6). These operations are presented in details in reference [1] when (2.3) is the system of equations obtained by linearization of the Euler equations discretized by a median-dual finite volume scheme [8].

- N is the Jacobian matrix of the system of discretized state equations with respect to the parameters of optimization. In our CFD applications [1], this involves:

  - The pre-processing of the mesh, which is the nodes-to-edges transformation, also described as primal-to-dual transformation; the transposed Jacobian maps gradients with respect to the dual data (the control surfaces attributed to an edge) to gradients with respect to the nodal data (nodal coordinates).
  - The deformation of the mesh, for a given deformation of the shape.

– The parameterization of the shape deformation, for given parameters of optimization.

Section 2.2 presents in more details the operations involved in equation (2.7).

2.1 Optimization algorithms

The optimization algorithms implemented so far in the numerical platform (see Figure 2.1) are textbook gradient-based methods supplemented by a line search algorithm using the Goldstein condition as globalization strategy [19]:

- conjugate gradient from Polak-Ribière;
- quasi-Newton method Broyden-Fletcher-Goldfarb-Shanno (BFGS) with Hessian updates;
- exact augmented Lagrangian method.

The Lagrangian method handles constraints whereas the conjugate gradient and quasi-Newton methods are used for unconstrained optimization. Note that the Lagrangian method approaches the solution of constrained problems of optimization through a sequence of unconstrained optimization problems. The unconstrained problems are approximately solved using either the conjugate gradient or the quasi-Newton algorithm above.

2.2 Functions and gradients computation

The following describes the algorithms or equations involved in the computation of the cost function, or of a constraint, depending on just one flow solution. Such a function is composed of five mappings:

- *Parameterization of shape deformations:* $\mathbf{a}_h \rightarrow \mathbf{d}_h$

  It mapps a vector of design parameters $\mathbf{a}_h$ into a vector of nodal displacements. Consider the mesh node with index i on the shape $\partial\Omega_w$. Its coordinates, denoted $\boldsymbol{x}_i$, are related to the initial shape $(\partial\Omega_w^0)$, through the nodal displacements $\mathbf{d}_h = \{\boldsymbol{d}_i\}_{1\leq i\leq N_w}$ as

  $$\boldsymbol{x}_i = \boldsymbol{x}_i^0 + \boldsymbol{d}_i\,, \quad \text{for } i \in \mathcal{V}(\partial\Omega_w)\,, \tag{2.8}$$

  where $\mathcal{V}(\partial\Omega_w)$ denotes the set of indexes of the mesh nodes on $\partial\Omega_w$. The mapping of the design parameters $\mathbf{a}_h = \{a_i\}_{1\leq i\leq N}$ into displacements $\mathbf{d}_h$ is expressed as

  $$\mathcal{S}_h\left(\mathbf{d}_h, \mathbf{a}_h\right) = \mathbf{0}\,. \tag{2.9}$$

  There are many strategies for the parameterization of shapes, or shapes deformations. Two kinds of parameterizations used here are presented below (see §3).

- *Mesh deformation:* $\mathbf{d}_h \rightarrow \mathbf{X}_h$

  To retain mesh quality, the coordinates of all mesh nodes $\mathbf{X}_h$ must be adapted to the displacements $\mathbf{d}_h$ of the nodes on the part of the boundary being optimized. This is accomplished here by a mesh deformation algorithm, a smoother as in [1] or a method of interpolation based on RBF as in [14]:

  $$\mathcal{M}_h\left(\mathbf{X}_h, \mathbf{d}_h\right) = \mathbf{0}\,. \tag{2.10}$$

- *Re-calculation of the control volumes/surface vectors:* $\mathbf{X}_h \rightarrow \mathbf{n}_h$

  By deforming the mesh we conserve the connectivities between the nodes, but the control volumes of the dual grid and their surface normals $\mathbf{n}_h$ (the vector of all control surface normals) need to be re-computed. Therefore, $\mathbf{n}_h$ is a function of the mesh coordinates, which we denote

  $$\mathbf{n}_h \equiv \mathbf{n}_h \left(\mathbf{X}_h\right) \ . \tag{2.11}$$

  General expressions for the surface vectors of the dual mesh control volumes $\mathbf{n}_h$ are given in 2D and 3D in reference [1].

- *Solution of the discretized Euler equations:* $\mathbf{n}_h \rightarrow \mathbf{w}_h$

  As mentioned previously, the program Edge [8] solves the Euler equations on a dual mesh. The discretized Euler equation in *steady state* yields a system of equations in residual form where the discrete flow state $\mathbf{w}_h$ (the density, velocity and pressure at all nodes) is the unknown and the dual mesh data $\mathbf{n}_h$ parameterize this system[1]:

  $$\mathcal{R}_h \left(\mathbf{w}_h, \mathbf{n}_h\right) = \mathbf{0} \tag{2.12}$$

- *Function evaluation:* $\mathbf{X}_h, \mathbf{n}_h, \mathbf{w}_h \rightarrow J$

  For the sake of the presentation, let us consider a function which minimization involves a reduction of the drag coefficient (C_D) and penalized changes in lift (C_L) and pitching moment (C_m):

  $$\begin{aligned} J\left(\mathbf{w}_h, \mathbf{n}_h, \mathbf{X}_h\right) =& \mu_D C_D \left(\mathbf{w}_h, \mathbf{n}_h\right) + \frac{1}{2}\mu_L \left(C_L\left(\mathbf{w}_h, \mathbf{n}_h\right) - C_L^0\right)^2 + \\ & \frac{1}{2}\mu_M \left(C_m\left(\mathbf{w}_h, \mathbf{n}_h, \mathbf{X}_h\right) - C_m^0\right)^2 \ , \end{aligned} \tag{2.13}$$

  where μ are positive scalars and the upperscript ^{0} denotes the values at initial design. The aerodynamic coefficients (C_D , C_L , C_m), also used in the examples (§4), are defined here as:

  $$\begin{aligned} C_D\left(\mathbf{w}_h, \mathbf{n}_h\right) &= \sum_{i \,\in\, \mathcal{V}(\partial\Omega_w)} \frac{p_i \boldsymbol{n}_i \cdot \boldsymbol{d}_D}{\frac{1}{2}\rho_\infty \boldsymbol{v}_\infty^2 S_{\text{ref}}} \ , \\ C_L\left(\mathbf{w}_h, \mathbf{n}_h\right) &= \sum_{i \,\in\, \mathcal{V}(\partial\Omega_w)} \frac{p_i \boldsymbol{n}_i \cdot \boldsymbol{d}_L}{\frac{1}{2}\rho_\infty \boldsymbol{v}_\infty^2 S_{\text{ref}}} \ , \\ C_m\left(\mathbf{w}_h, \mathbf{n}_h, \mathbf{X}_h\right) &= \sum_{i \,\in\, \mathcal{V}(\partial\Omega_w)} \frac{p_i \boldsymbol{d}_M \cdot \left(\boldsymbol{x}_i - \boldsymbol{O}_{\text{ref.}}\right) \times \boldsymbol{n}_i}{\frac{1}{2}\rho_\infty \boldsymbol{v}_\infty^2 S_{\text{ref}} L_{\text{ref}}} \ , \end{aligned} \tag{2.14}$$

  were $\boldsymbol{d}_D = \boldsymbol{v}_\infty / |\boldsymbol{v}_\infty|$, $\boldsymbol{v}_\infty$ is the far-field air velocity, $\boldsymbol{d}_L$ is an upward oriented unit vector orthogonal to $\boldsymbol{d}_D$, $\boldsymbol{d}_M$ is a unit vector orthogonal to $\boldsymbol{d}_D$ and $\boldsymbol{d}_L$, p_i is the pressure at node i, $\boldsymbol{n}_i$, an element of $\mathbf{n}_h$, is the outward-oriented surface normal at the boundary node i, ρ_∞ denotes the far-field air density, S_{ref} a reference surface, L_{ref} a reference length.

The gradient of J (2.13) with respect to $\mathbf{a}_h$, subject to the constraints (2.9)-(2.12), also called the reduced gradient of J, is denoted ∇J_a. All functions in expressions (2.9)-(2.13) being assumed continuously differentiable[2], the computation of ∇J_a can be formulated as:

[1] The discretized Reynolds Averaged Navier–Stokes equations also depend explicitly on the mesh nodal coordinates $\mathbf{X}_h$.

[2] Neglecting the artificial diffusivities used in the solution procedure of the flow equations [1].

- For a given design $\mathbf{a}_h$ and corresponding mesh $\mathbf{X}_h$ (2.9)-(2.10), dual grid data $\mathbf{n}_h$ (2.11) and discrete flow solution $\mathbf{w}_h$ (2.12), compute the adjoint flow solution $\mathbf{w}_h^*$ by solving (see [1])

$$\left(\frac{\partial \mathcal{R}_h}{\partial \mathbf{w}_h}\right)^* \mathbf{w}_h^* \quad = \quad \left(\frac{\partial J}{\partial \mathbf{w}_h}\right)^T . \tag{2.15}$$

- Calculate the gradient with respect to the dual grid data ∇J_n and the gradient with respect to the nodal coordinates ∇J_X as follows:

$$\nabla J_n \quad = \quad -\left(\frac{\partial \mathcal{R}_h}{\partial \mathbf{n}_h}\right)^* \mathbf{w}_h^* + \left(\frac{\partial J}{\partial \mathbf{n}_h}\right)^T , \tag{2.16}$$

$$\nabla J_X \quad = \quad \left(\frac{\mathrm{d}\mathbf{n}_h}{\mathrm{d}\mathbf{X}_h}\right)^T \nabla J_n + \left(\frac{\partial J}{\partial \mathbf{X}_h}\right)^T . \tag{2.17}$$

- Solve the adjoint mesh deformation equation, see [1, 14] for details:

$$\left(\frac{\partial \mathcal{M}_h}{\partial \mathbf{X}_h}\right)^* \mathbf{X}_h^* \quad = \quad -\nabla J_X \, . \tag{2.18}$$

- Calculate the gradient with respect to the shape deformations ∇J_d:

$$\nabla J_d \quad = \quad \left(\frac{\partial \mathcal{M}_h}{\partial \mathbf{d}_h}\right)^* \mathbf{X}_h^* \, . \tag{2.19}$$

- Solve the adjoint parameterization equation, details are given below for two examples of parameterization (§3):

$$\left(\frac{\partial \mathcal{S}_h}{\partial \mathbf{d}_h}\right)^* \mathbf{d}_h^* \quad = \quad -\nabla J_d \, . \tag{2.20}$$

- Finally, calculate the reduced gradient with respect to the parameters of design:

$$\nabla J_a \quad = \quad \left(\frac{\partial \mathcal{S}_h}{\partial \mathbf{a}_h}\right)^* \mathbf{d}_h^* \, , \tag{2.21}$$

where * denotes the adjoint of an operator or the unknown in an adjoint equation.

2.3 Example of geometrical constraint

Problems of shape optimization usually include constraints on the geometry. We give here the expressions used in the examples (§4) for computing the thickness of an airfoil section or its volume. Let us define the thickness of an airfoil: suppose, for the sake of the presentation, that the leading and trailing edges are on the x-axis as in Figure 2.2, we call $f\,(x, \mathbf{a}_h)$ the distance, measured perpendicularly to the x-axis, between the upper and lower sides of the airfoil and define the thickness of the airfoil as the maximum of this function:

$$T\,(\Gamma) = \max_{x_0 \le x \le x_1} f\,(x, \mathbf{a}_h) \tag{2.22}$$

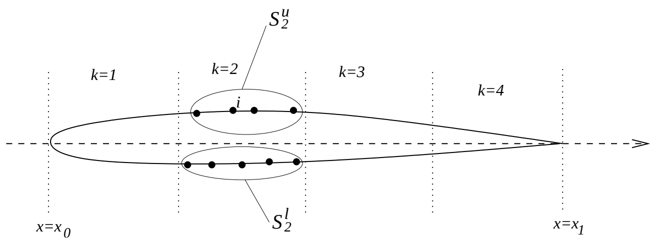

Figure 2.2: Sketch showing the principle of the streamwise discretization used for computing the thickness of airfoils or streamwise sections of wings. Here $m = 4$, $|S_2^u| = 4$, $|S_2^l| = 5$, but in the applications §4 $m = 20$

Applying definition (2.22) on the discretization of an airfoil, or a section of a wing is straightforward. However, it is not a differentiable function with respect to the parameters of design.

As an alternative we can approximate the maximum norm by the p-norm ($p < \infty$):

$$\|f\|_p = \left[\int_{x_0}^{x_1} |f(x, \mathbf{a}_h)|^p \mathrm{d}x\right]^{\frac{1}{p}} \tag{2.23}$$

because:

$$\|f\|_\infty = \lim_{p\to\infty} \|f\|_p \,, \tag{2.24}$$

The norm (2.23) is differentiable with respect to the shape even if the shape is discretized. The integral in (2.23) is approximated, after discretization of the x-axis in m intervals (Figure 2.2), here using the trapezoidal quadrature:

$$T_h = \left[\frac{1}{2}hf_1^p + \sum_{k=2}^{m-1} hf_k^p + \frac{1}{2}hf_m^p\right]^{\frac{1}{p}} \,, \tag{2.25}$$

where f_k approximates $f(x_k, \mathbf{a}_h)$:

$$f_k = \|\boldsymbol{u}_k - \boldsymbol{l}_k\| \,, \tag{2.26}$$

with $\boldsymbol{u}_k$ and $\boldsymbol{l}_k$ are the mass centers of the points on the upper or lower side of the airfoil. Denoting by S_k^u, respectively, S_k^l, the set of indices of the nodes in the interval k on the upper side, respectively, on the lower side, then:

$$\boldsymbol{u}_k = \frac{1}{|S_k^u|}\sum_{i\in S_k^u} \boldsymbol{x_i} \,, \quad \text{and} \quad \boldsymbol{l}_k = \frac{1}{|S_k^l|}\sum_{i\in S_k^l} \boldsymbol{x_i} \tag{2.27}$$

For $p = 1$ expression (2.23) is the cross-section area and expressions (2.25)-(2.27) give an approximation of this area.

Gradient of the thickness constraint

We detail here the expressions that are necessary in order to calculate explicitly the gradient of the thickness. Suppose that the coordinates of node i on the airfoil is displaced by $\delta\boldsymbol{x}_i$. We derive the first variation of the thickness (2.25) due to the perturbation $\delta\boldsymbol{x}_i$:

$$\delta T_h = \begin{cases} hT_h^{(1-p)} f_k^{p-1}\delta f_k \text{ for } \; i \in S_k^{u \text{ or } l} \text{ with } 2 \le k \le m-1 \\ \dfrac{1}{2}hT_h^{(1-p)} f_k^{p-1}\delta f_k \text{ for } \; i \in S_k^{u \text{ or } l} \text{ with } k = 1 \text{ or } m \end{cases} \tag{2.28}$$

and the first perturbation of the local thickness δf_k is for a node i on the upper side:

$$\delta f_k = \frac{1}{|S_k^u|} \frac{\boldsymbol{u}_k - \boldsymbol{l}_k}{f_k} \cdot \delta \boldsymbol{x}_i \tag{2.29}$$

or for a node i on the lower side:

$$\delta f_k = -\frac{1}{|S_k^l|} \frac{\boldsymbol{u}_k - \boldsymbol{l}_k}{f_k} \cdot \delta \boldsymbol{x}_i \tag{2.30}$$

which gives expressions for calculating $\partial T_h / \partial \boldsymbol{x}_i$ for all nodes on the airfoil. Using relations (2.20)-(2.21) gives the gradient of T_h with respect to all design variables.

2.4 Aspects of parallel computations

The numerical platform AESOP is parallelized in two ways. Either the loop is parallelized so that the flow and adjoint flow solution are computed "sequentially", but several at a time. Or, the flow and adjoint flow solutions are obtained on a partitioned domain, that is solving in parallel one flow (2.12) or adjoint flow problem (2.15) at a time. The first option enables to obtain a speed up even for small problems where there is no advantage in partitioning the computational domain. The second approach is of course advantageous for large computational meshes. The computation of ∇J_X (2.16)-(2.17) is performed in Edge following the scheme described in reference [1], which is also parallelized.

2.5 Discrete against continuous adjoint sensitivities

The purpose of this example is to show that the discrete approach for calculating gradients of the cost function or constraints (2.15)-(2.21) can sometimes be simplified without important change in the results of the optimization.

For instance the calculation of gradients requires adjoints of the mesh deformation (2.18) (one adjoint per cost function and constraint that depend on mesh deformations). If the shape optimization is based on inviscid flow equations, skipping the solution of equation (2.18) may not affect the optimization as Figure 2.3 shows the results of optimization for the M6 wing obtained by a complete gradient calculation 'Optim1-1' compared to the same optimization where the adjoint mesh equation is not solved 'Optim1-2'. The first optimization required a total of 155 flow equivalent solutions[3] against 163 for the second optimization, which indicates that the loss of accuracy did not destroy the convergence of the numerical optimization. Regarding the computer cost there is no savings here because the CPU time due to solving the adjoint mesh equations is less than the cost of the additional flow computations required in 'Optim1-2'. This could be different in optimization involving the RANS equations because the cost of deforming the RANS mesh can be considerably higher than deforming inviscid meshes [14] and the cost of solving the adjoint of the mesh deformation equation has a similar amplitude.

[3] Flow equivalent solution means here one flow or adjoint flow solution. The cost for one flow solution in this example is 1520 seconds CPU, and 24 seconds CPU for the solution of one adjoint mesh deformation equation.

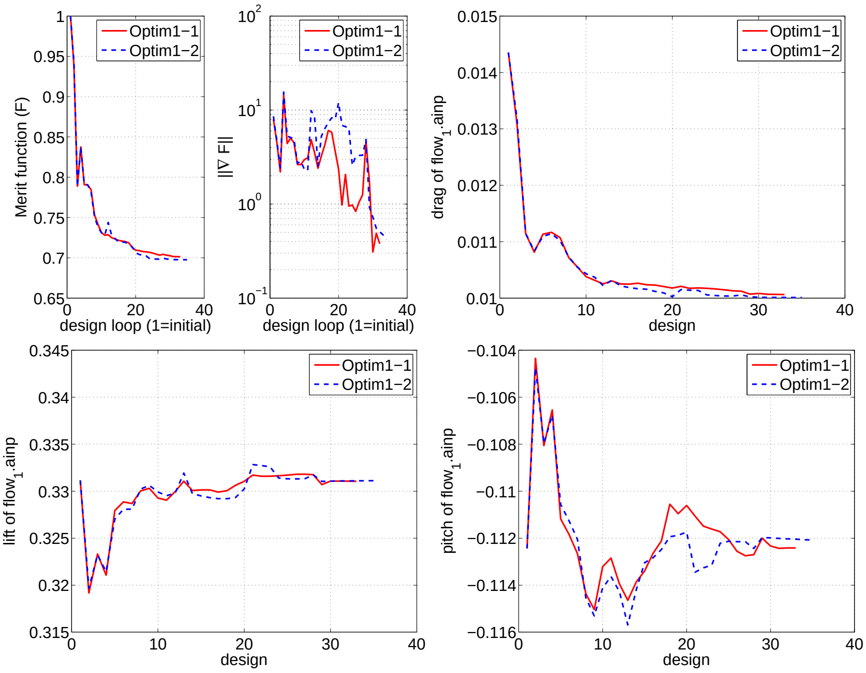

Figure 2.3: M6 wing optimization with complete gradient calculation 'Optim1-1' against a gradient calculation without adjoint mesh solution 'Optim1-2'.

3 RBF parameterization of shape deformations

The choice of the RBF method for the parameterization of shapes in aerodynamic shape optimization can be motivated by the following properties:

1. scattered data interpolation - Shapes are generally described by clouds of points without topological information. The primal idea of the parameterization defined below (§3.1) is to interpolate the displacements of control points scattered in $\mathbb{R}^2$ or $\mathbb{R}^3$ in order to define displacements everywhere.

2. regularization - In gradient-based optimization, the parameters are updated given shape gradients of the aerodynamic coefficients, which have no known regularity, an example is given in Figure 3.1. The parameters in our RBF parameterization are the displacements of control points. When the RBF is such that it can resolve the high frequencies of the shape gradient, the interpolated displacements would inevitably create wavy shape deformations. Therefore, a regularized approximation as defined in §3.2 is in this case preferred to an interpolant, see Figure 3.2.

3. extrapolation - Aeroelastic shape optimization requires deforming the structure model according to changes in the parameters that describe the shape. The same RBF can be used for the parameterization of the wetted surface deformations and for the deformation of the inner structure model.

4. 'arbitrary' dimensions - The RBF formulation involves a distance between the location of data, which is invariant with respect to the dimension.

The RBF interpolation and approximation (regularized RBF) are presented in the next two sections. This type of representation of shape deformations often requires to set boundaries to the region where the deformations are defined [14], for example defining cut-off functions as in section 3.3. Finally, the computation of gradients, the steps (2.20)-(2.21) above, is described in section §3.4.

3.1 RBF interpolant

For the sake of the presentation, the parameters of design used here $\{a_i \in \mathbb{R}\,,\, 1 \leq i \leq n\}$ are the displacements in y-direction of n control points. Denoting by $\boldsymbol{x} = [x, y, z, ...]$ the coordinates in $\mathbb{R}^d$ of a material point, and the superscript ^{0} denoting its position at rest, a displacement mapping $\boldsymbol{d}_h(\boldsymbol{x}) : \mathbb{R}^d \rightarrow \mathbb{R}^d$ is defined here, generalizing the definition (2.8) to all points in $\mathbb{R}^d$:

$$\boldsymbol{x} = \boldsymbol{x}^0 + \boldsymbol{d}_h\left(\boldsymbol{x}^0\right) \,. \tag{3.1}$$

The mapping is further defined by imposing the following interpolation condition at the control points:

$$d_y\left(\boldsymbol{x}_i\right) = a_i \,,\, 1 \leq i \leq n \,, \tag{3.2}$$

where $d_y(\mathbf{x})$ is the y-component of $\boldsymbol{d}_h$. The displacements $d_y(\boldsymbol{x})$, represented by a RBF such as the one derived in reference [14], are expressed as:

$$d_y\left(\boldsymbol{x}\right) = \sum_{i=1}^{n} c_{y,i}\phi_\epsilon\left(\boldsymbol{x} - \boldsymbol{x}_i\right) + \sum_{k=1}^{d+1} b_{y,k}Q_k\left(\boldsymbol{x}\right) \,, \tag{3.3}$$

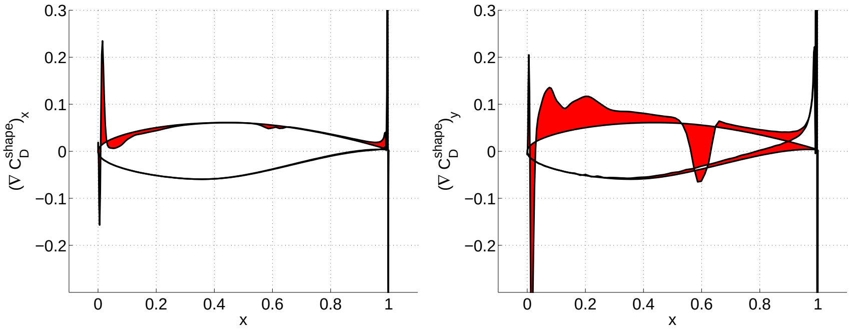

Figure 3.1: Gradient of the drag coefficient with respect to the nodal coordinates on the airfoil, including flow derivatives: x-coordinates (left) and y-coordinates (right)

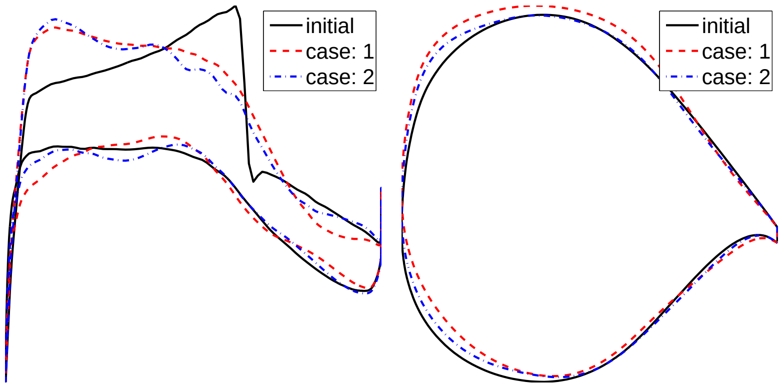

Figure 3.2: Comparison of inviscid optimization of airfoils using RBF parameterization: with regularization (case:1) and without regularization (case:2) - pressure coefficients ($-C_p$) on the left and geometries on the right.

where the radial function $\phi_\epsilon(\mathbf{r})$ is defined by a real valued function $\Phi(t)$ (see Table 3.1 for examples) via

$$\phi_\epsilon(\mathbf{r}) = \Phi(\epsilon\|\mathbf{r}\|) \ , \tag{3.4}$$

and $\|.\|$ denotes here the Euclidean norm, ϵ being the shape factor, a strictly

Basis function	$\Phi(t)$ where $t \geq 0$
Wendland ($W_{3,1}$)	$(4t+1)\times(1-t)^4$, if $0 \leq t \leq 1$, 0 otherwise
Inverse multi-quadric (IMQ)	$\frac{1}{\sqrt{1+t^2}}$
Multi-quadric (MQ)	$\sqrt{1+t^2}$
Gaussian (GS)	e^{-t^2}
Thin Plate Splines (TPS2)	$t^2\log(t)$, if $t > 0$, 0 otherwise
Thin Plate Splines (TPS4)	$t^4\log(t)$, if $t > 0$, 0 otherwise

Table 3.1: Example of “RBF” functions.

positive real number. For 3D applications ($d = 3$) the monomials Q are:

$$\begin{aligned} Q_1(\boldsymbol{x}) &= 1 \\ Q_2(\boldsymbol{x}) &= x \\ Q_3(\boldsymbol{x}) &= y \\ Q_4(\boldsymbol{x}) &= z \end{aligned} \tag{3.5}$$

The coefficients of the RBF (3.3), $\{\mathbf{c}_h;\mathbf{b}_h\} = \{c_{y,i} \in \mathbb{R}\,,\, 1 \le i \le n\,;\, b_{y,k} \in \mathbb{R}\,,\, 1 \le k \le d+1\}$, need to fulfill the system of $n + d + 1$ equations:

$$\begin{bmatrix} \mathrm{A} & \mathrm{P} \\ \mathrm{P}^{\mathrm{T}} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{c}_y \\ \mathbf{b}_y \end{bmatrix} = \begin{bmatrix} \mathbf{a}_h \\ \mathbf{0} \end{bmatrix}, \tag{3.6}$$

where A is the interpolation matrix:

$$\begin{aligned} \mathrm{A} &= \{A_{ij}\}_{\substack{1 \le i \le n \\ 1 \le j \le n}}\,, \\ A_{ij} &= \phi_\epsilon(\boldsymbol{x}_i - \boldsymbol{x}_j)\,,\ 1 \le i \le n\,,\, 1 \le j \le n\,, \end{aligned} \tag{3.7}$$

and P is the matrix of constraints

$$\begin{aligned} \mathrm{P} &= \{P_{ik}\}_{\substack{1 \le i \le n \\ 1 \le k \le d+1}}\,, \\ P_{ik} &= Q_k(\boldsymbol{x}_i)\,,\ 1 \le i \le n\,,\, 1 \le k \le d+1\,. \end{aligned} \tag{3.8}$$

The system (3.6) is not singular if the coordinates of the control points $\boldsymbol{x}_i$ are distinct and if A defined by (3.7) is not singular. The last condition is fulfilled for certain functions Φ, see reference [29] and the examples in Table 3.1. The $d+$ 1 constraints are imposed in reference [14] in order to obtain interpolations that are translation and rotation invariant, in addition to fulfilling the interpolation conditions (3.2).

3.2 Regularized RBF

The interpolation presented above may induce wiggles when increasing the number of control points n in (3.2)-(3.3). Even small oscillations on the geometry, as in the example shown in Figure 3.2, affect the pressure in a way that could trigger flow separation or (laminar to turbulent) transition, for example. A possible cure is to penalize a norm of the RBF (3.3) while approximating the displacements of the control points. By this approach the relation (3.2) does not necessarily hold. The standard approach, in order to define an approximation d_y of the form (3.3), is to observe that the solution of the system of equations (3.6) above is solution of the least-square approximation problem with constraints:

$$\begin{cases} \displaystyle\min_{\mathbf{c}_y \in \mathbb{R}^n, \mathbf{b}_y \in \mathbb{R}^{d+1}} \sum_{i=1}^{n} \left[d_y(\boldsymbol{x}_i) - a_i\right]^2 \\ \displaystyle\text{subject to } \sum_{i=1}^{n} c_{y,i}\, q(\boldsymbol{x}_i) = 0, \end{cases} \tag{3.9}$$

where $d_y(\boldsymbol{x})$ is defined by (3.3) and $q(\boldsymbol{x})$ is any first degree polynomial.

Adding a penalty on the (native space) norm of the RBF yields the following problem of approximation:

$$\begin{cases} \displaystyle\min_{\mathbf{c}_y \in \mathbb{R}^n, \mathbf{b}_y \in \mathbb{R}^{d+1}} \sum_{i=1}^{n} \left[d_y\left(\boldsymbol{x}_i\right) - a_i\right]^2 + \beta \sum_{i=1}^{n}\sum_{j=1}^{n} c_{y,i} c_{y,j} \phi_\epsilon\left(\boldsymbol{x}_i - \boldsymbol{x}_j\right) \\ \text{subject to } \displaystyle\sum_{i=1}^{n} c_{y,i}\, q\left(\boldsymbol{x}_i\right) = 0, \end{cases} \tag{3.10}$$

where d_y and q are defined as in problem (3.9) and $\beta \in \mathbb{R}^+$. If the matrix A defined by (3.7) is not singular, the solution of the problem (3.10) is given by the solution of the system:

$$\begin{bmatrix} \mathrm{A} + \beta \mathrm{I} & \mathrm{P} \\ \mathrm{P}^{\mathrm{T}} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{c}_y \\ \mathbf{b}_y \end{bmatrix} = \begin{bmatrix} \mathbf{a}_h \\ \mathbf{0} \end{bmatrix}, \tag{3.11}$$

The choice of the parameter β is of course determining the “quality” of the approximation in terms of the pointwise error:

$$e_{y,i} = d_y\left(\boldsymbol{x}_i\right) - a_i\,,\ 1 \leq i \leq n \tag{3.12}$$

and in terms of the oscillations of d_y. Small values of β give small pointwise errors $|e_{y,i}|$, with a risk of overshooting, whereas large values of β potentially eliminate overshooting at the cost of increased pointwise errors. When approximating functionals, there are no known optimal values of β but an attempt can be made to reduce the influence of “noise” and therefore reduce the integrated error (based on the L^2 norm for example) by cross-validation, as found in Rudholm and Wojciechowski [22].

3.3 Boundary conditions

In certain applications, such as the 3D examples presented in §4.3 or in reference [14], it is necessary to limit the support of the deformation defined by RBF (3.3). Given a plane $\mathcal{P}$ (defined by a point $\boldsymbol{x}_P$ and a normal vector $\boldsymbol{n}_P$), the displacements in the half-space $\mathcal{P}^-$ (defined by $\boldsymbol{x}_P - \boldsymbol{n}_P \in \mathcal{P}^-$) should be zero. We define a cut-off function $\varphi_P\left(\boldsymbol{x}\right)$ that is 0 for points in $\mathcal{P}^-$ and 1 away from the plane $\mathcal{P}$ by:

$$\begin{cases} \varphi_P\left(\boldsymbol{x}\right) = 1 - \Phi^W\left(\dfrac{\epsilon_P\left(\boldsymbol{x} - \boldsymbol{x}_P\right)\cdot \boldsymbol{n}_P}{\|\boldsymbol{n}_P\|}\right), \quad \text{for } \boldsymbol{x} \notin \mathcal{P}^- \\ \varphi_P\left(\boldsymbol{x}\right) = 0\,, \quad \text{for } \boldsymbol{x} \in \mathcal{P}^- \end{cases} \tag{3.13}$$

where Φ^W is the Wendland function from Table 3.1. A displacement d_y^P, null in $\mathcal{P}^-$, would for example be used instead of d_y, the latter being defined by (3.9) or (3.10):

$$d_y^P\left(\boldsymbol{x}\right) = \varphi_P\left(\boldsymbol{x}\right) d_y\left(\boldsymbol{x}\right)\ . \tag{3.14}$$

The same approach can be used in order to impose pointwise constraints.

3.4 Computation of shape gradients

The following gives the details of the operations (2.20)-(2.21) in cases where the parameterization equation (2.9) is defined by the RBF approach described above. In order to make the example more general suppose that regularization is used (§3.2) as well as boundary conditions (§3.3). Suppose that we are given the gradient of J with respect to the y-coordinates displacements on the shape $\mathrm{d}J/\mathrm{d}\mathbf{d}_y$, calculating the gradient of J with respect to the RBF coefficients $\mathbf{c}_y$ and $\mathbf{b}_y$ requires the following operations:

$$\frac{\partial J}{\partial \lambda_i^y} = \sum_{j=1}^{N_w} \varphi_P\left(\boldsymbol{x}_j\right) \psi_i\left(\boldsymbol{x}_j\right) \frac{\mathrm{d}J}{\mathrm{d}d_{y,j}}\,, \quad 1 \le i \le n+d+1\,, \tag{3.15}$$

where we included eventual cut-off functions in $\varphi_P\left(\boldsymbol{x}_j\right)$ as defined in §3.3 by relation (3.13). The following notations were used in (3.15):

$$\begin{aligned} \lambda_i^y &= c_{y,i}\,, \quad \text{and } \psi_i\left(\boldsymbol{x}\right) = \phi_\epsilon\left(\boldsymbol{x}-\boldsymbol{x}_j\right)\,, \quad 1 \le i \le n \\ \lambda_{n+i}^y &= b_{y,i}\,, \quad \text{and } \psi_{n+i}\left(\boldsymbol{x}\right) = Q_i\left(\boldsymbol{x}\right)\,, \quad 1 \le i \le d+1 \end{aligned} \tag{3.16}$$

Calling $\mathrm{d}J/\mathrm{d}\lambda^y$ the vector of all partial derivatives of J with respect to the RBF coefficients (3.15)-(3.16), we can calculate the gradient of J with respect to the vector of all control points (y-)displacements $\mathbf{a}_h$ by solving the system:

$$\left[\begin{array}{cc} \mathrm{A}+\beta\mathrm{I} & \mathrm{P} \\ \mathrm{P}^{\mathrm{T}} & 0 \end{array}\right] \left[\begin{array}{c} \mathrm{d}J/\mathrm{d}\mathbf{a}_h \\ \mathbf{0} \end{array}\right] = \left[\begin{array}{c} \mathrm{d}J/\mathrm{d}\lambda^y \\ \mathbf{0} \end{array}\right]\,, \tag{3.17}$$

4 Examples

All examples below are solved in AESOP using the augmented Lagrangian algorithm (see §2.1). The accuracy of the gradients computed by adjoint flow equations as exposed above is an important aspect of these optimizations. An estimation of the accuracy can be done comparing "adjoint gradients" with finite difference approximations (see for example [1, 2]). In Figure 4.1 the gradients of the drag, lift and pitch, used in the multipoint optimization test case below (see §4.1), are computed by the procedure above and by finite differences approximations where the perturbations of the design variables vary between 10^{-4} and 10^{-7}. The gradients obtained by the two methods argee within $1-2\%$ at design point 1, and within 0.5% at design point 2, at least for perturbation parameters smaller or equal than 10^{-5}. The difference of "accuracy" between the two design points probably depends on the difference in shock strength. The shock is stringer at design point 1 than at design point 2, which involves that the artificial viscosity is has a more important role at design point 1 thus leading to larger errors because the artificial viscosity was not differentiated when deriving the adjoint flow equations used here (see reference [1].

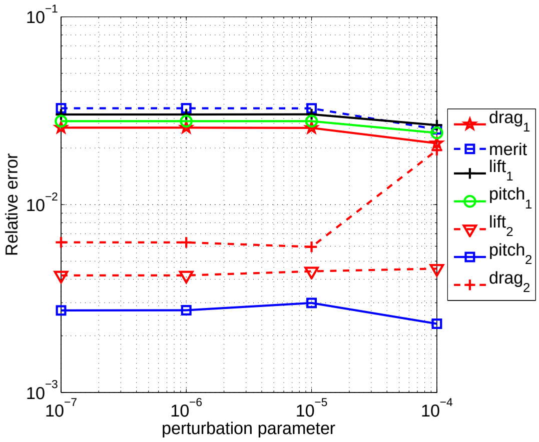

Figure 4.1: Comparison of the gradients at first iteration obtained by adjoint method vs finite differences with different perturbation of the design variables - The subscript indicates the design point (1 or 2).

4.1 Inviscid multipoint optimization

This is an example of airfoil design at cruise for Mach number 0.716, a study performed in the European project CESAR. The goal was here to minimize the drag due to shocks, in contrast to the next example (see §4.2) where the main objective is to reduce the viscous friction. It is therefore sufficient to use the Euler equations in order to model the flow. Note, however, that the performance of the final design was verified in the project, as it is customary, solving the RANS equations around the initial and final airfoils, but these results are not presented here. Constraints are imposed on the thickness, the pitching moment and the lift, whereas the goal is to minimize the drag. The

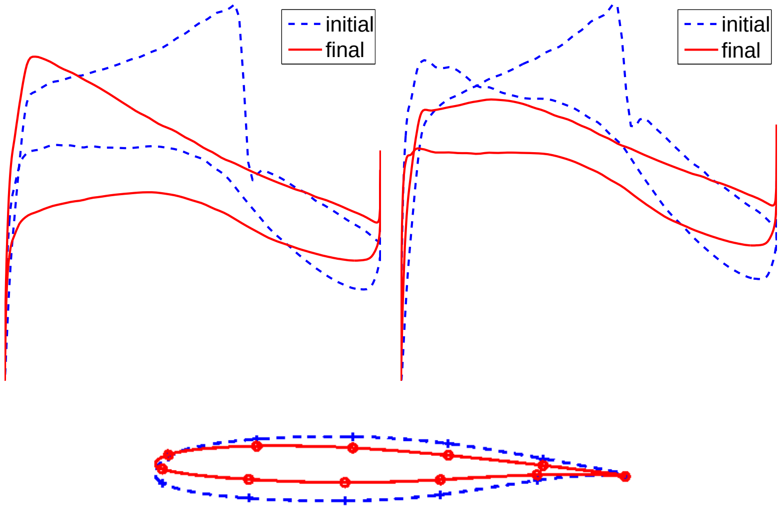

Figure 4.2: Multipoint inviscid airfoil optimization at Mach=0.716 - Pressure coefficients $-C_p$ at design point 1 (left) and design point 2 (right), unscaled airfoils (bottom), where the dashed blue line is for the initial design and the solid red line is for the final design.

multipoint approach consists here in defining a problem of optimization that depends on more than one flight condition, for instance two angles of attack $\{\alpha_1, \alpha 2\}$. Here α_2 gives the required lift coefficient at Mach= 0.716 , but the wave drag is only 20 drag counts. Chosing to minimize the drag at a higher angle of attack was the strategy chosen by the author in order to have better sensitivities of the cost function with respect to changes in the geometry. The cost function is thus the wave drag at α_1, an angle of attack at which the initial drag is about 80 drag counts. The lift coefficients at both design points are imposed to be larger or equal than the values for the baseline. Regarding the pitching moments, the project required values larger than those of the baseline, so that a lower bound was finally chosen at -0.12. Note that the pitching moments based only on the pressure can be very different from the values that include the effect of viscosity, so that the main objective is to increase the pitching moments (the convention here is negative pitch for nose down moment), not to achieve precise values.

To summarize, the optimization problem is formulated as:

$$\begin{cases} \min\limits_{\Gamma} \dfrac{C_{D1}}{C_{D1}^0} \quad \text{subject to} \\ C_{D2} \leq C_{D2}^0 \\ C_{L1} \geq C_{L1}^0 \\ C_{L2} \geq C_{L2}^0 \\ C_{m1} \geq C_{m1}^{min} \\ C_{m2} \geq C_{m2}^{min} \\ t \geq t^{min} \end{cases} \tag{4.1}$$

where the superscript ^{0} indicates the values at initial design (baseline), min indicates values chosen that are different from the baseline: $t^{min} = 9\%$, $C_{m1}^{min} = -0.12$, $C_{m2}^{min} = -0.12$. The parameterization of the airfoil Γ, or its deformations, is the regularized RBF method described in section 3.2. The radial function ϕ_ϵ in expression (3.3) is the Gaussian (see Table 3.1) with a shape factor $\epsilon = 2.5$, a regularization parameter $\beta = 1.1$ in (3.10)-(3.11), and 11 control points distributed on the surface. Note that imposing the constraints on the pitching moment, in the multipoint approach, has allowed here to achieve a better design at both design points $\{\alpha_1, \alpha 2\}$ than the designs that could be obtained by performing only single point optimization at α_1 and α_2, with less constraining conditions on the pitching moment. Not only the wave drag is reduced of 70 drag counts at α_1 and 12 drag counts at α_2, the lift at α_2 was increased as well as the pitching moments at both design points, all other constraints being satisfied (Figure 4.3). A possible explanation is that non-linear problems of optimization of this kind have local minima and imposing constraints away from the values at initial design may help moving away from local optimum solutions.

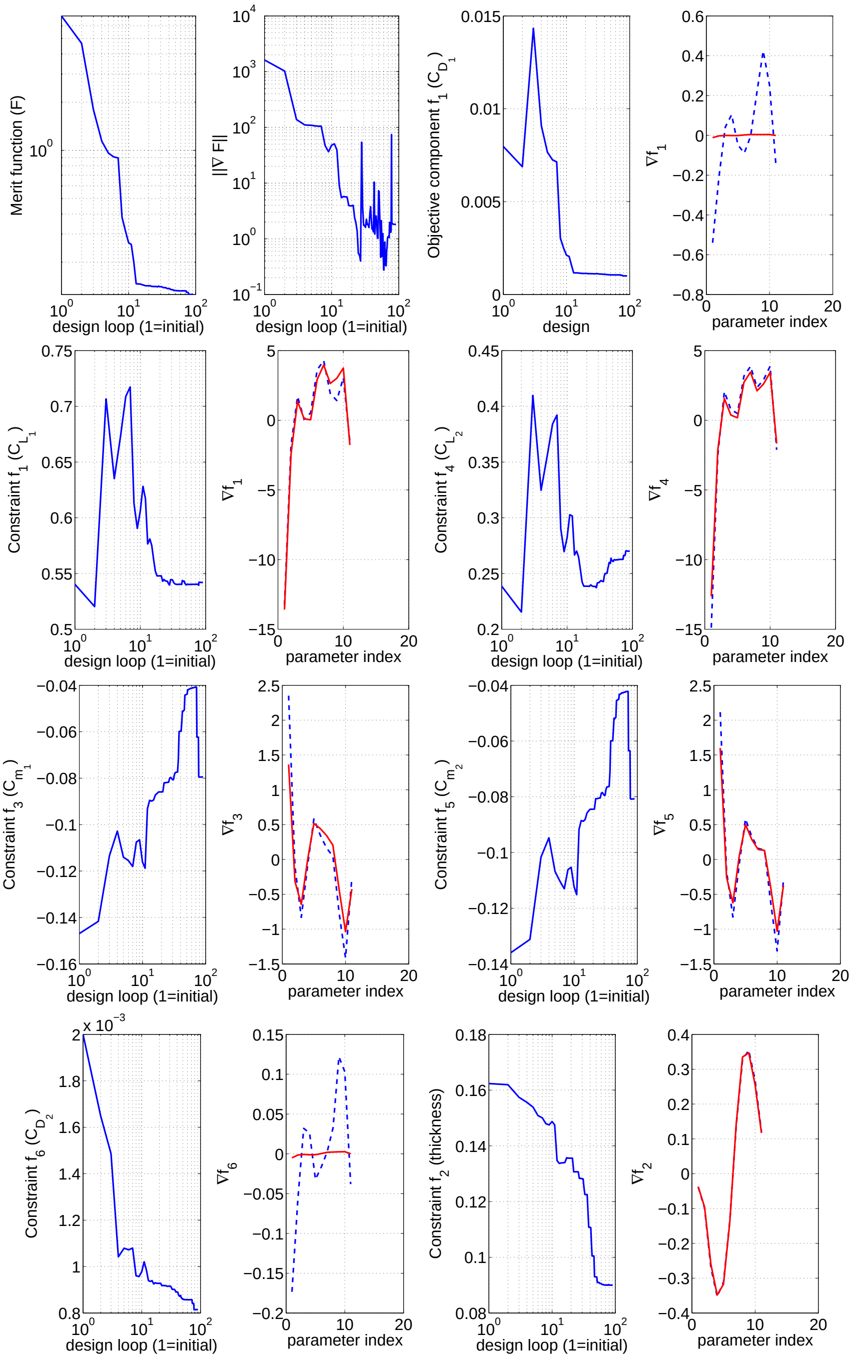

Figure 4.3: Multipoint inviscid airfoil optimization at M=0.716 - Merit function and gradient norm (first row left), drag and gradient at design point 1 (first row right), lift at design point 1 (second row left), lift at design point 2 (second row right), pitching moment at design point 1 (third row left), pitching moment at design point 2 (third row right), drag at design point 2 (bottom left), thickness (bottom right). The gradients are represented at initial (dashed blue lines) and final design. (solid red lines).

4.2 Natural Laminar Flow design optimization

This is an example of laminar airfoil design at cruise conditions, the Mach number is 0.372 and the Reynolds number is $12E6$. This work was also performed for the European project CESAR. In constrast to the previous example the goal is to reduce the viscous drag. The common strategy in the area, and the one applied here, is to increase the laminar portion of the flow. For this purpose the flow equations (Euler in this example) are complemented with a model of the propagation of disturbances in the laminar part of the boundary layer. The reason is that the mechanism of laminar-to-turbulent transition is generally governed by the growth of such disturbances. Therefore, damping the growth of those disturbances is expected to delay downstream the location of this transition, increasing the laminar portion of the flow where the friction coefficient is much lower than in the turbulent area. The strategy used here has been implemented for the design of laminar airfoils using adjoint sensitivities in reference [2]. For this purpose the platform AESOP communicates with additional solvers for the boundary layer flow and stability equations (and their adjoints). Following this strategy the cost function is the energy of a selected disturbance (E) propagated in the boundary layer on the suction side. Constraints are imposed on the thickness, the pitching moment and the lift. In order to avoid increasing the wave drag, around 10 drag counts for the baseline, a penalty term is added to the energy in the cost function. This is preferred to a usual constraint because a design with a slightly higher wave drag could be acceptable, if the laminar region of the flow is sufficiently increased. The lift and pitching moment coefficients are imposed to be larger or equal than the values for the baseline, the angle of attack being such that the initial lift is the minimum required lift.

To summarize, the optimization problem is formulated as:

$$\begin{cases} \min\limits_{\Gamma} \quad \log(E) + 0.1\dfrac{C_D}{{C_D}^0} \quad \text{subject to} \\ C_L \geq {C_L}^0 \\ C_m \geq {C_m}^0 \\ t \geq t^{min} \end{cases} \tag{4.2}$$

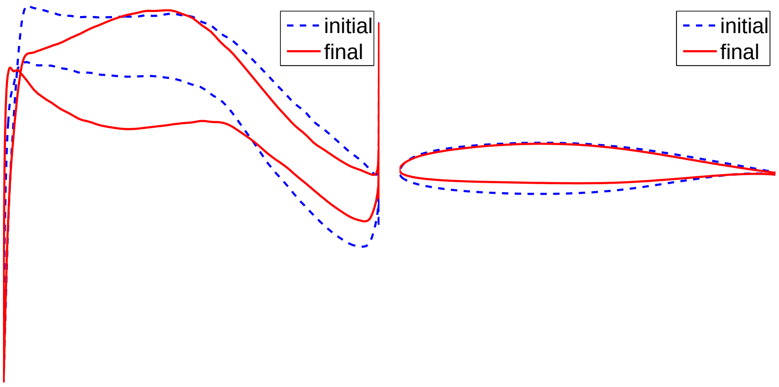

Figure 4.4: NLF airfoil optimization at Mach=0.312, Re=12E6 - Pressure coefficients $-C_p$ (left) and unscaled airfoils (right), where the dashed blue line is for the initial design and the solid red line is for the final design.

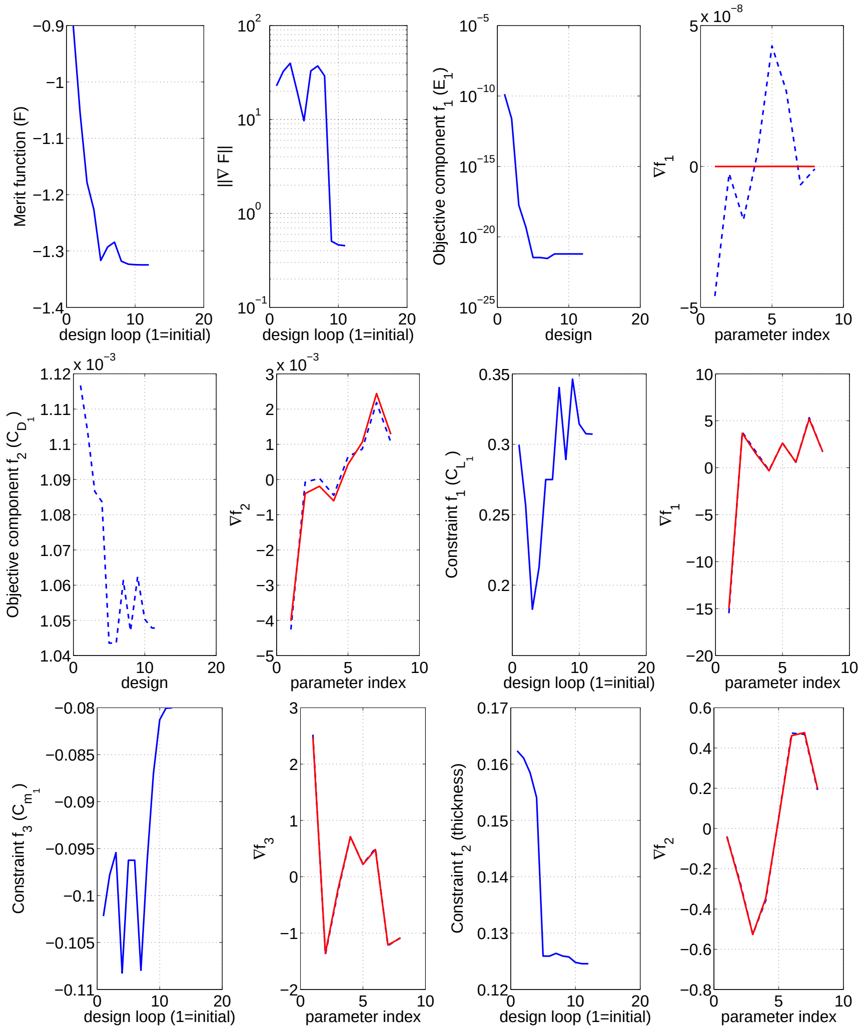

Figure 4.5: NLF airfoil optimization at Mach=0.312, Re=12E6 - Merit function and gradient norm (first row left), disturbance energy and gradient (first row right), wave drag (2nd row left), lift (2nd row right), pitching moment (bottom left), thickness (bottom right). The gradients are represented at initial (dashed blue lines) and final design (solid red lines).

where the superscript ^{0} indicates the values at initial design (baseline) and min indicates a different reference value than that of the baseline: $t^{min} = 12\%$. The parameterization of the airfoil Γ deformations is the regularized RBF method described in §3.2. The radial function ϕ_ϵ in expression (3.3) is the Gaussian (see Table 3.1) with a shape factor $\epsilon = 1.7$, a regularization parameters $\beta = 0.1$ in (3.10)-(3.11), and 8 control points distributed on the surface. As explained earlier, the computation of the energy of a disturbance involves here the solution of the flow in the laminar part of the boundary layer and the solution of the transport equations for the disturbance in the boundary layer (here the Parabolized Stability Equations). In order to calculate efficiently and accurately the gradients, the adjoint of the boundary layer and of the stability equations are solved, in addition to the adjoint of the Euler flow equations [2]. The reduction of the energy function (Figure 4.5) is due to the favorable pressure gradient on the suction side, which damps the growth of this disturbance (see Figure 4.4). Note that the use of the logaritm of the energy is necessary due to variations of several orders of magnitude of the function E from the baseline design to the final design (Figure 4.5).

4.3 Inviscid transonic wing optimization

The ONERA M6 wing, described in the AGARD report [24], is optimized here at Mach number 0.8395 and angle of attack $\alpha = 3.06^o$ using the Euler equations as flow model. The parameterization of the deformations of the wing Γ is the interpolation RBF method described in section 3.1 (without regularization). A similar optimization was carried out in reference [1] using a parameterization of the twist, camber and thickness along the spanwise direction. Another difference with those previous results is that the lift and pitching moment coefficients are here considered as constraints, which was not the case in [1] where changes in these coefficients were only penalized. The radial functions ϕ_ϵ in Table 3.1 are compared: $\mathrm{W}_{3,1}$ ('Optim1'), GS ('Optim2'), TPS2 ('Optim3') and TPS4 ('Optim4'). The shape factor is $\epsilon = 0.17$ for a chord that varies between $10m$ at the root to $5.5m$ at the tip (see Figure 4.6) and there are 52 control points distributed on the surface. Concerning the boundary conditions of the flow domain, a symmetry plane (x-z) is placed at the root of the wing. The root section geometry is fixed through a cut-off function as described in section 3.3 where $\boldsymbol{n}_P = (0, 1, 0)$ (orthogonal to the plane x-z) and $\epsilon_P = 0.33$.

To summarize, the optimization problem is formulated as:

$$\begin{cases} \min\limits_{\Gamma} \dfrac{C_D}{{C_D}^0} \quad \text{subject to} \\ C_L \geq {C_L}^0 \\ C_m \geq {C_m}^0 \\ t_{20\%} \geq t^0_{20\%} \\ t_{50\%} \geq t^0_{50\%} \end{cases} \tag{4.3}$$

where the superscript ^{0} indicates the values at initial design (baseline), $t_{20\%}$ and $t_{50\%}$ indicate the thicknesses of the streamwise wing sections at 20% and 50% of span.

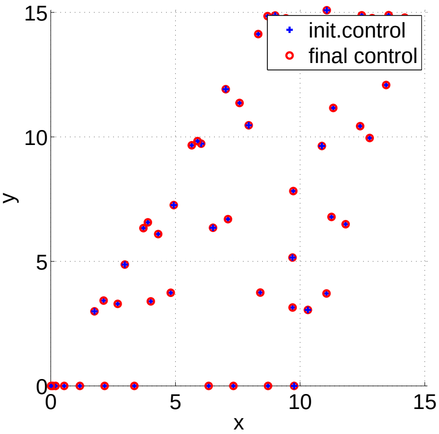

Figure 4.6: Distribution of the control points on the M6 wing (on both sides).

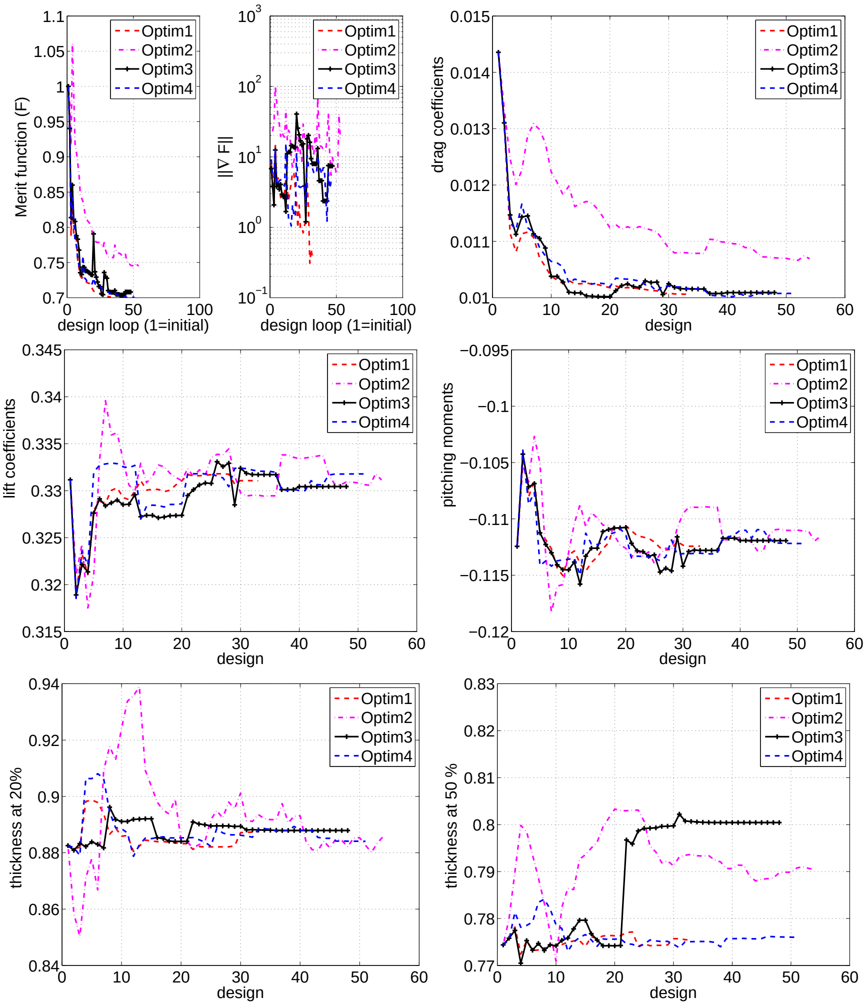

Figure 4.7: Optimization - Influence of the parameterization of the M6 wing deformations: Wendland (Optim1), Gauss (Optim2), TPS2 (Optim3), TPS3 (Optim4).

The trend for all four cases is visible in Figures 4.7-4.8: the drag is reduced due to the weakening of the shock, while the lift and pitching moment coefficients are kept to their initial values. The new shapes perform even better than in reference [1] without reducing the thickness of the wing. Unless for the Optim2 case, where the drag at final design is larger than in the three other cases, there is no large difference between the aerodynamic performances of the designs obtained by the three parameterizations used in Optim1, Optim3 and Optim4 (see Merit function, drag, lift and pitch in Figure 4.7). Comparing the airfoils from Optim1 (Figure 5.1) and Optim2 (Figure 5.2) shows important differences, for example in trailing edges thickness and leading edges radius, depending on the methods of parameterization (the Optim3 and Optim4 airfoils, not shown here, show similar trends as Optim1).These results are very different from the ones obtained in reference [1] because the reduction in drag is here around 40 drag counts, without violation of the constraints whereas the reduction obtained in [1] was at best 30 drag counts with up to 1% violation of the lift. This improvement is due to the shape parameterization used here. The use of RBF enables to change the shape of the "airfoils" without being limited to changes in twist and camber. Note that the drag induced by the lift for an elliptic lift distribution is $C_L^2/\pi AR$, where AR is the aspect ratio of the M6 wing (3.85), which is 90 drag counts for $C_L = 0.33$. The wings obtained by optimization (Optim1, Optim3 and Optim4) are thus quite closed to the theoretical optimal wing with the same aspect ratio.

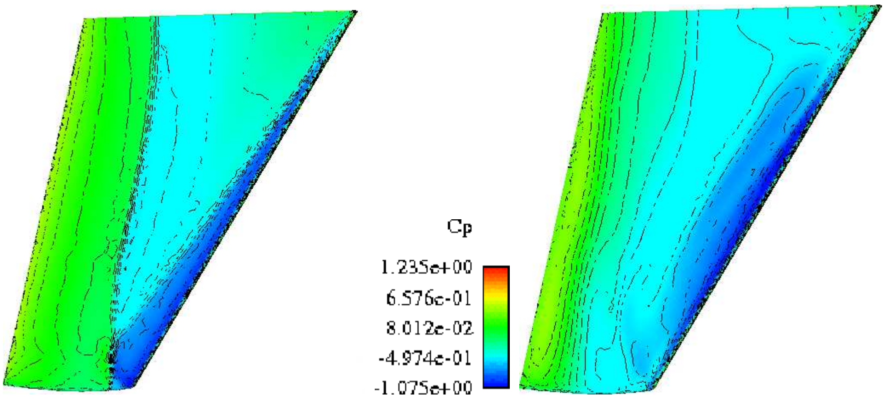

Figure 4.8: Pressure coefficients and iso-Mach lines of the M6 wing (left) and of the optimized 'Optim1' (right), at Mach=0.84, angle of attack of 3 degrees.

5 Summary and outlook

AESOP is a platform for large scale aerodynamic shape optimization and for the development of multidisciplinary design optimization (MDO). Large scale refers here to problems of optimization with many design parameters or with computationaly intensive functions evaluations. The main advantage of the program is to take fully advantage of the adjoint approach developed in recent years in CFD, for instance via the coupling with the unstructured flow and adjoint flow solver in Edge [1]. Several developments can improve the performance or broaden the fields of applications. The performance could for example be improved using other optimization algorithms, for example the Method of Moving Asymptotes from Svanberg [28] and NLPQLP (a Sequential Quadratic Programming implementation) from Schittkowski [23], which is the goal of current investigations.

Typical problems of aerodynamic shape optimization in aeronautics concern the design or re-design of airfoils, wings, turbine blades, canals, pylons, fairing of surfaces intersections, among others. Devising a strategy of parameterization that gives satisfying results for the optimization of all these shapes is not straightforward but it seems that the properties of the Radial Basis Functions (RBF) presented above could help building a general stategy.

An adjoint RANS solver is also being developed in Edge [11]. The capability to take into account viscous turbulent flows in the optimization will enable the optimization of systems where the viscous effects are dominant, such as high-lift systems. Another important aspect is that the use of the RANS equations instead of Euler will improve the reliability of the results of optimization. For instance, the pressure distribution around an optimized shape obtained with Euler may well lead to flow separation althgough the aerodynamic coefficients computed by the inviscid flow model can indicate that the shape has a low drag coefficient and the right lift.

Finally, aeroelasticity is being integrated in the aerodynamic shape optimization framework presented here, starting by superposing static deformations due to loads and shape deformations due to optimization.

Acknowledgements

The author wants to the reviewers for their constructive comments. Many thanks to Martin Berggren, professor at the university of Umeå, Stefan Jakobsson from FCC, as well as Jan Pralits, Ardeshir Hanifi and Mattias Chevalier from FOI, for the work done together on shape optimization. Finally, part of this work received fundings from PSCI (VINNOVA, the Royal Institute of Technology in Stockholm, Uppsala University and a consortium of Swedish industries) and the European research projects in aeronautics: Aeroshape, Eurolift II, SUPERTRAC, HISAC, CESAR and NACRE.

Bibliography

[1] O. Amoignon and M. Berggren. Adjoint of a median-dual finite-volume scheme: Application to transonic aerodynamic shape optimization. Technical Report 2006-013, Department of Information Technology, Uppsala University, Uppsala, Sweden, 2006.

[2] O.G. Amoignon, J.O. Pralits, A Hanifi, M. Berggren, and D.S. Henningson. Shape Optimization for Delay of Laminar-Turbulent Transition. *AIAA Journal*, 44(5):1009–1024, 2006.

[3] W.K. Anderson and D.L. Bonhaus. Airfoil design on unstructured grids for turbulent flows. *AIAA Journal*, 37(2):185–191, 1999.

[4] O. Baysal and K. Ghayour. Continuous adjoint sensitivities for optimization with general cost functionals on unstructured meshes. *AIAA Journal*, 39(1):48–55, 2001.

[5] F. Beux and A. Dervieux. Exact-gradient shape optimization of a 2D Euler flow. Technical Report RR-1540, INRIA Sophia Antipolis, 2004 Route des Lucioles, 06360 Valbonne, France, 1991.

[6] G. Bugeda and E. Oñate. Optimum aerodynamic shape design for fluid flow problems including mesh adaptivity. *Int. J. Numer. Meth. Fluids*, 30:161–178, 1999.

[7] G.W. Burgreen and O. Baysal. Three-dimensional aerodynamic shape optimization using discrete sensitivity analysis. *AIAA Journal*, 34(9):1761–1170, 1996.

[8] P. Eliasson. Edge, a Navier–Stokes solver, for unstructured grids. Technical Report FOI-R–0298–SE, Swedish Defence Research Agency, Stockholm, November 2001.

[9] J. Elliot and J. Peraire. Aerodynamic design using unstructured meshes. *AIAA Paper*, (96-1941), 1996.

[10] O. Enoksson and P. Weinerfelt. Numerical methods for aerodynamic optimization. Proceedings to the 8th International Symposium on Computational Fluid Dynamics, 5-10 Sept., Bremen, 1999.

[11] N. Gardberg and O. Amoignon. Development of the discrete adjoint RANS for a median-dual finite volume scheme - Application in aerodynamic shape optimization, 2008. Unpublished preprint, FOI.

[12] M.B. Giles and N.A. Pierce. An introduction to the adjoint approach to design. *Flow, Turbulence and Combustion*, 65:393–415, 2000.

[13] M.D. Gunzburger. *Perspectives in Flow Control and Optimization.* Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2002.

[14] S. Jakobsson and O. Amoignon. Mesh Deformation using Radial Basis Functions for Gradient-based Aerodynamic Shape Optimization. *Computers and Fluids*, 36(6):1119–11136, July 2007.

[15] A. Jameson and S. Kim. Reduction of the adjoint gradient formula for aerodynamic shape optimization problems. *AIAA Journal*, 41(11), 2003.

[16] B. Mohammadi. A new optimal shape procedure for inviscid and viscous turbulent flows. *Int. J. Numer. Meth. Fluids*, 25:183–203, 1997.

[17] B. Mohammadi. Dynamical approaches and incomplete gradients for shape optimization and flow control. *AIAA Journal*, (99-3374), 1999.

[18] S.K. Nadarajah. *The discrete adjoint approach to aerodynamic shape optimization.* PhD thesis, Aeronautics and Astronautics Department, University of Stanford, 2003.

[19] J. Nocedal and S. Wright. *Numerical Optimization.* Springer Series in Operations Research, 1999.

[20] O. Pironneau. *Optimal Shape Design for Elliptic Systems.* Springer Verlag, 1984.

[21] J. J. Reuther, A. Jameson, J. J. Alonso, M. J. Rimlinger, and D. Saunders. Constrained multipoint aerodynamic shape optimization using an adjoint formulation and parallel computers, part 1. *J. Aircraft*, 36(1):51–60, 1999.

[22] J. Rudholm and A. Wojciechowski. A method for simulation based optimization using radial basis functions. Technical report, Department of Mathematical Sciences, Division of Mathematics, Chalmers University of Technology, Göteborg University, Göteborg, Sweden, August 2007.

[23] K. Schittkowski. NLPQLP: A new Fortran implementation of a sequential quadratic programming algorithm for parallel computing. Technical report, Department of Mathematics, University of Bayreuth, 2001.

[24] V. Schmitt and F. Charpin. Pressure distributions on the ONERA-M6-WING at transonic mach numbers. In *Experimental Data Base for Computer Program Assessment*, pages B1–1–B1–44. AGARD-AR-138, May 1979.

[25] B.I. Soemarwoto. *Multi-Point Aerodynamic Design by Optimization.* PhD thesis, Delft University of Technology, Faculty of Aerospace Engineering, P.O. Box 5058, 2600 GB Delft, Netherlands, 1996.

[26] O. Soto and R. Löhner. Cfd shape optimization using an incomplete-gradient adjoint formulation. *International Journal for Numerical Methods in Engineering*, 51:735–753, 2001.

[27] C. Sung and J.H. Kwon. Accurate aerodynamic sensitivity analysis using adjoint equations. *AIAA Journal*, 38(2):243–250, 2000.

[28] K. Svanberg. The method of moving asymptotes—a new method for structural optimization. *International Journal for Numerical Methods in Engineering*, 24:359–373, 1987.

[29] H. Wendland. *Scattered Data Approximation.* Cambridge monographs on applied and computational mathemetics, 2005.

Appendix

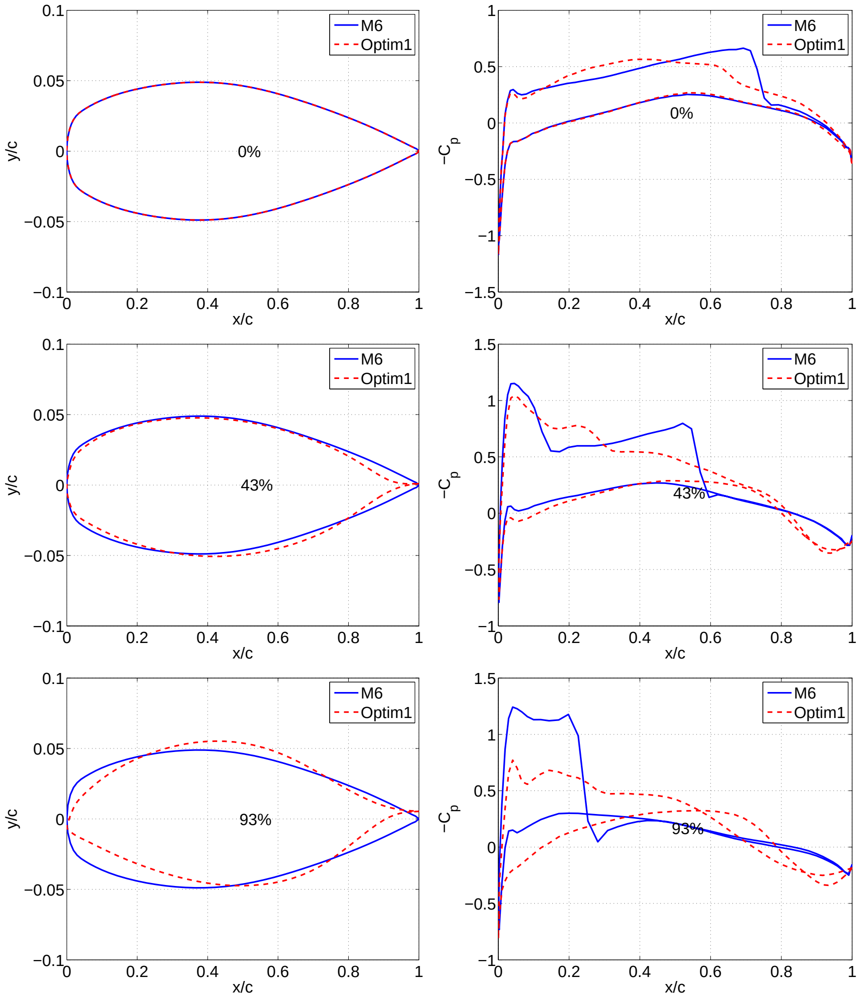

Figure 5.1: Optimization - Comparison of the M6 wing shape and Cp with Optim1 wing at 0, 43 and 93 percent of span.

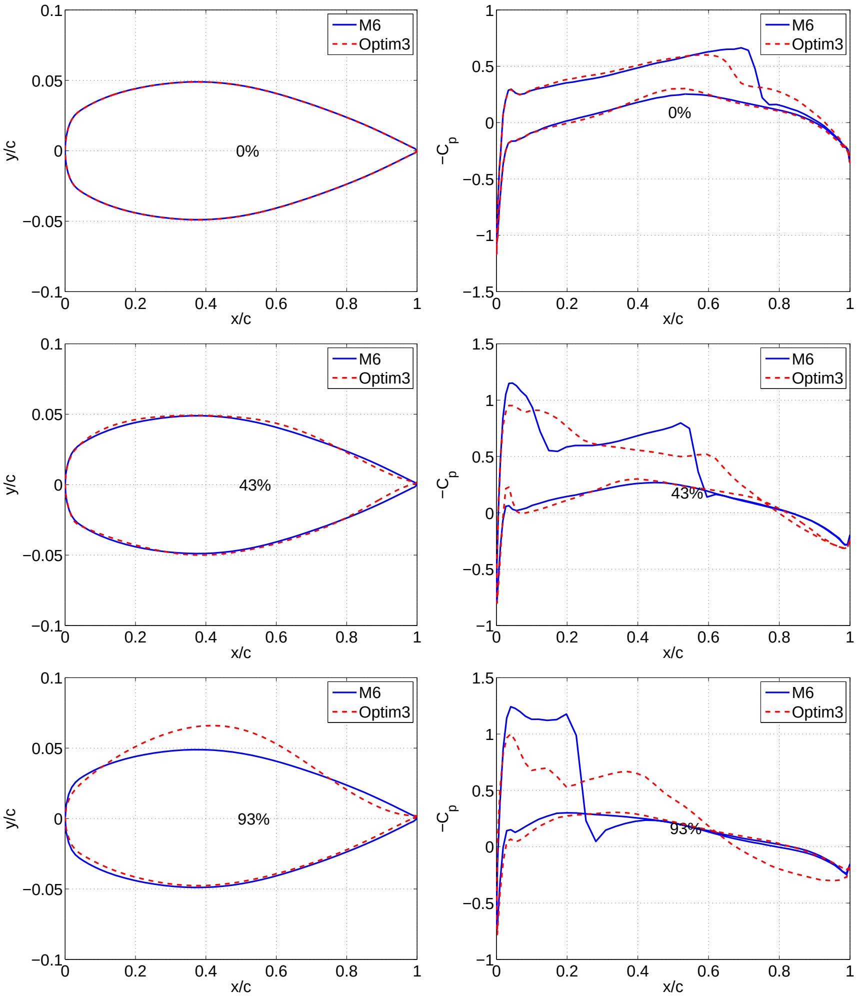

Figure 5.2: Optimization - Comparison of the M6 wing shape and Cp with Optim2 wing at 0, 43 and 93 percent of span.