ONERA
THE FRENCH AEROSPACE LAB

## UNIVERSITÉ DE NANTES

Synthèse des travaux en vue d'obtenir l'Habilitation à Diriger des Recherches
Spécialité : Mécanique, génie mécanique, génie civil.

—

# CONTRIBUTIONS TO DISCRETE ADJOINT METHOD IN AERODYNAMICS FOR SHAPE OPTIMIZATION AND GOAL-ORIENTED MESH ADAPTATION

—

Mémoire préparé et soutenu par
**Jacques PETER**

JURY :
**M. Frédéric ALAUZET**
Directeur de Recherche, INRIA, Rapporteur
**M. Richard DWIGHT**
Professeur, Université de Delft, Rapporteur
**M. Nicolas GAUGER**
Professeur, Université de Kaiserslautern, Rapporteur
**M. François JAUBERTEAU**
Professeur, Université de Nantes, Examinateur
**M. Jens-Dominik MÜLLER**
Professeur, Queen Mary University of London, Examinateur
**M. Michel VISONNEAU**
Directeur de Recherche, CNRS, Examinateur

le 25 septembre 2020

*(Le professeur Aronnax, Conseil et Ned Land s'inquiètent de la marche du Nautilus menacée par les mouvements d'un iceberg)*

Je me promenai pendant quelques instants du salon à la bibliothèque. Mes compagnons assis se taisaient. Je me jetai bientôt sur un divan, et je pris un livre que mes yeux parcoururent machinalement. Un quart d'heure après, Conseil s'étant approché de moi me dit :
– Est-ce intéressant ce que lit monsieur ?
– Très intéressant, répondis-je.
– Je le crois. C'est le livre de monsieur que lit monsieur !
– Mon livre ?
En effet je tenais à la main l'ouvrage des *Grands Fonds sous-marins*

*Vingt mille lieux sous les mers (IIe partie, chapitre XV).* Jules Verne

# Remerciements

Je remercie tout d'abord Frédéric Alauzet, DR INRIA, et les professeurs Richard Dwight et Nicolas Gauger qui ont expertisé ce document. Aussi bien sûr, Jens-Dominik Müller, professeur, Michel Visonneau, DR CNRS, et François Jauberteau, professeur à l'Université de Nantes qui ont contribué à l'évaluation des recherches. Je sais gré encore à Michel Visonneau et François Jauberteau de m'avoir guidé dans les démarches administratives et d'avoir organisé la soutenance en cette période compliquée de l'année 2020.

Je remercie chaleureusement les étudiants qui ont travaillé pendant tout ou partie de leur thèse à l'ONERA sur la méthode adjointe pour la simulation aérodynamique, Frédérique Drullion, Chi-Tuân Pham, Antoine Dumont, Meryem Marcelet, Maxime Nguyen-Dinh, Manuel Bompard, Sébastien Bourasseau et Andrea Resmini. Et encore les deux étudiants de TU Delft venus à l'ONERA réaliser leur stage de *Master of Science*, Giovanni Todarello et Floris Vonck. Je leur sais gré, et de leur entrain au travail, et de leur exigence scientifique.

Je dois associer à leurs travaux Jean-Antoine Désidéri, Directeur de Recherches INRIA, consultant scientifique à l'ONERA, qui a contribué à l'encadrement ou à l'expertise des travaux de presque tous ces étudiants, apportant sa vue très complète sur les méthodes numériques, sa connaissance des méthodes d'optimisation, des écoulements supersoniques et encore des finesses de la langue anglaise. Je lui en sais gré de même que je sais gré à Florent Renac de ses indications sur le sujet de la section 2.2.

Les travaux de recherche présentés dans les pages qui suivent reposent sur le module de calcul de gradient du grand code aérodynamique $elsA$ de l'ONERA auquel ont largement contribué Florent Renac et Pierre Trontin. Par ailleurs, beaucoup de calculs et développements ont été rendus possibles grâce à l'assistance efficace de Sébastien Heib et Julien Mayeur. Je les remercie vivement.

# Acknowledgements

I warmly thank Frédéric Alauzet, DR INRIA, and professors Richard Dwight and Nicolas Gauger who reviewed this document. Also of course, Jens-Dominik Müller, professor, Michel Visonneau, DR CNRS, and François Jauberteau, professor at Nantes University who contributed to the evaluation of the researches. I am also grateful to Michel Visonneau and François Jauberteau for organizing the defense in this complex period of time.

I am very grateful to the students who performed part or all of their PhD thesis at ONERA on the adjoint method for aerodynamics: Frédérique Drullion, Chi-Tuân Pham, Antoine Dumont, Meryem Marcelet, Maxime Nguyen-Dinh, Manuel Bompard, Sébastien Bourasseau and Andrea Resmini. Also to the two students of TU Delft who did their internship of *Master of Science* at ONERA, Giovanni Todarello and Floris Vonck. I did really appreciate their scientific and technical commitment.

I must thank here Jean-Antoine Désidéri, Research Director at INRIA, scientific consultant at ONERA, who contributed to the supervision of many of these students or to the reviewing process of their PhDs. His broad expertise on numerical methods, his knowledge about optimisation methods, supersonic flows and subtleties of English were extremely useful. I also need to thank Florent Renac for his indications on the topic of section 2.2.

The research activities presented hereafter rely on the adjoint module of the $elsA$ code for compressible CFD. Florent Renac and Pierre Trontin made great contributions to this module. Sébastien Heib and Julien Mayeur fixed numerous small or complex issues making many direct and adjoint simulations feasible. I am very grateful to them.

# Contents

# Introduction

When I was a student in “Math Spé”, the Maths teacher one day made a remarkable hand-drawing on the blackboard. He drew a frame of reference and a perfect paraboloïd – let us say it was

$$x^2 + 2y^2 - z - 6 = 0 \qquad -$$

before drawing a set of three or four parallel oblique planes – let us say these were

$$x + y + z = K \qquad \text{planes.}$$

The question was to find Min $(x+y+z)$ on the paraboloïd. By some kind geometrical evidence (this was one of the few non-demonstrated properties during this year), he inferred that the limiting plane crossing the paraboloïd when decreasing $K$ would be tangent to it at the point we were looking for. So there was a $\lambda$ coefficient, that we could call Lagrange multiplier, such that, at the target solution point, the gradients of the two surfaces would be proportional,

$$\lambda \begin{pmatrix} 2x \\ 4y \\ -1 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

which quickly lead to the calculation of $\lambda$, then $x$ and $y$ and finally $z$. This $\lambda$ was something nice and efficient that had made a tough problem simple... and I had then no idea I would later spend about fifteen years calculating and manipulating adjoint vectors.

The next time I heard about Lagrange multipliers was in 2001 as Airbus and ONERA prepared a joint project on discrete adjoint and local optimization (and it quickly appeared to me that the $\lambda$ would not be a single real number but possibly a very large vector). Looking back at what had been done before, I realized that Computational Fluid Dynamics (CFD) based aerodynamic optimization was about as old as CFD. It actually started in the 70’s and 80’s with simple parametrizations of aerofoils and wings, and gradient based local optimization where the gradient of the functions of interest were calculated by finite differences [71, 72, 156]. This field then underwent a very deep change in 1988: Fifteen years after a famous article by Pironneau, that introduced shape optimization via optimal control for incompressible flows [151], Jameson introduced the continuous adjoint method for aerodynamics [78]. This method allows the calculation of the objective and constraints sensitivities with respect to (w.r.t.) design parameters at a cost scaling with the number of functions to differentiate which arose a strong interest in the CFD community. A few years later, Shubin and Frank demonstrated a discrete version of Jameson’s adjoint method [164, 165, 54] that they called the “implicit gradient approach” and that, soon after, was called “discrete adjoint method”. It did not require the manipulation of partial differential equations (p.d.e.) of fluid dynamics with a mapping of the fluid domain and was hence much simpler from a mathematical point of view... but involved more operations to differentiate the discrete fluxes of the direct calculation. This was probably the beginning of an unlimited debate between those in favor of continuous adjoint (linearize-then-discretize) and those in favor of discrete adjoint (discretize-then-linearize) ([139]§2.7) formulations.

Both methods have assets but it seems today that the discrete approach is the dominant one for large CFD codes. Probably the main reason for this is the consistency of the gradients w.r.t. finite differences at fixed mesh size that is important for shape optimization and other applications like gradient-enhanced metamodeling [1]. Besides, not only this theoretical property is desirable but NASA Langley teams have demonstrated at the end of the 90's that gradients with almost perfect accuracy could indeed be obtained, even in the quite complex case of (RANS) flows and unstructured meshes [5, 123], at the price of an extremely rigorous process for the development of the discrete adjoint code. Finally the possibility to use Automatic Differentiation (AD) tools to linearize the residual without bug may be an important second asset for the discrete approach when working in a large CFD code.

In 2002, it was actually a discrete adjoint module, named $elsA/Opt$ [145], that had to be added to the $elsA$ code [23, 25, 24]. This activity was started for aircraft "options", that is fluxes, join and boundary conditions generally used for external aerodynamics simulations. The counterpart work for turbomachinery started a few years later and, as discrete adjoint development went on in the CFD Department, all the other components of a local optimization chain were developed or improved at the Applied Aerodynamics Department [2] of ONERA : parametrization, volume mesh deformation, descent method and advanced post-processing providing partial derivatives of the functions of interest.

At the same time, the engineers involved in development of the adjoint module of the $elsA$ code have developed related research activities on implicit stages for the adjoint equation, extension of classical discrete adjoint to hovering rotor and optimization based on the so-called "shape gradient". These research activities are summarized in the first chapter of the manuscript.

In 2005, roughly speaking, it appeared evident that the sensitivities of the volume mesh w.r.t. the design parameters would become the memory bottleneck of the adjoint method. At the cost of calculating the Jacobian of the discrete residual w.r.t. volume mesh nodes coordinates, it was possible to get rid of the mesh sensitivities and, more generally, of the design parameters in the CFD adjoint module. The outputs of this module are then the total derivative of the functions of interest w.r.t. volume mesh node coordinates that can later be processed to calculate wall mesh sensitivities ("shape gradients") then the sensitivities w.r.t. one or several sets of design variables. In 2008, after these total derivatives were available in $elsA$, a very significant effort has been undertaken to use them for "goal-oriented" mesh refinement, that is mesh-refinement aiming at the accurate calculation of a specific output functional. These research activities are summarized in the second chapter of this document.

---

[1] On this subject, see references in [12, 13]

[2] The teams of these two former Departments are now gathered in a single Department called "Département Aérodynamique, Aéroélasticité, Acoustique" (DAAA)

## Nomenclature

| | |
|---|---|
| AoA | Angle of attack (external flow) |
| $CD$ | Drag coefficient |
| $CDp$ | Pressure drag coefficient |
| $CDw$ | Wave drag coefficient |
| $CL$ | Lift coefficient |
| $CLp$ | Pressure lift coefficient |
| $d$ | Normal component of shape gradient $d\overline{J}/dX_S$ |
| $D(X, X_S)$ | Implicit volume mesh deformation |
| $D_\alpha$ | Domain of $\alpha$, vector of design variables |
| $E$ | Total energy of the fluid per unit mass |
| $(\mathcal{F}_x, \mathcal{F}_y, \mathcal{F}_z)$ | Euler fluxes in $(x, y, z)$ directions |
| $\mathrm{J}_k$ | Functions of interest as function of $W$ and $X$ |
| $J_k$ | Functions of interest as function $X$ |
| $\overline{J}_k$ | Functions of interest as function $X_S$ |
| $\mathcal{J}_k$ | Functions of interest as function of $\alpha$ |
| $k$ | Curvature |
| $\mathrm{M}_\infty$ | Far-field Mach number (external flow) |
| $n_W$ | Number of cells times number of equations (size of $R$ and $W$) |
| $n_X$ | Number of mesh points (size of $X$) |
| $\mathcal{P}$ | Operator removing components of $dJ/dX$ |
| $P_a$ | Mean value of stagnation pressure over the wall(s) |
| $n_\alpha$ | Number of desing variables |
| $R(W, X)$ | Explicit stage of the numerical scheme |
| R | Helicopter blade radius |
| Re | Reynolds number |
| $s$ | Curvilinear abscissa |
| $S$ | Akima's spline fitted to $X_S$ |
| $\mathcal{S}$ | Smoothing operator for $d\overline{J}/dX_S$ or its normal component |
| $\overline{V}$ | Velocity $\overline{V} = (u, v, w)$ |
| $W$ | Conservative variables $W = (\rho, \rho u, \rho v, \rho w, \rho E)$ |
| $X$ | Mesh |
| $X_S$ | Wall mesh |
| $(x, y, z)$ | Physical coordinates |

| | |
|---|---|
| $\alpha$ | Design variables |
| $\Gamma_k$ | Adjoint vector (column) of function $\overline{J}_k$ for mesh deformation $D$ |
| $\Lambda_k$ | Adjoint vector (column) of function $J_k$ for scheme $R$ |
| $\theta[J]_m$ | Local indicator for $J$-oriented mesh refinement (based on $\mathcal{P}(dJ/dX)$) |
| $\overline{\theta}[J]_m$ | Local indicator for $J$-oriented mesh refinement (based on spatial mean of $\mathcal{P}(dJ/dX)$) |
| $\theta[J]$ | Estimator of suitability of current mesh to evaluate $J$ (based on $\mathcal{P}(dJ/dX)$) ) |
| $\overline{\theta}[J]$ | Estimator of suitability of current mesh to evaluate $J$ (based on spatial mean of $\mathcal{P}(dJ/dX)$) |
| $\mu[J]_m$ | Norm of $\mathcal{P}(dJ/dX_m)$ |
| $\mu[J]$ | Mean of $\mu[J]_m$ values over the mesh |
| $\Psi$ | Modified total variation of curvature of a current wall mesh $X_S$ |
| $\rho$ | Density |

# Chapter 1

# Discrete gradient calculation methods for shape optimization

This chapter starts with a general subsection (§I.1) that provides all the classical equations of discrete gradient method for finite volume (FV) CFD for compressible flows (§I.1.1) to (§I.1.7) [112, 139]. This subsection goes on with two technical pages (§I.1.8) summarizing the fluxes, models and boundary conditions that have been linearized with respect to state variable and/or volume-mesh coordinates in the $elsA/Opt$ module [145]. Its ends (§I.1.9) with a note about mesh convergence of discrete adjoint fields.

On the contrary, the next three subsections, (§I.2) to (§I.4), refer to specific research activities. A family of implicit stages that has been defined by me for the solution of steady state flows [134] is presented in (§I.2). Their description includes reference to stability analysis carried out with Drullion [40, 138] and extension to the solution of direct differentiation and discrete adjoint equations. Subsection (§I.3) summarizes the work done by Dumont [41, 42] and his supervisors to extend ONERA's discrete adjoint capability from aircraft mechanical formulation, to the "absolute velocity/relative frame of reference" mechanical formulation that is classically used for rotor simulations. It includes also a short presentation of an adjoint-based shape optimization of a rotor. The next and last subsection (§I.4) is a presentation of part of Bompard's research during his PhD [12], namely effort to optimize airfoil based on the so called shape gradients, that is the total derivative of the function of interest (objective and constraints) with respect to wall mesh node coordinates.

This work has been presented in the following journal articles:

• Numerical sensitivity analysis for aerodynamic optimization: a survey of approaches. *Computers and Fluids* 39 (2010) (J. Peter and R.P. Dwight)

• Large stencil viscous flux linearization for the simulation of 3D turbulent compressible flows with backward-Euler schemes. *Computers & Fluids* 36 (2007) (J. Peter and F. Drullion)

• Aerodynamic shape optimization of hovering rotors using a discrete adjoint of the Reynolds-Averaged Navier-Stokes Equations. *Journal of the American Helicopter Society* 56 (2011) (A. Dumont, A. Le Pape, J. Peter, S. Huberson)

and in lecture notes and conference papers that are referred in each specific section.

## 1.1 Reminder: Discrete gradient calculation methods

The purpose of this section is to recall the equations of discrete gradient calculation for finite volume CFD (subsection 1.1 to 1.4), which are the basis of the research activities to be then described. For the sake of completeness, the equations of the alternative point of view, the continuous approach, are presented in

Annex A for a simple case and a short discussion about respective assets of both approaches is presented in subsection 1.9. No extended bibliography is provided here but the reader can find a number of relevant references in publications by R.P. Dwight and me [139] or M. Bompard and me [14].

### 1.1.1 Notation

In the framework of shape optimization by mean of Computational Fluid Dynamics, a set of parametrized meshes has first to be defined. These meshes surround the various shapes that the aeronautical item of interest (possibly wing, blade, tail, aircraft...) may take. We do not here enter the detail of meshing/mesh deformation techniques on the design space [1]. The vector of design parameters is denoted $\alpha$ (size $n_\alpha$) ; its domain of variation is denoted $D_\alpha$. The volume mesh is denoted $X$ (size $n_X$) and the wall surface mesh $X_S$ (size $n_S$). The volume mesh $X(\alpha)$, and hence also $X_S(\alpha)$, are supposed to at least continuously differentiable.

The finite volume scheme of interest defines the steady-state flow $W$ (size $n_W$) as the solution of a set of $n_W$ non linear equations involving $W$ and $X$,

$$R(W, X) = 0,$$

where $R$ is supposed to be a $C^1$ function of its both arguments. This assumption may seem strong as numerical fluxes and limiters possibly involve min, max or absolute value and as update stage in industrial codes involve cuts-off. This point is further discussed for the selected schemes in subsection 1.1.8.

Besides at every pair of flow and mesh $(W_i, X_i)$ where variation calculus is to be performed, it is assumed that the flow is perfectly converged

$$R(W_i, X_i) = 0 \quad \text{and that} \quad det\left(\frac{\partial R}{\partial W}\right)[W_i, X_i] \neq 0.$$

The implicit function theorem then guarantees the existence of an open set $D_X$ including $X_i$ and a $C^1$ unique function W such that

$$\mathrm{W}(X_i) = W_i \quad R(\mathrm{W}(X), X) = 0 \;\; \forall X \in D_X.$$

This yields a straightforward local definition of $W$ as a continuously differentiable function of the vector of design parameters $\alpha$. For the sake of simplicity, it assumed hereafter that this $C^1$ dependence of $W$ w.r.t. $\alpha$ is valid all over $D_\alpha$.

Typical functional outputs involved in local optimization are objectives like drag, component of drag... and constraints like lift, pitching moment... When using a gradient-based local optimization algorithm, derivatives of the objective and active constraints are needed. Depending on the algorithm, the derivatives of non-active constraints may or may not be required. These $n_f$ functions to be differentiated w.r.t. the design parameters are denoted $\mathrm{J}_k$ when considered as functions of flow and mesh $W$ and $X$ and $\mathcal{J}_k$ when considered as functions of $\alpha$, with the obvious link

$$\mathcal{J}_k(\alpha) = \mathrm{J}_k(W(\alpha), X(\alpha)) \qquad k \in \{1...n_f\}.$$

Besides expressing $W$ as a function of $X$ through the implicit function theorem, allows to define the functions of interest as function of the mesh only:

$$J_k(X) = \mathrm{J}_k(\mathrm{W}(X), X) \qquad k \in \{1...n_f\}$$

---

[1] Many options exist : working on difference with respect to a reference shape or directly on the shape itself ; directly meshing the fluid domain, or first the wall of the item then the fluid domain ; use aeronautical expert parameters or a geometrical approach...

Older discrete gradient calculation methods aim at computing all $n_f \times n_\alpha$ derivatives

$$\frac{\partial \mathcal{J}_k(\alpha)}{\partial \alpha_i} \qquad i \in \{1, ..n_\alpha\} \quad k \in \{1, ..n_f\}$$

A more recent method aims at first calculating the total derivative

$$\frac{dJ_k}{dX} \qquad k \in \{1...n_f\},$$

that includes both the direct influence of node position of the function and its indirect influence through change of discrete flow to reach steady state convergence.

### 1.1.2 Finite differences and complex variable method

The application of finite differences to an entire flow solver is the oldest and, by far, the simplest means of obtaining solution gradients, as it requires no modification of the solver itself. To proceed, the numerical flow solution corresponding not only to $\alpha$ but also to perturbed states $\alpha + \delta\alpha$ and possibly $\alpha - \delta\alpha$ is calculated. For the typical case of second order finite difference $\delta\alpha_l$ ($l \in \{1...n_\alpha\}$) representing a geometry modification in direction $l$ of $D_\alpha$, this implies two mesh deformations $X(\alpha+\delta\alpha_l)$ $X(\alpha-\delta\alpha_l)$, and two new flow solutions on the modified meshes satisfying

$$R\left(W(\alpha - \delta\alpha_l), X(\alpha - \delta\alpha_l)\right) = 0\ , \quad R\left(W(\alpha + \delta\alpha_l), X(\alpha + \delta\alpha_l)\right) = 0.$$

An approximation of all functions derivatives in the direction $\delta\alpha$ can then be approximated by classical centred finite difference formula

$$\begin{aligned} \frac{\mathrm{d}\mathcal{J}_k(\alpha)}{\mathrm{d}\alpha_l}\delta\alpha_l &\simeq \frac{1}{2}\left[\mathcal{J}_k(\alpha+\delta\alpha_l) - \mathcal{J}_k(\alpha-\delta\alpha_l)\right] \qquad \forall\, k\ \in\ \{1...n_f\} \\ &= \frac{1}{2}\left[\mathrm{J}(W(\alpha+\delta\alpha_l), X(\alpha+\delta\alpha_l)) - \mathrm{J}(W(\alpha-\delta\alpha_l), X(\alpha-\delta\alpha_l))\right]. \end{aligned} \tag{1.1}$$

The entire matrix $\mathrm{d}\mathcal{J}(\alpha)/\mathrm{d}\alpha$ may be evaluated at a cost of $2 \times n_\alpha$ flow solutions [2] making the method impractical for large design spaces. Another serious disadvantage is that the choice of the step size $\|\delta\alpha\|$ is critical to the accuracy of the result. If it is too small then rounding errors become significant; if it is too large the neglection of higher order terms in the Taylor expansion in (1.1) is not valid. Moreover, balancing these two errors requires the third order derivative of $\mathcal{J}_k$ in the direction $d\alpha_l$ that is unknown so that in practice only very expensive parametric studies lead to the best finite difference value. See [139] for more details.

All these issues may be alleviated by using a complex finite difference formula, such as

$$\frac{\mathrm{d}\mathcal{J}}{\mathrm{d}\alpha_l}\delta\alpha_l \simeq \Im\left[\mathcal{J}(\alpha + i\delta\alpha_l)\right],$$

where $\Im$ represents the imaginary part. As there is no longer any difference of $\mathcal{J}$ in this expression, it does not suffer from cancellation error, and $\|\delta\alpha_l\|$ may be chosen as machine zero with no loss of accuracy [6]. However the solver must be modified to accept complex variables throughout, negating the main advantage of finite differences.

Finite differences have been used since the 70s in the context of shape optimization. Early contributions include works of Hicks, Henne and VanderPlats [73, 177, 71], also Destarac, Reneaux and Thibert [155, 156, 36]. Complex variable method is the verification method of NASA's FUN3D adjoint module [6, 124].

However these methods have cost scaling with the number of design parameters, $n_\alpha$, and solve numerous non linear sets of equations. This has lead to the investigation of alternative means of gradient evaluation.

---

[2] or if a first-order difference is used $n_\alpha + 1$ flow solutions

### 1.1.3 Direct differentiation method

Under the assumptions detailed in subsection 1.1.1,

$$\forall \alpha \in D_\alpha \quad R(W(\alpha), X(\alpha)) = 0$$

and this form of the governing equations may be differentiated with respect to all $\alpha_l$ to give

$$\frac{\partial R}{\partial W}\frac{\mathrm{d}W}{\mathrm{d}\alpha_l} = -\frac{\partial R}{\partial X}\frac{\mathrm{d}X}{\mathrm{d}\alpha_l} \qquad \forall l \in \{1...n_\alpha\}. \tag{1.2}$$

This may be regarded as a linear system in unknowns $\mathrm{d}W/\mathrm{d}\alpha_l$. The dimension of the system is the number of degrees of freedom in the non-linear equations $n_W$, and it can be regarded as a linearization of those equations.

Given the $n_\alpha$ solutions $\mathrm{d}W/\mathrm{d}\alpha$, the derivatives of $\mathcal{J}$ are

$$\frac{\mathrm{d}\mathcal{J}_k(\alpha)}{\mathrm{d}\alpha_l} = \frac{\partial \mathrm{J}_k}{\partial W}\frac{\mathrm{d}W}{\mathrm{d}\alpha_l} + \frac{\partial \mathrm{J}_k}{\partial X}\frac{\mathrm{d}X}{\mathrm{d}\alpha_l} \qquad \forall l \in \{1...n_\alpha\} \tag{1.3}$$

where again the partial derivatives are in principle easy to evaluate, as $\mathrm{J}_k$ is a known, explicit function of $W$ and $X$. The cost of getting flow and aerodynamics function derivatives is one non-linear and $n_\alpha$ linear solutions, all of dimension $n_W$.

This method was considered as early as 1982 by Bristow and Hawk for a subsonic panel method [20, 21], and again in 1989 for the transonic perturbations equations by Elbanna et al. [49]. In the early 90s it was applied to the compressible Euler equation by two teams at Old Dominion University; that of Baysal [9] and that of Taylor and Hou [169]. On unstructured grids the idea was pursued by Newmann, Taylor et al. from 1995 onwards [120].

### 1.1.4 The Discrete Adjoint Method

There are many ways to derive the discrete adjoint equations, the one given here is chosen for its similarity to the derivation of the continuous adjoint presented in annex A. Let the direct linearization (1.2) be premultiplied by an arbitrary line vector $\Lambda^T$ of dimension $n_W$, so that

$$\Lambda^T \frac{\partial R}{\partial W}\frac{\mathrm{d}W}{\mathrm{d}\alpha_l} + \Lambda^T\left(\frac{\partial R}{\partial X}\frac{\mathrm{d}X}{\mathrm{d}\alpha_l}\right) = 0, \quad \forall \Lambda \in \mathbb{R}^{n_W}.$$

Adding this expression to (1.3)

$$\frac{\mathrm{d}\mathcal{J}(\alpha)}{\mathrm{d}\alpha_l} = \frac{\partial \mathrm{J}_k}{\partial X}\frac{\mathrm{d}X}{\mathrm{d}\alpha_l} + \frac{\partial \mathrm{J}_k}{\partial W}\frac{\mathrm{d}W}{\mathrm{d}\alpha_l} + \Lambda^T\frac{\partial R}{\partial W}\frac{\mathrm{d}W}{\mathrm{d}\alpha_l} + \Lambda^T\left(\frac{\partial R}{\partial X}\frac{\mathrm{d}X}{\mathrm{d}\alpha_l}\right), \quad \forall \Lambda \in \mathbb{R}^{n_W}, \tag{1.4}$$

and factorizing

$$\frac{\mathrm{d}\mathcal{J}_k(\alpha)}{\mathrm{d}\alpha_l} = \left(\frac{\partial \mathrm{J}_k}{\partial W} + \Lambda^T\frac{\partial R}{\partial W}\right)\frac{\mathrm{d}W}{\mathrm{d}\alpha_l} + \frac{\partial \mathrm{J}_k}{\partial X}\frac{\mathrm{d}X}{\mathrm{d}\alpha_l} + \Lambda^T\left(\frac{\partial R}{\partial X}\frac{\mathrm{d}X}{\mathrm{d}\alpha_l}\right), \quad \forall \Lambda \in \mathbb{R}^{n_W},$$

isolates the term $\mathrm{d}W/\mathrm{d}\alpha$, which may be eliminated by choosing the arbitrary vector $\Lambda$ to satisfy

$$\left(\frac{\partial \mathrm{J}_k}{\partial W} + \Lambda^T\frac{\partial R}{\partial W}\right) = 0$$

Actually, the vector $\Lambda$ is then fixed and associated to the $k$-th function of interest so that so that a suitable notation is now $\Lambda_k$. It is the solution of

$$\left(\frac{\partial \mathrm{J}_k}{\partial W} + \Lambda_k^T\frac{\partial R}{\partial W}\right) = 0 \quad \text{or equivalently} \quad \left(\frac{\partial R}{\partial W}\right)^T \Lambda_k = -\left(\frac{\partial \mathrm{J}_k}{\partial W}\right)^T, \tag{1.5}$$

the *discrete adjoint equation*, a linear system in unknowns $\Lambda_k$ the *adjoint vector* associated to function $J_k$ for scheme $R$. Given $\Lambda_k$ the sensitivities of function $\mathcal{J}_k$ may be written

$$\frac{\mathrm{d}\mathcal{J}_k(\alpha)}{\mathrm{d}\alpha} = \frac{\partial \mathrm{J}_k}{\partial X}\frac{\mathrm{d}X}{\mathrm{d}\alpha} + \Lambda_k^T \left( \frac{\partial R}{\partial X}\frac{\mathrm{d}X}{\mathrm{d}\alpha} \right).$$

The critical point is that, because $\alpha$ does not appear in equation (1.5), that linear system must only be solved once for each $J_k$. Hence the full matrix $\mathrm{d}\mathcal{J}/\mathrm{d}\alpha$ may be evaluated at a cost of $n_{\mathcal{J}}$ linear system solutions, substantially independent of $n_f$. Perhaps the first application of this method was given by Shubin and Frank in 1991 for a quasi one-dimensional nozzle flow using the compressible Euler equations [165, 164, 54], and was denoted there the “implicit gradient approach” as contrast to the direct approach. Baysal et al. also recognized its potential, and offered it as an alternative to the direct approach when $n_{\mathcal{J}} \gg n_f$ [9].

### 1.1.5 Identification of adjoint vector. Numerical and physical point of view

The equations of this section require a cell index ($m$), a component index ($a$), and an index going from 1 to the number of equations of the initial p.d.e.. For the sake of readability, the function index $k$ is dropped here from the function and and adjoint vector; $\Lambda$ is the discrete adjoint vector associated to the function $J$ for discretization $R$.

In the previous subsection the discrete adjoint vector appeared as a multiplier of direct differentiation equation that can be specified to remove the flow sensitivity $dW/d\alpha_l$ from the derivatives $d\mathcal{J}_k(\alpha)/d\alpha_l$ for one specific function $\mathcal{J}_k$ but for all design parameters $\alpha_l$. It could also have been defined with a Lagrangian in an closely related way. In both presentations, the discrete adjoint vector is linked to the numerical scheme $R$ and the current mesh $X$ and no insight is gained about a possible mesh convergence of this field. The following characterization of discrete adjoint is interesting from this point of view [51]. It is assumed that small arbitrary changes $\delta R$ are done in $R$ (think of very small fixed numbers added to some of $R$ components). The corresponding solution perturbation $\delta W$ is such that $W + \delta W$ satisfies new discrete flow equations

$$(R + \delta R)(W + \delta W, X) = 0,$$

or at first order

$$[\ R(W, X) + \ ]\ \ \delta R + \frac{\partial R}{\partial W}\delta W = 0$$

The first order change in the function of interest $J$ due to change in flow $\delta W$ is

$$\delta J = -\frac{\partial J}{\partial W}(\frac{\partial R}{\partial W})^{-1}\delta R \qquad \text{since} \qquad J(W + \delta W, X) \simeq J(W, X) + (\frac{\partial J}{\partial W})\delta W$$

Involving discrete adjoint vector $\Lambda_k$, defined by equation (1.5), yields

$$\delta J = \Lambda^T \delta R$$

If only the $a$-th component of $R$ at cell index $m$ has been arbitrarily altered by a small number $\delta R_m^a$ then previous equation yields

$$\Lambda_m^a = \delta J / \delta R_m^a \tag{1.6}$$

This defines the $a$-th component of $\Lambda$ at cell index $m$ as the limit ratio of change in $J_k$ divided by infinitesimal change in the residual $R$ at the corresponding cell & component who caused the change in flow and function value.

In the context of $J$-oriented mesh refinement (successive computations aiming at accurate calculation of $J$ output), we may assimilate $\delta R$ to a lack of accuracy in space discretization and the adjoint vector would indicate the importance of local accurate space discretization for the calculation of $J$. Actually, in the discrete adjoint method only expressions involving products of adjoint vector times residual $R$ or derivatives of $R$ have an intrinsic value independent of the definition of $R$ and this point will have to be more precisely discussed in chapter 2.

Up to now the adjoint vector appears as a mathematical object, dual of the residual $R$. However, in a well-known conference paper Giles and Pierce proposed a physical point of view [60]. As their discussion is based on continuous adjoint, we transpose their ideas in discrete adjoint (and refer in parenthesis to the counterpart in [60]). For this discussion, the residual $R$ is a consistent discretization of Euler or (RANS) equations flux balance (not divided by volume) so that the discrete and continuous adjoint fields are similar [3].

Giles and Pierce consider four local perturbations $\delta R$ (Dirac perturbation $f_n(\xi)\delta(x-\xi)$) in all cells of index $m$ (location $\xi$) for 2D Euler equations (linearized Euler equations with homogeneous boundary conditions) about an aerofoil: (1) mass source at fixed stagnation pressure and total enthalpy ; (2) local normal force ; (3) change in total enthalpy at fixed static and total pressure ; (4) change in total pressure at fixed total enthalpy and static pressure. The reader is referred to reference [60] for details but, as a starting point, the expression of $\delta R^1$ (mass source at fixed stagnation pressure and total enthalpy) is

$$\delta R^1_m = \epsilon \begin{pmatrix} 1 \\ u \\ v \\ H \end{pmatrix}$$

which indeed expresses the presence of mass source in cell $m$ (mass flow $\epsilon$ kg/s$^{-1}$) and compensates all disequilibrium caused in Euler equations by the flow injection.

The function of interest is the integral over the airfoil of the pressure times a local constant (denoted $h$) which involves inviscid drag and lift. By the mean of approximate mechanical analysis, its variations $\delta J^{(1m)}$, $\delta J^{(2m)}$, $\delta J^{(3m)}$, $\delta J^{(4m)}$ due to the source terms in cell $m$, $\delta R^1_m$, $\delta R^2_m$, $\delta R^3_m$ and $\delta R^4_m$, are calculated and (assuming they are exact) from (1.6),

$$(\delta J^{(1m)}, \delta J^{(2m)}, \delta J^{(3m)}, \delta J^{(4m)}) = (\Lambda_{1,m}, \Lambda_{2,m}, \Lambda_{3,m}, \Lambda_{4,m}) \begin{pmatrix} \delta R^1_{1,m} & \delta R^2_{1,m} & \delta R^3_{1,m} & \delta R^4_{1,m} \\ \delta R^1_{2,m} & \delta R^2_{2,m} & \delta R^3_{2,m} & \delta R^4_{2,m} \\ \delta R^1_{3,m} & \delta R^2_{3,m} & \delta R^3_{3,m} & \delta R^4_{3,m} \\ \delta R^1_{4,m} & \delta R^2_{4,m} & \delta R^3_{4,m} & \delta R^4_{4,m} \end{pmatrix}$$

where the $(dm)$ superscript for $\delta J$ means change in $J$ function due to $d$-th change of $R$ at cell $m$, reconverging flow $W$ with global change $\delta W^d$ which trace on the aerofoil leads to change $\delta J^{(dm)}$. As the four changes in $R$ are linearly independent Giles and Pierce may define the adjoint vector at cell $m$ as

$$(\Lambda^1_m, \Lambda^2_m, \Lambda^3_m, \Lambda^4_m) = (\delta J^{(1m)}, \delta J^{(2m)}, \delta J^{(3m)}, \delta J^{(4m)}) \begin{pmatrix} \delta R^1_{1,m} & \delta R^2_{1,m} & \delta R^3_{1,m} & \delta R^4_{1,m} \\ \delta R^1_{2,m} & \delta R^2_{2,m} & \delta R^3_{2,m} & \delta R^4_{2,m} \\ \delta R^1_{3,m} & \delta R^2_{3,m} & \delta R^3_{3,m} & \delta R^4_{3,m} \\ \delta R^1_{4,m} & \delta R^2_{4,m} & \delta R^3_{4,m} & \delta R^4_{4,m} \end{pmatrix}^{-1} \tag{1.7}$$

This of course is derived from (1.6) and close to it except that the residual perturbation $\delta R$ have been physically defined and which makes the local discrete adjoint vector physically defined. For all system of equation for which a similar demonstration can be done, the adjoint vector gets intrinsic and we can expect similar solutions from different discretization and also mesh convergence.

Let us finally note that not only the $\Lambda$ components can plotted, as usually done but also, in the spirit of Giles and Pierce, the limit ratio of the $\delta J^{(d)}$ divided by $\epsilon$ (the small parameter of the physical source term). The corresponding plot is a view of the influence of the source term on the output of interest plotted at source term location – see fig. 3 in [60].

---

[3] this technical requirement in presented with more details in §2.1.2

### 1.1.6 Adjoint-mesh gradient computation

The technique that we call "adjoint-mesh" computation is a way to gather the derivative terms that avoids to manipulate any quantity related to the design parameters inside the CFD code during the adjoint step. It unfortunately does not seem to have a unique name (it is sometimes called "full-adjoint" or "full-reverse-mode").

In the mid 2000's the cost of evaluating $dX/d\alpha$ was becoming a burden in case $X$ is an implicit function of the wall mesh [127]. Moreover, successively reading (for all functions) or loading in memory all $dX/d\alpha_l$ was becoming a time and/or memory burden limiting in practice the cost independence-of-design-parameters of the adjoint method. Finally the cost of the calculation of geometrical sensitivity of the residual, even by finite differences like,

$$\left(\frac{\partial R}{\partial X}\frac{\mathrm{d}X}{\mathrm{d}\alpha_l}\right) \simeq \frac{R(W(\alpha),X(\alpha)+\epsilon(dX/d\alpha_l)) - R(W(\alpha),X(\alpha)-\epsilon(dX/d\alpha_l))}{2\epsilon}\,, \tag{1.8}$$

was possibly not neglectible for the very large numbers of design parameters $n_\alpha$ that were then more and more often considered. All these costs scaling with $n_\alpha$ were not satisfactory of course in the discrete adjoint method that is expected to scale with the number of functions $n_J$.

To overcome this issue, in case $X$ is an explicit function of the design parameters $\alpha$, the adjoint community has used a rewriting of adjoint gradient (see [136, 146])

$$\frac{\mathrm{d}\mathcal{J}_k(\alpha)}{\mathrm{d}\alpha} = \left(\frac{\partial \mathrm{J}_k}{\partial X} + \Lambda_k^T\frac{\partial R}{\partial X}\right)\frac{\mathrm{d}X}{\mathrm{d}\alpha}, \tag{1.9}$$

where the two factors of the right-hand-side of (1.10) may be calculated independently. The factor in the bracket is calculated by the CFD code that solved the adjoint equation and is saved on files ; the CFD code has then no knowledge of parametrization. The product is carried out besides by a basic code that does not need to be run on a supercomputer.

The term inside the bracket is easily identified as the total derivative of the function of interest w.r.t. the volume mesh nodes

$$\frac{dJ_k}{dX} = \frac{\partial \mathrm{J}_k}{\partial X} + \Lambda_k^T\frac{\partial R}{\partial X} \tag{1.10}$$

A component of the first term, $(\partial \mathrm{J}_k/\partial X_m)$, corresponds to the direct dependency of function $\mathrm{J}_k$ on the location of node $m$, whereas a component of the second term, $\Lambda_k^T(\partial R/\partial X_m)$, corresponds to changes of the flow field on the support of function $\mathrm{J}_k$, due to driving residual $R$ to zero after changing node $m$ location.

The $dJ/dX$ vector field may be checked by finite differences. For a second-order check at a specific node $X_m$, two opposite individual small displacements of $X_m$ along each frame axis need to be considered. For example, let $\delta y_m$ and -$\delta y_m$ be small displacement of node $X_m$ along $y$ axis and $X^{(+\delta y_m)}$ $X^{(-\delta y_m)}$ be the shifted meshes resulting of these single alterations of the nominal mesh $X$. Discrete flows denoted $W^{(+\delta y_m)}$ and $W^{(-\delta y_m)}$, need then to be converged for the shifted meshes by solving

$$R\left(W^{(+\delta y_m)},X^{(+\delta y_m)}\right) = 0 \quad \text{and} \quad R\left(W^{(-\delta y_m)},X^{(-\delta y_m)}\right) = 0. \tag{1.11}$$

Then of course $dJ/dy_m$ can be checked based on

$$\frac{dJ}{dy_m} = \frac{J(W^{(+\delta y_m)},X^{(+\delta y_m)}) - J(W^{(-\delta y_m)},X^{(-\delta y_m)})}{2\delta y_m} + O(\delta y_m^2) \tag{1.12}$$

### 1.1.7 Shape gradient

Finally, let us note that if the $X$ explicitly depends on the wall mesh, $X_S$, that depends on the design parameters, a simple and efficient way to compute the derivatives of interest is to solve the $n_f$ adjoint equations and then to calculate the sensitivity of the functions w.r.t. wall mesh nodes:

$$\left(\frac{\partial R}{\partial W}\right)^T \Lambda_k = -\left(\frac{\partial \mathrm{J}_k}{\partial W}\right)^T \qquad \frac{dJ_k}{dX} = \frac{\partial \mathrm{J}_k}{\partial X} + \Lambda_k^T \frac{\partial R}{\partial X} \qquad \frac{d\mathcal{J}_k}{d\alpha_l} = \left[\frac{dJ_k}{dX}\frac{dX}{dX_S}\right]\frac{dX_S}{d\alpha_l}$$

(where the term in brackets is to be computed first).

In case $X$ depends implicitly on $X_S$ which directly depends on the design parameters $\alpha$, Nielsen and Park proposed an elegant solution [127]. Let us denote $D(X, X_s) = 0$ the implicit link between volume and volume mesh. In reference[127] for example, $D$ was a simplified linear elasticity solver and the cost of solving the sensitivity equation

$$\frac{\partial D}{\partial X}\frac{dX}{d\alpha_l} + \frac{\partial D}{\partial X_S}\frac{dX_S}{d\alpha_l} = 0 \qquad \forall l \in \{1...n_\alpha\}, \tag{1.13}$$

for $dX\ d\alpha_i$ was not neglectible. Nielsen and Park proposed to add an adjoint equation for the mesh deformation to get rid of this issue. The sequence of operations is the following:

$$\begin{aligned}
\left(\frac{\partial R}{\partial W}\right)^T \Lambda_k &= -\left(\frac{\partial \mathrm{J}_k}{\partial W}\right)^T \\
\left(\frac{\partial D}{\partial X}\right)^T \Gamma_k &= -\left(\frac{\partial \mathrm{J}_k}{\partial X} + \Lambda^T \frac{\partial R}{\partial X}\right)^T = -\left(\frac{dJ_k}{dX}\right)^T \\
\frac{d\mathcal{J}_k}{d\alpha_l} &= \left[\Gamma_k^T \frac{\partial D}{\partial X_S}\right]\frac{dX_S}{d\alpha_l}
\end{aligned}$$

(where the term in brackets is to be computed first). Finally, as $J_k(X)$ has been defined from $\mathrm{J}_k(W, X)$ where the residual of the discrete flow is converged ($R(W, X) = 0$), we can define $\overline{J}_k(X_S)$ from $J_k(X)$ where $X$ is the volume mesh that corresponds to surface mesh $X_S$, namely (in the implicit case)

$$\overline{J}_k(X_S) = J_k(X) \quad \text{where} \quad D(X, X_s) = 0 \qquad k \in \{1...n_f\}. \tag{1.14}$$

The gradient of $\overline{J}$ w.r.t. $X_S$, $d\overline{J}/dX_S$, is often called *shape gradient*. Previous equations yield

$$\frac{d\overline{J}_k}{dX_S} = \frac{dJ_k}{dX}\frac{dX}{dX_s} \quad \text{(explicit link between } X \text{ and } X_s\text{)} \tag{1.15}$$

$$\frac{d\overline{J}_k}{dX_S} = \Gamma_k^T \frac{\partial D}{\partial X_S} = -\frac{dJ_k}{dX}\left(\frac{\partial D}{\partial X}\right)^{-1}\frac{\partial D}{\partial X_S} \quad \text{(implicit link } \ D(X, X_s) = 0\text{)}. \tag{1.16}$$

The shape gradient is a field of interest for design engineers. It should confirm their knowledge about how to modify the solid walls to improve the values of the output of interest and possibly give them new intuitions.

### 1.1.8 Discrete versus continuous adjoint. ONERA options for discrete adjoint

For the sake of completeness, annex 1 presents the equations of continuous adjoint method in the simple case of a 2D Euler flow, although only discrete adjoint was developed ar ONERA. This choice was made in agreement with Airbus experts [111] and for the following reasons:
(1) to get the gradient of the discrete objective and be fully consistent with the optimizer in the optimization process ;

(2) to work on well posed problems for all types in functions. More specifically, ONERA has developed a strong expertise on far-field drag extraction under the lead of Destarac [35] and continuous adjoint equations do not exist for functions defined as integrals inside the fluid domain (typically, in the calculations of annex 1, the flow sensitivity can not be cancelled along the integration contour of the far-field functions) ;
(3) to possibly use automatic differentiation [66] to produce routines calculating Jacobian of fluxes multiplied, on the right, by flow sensitivity or, on the left, by an adjoint vector. This technique is more and more commonly used even concurrentlty with the continuous approach [48, 3, 191] ;
(4) to get a theoretically simple definition of adjoint systems even for complex flow equations involving complex turbulence models source terms.
The accepted drawback of this choice is that the discrete “explicit part”of the adjoint code, that linearizes the discrete fluxes entering the flux balance $R$, is considerably longer and more complex than its continuous adjoint counterpart would have been. The discrete adjoint code has been essentialy written by hand although INRIA’s Automatic Differentiation code Tapenade has been used for tests and checks [70]. This approach is more error-prone than automatic differentiation but leads to a more efficient code [114].

ONERA’s aerodynamic discrete adjoint module is part of the $elsA$ code [23, 25, 24]. This module is named $elsA/Opt$ [145]. End of 2015, its number of lines was about 300000 which was about 20 percent of $elsA$’s total number of lines. $elsA$ is a very large finite-volume cell-centred code and it has not been planned to differentiate all schemes and turbulence models. The considered types of equations / schemes / linearization are the following:
– Euler flows discretized by Roe flux [160] (with parabolic entropy fix) plus MUSCL approach with van Albada limiting function [174, 173] or by Jameson-Schmidt-Turkel flux [85].
– RANS equations discretized by the above mentioned convective fluxes and so called 5P-COR viscous fluxes (cell-centred gradients which arithmetic mean at interface is corrected in the direction connecting the centers of the two cells adjacent to the face) with frozen eddy-viscosity in the linearization.
– RANS and Spallart-Almaras model equations with linearization of the equation of the discrete model. Discretization and linearization for mean flow are as presented before. Discretization of one-equation turbulence model involves first order convective Roe flux (with parabolic entropy fix), 5P-COR viscous flux formula and source term calculated from cell-centered gradients.

We recall here the assumption that the discrete steady residual of the scheme should be continuously differentiable w.r.t. to flow and mesh to perform discrete gradient calculation (§1.1.1). Actually most of the results presented in later sections were obtained with Roe flux [160] with parabolic entropy fix plus van Albada limiting function [173]. This flux possibly entering $R$ is actually $C^1$ (but not $C^2$) w.r.t. the flow field $W$. Some results have been obtained with Jameson-Schmidt-Turkel scheme [85] that is not $C^1$ where the velocity at a specific face is orthogonal to its normal vector [4] or possibly where the classical pressure-velocity sensor for high gradients is zero [5] . The reader is referred to [121] (§1.2) for an example with a symmetric-mesh and symmetric-flow where the violation of this regularity requirement has clear consequences. The same section of the PhD thesis of Nguyen-Dinh [121] discusses a possible regularization to make this scheme continuously differentiable. Considered viscous flux formula, and source term discretization are both $C^1$ w.r.t flowfield.
Besides, all mentioned fluxes and source terms are $C^1$ w.r.t mesh $X$.

As concerning the resolution of direct differentiation and adjoint equation with the $elsA$ code, the method that has been generally used is LU relaxation method for an iterative scheme based on an approximate Jacobian of the scheme. Actually, although direct differentiation and adjoint sets of equations are linear, their size is too big for large 3D test cases to use direct resolution and iterative solve is the classical way

---

[4] the spectral radius involved in numerical dissipation then includes a $|\overline{V}.\overline{S}|$ term with $\overline{V}.\overline{S} = 0$

[5] in its usual form involving only static pressure, this sensor has the following form $\nu_i = |\frac{p_{i+1}-2p_i+p_{i-1}}{p_{i+1}+2p_i+p_{i-1}}|$

of processing. The selected method has been derived from the classical backward-Euler schemes [22]. The approximate Jacobian and LU relaxation have been proposed and studied by Drullion and me [138] and adapted for adjoint. More information is given in section §1.2 about these implicit stages. Recently, Cerfacs engineers and then Blondeau at ONERA proposed advanced GMRES methods to solve adjoint equation [150]. More advanced methods involving Runge-Kutta steps and multigrid have been presented by Xu et al. [186] and could be considered for $elsA/Opt$ future improvements.
The generalization of iterative resolution of direct differentiation and adjoint equation raised the questions of "exact duality" at the beginning of the 2000's:
– under what conditions would the asymptotic rate of convergence of direct non-linear problem and direct differentiation method be the same ?
– under what conditions the gradients evaluated at any step of the iterative process for direct differentiation and adjoint method would be strictly equal (so that their convergence rate would be the same) ?
Giles and coworkers, Nielsen and coworkers answered to these two questions for different types of schemes [57, 58, 59, 126]. The iterative methods for direct and adjoint gradient methods in $elsA/Opt$ where no built with this strict requirement. It nonetheless very rarely happened that, with corresponding parameters and options, one calculation would diverge and the other one would converge.

The $elsA$ software is a cell-centred finite-volume code and specific boundary states (denoted $W_b$ in the $elsA$ documentation) are calculated at boundary faces. For standard boundary conditions, $Wb$ depends on the flowfield in the adjacent cell and on the local normal via discrete characteristic relations. It is needed to differentiate $W_b$ w.r.t. flow for all discrete gradient methods (terms entering the Jacobian $\partial R/\partial W$) and also w.r.t. mesh for the adjoint-mesh method. Besides, the support of the functions of interest is most often a set of boundary faces and terms like $(\partial \mathrm{J}/\partial W_b)(\partial W_b/\partial W)$ or $(\partial \mathrm{J}/\partial W_b)(\partial W_b/\partial X)$ appear in the right-hand of adjoint equation (first term) and in the expression of $dJ/dX$ (second term). The complete gradient equations are easily derived with these actual dependancies [142]. All classical local boundary conditions have actually been differentiated plus radial equilibrium. Besides, $elsA$ involves a large number of matching conditions (1 to 1 matching, 1 to N matching, general surface matching...) that have been taken into account in the discrete gradient module $elsA/Opt$ by a ghost-cell filling process (using the same routines for adjoint vector or flow sensitivity as for flowfield).

As concerning mechanical formulations for Euler and (RANS) equations, the basic option is the simplest "aircraft" formulation where flow equations involve velocity in absolute frame of reference and where this set of axis is used for the projection of the momentum conversation law. For rotating flows, two formulations are avalaible in $elsA$: equations with relative velocity in the relative frame of reference (for turbomachinery flows) and equations with absolute velocity projected in the relative frame of reference (for rotor flows ; absolute velocity avoids numerical issues in far-field cells) although the second formulation tends to replace the first for turbomachinery flows. This framework was fixed by Jean-Christophe Boniface [16]. Extension of direct differentiation method and discrete adjoint parameter method for absolute velocity / relative frame was done by Dumont [41] with my contribution for the analysis of the numerical terms and the debugging. It is described in §1.2 that is completed with a short presentation of an adjoint-based local shape optimization. Extension of direct differentiation method and discrete adjoint parameter method for relative velocity / relative frame was carried out by Pham and Renac [148, 154].

Finally, lets us note that advanced post-processing tools have been developed at ONERA for the aircraft (FFD series of codes, under the lead of Destarac [35]), helicopters (HeliOpt, under the lead of Dumont) and engines (Xopt now named TurbOpt, under the lead of Castillon). The specific functions calculated be these tools are precisely the one that are to be involved as objective and constraints in the local optimization process. For the sake of consistency and efficiency, the partial derivatives $\partial \mathrm{J}/\partial W$, $\partial \mathrm{J}/\partial X$ are calculated by these tools.

### 1.1.9 Note on mesh convergence of discrete adjoint fields

Discrete adjoint mesh convergence is connected to both the discrete adjoint overview (§I.1) and the review of goal oriented mesh refinement methods (§II.1). The subject is rather discussed here for the clarity of the next chapter. Actually all classical goal-oriented mesh refinement methods implicitely assume adjoint mesh convergence. The best illustration is probably the rewriting of the well-known dual-weighted-residual as proposed by Venditti and Darmofal [182] ($H$ denoting current grid, $h$ denoting finer embedded grid),

$$(\Lambda_h\Big|_{W_h^H})^T R_h(W_h^H) = \Lambda_h^H R_h(W_h^H) + ((\Lambda_h\Big|_{W_h^H})^T - \Lambda_h^H) R_h(W_h^H),$$

where it is implicitly assumed that $\Lambda_h^H$ is a satisfactory approximation of $\Lambda_h\Big|_{W_h^H}$ which converges towards $\Lambda_h$ as $W_h^H$ converges towards $W_h$. Unfortunately the question of discrete adjoint mesh convergence is not so simple when singularities are expected in the adjoint field, which is the case for Euler flows, under some far-field and geometrical conditions, at stagnation streamline and at the wall [60].

Let us first recall the end of section §1.1.5: following a paper of Giles and Pierce [60], under the existence of $d$ [6] independent physical source terms $\delta R$ that can be added everywhere in the fluid domain to the right-hand side of the evolution equations of interest to define well-posed problems, we may expect the discrete adjoint fields to converge as the mesh is refined as numerical solution of a series of well-posed physical problems (1.7).

It is not absolutely clear whether mass injection or normal force, for example, can be applied in all the fluid domain including contact discontinuities and shock-waves to define a sound physical problems and we move to the numerical point of view. Adjoint mesh convergence is obviously not the simple counterpart of flow mesh convergence as the discrete adjoint field is the solution of an equation which depends on the discrete flow calculated on the current grid. The question of mesh convergence could hence be possibly addressed with a "perfect" flow solution interpolating a limiting flow field to the current grid to make independent flow and adjoint grid convergence. Actually, for all the following discussion, only the practical point of view with converged discrete flow then converged discrete adjoint based on this flow, is considered

A technical point is made before moving to the bibliography on the subject of discrete adjoint mesh convergence : In the discrete adjoint module of a finite volume CFD code, $R$ may be defined as the flux-balance or possibly the flux balance divided by the cell-volume (often called "explicit residual" and used as right-hand-side for implicit stages). [7] Discretizing Euler flux balance in one cell of a structured mesh, assuming the flux is simply the physical flux at the arithmetic mean of left and right state, is one simple way to establish a link between discrete adjoint equation (1.5) and continuous adjoint equation (3.6). In this comparison, it clearly appears that the residual $R$ of the discrete adjoint should be the flux balance so that the discrete adjoint vector would be the counterpart of the continuous adjoint vector. It is then actually observed that the discrete adjoint field has similar general aspect when moving to coarser or finer grid.

Maybe the first contribution to the subject of discrete adjoint mesh convergence is the one of Giles, Duta, Müller and Pierce [58, 59]. They considered a transonic flow about the NAC0012 with a strong shock which wandered between mesh points as the angle of attack was varied. The lift of the aerofoil was seen to be somewhat dependent on the position of the shock within the local mesh cell, as a result of which the

---

[6] dimension of the considered system of e.d.p. that is 4 in 2D and 5 in 3D

[7] From a strict mathematical point of view, any other definition ensuring null flux balance at convergence would lead to consistent discrete adjoint gradients. Flux balance divided by cell-index, or multiplied by $z$ coordinate of center would work... but add useless calculations and have no connection with continuous adjoint.

design parameter-lift curve took on a slightly scalloped shape. As the mesh was refined, the amplitude of the scalloping was reduced as expected – but, as it always took the same shape, the amplitude of its gradient with respect to the angle of attack was not reduced. A counterpart issue for internal flows was discussed and illustrated with a shocked nozzle flow with varying outlet pressure and "lift" as function of interest. From this numerical issue for the gradient, Giles and his co-authors concluded that "*the fact that grid convergence of non linear flow calculation does not guarantee convergence of linear sensitivities is a fundamental problem for the discrete approach to adjoint calculations*".
Altough they relate to continuous adjoint, I mention here recent contributions by Lozano: in two papers published in 2012 [105, 106], he discusses the assets and drawbacks of different continuous adjoint formulations for Euler and Navier-Stokes equations. One test case is the NACA0012 airfoil represented by a NURBS and mesh convergence of lift/drag sensitivities w.r.t. to the parameters of the NURBS (coordinates and weights of the 18 selected control points) is studied. Discrepancies are observed between gradients issued from ajoint sensitivities and finite differences even on the finest grid. In a more recent contribution paper [108, 109], the same author points out a strong lack of adjoint mesh convergence for a simple case: a transonic ($M_\infty = 0.8$, AoA = $1.25^o$) flow about the NACA0012 aerofoil is considered and lift-adjoint is computed [8]. Mesh convergence is not observed at the wall for this lifting flow whereas it is demonstrated for a non-lifting flow. A specific issue appears at the trailing edge *where the values of the adjoint variables at the next-to-trailing-edge nodes grow continually as the mesh is refined.*

I am not aware of any article that had further discussed the issues pointed ot in reference [58, 59]. As concerning the question recently raised by Lozano [108, 109], it deserves a specific in-depth study. Nevertheless, as together with PhD students and colleagues, we have calculated series of Euler flows and corresponding $CLp/CDp$-adjoint fields for various verifications, a few plots a presented in this section for a simple flow about the NACA0012 airfoil at transonic flow conditions ($M_\infty = 0.8$, AoA = $2.^o$). Once again the meshes are those of the classical article [180]. The selected scheme is the one defined by Jameson et al. [85] [9]. The isolines of the first component of lift-adjoint are plotted in the fluid domain – figure 1.1 – and no sign of lack of mesh convergence can be observed in this first image. It is also plotted in the cells adjacent to the wall and along a line normal to the stagnation streamline, distant of half a chord to the trailing edge – figure 1.1 upper part. The reason for this second extraction is that close adjoint-isolines are observed in the vicinity of the stagnation streamline where Giles and Pierce predicted an inverse square-root singularity [60]. The drawback of the plots for cells adjacent to the wall is that the location of the variables vary when the mesh is refined. This is the reason why first and third component of lift -adjoint are plotted and along the vertical line ($x$=0.5 $z$>0.) as function of $z$ components – figure 1.1 lower part. – figure 1.1 left and right. Anyhow, it clearly appears in this series of figures, that the adjoint mesh convergence is not reached at oblique characteristic line or at the trailing edge even for that last two very dense meshes.

---

[8] direct and adjoint computations with DLR's unstructured solver TAU also used for the two previously quoted references

[9] Actually, the convergence of flows equations to machine zero could be achieved for the (129×129), (257×257), (513×513) and (1025×1025). Reduction of the explicit residual was $10^{-9}$ for the (2049×2049) mesh

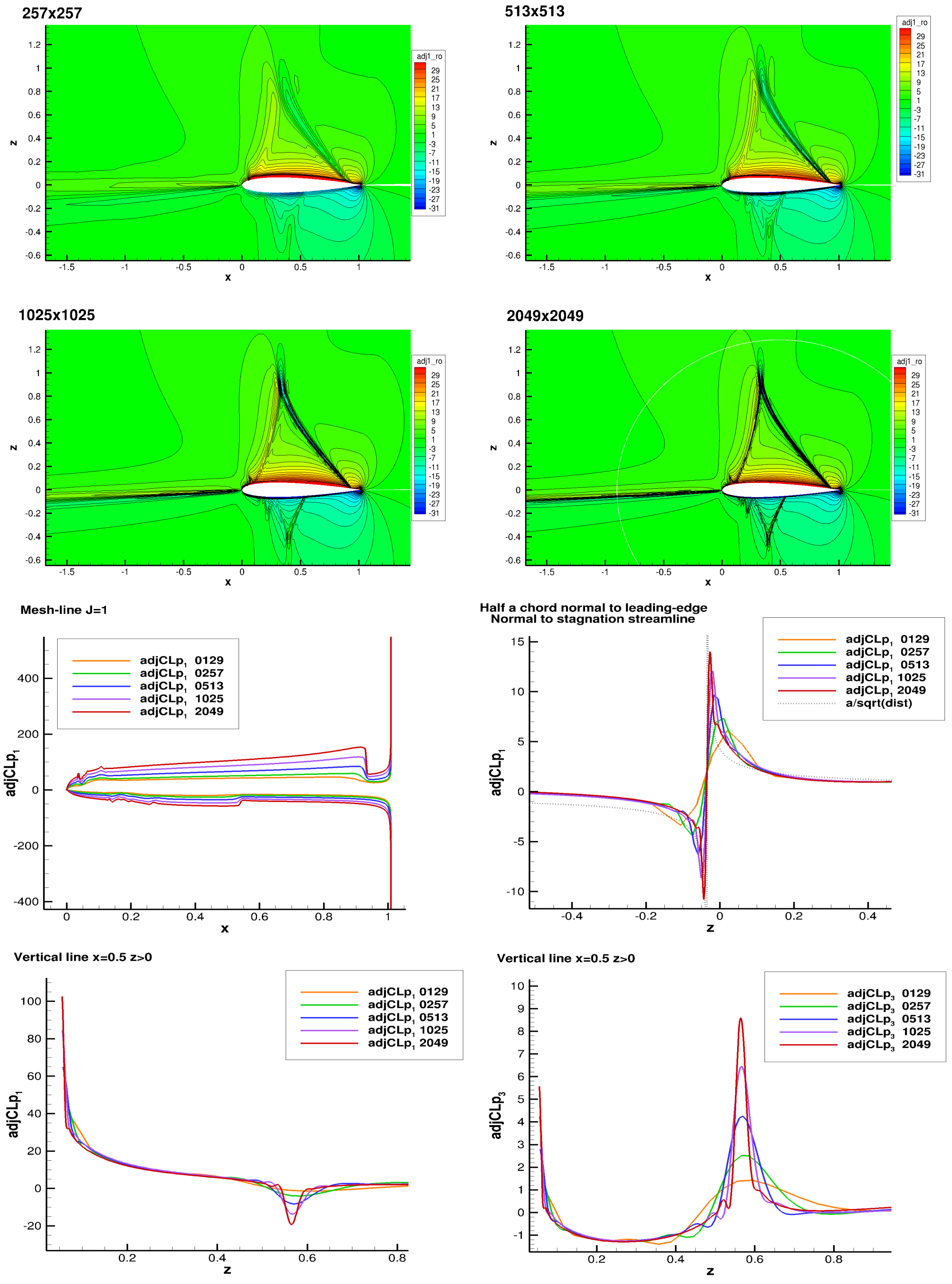


Figure 1.1: NACA0012 $M_\infty$=0.8, AoA = 2.$^o$ inviscid flow – Isolines: first component of $CLp$-adjoint vector – Curves: – Up left: cells adjacent to the wall (first component) – Up right: along a line normal to stagnation streamline, half a chord from stagnation point (first component) – Down left: vertical cell-line, suction side (X=0.5, Z>0) (first component) – Down left: same location (third component)

## 1.2 Implicit stages for flow solution and gradient calculation methods

The purpose of this section is first to define the approximate Jacobians that have been proposed and studied for flow calculation by backward-Euler schemes [138]. A second subsection then describes the way these implicit stages have been adapted to direct differention method and discrete adjoint method.

### 1.2.1 Implicit stages for backward-Euler schemes for steady-state flows

The implicit stages of $elsA$ that are most often used to solve direct and adjoint discrete equations on structured meshes, derive from those proposed by me [134] and studied with Drullion for the computation of steady state flows with backward-Euler schemes [138]. These implicit stages (named "LURELAX" in the framework of the $elsA$ project) are briefly rewieved in this section, whereas the next section indicates which ones were selected for discrete gradient computation and how they were adapted for adjoint. The generic form of a backward-Euler scheme is simply:

$$\left(I + \frac{\Delta t}{Vol}\frac{\partial R}{\partial W}^{(APP)}\right)\left(W^{(l+1)} - W^{(l)}\right) = -\frac{\Delta t}{Vol}R(W^{(l)}) \tag{1.17}$$

Most often, an approximate Jacobian of the scheme $\frac{\partial R}{\partial W}^{(APP)}$ is first defined and an approximate resolution of the linear system (approximate factorization method, relaxation method, Krylov method...) is selected which completes the definition of the implicit stage.

Of course, approximate Jacobians of both upwind and centred schemes is an old topic in CFD [19]. The specific features of those proposed in reference [138] are the following: (1) all terms are gathered at cell centers (avoiding the complexity of Jacobian absolute value or spectral radius calculated at interfaces) ; (2) if the viscous flux is calculated from cell-centred gradient, it uses a five point approximate linearization of viscous flux balance.

The basic matrix approximate linearization of convective and diffusive flux balance entering $(\partial R/\partial W)^{APP}$ are hence the following :

• for the inviscid flux

$$d(F_{i+\frac{1}{2}} - F_{i-\frac{1}{2}}) \simeq (\delta_i^+ A^- + \delta_i^- A^+)dW = -A^+_{i-1}dW_{i-1} + |A_i|dW_i + A^-_{i+1}dW_{i+1}, \tag{1.18}$$

$|A|$, $A^+$, $A^-$ matrices being the absolute value, positive and negative part of (diagonalizable) physical Jacobian $A$ in the mesh line direction (evaluated at cell-centers taking the mean of adjacent faces surface vectors) ;

• for the viscous flux

$$-\,d(F^v_{i+\frac{1}{2}} - F^v_{i-\frac{1}{2}}) \simeq -M^v_{i-1}dW_{i-1} + 2M^v_i dW_i - M^v_{i+1}dW_{i+1} \tag{1.19}$$

if $F^v$ is calculated from face centred gradients, but if cell-centred gradients enter $F^v$, then

$$-\,d(F^v_{i+\frac{1}{2}} - F^v_{i-\frac{1}{2}}) \simeq \frac{1}{4}(-M^v_{i-2}dW_{i-2} + 2M^v_i dW_i - M^v_{i+1}dW_{i+2}) \tag{1.20}$$

where $M^v$ is the classical Jacobian of the thin layer-approximation of viscous flux [27, 148].

Actually, reference [138] also considers two possibly efficient variants in these implicit phases: the classical scalar approximation of convective and diffusive terms [190] and the inclusion of a fourth order artificial dissipation term in the original linearizations (without scalar approximation). The linear systems were solved by LU relaxation (two or four steps). Drullion presented the scalar linear stability analysis of the schemes which we checked together, whereas I conducted series of (RANS)&(k,$\omega$) monogrid and multigrid calculations for a wing and a wing-body-pylon-nacelle configuration for three centred schemes [85, 110, 168] for the studied implicit stages. The reader is refered to [138] for the analysis of the most efficient associations of centred schemes and implicit stages.

### 1.2.2 Implicit stages for the recursive solution of discrete adjoint equation

Until recently, the discrete direct differentiation equation and discrete adjoint equations were solved in $elsA$ by iterative methods that read

$$\left(\frac{\partial R}{\partial W}\right)^{(APP)\,T}\left(\lambda_k^{(l+1)}-\lambda_k^{(l)}\right)=-\left((\frac{\partial R}{\partial W})^T\lambda_k^{(l)}+(\frac{\partial J_k}{\partial W})^T\right), \tag{1.21}$$

$$\left(\frac{\partial R}{\partial W}\right)^{(APP)}\left((\frac{dW}{d\alpha_i})^{(l+1)}-(\frac{dW}{d\alpha_i})^{(l)}\right)=-\left((\frac{\partial R}{\partial W})\frac{dW}{d\alpha_i}^{(l)}+\frac{\partial R}{\partial X}\frac{dX}{d\alpha_i}\right). \tag{1.22}$$

Comparing equation (1.17) (backward-Euler scheme) and equation (1.22) (direct differentiation method), it is clear that the involved linear systems may be almost the same in both contexts – there is only a $\Delta t/Vol$ factor that should be inverted for all lines to produce a suitable routine for direct differentiation equation. This is actually the way the first routines for the solution of direct differentiation were derived. The basic features were kept – same approximate Jacobian definition, few LU-relaxation steps to approximately solve equation (1.21), multi-domain correction at matching-boundaries between two L or U relaxation steps – but not all approximate Jacobian routines were transfered to the gradient calculation module. The first reason was the growing use of the discrete viscous flux based on cell-centred (velocity and temperature) gradients corrected at interface in the direction of the two adjacent centers [137] (that is called 5P-COR in the framework of the $elsA$ project). For structured meshes, this discrete flux exhibits an accuracy close to the one obtained with interface-centred gradients, whereas its memory storage is the low one of the classical formula based on cell-centred gradients. As the approximate Jacobian of its flux balance is equation (1.19), only this formula was coded in the gradient calculation module. Besides, as up to now Roe's flux is the most used inviscid discretization with $elsA/Opt$ and as it is best combined with block-matrix approximate Jacobians in the iterative procedure (1.22), the scalar-approximated matrices studied in [138] have not yet been transfered in the gradient calculation module.

Concerning the discrete adjoint equation, a technical question araised: formulas like (1.18),(1.19) or (1.20) imply that all block-matrices of a specific column of the approximated Jacobian are evaluated at the corresponding same point. This allows a lot of simplifications when coding the L- and U-relaxation routines but this property is lost of course when moving to adjoint. Keeping this coding would have meant to change the approximate Jacobians to have all block-matrices of a specific line evaluated at the corresponding same point. It was decided to keep the same approximate Jacobian and define a new coding for LU relaxation adjoint routines.

## 1.3 Extension of discrete gradient capability to relative frame/asbsolute velocity

To our knowledge, Lee and Kwon were the first to develop an adjoint approach for a rotor flow [96]. In their method, the 3D Euler equations were solved on unstructured meshes and they used a continuous adjoint approach. More recently ONERA (with the $elsA$ code [23, 25, 24]) and NASA Langley (with the FUN3D code [11]) pursued the objective to carry out adjoint-based local optimization with (RANS) equations. Nielsen et al. [125] developed a discrete adjoint for (RANS) discretized on unstructured meshes including full linearization of Spallart-Almaras turbulence model. They presented the optimization of a tilt rotor in purely axial flow. At the same time, Dumont and his PhD supervisors developed the same kind of capability with structured meshes and presented optimization of ONERA 7A rotor and ONERA-DLR ERATO rotor [42, 43, 41, 44].

### 1.3.1 Flow equations, discretization

The classical Navier-Stokes equations read

$$\begin{aligned}
\frac{\partial \rho}{\partial t} + div(\rho \overline{V}) &= 0 && (1.23)\\
\frac{\partial \rho \overline{V}}{\partial t} + div(\rho \overline{V} \otimes \overline{V} + p\overline{\overline{I}}) &= div(\overline{\overline{\tau}} + \overline{\overline{\tau_R}}) && (1.24)\\
\frac{\partial \rho E}{\partial t} + div(\rho E \overline{V} + p\overline{V}) &= div((\overline{\overline{\tau}} + \overline{\overline{\tau_R}})\overline{V}) - div(\bar{s}) - div(\bar{s_t}) && (1.25)
\end{aligned}$$

where all variables are defined according to Favre [10], where $\overline{\overline{\tau_R}}$ is the Reynolds stress tensor and $\overline{s_t}$ is the diffusive flux of turbulent enthalpy. We consider here only models relying on Boussinesq assumption where $\overline{\overline{\tau_R}}$ and $\overline{s_t}$ are respectively proportional to $\overline{\overline{\tau}}$, the tensor of viscous stresses and $\bar{s}$, the heat flux. The retained turbulence model is the $k - \omega$ model of Kok [93] that defines the turbulent viscosity involved in the equations defining $\overline{\overline{\tau_R}}$ and $\overline{s_t}$.

The system of equations (1.23)-(1.25) is reformulated in a relative frame of reference with absolute velocity. The rotor is assumed to rotate at constant angular speed $\Omega$ and, when using a frame of reference attached to the rotor, we may still consider the absolute velocity but calculate, gradient, integrals... and write the momentum equation in the rotating frame of reference. The use of absolute velocity avoids to manipulate large non uniform velocity fields in the far-field and prevents the apparition of numerical noise. Conversely, the use of the rotating frame of reference makes the rotor and mesh steady. The no-slip condition on the rotor blade is then formulated as the equality between the flow velocity and the local entrainment velocity of the rotating frame. For these questions of velocity and frame formulation, the reference document at ONERA is the PhD thesis of Boniface [16].

The velocity of a fixed point of the rotating frame in the inertial frame is from now on denoted $\overline{V_e}$ ( in French *"vitesse d'entrainement")*. The (RANS) equations with this choice for velocity and frame read

$$\begin{aligned}
\frac{\partial \rho}{\partial t} + div(\rho(\overline{V} - \overline{V_e})) &= 0 && (1.26)\\
\frac{\partial \rho \overline{V}}{\partial t} + div(\rho \overline{V} \otimes (\overline{V} - \overline{V_e}) + p\overline{\overline{I}}) &= div(\overline{\overline{\tau}} + \overline{\overline{\tau_R}}) + \bar{C} && (1.27)\\
\frac{\partial \rho E}{\partial t} + div(\rho E(\overline{V} - \overline{V_e}) + p\overline{V}) &= div((\overline{\overline{\tau}} + \overline{\overline{\tau_R}})\overline{V}) - div(\bar{s}) - div(\bar{s_t}) && (1.28)
\end{aligned}$$

where $\overline{C}$ is a source term arising from the definition of velocity and frame

$$\bar{C} = -\rho \overline{\Omega} \wedge \overline{V},$$

and $\overline{\Omega}$ is the vector defining the constant rotation w.r.t. the rotor-axis (which norm is equal to the angular velocity).

---

[10] in compressible (RANS) equations, $\rho$ and $p$ are defined as statistical repetition mean of the corresponding physical quantities, $\overline{V}$ is defined by Favre mean, that is the statistical repetition mean of $\rho V$ divided by the one of $\rho$

### 1.3.2 Discretization and linearization in direct and adjoint modes

Among all the space-discretizations available in the $elsA$ code, the ones that have been linearized have been recalled in subsection §1.1.8. Among the two linearized inviscid fluxes, the one retained for the steady state calculation was Roe flux with van Albda limiting function [160, 174, 173]. The frozen turbulence assumption was retained in the linearization ; its influence on the accuracy of flow and functions of interest sensitivity was eventually checked ([41] §2.7 [42] §VI.B) and found acceptable.
The steady state simulation capability with absolute velocity / relative frame has been implemented in the $elsA$ software by Boniface. The adaption of the direct differentiation / adjoint-parameter capability [11] to this formulation has been carried out by Dumont with my contribution for the analysis of discrete term, coding options and debugging. Five steps were required ([41] §2.6):
– adapting Roe flux linearization, correcting Jacobian eigenvalues and Harten fix w.r.t. absolute velocity/absolute frame formulation ;
– linearizing the source term of the formulation that depends on $\rho\overline{V}$ ;
– adapting the linearization of inviscid wall boundary condition (for objective and residual linearization) accounting for wall displacement with velocity $\overline{V_e}$;
– adapting the linearization of viscous wall boundary condition (for objective and residual linearization) accounting for wall displacement with velocity $\overline{V_e}$ ;
– adapting the linearization of far-field boundary conditions (that depends on $\overline{V_e}$) for residual linearization ;
– adapting to the formulation the approximate Jacobian of the Newton/relaxation method and the additive-Schwartz multi-domain correction between two relaxations steps – equations (1.22) and (1.21). Actually, all Jacobians, positive/negative/ absolute value of Jacobians have their eigenvalues shifted by $\overline{V_e}.S$ where $S$ is the relevant surface vector.

### 1.3.3 Example of application : ERATO rotor

This test case is selected to provide an example of application of previous adjoint capability and a glance at adjoint-based aerodynamic optimization at ONERA. More general information about that second activity can be found in [14] or [145] but, in a nutshell:
– for parametrization, either the mesh is globally parametrized by free-form [188] or the solid shape is parametrized by any of the numerous classical techniques and the wall deformation is propagated in the fluid domain by an analytic method [111] or quaternions. In the later case, the method is differentiated to possibly calculate the shape gradient (§1.1.7)
– $elsA$ is used for solving direct [23, 25, 24] and adjoint equations (§1.1.7 and [145]) ;
– objective, constraints and their partial derivatives are calculated by application-specific codes named FFD72 for aircraft [35], HeliOpt for rotor flows [41] and TurbOpt for internal engine flows ;
– most used descent methods are conjugate gradient method for unconstrained problems and feasible direction method for constrained problems [176]. Least Square Sequential Quadratic Programming has also been successfully tested [28].

The considered rotors are made of four blades. The flow about one blade is calculated and the influence of the other blades (in particular vortex shed by front blade) is obtained by angular join conditions expressing the angular periodicity of the flow. A CH structured mono-domain mesh is designed about the simulated blade. It is checked after calculation of the (RANS) steady state flow that $y^+$ of first cell center (adjacent to the wall) is close to 1.
Concerning the parametrization of blades and mesh deformation, Dumont and Le Pape decided to never

---

[11] The adjoint-mesh capability was not developed at that time. See §III Conclusion and perspectives

deform the profiles of the initial blade, as it is very difficult to define profiles that are suitable for helicopter blades, especially for forward flight were the blade encounters an unsteady regime. The design parameters were then twist, chord, dihedral and sweep. The laws for the variations of these geometrical quantities w.r.t. radius can be either linear by part functions or Bézier curves. The mesh deformation method that defines a new volume mesh $X(\alpha)$ from a new surface mesh $X_S(\alpha)$ is an analytic $C^1$ function that propagates full deformation up to a distance $d_1$, no deformation after distance $d_2$ and makes a regular transition between $d_1$ and $d_2$ (see [41] §2.4 for more details). The objective function for the local optimization is the classical one for aerodynamic optimization of hovering rotors, the Function of Merit $FM$ ([41] §1.5 and Annex B).

The ERATO rotor has been designed in the framework of an aeroacoustic ONERA-DLR joint initiative [166, 153]. Its radius is R $=$ 2.1 m, its mean chord 0.14 m and its twist varies linearly w.r.t. to span and is egal to -10 degree at the tip. Its aeroacoustic performance has been assessed in windtunnel. Its aerodynamic performance in forward flight has been proved to be good but its performance in hovering flight has been considered as not very good, in particular for high loads [41].

For this reason, an adjoint-based local optimisation has been carried out for this geometry. The tip Mach number of the flow was 0.617 and the Reynolds number was the one derived from radius, velocity of the tip, density and dynamic viscosity of the air at the ground. Laws describing changes in twist, chord and sweep on the external part of the rotor, [0.45 R, R], have been considered. These are degree nine polynomials curves like

$$\mathbf{TW}(t) = \sum_{i=0}^{i=9} \mathbf{TW_i}\, B_{i,n}(t) \qquad t = \frac{(r - 0.45\mathrm{R})}{.55\mathrm{R}} \qquad B_{i,n}(t) = \begin{pmatrix} n \\ i \end{pmatrix} t^i (1-t)^{n-i},$$

where the first of the $\mathbf{TW_i}$ is fixed and the eight others have fixed abscissae (see the red points, figure1.2) and varying ordinate (that are precisely the design parameters). Using the classical properties of Bernstein polynomials, it is possible to bound the maximum twist (resp. chord and sweep) by bounding identically the coefficients in the Bézier functions. On top of these 24 ($3\times8$) parameters, the angle of rotation of the blade along its span, that is called collective pitch for rotors, has also been included as design parameter in the optimization process. The objective function was the $FM$. No constraint is involved in the local optimization but considering the definition of $FM$, it is unlikely that the lift would be decreased during the optimization. Besides, a full polar for initial and final shape was eventually calculated.

Seven steps of adjoint based local optimization have been done. The (discrete adjoint) gradient vector of $FM$ at each step is used by the conjugate gradient method of Fletcher and Reeves [53, 176] to define the successive descent directions. Based on an estimation of step size, three configurations (geometry and collective pitch) in the fixed direction of descent are defined for which flow and Figure of Merit are computed. A third order polynomial interpolation for $FM$ is then used to estimate by 1D maximization the descent step and the configuration where the optimization process is to be pursued. Seven optimization hence mean 28 flow computations (possibly restarted from neighboring flow) and seven adjoint solves.

The global increase of $FM$ is 6.6 points. The changes in shapes is described by the Bézier curves for twist chord and sweep – see figure (1.3) the three final curves. The variation of shape can be directly observed in Figure (1.5) that also presents the change in pressure distribution in three span sections. Finally looking at polar curve of the rotor – $FM$ as a function of collective pitch, figure (1.4) – it is observed that the optimized shape has led to both higher lift and higher $FM$ (at constant lift, the optimized shape has an higher $FM$ in all the domain of lift/collective pitch).

Finally, the CPU cost of the optimization was calculated and also its counterpart assuming gradients of $FM$ would have been calculated by finite difference (all time measurements refer to NEC SX8, former large computing facility of ONERA). The time needed to get a steady state flow solution was 1h20min, the time to get the adjoint gradients was 1h40min. The total time of adjoint-based optimization was hence 49 CPU hours. Would the gradient of $FM$ w.r.t. the 25 parameters have been calculated by first order finite difference, the cost would have been 270 CPU hours. The corresponding cost with second order finite difference being 504 CPU hours. The advantage of adjoint method was then obvious.

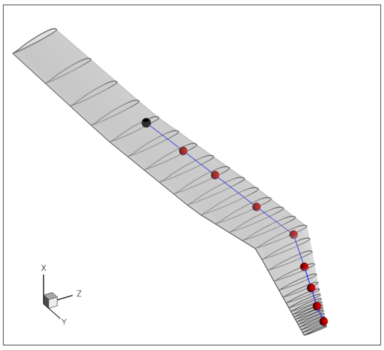


Figure 1.2: Position of the 8 Bézier poles of the laws for geometrical changes

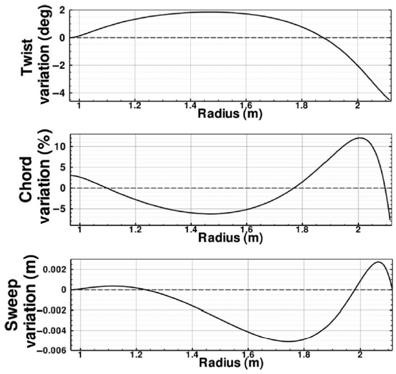


Figure 1.3: Variation of geometrical laws

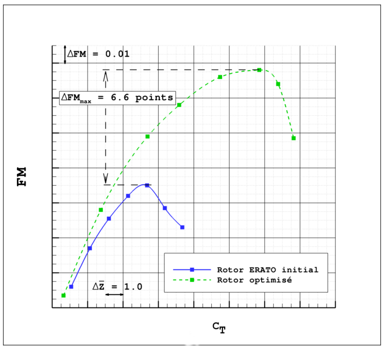


Figure 1.4: Comparison of Figures of Merit (as functions of lift varying collective pitch)

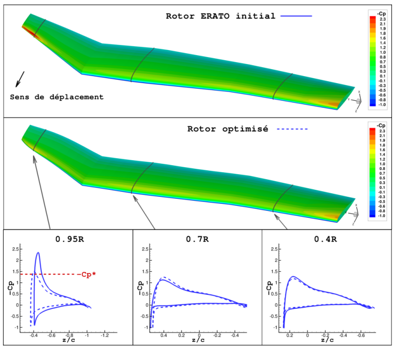


Figure 1.5: Comparison of Cp contours

## 1.4 Airfoil optimisation based on shape gradient

Defining the number of parameters that is suitable to drive a shape optimization is not a trivial task. Obviously, very few or too global parameters is not a good option as the optimal shape(s) may not be part of the design space. Castonguay et al., Vassberg et al. and Carrier et al., for example, illustrated this issue [29, 179, 28]. Conversely, it is known from a long time that very large design spaces may introduce high-frequency noise in the gradients of the functions of interest which may lead to weavy shapes and failure of descent methods – see Reuther et al. [158, 159], Li et al [100, 101], or Stück et al. [167]. "Noise" does not mean here that the gradient is numerically unstable and prone to numerical errors but that it exhibits high frequencies for many simple problems and can hence not be safely used in descent methods [101] [12].

---

[12] This can be considered as obvious. From a mechanical point of view, one can argue that moving towards wavy shapes will reduce the aerodynamic performance of the airfoil because (if the flow is attached) as the regularity of the velocity and pressure distribution is directly related to the airfoil curvature as easily understood by examining the Euler equations written in the local streamwise reference frame [34]

One way to go forward, is to nevertheless use the richest possible design space for a given mesh, that is the set of all wall nodes coordinates $X_S$, but smooth the shape gradient $d\overline{J}/dX_S$ – or more commonly, its component normal to the wall $(d\overline{J}/dX_S, n)$ – before using it in a local optimization algorithm. Different research teams have studied several types of smoothers. At ONERA, Bompard, Maugars, A. Costes, Désidéri and I worked on this topic [12, 15, 33]. Bompard and his supervisors proposed an original smoother and an original criterion to assess the validity of new shapes [12, 15]. This work was unfortunately not published in an international journal due to uncomplete proof of superiority over existing methods and inability to include friction in the adjoint step.

The state of the art about shape gradient smoothing, Bompard et al. smoother and criterion for shape suitability [15] are presented here after.

### 1.4.1 Profile smoothing

For the sake of completeness, let us first note that profile smoothing is not always related to design. Campbell defined a curvature smoothing procedure in the framework of its inverse design tool CDISC [26, 99]. It is based on Akima's interpolation scheme [1]. The CFAST method of Li and Krist, based on Dierckx's spline [37], is meant to remove curvature oscillations with extremely small geometry changes when defining a cubic spline description of an airfoil from a set of points [99].

### 1.4.2 Implicit shape gradient smoothing. State of the Art

The focus is here on the smoothing of the shape gradient for profile optimization by descent method. In practice, the field of normal shape gradient,

$$d = (\frac{d\overline{J}_k}{dX_S}, n),$$

(that is a scalar field defined at each node of the aerofoil) is smoothed and the corresponding operator is denoted $\mathcal{S}$. Among all proposed methods for the regularization of field $d$, the one defined by Jameson is the most commonly used : it is based on an implicit residual smoothing method (IRS) [98, 83], previously introduced as an implicit stage for steady state flow calculations. It results in a smoothed normal shape derivative also called "Sobolev gradient" as it exhibits a property of decreasing the objective in a relation where the right-hand-side is expressed as a weighted dot product of $H^1_0$ [81, 92]. The method has been presented by Jameson and coworkers in numerous articles and conference papers [79, 80, 82, 84, 86, 87, 81, 92]. It was used among others by Mohammadi et al. in their CAD-free framework [112], Mousavi, Castonguay and Nadarajah [113, 29] and Schmidt, Schultz, Gauger et al. [162] who presented the operator as an approximate Hessian. In [82, 84] the authors noted that, when using a smoothed information, incomplete convergence of the flow and the adjoint solution is sufficient thus permitting large savings in computational costs.

The interior point equation for the smoothing of 1D field $d$ in $\mathcal{S}(d)$ is

$$-\epsilon_J\ \mathcal{S}(d)_{l-1} + (1+2\epsilon_J)\mathcal{S}(d)_l - \epsilon_J\ \mathcal{S}(d)_{l+1} = d_l$$

The first and last equations can be adapted in particular to ensure that the sum of the element of the $\mathcal{S}(d)$ is equal to the sum of the $d$ terms. It is also possible to remove the $\epsilon_J$ terms from one line to equalize the corresponding left-hand-side and right-hand-side components. This is often done for leading-edge and trailing edge after the value of $d$ at these points has been set to zero in order to keep the chord unchanged.

Finally, a method associating ideas of parameter-based and parameter-free shape optimization is mentioned : Castonguay and Nadarajah used implicitely smoothed shape gradients in conjunction with local shape parameters [29]. In the classical reduced form of discrete adjoint gradient

$$\frac{d\mathcal{J}_k}{d\alpha_l} = \frac{d\bar{J}_k}{dX_S}\frac{dX_S}{d\alpha_l} \qquad\qquad l \in \{1...n_\alpha\} \quad k \in \{1...n_f\},$$

the shape gradient $d\bar{J}_k/dX_S$ is replaced by a smoothed counterpart field. Reference [29] discusses the benefit of this approach for aerofoil optimization with B-splines and Hicks-Henne bumps.

### 1.4.3 Explicit shape gradient smoothing. State of the Art

In [80] Jameson also considered a simple two-pass three-point-per-mesh-direction explicit smoothing, one pass being

$$\mathcal{S}(d)_l = \epsilon_J d_{l-1} + (1-2\epsilon_J)d_l + \epsilon_J\, d_{l+1}.$$

The same operator was used recursively by Jaworski and Müller [88]. Recently, Stück et al. used an explicit operator based on a convolution of the shape gradient with a truncated Gaussian kernel and established a link between this operator and Jameson's implicit operator [167]. In the slightly different context of multi-level shape optimization, Vasquez, Dervieux and Koobus defined an explicit smoothing operator that projects the current shape change to a smaller space [178].
More geometrical methods have been proposed and assessed. In particular, the POSSEM method of Li et al. controls the curvature of the airfoil induced by a shape modification in case it is defined by cubic B-splines [100].

Besides, an advanced explicit smoothing method has been proposed by Bletzinger and co-workers [76]. It acts on both the tangential and normal components of the shape to define suitable shapes and meshes in the optimization process. As concerning the smoothing of normal shape gradient, the method is not strictly parameter-free. Design parameters denoted $\alpha$ need to be defined. A piecewise linear filter $A^T$ on shape gradients is used and then a transposed filter $A$ for displacement of the wall nodes $\delta X_S$

$$\frac{dJ}{d\alpha} = A^T\frac{dJ}{dX_S} \qquad \text{[descent algorithm for } \alpha] \qquad \delta X_S = A\delta\alpha$$

### 1.4.4 Proposed criterion for shape gradient smoothing. Curvature-control

As mentioned before, $d\bar{J}/dX_s$ and its components orthogonal to the wall, $(d\bar{J}/dX_S, n)$, are very wavy fields even for usual airfoils, flow conditions, meshes and functions of interest, whereas the airfoils considered all along the optimization path must be regular. This is the reason why smoothing operators are applied to $d\bar{J}/dX_S$ or $(d\bar{J}/dX_S, n)$ before changing the shape of the airfoil. As curvature changes most often result in perturbations of the flow-field, a control of the curvature total-variation of targeted airfoils is proposed : the curvature $k(s)$ as a function of the arc length $s$, is first computed over the airfoil of interest $X_S$, using Frenet formulas. More precisely, Frenet formulas define the curvature as the derivative of the angle of the local tangent with a fixed direction w.r.t. the arc length. Actually, the local angle of the curve w.r.t. the Ox axis is calculated for the points of the discrete curve $X_S$. Then an Akima's spline [1], denoted $S$, is fitted to this function and the curvature is obtained by differentiating the spline formula. This yield an accurate estimation of the curvature $k(s)$ even where the discretization nodes tend to cluster). The total variation of $k(s)$ is then

$$\int_S |k'(s)|ds.$$

In order to select suitable shapes when carrying out shape optimization, a smaller and more significant variable was found to be obtained by subtracting the difference in curvature between the upper side trailing edge (arc length $s_{tu}$ ) and the leading edge (arc length $s_{le}$) and the analogous quantities on the lower side:

$$\Psi(S) = \int_S |k'(s)|ds - |k(s_{le}) - k(s_{tu})| - |k(s_{le}) - k(s_{tl})|.$$

The curvature of the classical RAE2822 profile is presented at figure 1.6 as a function of the arc length $s$. It appears that $\Psi(S)$ does remove the influence of the very strong negative curvature at the leading edge, accounting only for changes of curvature at the upper side and lower side of the airfoil.

The current airfoil at a specific step of the local optimization process is denoted $X^c_s$, its local normal vector is denoted $n^c$, the characteristic step-size of the descent method is $\tau$ and $d$ stands for $(d\bar{J}/dX^c_s, n^c)$. Finally, $\mathcal{S}$ denoting the smoothing operator applied to $d$, the targeted airfoil is defined at each point of the surface mesh by [13]:

$$X^\tau_S = X^c_S + \tau \mathcal{S}(d)\mathbf{n}_c \tag{1.29}$$

The proposed smoothing operator that is described in next subsection is recursive and uses as stopping criterion

$$\Psi(S^\tau) < q\ \Psi(S^c)\ \ , \tag{1.30}$$

where $S^\tau$ and $S^c$ are the Akima's splines corresponding to $X^\tau_S$ and $X^c_S$. A suitable value for constant $q$ was found to be about 2. Actually this criterion could be used in any other smoothing method to check if a proposed new profile is suitable or not.

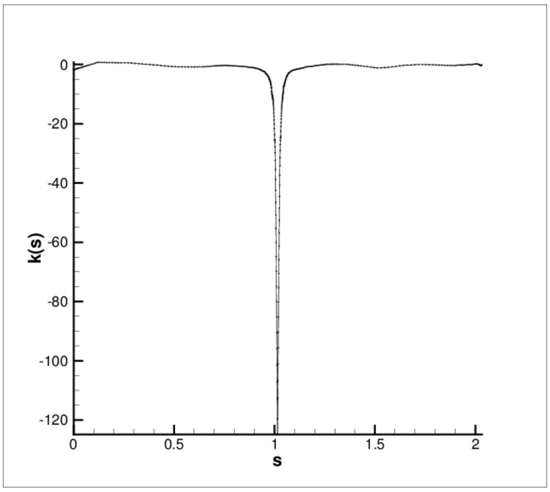


Figure 1.6: Curvature of RAE2822 airfoil as a function of curve length

### 1.4.5 Proposed recursive shape gradient smoothing based on Dierckx's spline

The method of Dierckx is based on the search of a balance between fitting the data and minimizing a smoothness measure. It enforces that a data fitting error is lower than a given parameter $\epsilon$, and seeks the smoothest spline curve subject to this constraint.

In this study, order-three splines are used and the smoothness is achieved by minimizing the discontinuity jumps of the third derivatives at the interior knots of the spline. In this procedure, the number of knots and their position is automatically chosen by the algorithm. The fitting criterion is the weighted sum of the squares of the differences between the input function and its spline approximate. In the input distribution $d(s)$, the three values corresponding to the leading edge and trailing edges (upper and lower sides) are set to zero and their weights in the fitting criterion are assigned much larger values than the

---

[13] in practice, leading and trailing edge are never displaced which means that anyhow $\mathcal{S}(d)$ is set to zero at theses points.

analogous weights associated to the other inputs so that the output distribution $\mathcal{S}(d)$ also has null values at these three points. The recursive algorithm is defined as follows [15]:

1. Set $l$ = 1 ; Compute $\Psi(S^c)$ the total variation of curvature indicator of current airfoil shape $S^c$ (defined by the set of coordinates $X^c_S$). Set $d^0 = (d\overline{J}/dX^c_s, n_c)$ . Compute the descent step $\tau$ that would lead to a specified decrease of $\overline{J}$ in case $\overline{J}$ was varying linearly in direction $d^0$.
2. Apply Dierckx's spline interpolation with tolerance $\epsilon$ to the field $d^{l-1}$ to get $d^l$.
3. Compute the target airfoil based on the step size $\tau$ and the current normal displacement $d^l$: $X^l_S = X^c_S + \tau\ d^l\ \mathbf{n}_c$ .
4. Compute the curvature of the Akmina's spline $S^l$ corresponding to target airfoil $X^l_S$ and test if $\Psi(S^l) < q\ \Psi(S^c)$. If condition satisfied, stop ; otherwise restart at step 2 with $l = l + 1$.

### 1.4.6 Test case for smoothing operator and parameter-free optimization chain

Although the next two subsections are focuses on smoothing operator and global efficiency of the parameter-free aerofoil optimization chain proposed by Bompard et al [15], a short presentation of all building blocks of this chain is given hereafter. The main reason is that the complete definition of the shape gradient requires the one of the space discretization residual $R$, the function of interest $J$ and link between volume mesh $X$ and wall mesh $X_S$. The studied parametrization-free optimization chain included six steps [12, 15]:

1. Solve the (RANS)&(SA) equations around the current shape. The finite-volume cell-centred $elsA$ software [23, 25, 24] was used for this stage. Second order Roe's flux [160] (using the MUSCL approach [174] with the Van Albada limiting function [173]) was used for the mean flow convective terms. Centered fluxes with interface-centered evaluation of gradients were used for both diffusive terms. Cell-centred gradients were used for the source terms.
2. Extract the relevant aerodynamic coefficient and evaluate the objective function. The FFD code was used [35]. For the applications, the wave drag was differentiated. The sum of wave drag and friction was the function of interest.
3. Compute the total derivative of the objective function (w.r.t. mesh coordinates) by the discrete adjoint method. This adjoint module of the $elsA$ software was used [135, 145].
4. Compute the derivatives of the objective function w.r.t. surface nodes. The method deforming the volume mesh based on the wall mesh displacements was the analytic method proposed by Meaux and coworkers [111].
5. Smooth the normal component of shape derivative. Bompard et al. method was used.
6. Update the shape using a descent algorithm. The optimization method was the CONMIN [175] iterative method that combines Zoutendijk's method of Feasible Directions [192] (when some constraints are active) and the conjugate direction method of Fletcher and Reeves (otherwise) [53].

The RAE2822 aerofoil is considered. The flow conditions are M$_\infty$ $= 0.730$, AoA= $2.79^o$, Re= $6.5\quad 10^6$ (as concerning the angle of attack, see footnote [14] below). Three two-domain meshes, including 32832,

---

[14] This test case corresponds to Cook's et al. [32] $9^{th}$ experiment related to the RAE2822 airfoil M$_\infty$ $= 0.730$, AoA= $3.19^o$, Re= $6.5\quad 10^6$ (this Reynolds number is based on the density, velocity, eddy viscosity at infinity and the chord of the 2D profile). Within the EUROVAL project [67], it was suggested that the influence of the wind tunnel walls could be accounted for in the calculations by correcting the angle of attack (AoA= $2.79^o$ instead of AoA= $3.19^o$). This correction, that has been widely accepted by the CFD community, is retained for our computations.

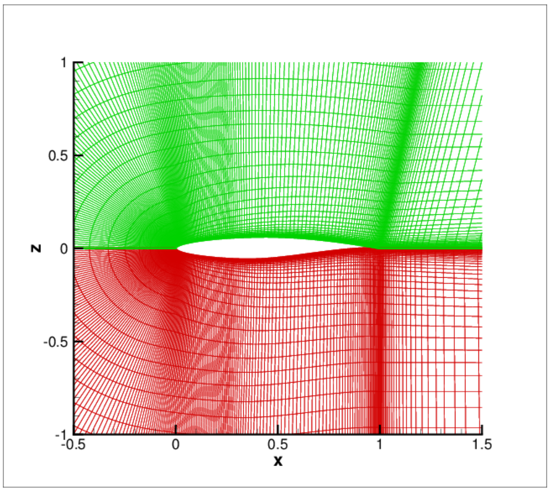


Figure 1.7: Two domains, 32832 cells grid about the RAE2822 airfoil

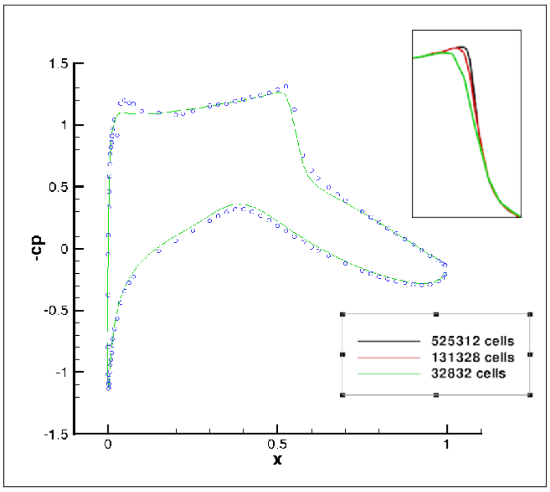


Figure 1.8: Validation of flow solver on the 32832 cells grid

131328 and 525312 cells, are used (see figure 1.7). The pressure distributions on the airfoil are calculated and only a slight difference is observed on the suction side upwind the shockwave. Besides the $C_p$ curve obtained with the 32832 cells grid is consistent with the experimental data (figure 1.8). Hence, this grid is retained for the assessment of smoothing operators and parameter-free shape optimizations.

### 1.4.7 Assessment of proposed smoothing operator

The smoothing operators are tested for the nominal shape, with the scheme described in previous subsection, on a distribution of normal component of wave drag shape gradient, $(dCDw/dX_S{,}n)$. The adjoint for $CDw$ has been run with both the frozen-turbulence assumption and linearization of the turbulence model (see [15] figures 4 and 5). The two curves – with and without turbulence-freezing – are quite close for both sides of the aerofoil. They are considered as representative of those appearing during the optimization process. Finally, the distribution of $(dCDw/dX_S{,}n)$ obtained with the frozen-turbulence are retained for the assessment of smoothing methods. This distribution is presented in figures 1.9 (lower side) and 1.10 (upper side).

The algorithm defined in section 1.4.5 has been tested with different values of $\epsilon$. While increasing $\epsilon$, the fitting constraint of Dierckx's spline definition is released and more smoothed curves are derived from the minimization of the smoothing criterion. It is also observed that the number of recursive iterations needed to satisfy criterion (1.30) gets lower. Figures 1.11 and 1.12 present the output curves obtained for different values of $\epsilon$. It clearly appears that the lower values of $\epsilon$ lead to better curves. It is observed that most satisfactory curves (corresponding to $\epsilon \in [3.2\ 10^{-3}, 10^{-4}]$) actually retain the global features and zones of positive and negative values of the input field. Numerically, it is checked that the dot product between $d$ and $\mathcal{S}(d)$ is positive so that $\mathcal{S}(d)$ is an actual descent direction. Also monitored is $\arccos(d, S(d))$ that is almost constant in the interval of $\epsilon$ mentioned above. The smoothed distributions obtained with $\epsilon = 10^{-4}$ are retained for the comparison with other methods.

For the sake of completeness, it is checked that recursivity is needed to define satisfactory smoothed curves. If only one step of smoothing via Dierckx's spline is applied, way larger values of $\epsilon$ are needed to satisfy the stopping criterion (1.30). The lowest corresponding value is 3.2. The corresponding $\mathcal{S}(d)$ distribution is plotted in figure 1.13, together with those obtained for $\epsilon$ equal to 2 and 4 and the reference curve selected above. It clearly appears that these curves cannot be considered as smooth approximations of the original distribution.

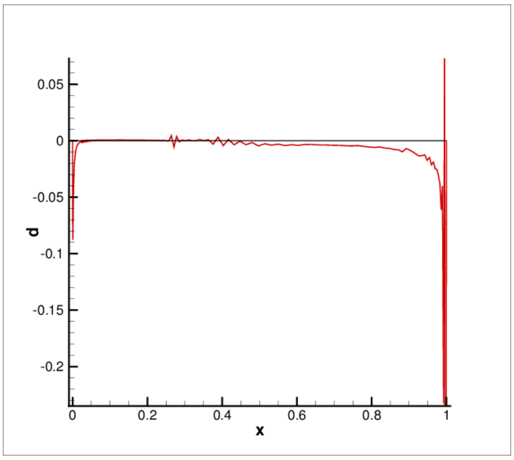


Figure 1.9: Distribution of (dCDw/d$X_S$,$n$). Lower side

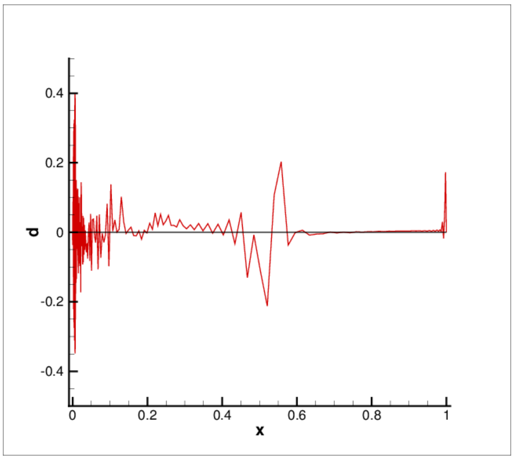


Figure 1.10: Distribution of (dCDw/d$X_S$,$n$) Upper side

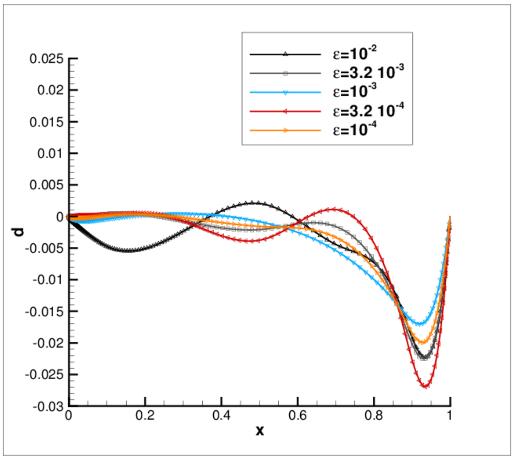


Figure 1.11: Recursively smoothed distribution of (dCDw/d$X_S$,$n$) (figure 1.9) using Dierckx's spline up to the satisfaction of criterion (1.30) Lower side

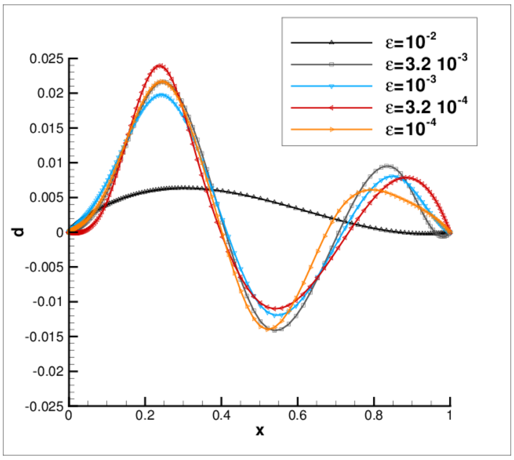


Figure 1.12: Recursively smoothed distribution of (dCDw/d$X_S$,$n$) (figure 1.10) using Dierckx's spline up to the satisfaction of criterion (1.30) Upper side

Besides, (IRS) was tested for standard values of $\epsilon_J$. This parameter was set to zero in the equations corresponding to leading edge and trailing edge in order to get zero value of $\mathcal{S}$ at these points. This also makes the smoothing of $(d\bar{J}/dX_S, n)$ independent on the upper and the lower sides. The curves resulting of the application of this operator to the selected distribution are presented by figure 1.14. In our test-case, the gradient at the leading and trailing edges are very high and difficulties to apply (IRS) have been experienced.

For the sake of completeness (IRS) was also tested inside the recursive algorithm defined for Dierckx's spline smoothing. Very low values of $\mathcal{S}(d)$ were obtained (which was the reason why (1.30) was satisfied) whereas the resulting $\mathcal{S}(d)$ curves did not appear as approximations of the original $d$ distribution.

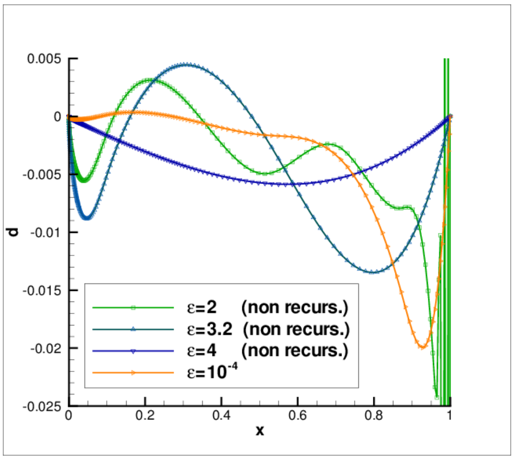


Figure 1.13: Smoothed distribution of (dCDw/d$X_S$,$n$) using once Dierckx's spline. Lower side

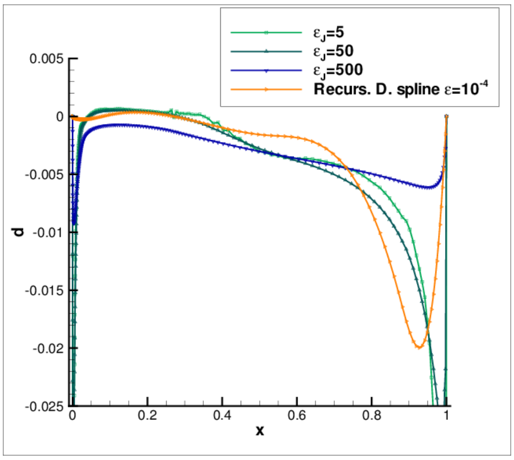


Figure 1.14: Smoothed distribution of (dCDw/d$X_S$,$n$) resulting from applying once (IRS). Lower side

### 1.4.8 Assessment of proposed parameter free optimization chain

In every test, the computation is initiated with the RAE2822 airfoil geometry and carried out at the flight conditions of the experiment of Cook et al. [32] mentioned above (M$_\infty$ $= 0.730$ and Re$= 6.5 \quad 10^6$). Four design cases, similar to those introduced by Kim et al [91], have been considered :

1. drag minimization at fixed angle of attack AoA=$2.79^o$;
2. lift maximization at fixed angle of attack AoA=$2.79^o$;
3. drag minimization at fixed lift $CL_0 = 0.8$;
4. lift maximization at fixed drag $CD_0 = 0.011$.

In the study of Kim et al, the surface of the airfoil was parametrized using 50 Hicks-Henne bump functions spread evenly on the upper and lower sides. In this paper, the same computations are made by the design procedure presented in subsection §1.4.6. The satisfactory results, described in table 1.1 were obtained with the proposed method.

| | min $CD$ | max $CL$ | min $CD$ s.t. $CL \geq CL_0$ | max $CL$ s.t. $CD \leq CD_0$ |
|---|---|---|---|---|
| Parameter-free frozen $\mu_t$ (S.A. model)<br>$CD_0 = 0.0110$<br>$CL_0 = 0.8$ | 0.0143 ⇒ 0.0098 | 0.756 ⇒ 0.901 | 0.0169 ⇒ 0.0120 | 0.8 ⇒ 0.998 |
| Parameter-free lin. of S.A.<br>$CD_0 = 0.0110$<br>$CL_0 = 0.8$ | 0.0143 ⇒ 0.0102 | 0.756 ⇒ 0.916 | 0.0169 ⇒ 0.0118 | 0.8 ⇒ 0.818 |
| 50 Hicks-Henne frozen $\mu_t$ (B.L. model) [91]<br>$CD_0 = 0.0143$<br>$CL_0 = 0.83$ | 0.0152 ⇒ 0.0100 | 0.799 ⇒ 0.857 | 0.0167 ⇒ 0.0109 | 0.799 ⇒ 0.976 |

Table 1.1: Results of parameter-free and parameter-based optimization chain on the four test cases

# Chapter 2

# Goal-oriented mesh adaptation

Goal-oriented mesh adaptation aims at defining suitable meshes for the calculation of a functional output (typically a force or moment for external flows ; loss, efficiency, massflow or outlet total pressure for internal flows). Concerning finite-difference/finite-volume methods, the subject was initiated by Giles and Pierce [61] with the goal of improving the accuracy of integral functionals. Venditti and Darmofal then defined the most popular method for non-linear outputs focusing on both estimates improvement and goal-oriented mesh refinement [181, 182, 183].

The scientific community became aware that the resulting adapted meshes (typically for lift of drag, for a transonic 2D flow) were very different from those defined by the older classical feature-based methods. In particular, goal-oriented adjoint-based methods effectively refine the mesh upwind the function support and features of interest which feature-based methods typically fail to do. Feature-based methods also refine zones that have no influence on the flow close to the function support. Figure 2.1 provides a visual illustration of this matter. Moreover, focusing on convergence of output functionals, it appeared that classical feature-based indicators sometimes fail to drive the output value towards the reference one, obtained using very fine regular meshes. This deficiency, that was early demonstrated in [184] and then in [182] (four times) and [47], has strengthened the interest in goal-oriented mesh refinement. Finally, as noted recently by Park in a recent paper [130], maps of goal-oriented error indicator and output adapted grids have influenced the design of new meshes [18].

After coworkers and I coded the adjoint mesh mode in the $elsA$ code for one of ONERA's industrial partners [136, 146], we considered the usage of the total derivative $dJ/dX$ for $J$-oriented mesh refinement. In a nutshell, the norm of this field times the local characteristic size of the mesh appeared to be a suitable indicator for refinement aiming at an accurate $J$ computation. This research was presented in conference papers [147, 144, 140] and in the following journal articles [144, 122, 171, 157] :

• Goal-oriented mesh adaptation using total derivative of aerodynamic functions with respect to mesh coordinates – With application to Euler flows. *Computers and Fluids* 66 (2012) (J. Peter, M. Nguyen-Dinh, P. Trontin)

• Mesh quality assessment based on aerodynamic functional output total derivative. *European Journal of Mechanics B/Fluids* 45 (2014) ( M. Nguyen-Dinh, J. Peter, R. Sauvage, M. Meaux, J.-A. Désidéri)

• Finite-volume goal-oriented mesh-adaptation using functional derivative with respect to nodal coordinates. *Journal of Computational Physics* 313 (2016) (G. Todarello, F. Vonck, S. Bourasseau, J. Peter, J.-A. Désidéri)

• Mono-block and non-matching multi-block structured mesh adaptation based on aerodynamic functional derivatives for RANS flows. *International Journal for Numerical Methods in Fluids* 83 (2017) (A. Resmini, J. Peter, D. Lucor)

The classical methods for (FV) goal-oriented mesh refinement are recalled in section §II.1. After the asymptotic behavior of $dJ/dX$ has been discussed in section §II.2, the proposed criteria for mesh adaptation and mesh assessment are derived in section §II.3. Applications to 2D and 3D, Euler and RANS flows with structured meshes are presented in section §II.4. Most conclusive results have been obtained with unstructured meshes which define the most suitable setting for mesh adaptation : for classical exercises of 2D Euler lift/drag-oriented mesh adaptation, the proposed method efficiently defined adapted meshes with the same refined zones as the most used method [182, 183] but without the need of a finer grid. These results are presented in section §II.5. In section §II.6, the perspectives of the $dJ/dX$-based method are discussed in relation with the maturity of the classical methods presented in §II.1.

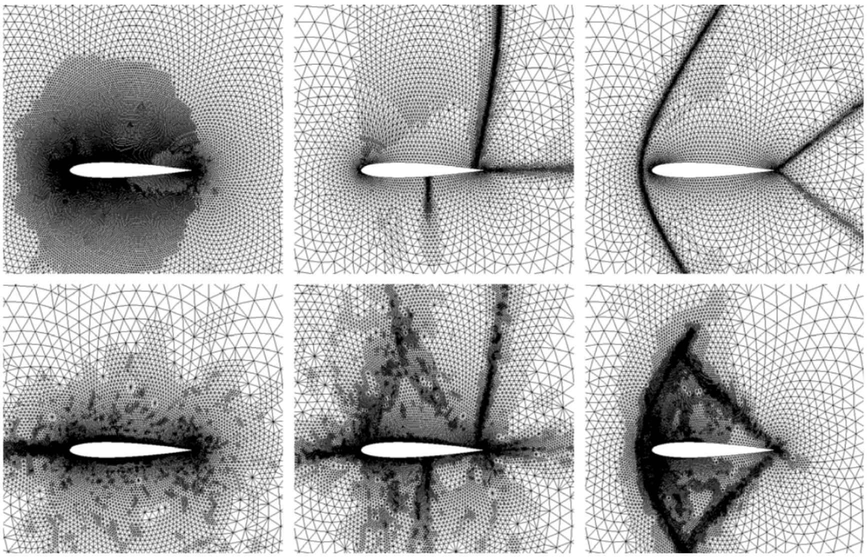


Figure 2.1: NACA0012 inviscid flows. From left to right ($M_\infty$,AoA) = ($0.5,0^o$) ($0.85,2^o$) ($1.5,1^o$). Up: classical feature-based refinement. Down: CL-oriented refinement. Reproduced with permission from reference [47]

## 2.1 Classical methods for finite-volume goal-oriented mesh-adaptation

### 2.1.1 Classical FV goal-oriented error estimators

A complete review of output-based error estimation and mesh adaptation methods has been published in by Fidkowski and Darmofal [51]. The reader is referred to this article for a detailed discussion covering variational formulations, mesh types and mesh adaptation strategies whereas this subsection focuses on the presentation of the classical methods for (FV) goal-oriented error estimation. As a complement the next one presents a few references about isotropic and anisotropic refinement.

At the end of the 90's, Pierce and Giles introduced adjoint-based error estimation for functions in a very broad framework [60, 61, 149, 65]. It is simply supposed that, in a Hilbert space $\mathcal{H}$, whose inner product is denoted $(.,.)$ [1]: (a) a well-posed "direct" linear differential equation $Lw = f$ can be solved, exactly or approximately, before computing the scalar product of the (possibly approximate) solution by another vector $g$ of $\mathcal{H}$; (b) a corresponding adjoint problem $L^*\lambda = g$ is well defined and can be solved exactly or approximately, before computing the dot product of the (possibly approximate) solution by $f$.

The error in the estimation of the common goal, $(g, w) = (L^*\lambda, w) = (\lambda, Lw) = (\lambda, f)$, by the approx-

---

[1] We keep the notations of the original papers by Giles and co-workers. They are consistent with those introduced earlier but typed in lower case. This seems satisfactory as the method has been largely demonstrated for scalar equations before being extended to systems of equations

imate solution $w_h$ ($h$ denoting the average mesh size) of the direct problem is

$$(g, w) - (g, w_h) = (g, (w - w_h)) = (L^*\lambda, (w - w_h)) = (\lambda, L(w - w_h)) = (\lambda, f - Lw_h). \tag{2.1}$$

If the adjoint problem has also been approximately solved, the error can be expressed as

$$(g, w) - (g, w_h) = (\lambda_h, f - Lw_h) + (\lambda - \lambda_h, f - Lw_h) \tag{2.2}$$

Pierce and Giles have given a detailed analysis of these formulas for a wide range of problems and numerical methods (including both finite-difference and finite-element methods). For simple linear differential equations like 1D or 2D Poisson equation, they presented complete proof and illustration that $(\lambda_h, f - Lw_h)$ is the dominant error term and the order of $(\lambda - \lambda_h, f - Lw_h)$ is twice the order of this first term. In this case, they proposed

$$\mathrm{j} \simeq (g, w_h) + (\lambda_h, f - Lw_h)$$

as (superconvergent) function estimation [61, 149]. In 2003, they involved in their theorems the accuracy of the reconstruction operator [2] that builds continuous functions from the discrete $w_h$, $\lambda_h$ before estimating the outputs and error terms in equations (2.1) (2.2) [64]. Giles, Pierce and Müller presented equivalently good results for a shock-free nozzle and an airfoil [61, 149, 115] but the theory they proposed for non-linear equations and non-linear outputs involves averaged Fréchet derivatives of considered system of partial differential equations (and boundary conditions). It also requires the evaluation of the the residual of the discrete flow for these operators. This theory has not been as broadly tested and used as the method proposed a little later by Venditti and Darmofal. [3].

In a series of three articles [181, 182, 183], Venditti and Darmofal have proposed similar formulas but for non-linear outputs, for FV (or FD) schemes and for discrete adjoint. They presented applications to 2D compressible flow computations [182, 183]. For the description of their method, we drop the index $k$ attached in chapter 1 to function-to-differentiate and adjoint vector and use subscripts and superscripts corresponding to grid level. The method involves two grids: a coarse one of characteristic mesh size $H$, and a fine one of characteristic mesh size $h$. The full computation of the flow field and the output of interest on level $H$ is supposed to be affordable, whereas it would be prohibitively expensive on level $h$. The subscripts $h$ and $H$ will be attached here to $R$, $X$ and $W$. Lastly, $W_h^H$ and $\lambda_h^H$ represent the coarse-grid flow-field and adjoint vector reconstructed on the fine grid *via* some consistent projection operator. A Taylor's expansion of the functional output of interest $\mathrm{J}_h$ about the interpolated coarse-grid solution yields

$$\mathrm{J}_h(W_h, X_h) = \mathrm{J}_h(W_h^H, X_h) + (\frac{\partial \mathrm{J}}{\partial W}\Big|_{W_h^H})(W_h - W_h^H) + \mathcal{O}(||W_h - W_h^H||^2)$$

Using an adjoint-like equation solved on the fine grid (2.3) and then a Taylor's expansion of $R$ about $W_h^H$ yields

$$(\Lambda_h|_{W_h^H})^T \left(\frac{\partial R_h}{\partial W_h}\Big|_{W_h^H}\right) = -\frac{\partial \mathrm{J}_h}{\partial W_h}\Big|_{W_h^H} \tag{2.3}$$

$$\begin{aligned}\mathrm{J}_h(W_h, X_h) &= \mathrm{J}_h(W_h^H, X_h) - (\Lambda_h|_{W_h^H})^T(\frac{\partial R_h}{\partial W_h}\Big|_{W_h^H})(W_h - W_h^H) + \mathcal{O}(||W_h - W_h^H||^2) \\ &= \mathrm{J}_h(W_h^H, X_h) + (\Lambda_h\Big|_{W_h^H})^T R_h(W_h^H) + \mathcal{O}(||W_h - W_h^H||^2) \qquad (2.4)\end{aligned}$$

---

[2] actually, linear or cubic spline interpolation

[3] Let us nevertheless note that reference [60] presents in the last equation of section I the adjoint weighted residual of Venditti and Darmofal theory except it involves the analytic flow solution at discrete grid points instead of a discrete flowfield interpolated from a coarser grid

If the flow computation is not affordable on the fine grid, neither is the solution of equation (2.3) for $(\Lambda_h|_{W_h^H})$. The alternative is to replace this adjoint field by the interpolated coarse-grid adjoint,

$$\mathrm{J}_h(W_h, X_h) \simeq \mathrm{J}_h(W_h^H, X_h) + \underbrace{(\Lambda_h^H)^T R_h(W_h^H)}_{computable\ correction} + \underbrace{((\Lambda_h\Big|_{W_h^H})^T - (\Lambda_h^H)^T) R_h(W_h^H)}_{error\ in\ computable\ correction\ (ECC)} \tag{2.5}$$

The authors recommend to take $\mathrm{J}_h(W_h^H, X_h) + \Lambda_h^H R_h(W_h^H)$ as the function estimate and adapt the mesh by reducing uniformly the simple upper bound of $ECC$ that is obtained taking the sum of the absolute value of elemental contributions. When doing so, the bound of $ECC$ term is actually approximated using two interpolations from coarse grids with different polynomial accuracy (typically, linear for $\Lambda_h^H$ and quadratic for approximate $\Lambda_h\Big|_{W_h^H}$). Finally, let us note that the error in computable correction, $ECC$, has an other expression and a first order approximation: a residual of adjoint equation at interpolated flow is first defined:

$$R_h^\Lambda(\Lambda) = \left[\frac{\partial R_h}{\partial W_h}\Bigg|_{W_h^H}\right]^T \Lambda - \left(\frac{\partial J_h}{\partial W_h}\Bigg|_{W_h^H}\right)^T = \left[\frac{\partial R_h}{\partial W_h}\Bigg|_{W_h^H}\right]^T (\Lambda - \Lambda_h|_{W_h^H}).$$

The $EEC$ is then easily expressed as

$$ECC^b = ECC = -\left(R_h^\Lambda(\Lambda_h^H)\right)^T \left[\frac{\partial R_h}{\partial W_h}\Big|_{W_h^H}\right]^{-1} R_h(W_h^H), \tag{2.6}$$

Finally, using the straightforward equality $R_h(W_h^H) = R_h(W_h^H) - R_h(W_h)$,

$$ECC^c = \left(R_h^\Lambda(\Lambda_h^H)\right)^T (W_h - W_h^H) \simeq ECC = ECC^b$$

These formulas have raised a deep interest in the aeronautical CFD community. The main applications of this method are described in the first section of reference [144]. Let me finally note that if the interpolation operator is carefully coded so that $\mathrm{J}_h(W_h^H, X_h) \simeq \mathrm{J}_H(W_H, X_H)$, the correction term accounts for the difference between fine and coarse grid output functional value so that refinement based on the correction may also be considered (see [115] and [65] §5).

Later on, Dwight has proposed a very different adjoint-based method attached to Jameson et al. scheme [85]. In a series of two articles [46, 47], he considered classical test cases for Euler flows. He ran computations using Jameson et al. scheme [85] on hierarchies of grids and for different sets of artificial dissipation coefficients $(k^2, k^4)$. The error for the functions of interest appeared to be mainly due to artificial dissipation. On this basis, the following measure for the approximation error in the Jameson et al. scheme has been proposed:

$$k^2 \frac{d\mathrm{J}}{dk^2} + k^4 \frac{d\mathrm{J}}{dk^4}$$

The dissipation coefficients are then interpreted as being defined independently for each control volume. This leads to a local indicator for dissipation-error in cell $m$:

$$k^2 \frac{d\mathrm{J}}{dk_m^2} + k^4 \frac{d\mathrm{J}}{dk_m^4}$$

The derivatives $d\mathrm{J}/dk_m^2$ and $d\mathrm{J}/dk_m^4$ can only be computed by the adjoint method with the standard adjoint equation and a specific gradient formula

$$\left(\frac{\partial R}{\partial W}\right)^T \Lambda = -\left(\frac{\partial \mathrm{J}}{\partial W}\right)^T \qquad d\mathrm{J}/dk_m^2 = \Lambda^T \left(\frac{\partial R}{\partial k_m^2}\right) \quad d\mathrm{J}/dk_m^4 = \Lambda^T \left(\frac{\partial R}{\partial k_m^4}\right) \tag{2.7}$$

The field of local indicator for dissipation error is used as a mesh refinement indicator and $\mathrm{J} - k^2 d\mathrm{J}/dk^2 - k^4 d\mathrm{J}/dk^4$ is considered as the corrected output value.

In 2010, Fidkowski and Roe made and illustrated an original contribution exhibiting specific functions for which the adjoint field is known [52]. For subsonic Euler flows, considering continuous equations, they first demonstrated that the entropy variables **v**, defined as the derivatives of the entropy $s$ [4] with respect to conservative variables

$$s = -\rho S/(\gamma - 1) \qquad (S \text{ being the physical entropy } \; S = \ln p - \gamma \ln \rho \;),$$

$$\mathbf{v} = ds/dW = \left( \frac{\gamma - S}{\gamma - 1} - \frac{\rho V^2}{2p}, \frac{\rho u}{p}, \frac{\rho v}{p}, \frac{\rho w}{p}, -\frac{\rho}{p} \right)^T ,$$

are the adjoint vector of the entropy flux

$$J_e = \int_{\partial\Omega} s\rho V.ndS \; .$$

They could then derive $J_e$-oriented mesh refinement without solving a system of equations to get the adjoint vector. In case of a subsonic inviscid flow, discrete non-zero value of $J_e$ is due to numerical dissipation and $J_e$-based refinement is relevant. The method could also be demonstrated for a transonic inviscid flow although the shock-wave abruptly creates entropy ([52] §6.5). We refer to [52] for the extension of the method to viscous flows [5]. The method was demonstrated with a discontinuous Galerkin code but the free-adjoint-field property could also be exploited with a Finite Volume or Finite Element code.
In a later reference, Fidkowski, Ceze and Roe took benefit of far-field drag breakdown theory in 2D. They derived mechanical conditions under which $J_e$ is proportional to an approximate form of Oswatitsch drag in 2D. They could then perform approximate-drag oriented mesh adaptation without solving the adjoint equation for the drag [50].

I finally mention the work of Alauzet, Dervieux, Loseille and Belme, although the numerical method they use is an hybrid of finite-volume (for inviscid fluxes) and finite-elements (for viscous fluxes) [104, 10]. Let me first recall that INRIA scientists have extensively studied this formalism of continuous mesh, derived from Riemannian geometry, where a symmetric metric tensor is defined all over the geometrical domain. The mesh sizes prescribed in the directions of the eigenvectors are derived from the eigenvalues (one over the square of the eigenvalues) [30, 56, 102], the idea being that in the actual discrete mesh, all edge lengths should have unit measure under the metric. This description is hence intrinsically anisotropic. In this framework and for quadratic functions, Loseille and Alauzet presented exact evaluations of the linear interpolation error and derived the optimal continuous mesh that minimizes the interpolation error [102, 103].
Concerning goal-oriented simulations, Alauzet, Dervieux, Loseille and Belme carried out the mathematical analysis of their numerical scheme in order to derive a bound for the a priori error estimate of the goal like

$$|J(W) - J(W_h)| \leq \int_{\Omega_h} |\nabla\Lambda||\mathcal{F}(W) - \Pi_h \mathcal{F}(W)| d\Omega_h + \int_{\Gamma_h} |\Lambda||\overline{\mathcal{F}(W)} - \Pi_h \overline{\mathcal{F}(W)}| d\Omega_h \qquad (2.8)$$

where $W$ is the exact flow solution, $\Lambda$ is the exact adjoint of $J$, $\Omega_h$ the computational domain, $\Gamma_h$ its boundary, $\overline{\mathcal{F}}$ is a function of Euler physical flux and the boundary conditions and $\Pi_h$ is the linear interpolation based on the current mesh. (This is the case of Euler equations, [104] equation (16), see

---

[4] this is for a perfect gas

[5] The entropy variables are then no longer the adjoint variables of the entropy flux but of a more complex function with two new terms involving the viscous tensor

corresponding bound, equation (22) in [10], for laminar flows.) The minimization of the weighted interpolation errors that appear in the bound of the exact error on $J$, is then solved in the continuous metric space [104] with a fixed goal complexity (continuous counterpart of number of points). Once the exact solution for the metric has been derived using current discrete flow and adjoint, a mesh is finally derived by a metric-conforming mesh-generator.

Also using the continuous mesh framework, Yano and Darmofal have presented a method for anisotropic $h$-adaptation of simplex meshes [189]. The method can be applied when the error to be minimized is a local function of the mesh; it is first demonstrated for $L^2$ projection error of regular functions (locality is obvious) then for dual-weighted-residual mesh-refinement with a DG scheme (locality is not obvious – see [189] §4.3). The principle is to apply a series of local changes to each element[6] and calculate/estimate the new local error for each of the considered configurations. The continuous metric formalism allows to build a local surrogate of the error on the metric space as each of the modified elements is naturally described by a new local metric matrix (more precisely, the error function depends on the logarithmic map matrix describing the change in the current metric. See [189] §3 and §4.3). Finally, a surrogate cost function and surrogate global error model are formulated by summing the element-contributions and the problem is solved approximately by a descent algorithm derived from the optimality conditions. The new local metric is then derived from the calculated metric change by the so-called exponential map relation and the new mesh is calculated by a metric conforming mesh-generator.
The model is demonstrated for $L^2$ projection error and for a model advection-diffusion problem in the original reference.

This discussion focuses on FV methods. For adjoint-based error estimation and mesh adaptation with Discontinuous Galerkin schemes, I refer to the articles of Hartmann et al. [69, 97, 68].

### 2.1.2 Isotropic vs anisotropic FV goal-oriented mesh refinement

Local adaptation strategies try to equidistribute the error estimator they rely on. Various ways of selecting cells to divide (in the standard case of unstructured meshes) or locations where to add mesh lines/planes (for structured mesh refinement) exist – fixed threshold, decreasing threshold, fixed fraction of the degrees of freedom... see [51] §IVC for an overview. In the common situation where remeshing is performed by a dedicated tool, the last major item of the adaptation is the question whether isotropic or anisotropic meshes should be built [130] .

I briefly review the methods presented in the previous subsection from this point of view :
– Venditti and Darmofal used isotropic mesh refinement when dealing with Euler flows [182] and anisotropic mesh refinement when dealing with Navier-Stokes flows [183]. In the later case, they combined their error estimator with the Hessian of the Mach number to introduce anisotropy in their procedure. Besides, Rogé and Martin derived another anisotropic version of Ventitti et al. original method [182] by introducing finer meshes refined in one physical direction only [161];
– More recently Darmofal et al. have used the method proposed by Yano and Darmofal that is intrinsicaly anisotropic;
– the contribution of Dwight involves isotropic mesh refinement for Euler flows [47];
– Fidkowski et al. used isotropic refinement for Euler, laminar and turbulent flows [52, 50]. The mesh adaptation starts from a structured mesh and builds locally adapted meshes with one level hanging-nodes. It is isotropic in the sense that the cells flagged by the error indicator are divided in all mesh-directions but the anisotropy of the initial mesh is locally retained ;
– Alauzet, Dervieux, Loseille and Belme use metric matrices that naturally describe anisotropic meshes [104, 10]. In the work carried out at ONERA, when dealing with structured meshes, anisotropy was

---

[6] for triangles: division in two triangles following the medians, isotropic division in four triangles

supported by mesh parametrization or by an elliptic equation method to be presented. When dealing with unstructured meshes for Euler flow, only isotropic mesh refinement was performed.

## 2.2 Asymptotic analysis of $dJ/X$ for 2D Euler flows

In the later sections, a local mesh refinement indicator based on $dJ/dX$ is proposed and assessed. It is hence useful to establish the order of $dJ/dX$ in the local space step as this allows to estimate the decrease of the indicator when refining the mesh and calculate a local goal mesh size for the next adapted mesh.

This question has been considered from a practical point of view when dealing with unstructured mesh adaptation for 2D Euler flows (see §2.5.2 and figure 2.21).
Besides, Nguyen-Dinh and me have considered this question from a mathematical point of view for regular structured meshes. The total derivative [7]

$$\frac{dJ}{dX} = \frac{\partial \mathrm{J}}{\partial X} + \Lambda^T \frac{\partial R}{\partial X} \,.$$

includes a simple explicit term $\frac{\partial \mathrm{J}}{\partial X}$ and the more complex $\Lambda^T \frac{\partial R}{\partial X}$ that involves the discrete adjoint vector, geometrical dependancies of the scheme and the usual FV metrics formula. Even restricted to the case of 2D Euler flows, this discussion is intricate due to the complexity of $\partial R/\partial X$ and the discontinuities of limiting flow and adjoint fields (§2.2.2 and §2.2.3).

### 2.2.1 Space-order of $(\partial \mathbf{J}/\partial X)$

As concerning $J$ functions, the discussion is restricted to the case of classical outputs of external aerodynamics calculated from flow-field at the wall (without gradient terms) and wall metric. This includes pressure drag, pressure lift, pressure pitching moment, wall-integral of stagnation pressure, temperature...
As concerning the metric, there is no variant in the calculation of 2D edge vectors and, when evaluating the 3D surface vectors, it is assumed that they are calculated as half the cross product of the face diagonals.

Under these assumptions, it is easily verified that $\partial \mathrm{J}_k/\partial X_m$ is
– for a 2D problem, first order in space, except if $X_m$ is the extremity of a non-closed integration curve were it would be zero-th order, but this is typically a case where the node would be fixed and where the projection $\mathcal{P}$ would set its components to zero.
– for a 3D problem, second order in space, except if $X_m$ is at the border of the integration domain (typically symmetry plane in wing or wing-body calculation) where first-order components appear. In case of a symmetry plane, these components will not be all cancelled by the projection operator $\mathcal{P}$.

### 2.2.2 Regularity of flow, $CLp-$ and $CDp-$adjoint for 2D Euler flows

The asymptotic study of $\Lambda^T(\partial R/\partial X)$ has been made for a specific type of schemes, for 2D Euler flows, under a strong assumption of local $C^2$ regularity of the discrete fluxes w.r.t. its two arguments, the flow $W$ and mesh coordinates $X$. Moreover, it is also assumed that the limiting flow and adjoint fields are $C^1$.

This, of course, prevents considering transonic or supersonic flows involving shock-waves. Besides, for a lifting subcritical flow, it is observed that $CLp$ adjoint and $CDp$ adjoint fields look very different close to stagnation streamline. The $CLp$-adjoint field exhibits a singularity at the stagnation streamline as predicted by Giles and Pierce ([60] fourth perturbation, $I_4$, and figure 3d). On the contrary $CDp$ adjoint

---

[7] where for the sake of readability of the technical calculations to come, I have dropped the function and adjoint vector index, $k$

only exhibits a small (physical or numerical) perturbation at this location. This is illustrated in figure 2.2 for the inviscid flow about a NACA0012 at $M_\infty$=0.4 and AoA=$5^o$. Finally, for this subcritical flow, no slip-line at the trailing edge is observed, and this case is selected to illustrate the developments of next subsection due to the regularity of its (numerical) flow and drag adjoint fields.

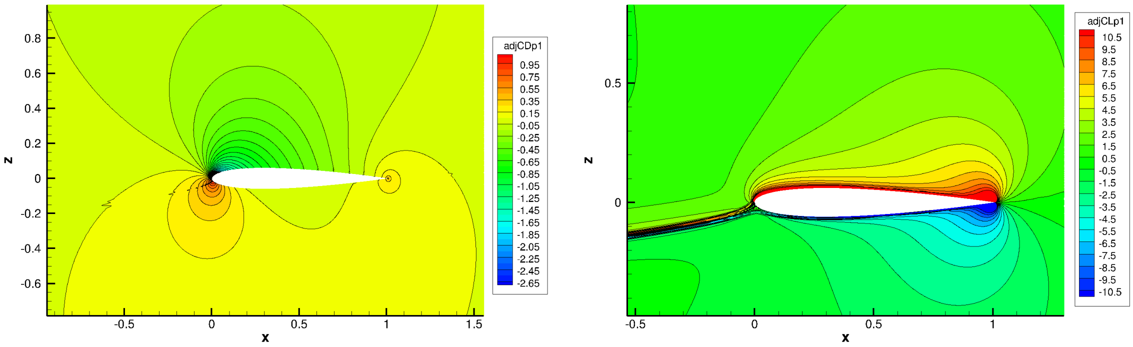


Figure 2.2: NACA0012 $M_\infty$=0.4 AoA=$5^o$ inviscid flow – Left: first component of drag adjoint. Right: first component of lift adjoint

### 2.2.3 Space-order of $\Lambda^T(\partial R/\partial X)$. Link with limiting flow/adjoint fields

Before looking at the theoretical order of $\Lambda^T(\partial R/\partial X)$, the mean norm of $\Lambda^T_{CDp}(\partial R/\partial x)$ and $\Lambda^T_{CDp}(\partial R/\partial z)$ over the interior points of the fluid domain has been calculated for the selected flow. Once again, the considered meshes are those of the classical study of Vassberg and Jameson [180] – from (129×129) to (1025×1025) nodes. The L2-norm of the

$$\Lambda^T_{CDp}(\partial R/\partial x) \quad \text{and} \quad \Lambda^T_{CDp}(\partial R/\partial z)$$

fields has been calculated over the interior nodes of the meshes and divided by the number of nodes. It exhibits $O(h^2)$ behavior. More precisely, the slopes of the very regular log-log plots presented in figure 2.3 are 2.15 and 2.05. When less regular flow and adjoint fields are considered the log-log slope coefficients are significantly lower.

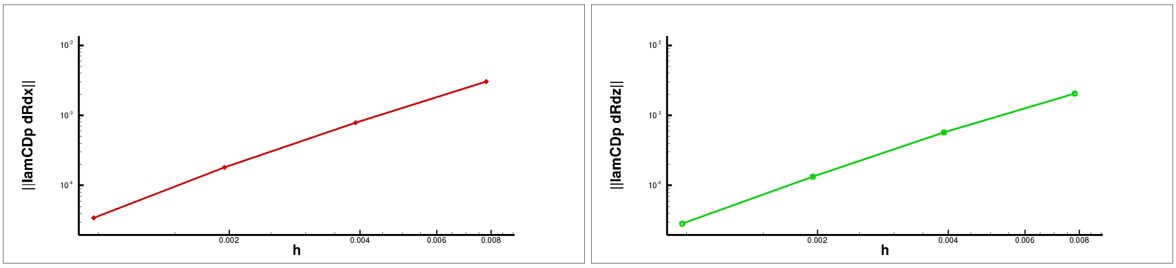


Figure 2.3: NACA0012 $M_\infty$=0.4, AoA=$5^o$, inviscid flow – Left: mean over interior nodes of $||\Lambda^T_{CDp}(\partial R/\partial x)||$ a function of mesh characteristic size (log-log scale). Right: corresponding plot for $||\Lambda^T_{CDp}(\partial R/\partial z)||$

The discussion of the space order of $\Lambda^T(\partial R/\partial X)$ is obviously dependent on scheme $R$. In a first attempt to deal with the subject, the following statement was demonstrated for 2D Euler flows [122, 121].

**Statement:** *A 2D finite-volume cell-centered scheme for Euler flows and for structured grids is considered. The numerical flux is supposed (a) to directly depend on the local surface vector and on two or four states of the corresponding mesh line on either side of the interface; (b) to be $C^2$ except at marginal locations where the absolute value has a zero argument, or min or max functions have equal arguments. The fixed node of interest $X_{i,j}$, is assumed to be located (a) outside of the support of $J$ ; (b) in a zone of the fluid domain where the discrete flow-field $w$ and the adjoint vector tend toward $C^1$ limiting functions $w$ and $\lambda$ ; (c) in a location such that the fluxes of the four surfaces attached to $X_{i,j}$ are $C^2$ functions of their aerodynamic and geometric arguments at the limit of small step sizes. Under these assumptions, the indirect aerodynamic dependence of $J$ w.r.t. Xi,j has the following asymptotic behavior as the mesh is refined:*

$$\begin{pmatrix} \Lambda \dfrac{\partial R}{\partial x_{i,j}} \\ \\ \Lambda \dfrac{\partial R}{\partial z_{i,j}} \end{pmatrix} = ds_{i,j} \sum_{d=1}^{4} \begin{pmatrix} \dfrac{\partial \lambda^d}{\partial z}\dfrac{\partial \mathcal{F}_Z^d}{\partial w}\dfrac{\partial w}{\partial x} - \dfrac{\partial \lambda^d}{\partial x}\dfrac{\partial \mathcal{F}_Z^d}{\partial w}\dfrac{\partial w}{\partial z} \\ \\ -\dfrac{\partial \lambda^d}{\partial z}\dfrac{\partial \mathcal{F}_X^d}{\partial w}\dfrac{\partial w}{\partial x} + \dfrac{\partial \lambda^d}{\partial x}\dfrac{\partial \mathcal{F}_X^d}{\partial w}\dfrac{\partial w}{\partial z} \end{pmatrix} + o(ds) \tag{2.9}$$

*where $\mathcal{F}_X$ (resp. $\mathcal{F}_Z$) is the continuous Euler flux density in direction x (resp. z) evaluted at $w$ and $ds_{i,j}$ is the surface attached to node $X_{i,j}$ (one quarter of the surface of the four neighboring cells).*

This result seemed satisfactory and consistent with the numerical result presented before in figure 2.3 Unfortunately, Nguyen-Dinh and me missed that, if the limiting adjoint field $\lambda$ is solution of the continuous adjoint equation (in Cartesian coordinates [7])

$$\left(\frac{\partial \mathcal{F}_x}{\partial w}\right)^T \frac{\partial \lambda}{\partial x} + \left(\frac{\partial \mathcal{F}_z}{\partial w}\right)^T \frac{\partial \lambda}{\partial z} = 0 \tag{2.10}$$

the term in brackets in the right hand side of equation (2.9) is zero [8]. This is derived from flow equations,

$$\left(\frac{\partial \mathcal{F}_x}{\partial w}\right) \frac{\partial w}{\partial x} + \left(\frac{\partial \mathcal{F}_z}{\partial w}\right) \frac{\partial w}{\partial z} = 0,$$

multiplied by $\partial \lambda/\partial x$ (dot product):

$$\sum_d \left[ \frac{\partial \lambda^d}{\partial x} \left(\frac{\partial \mathcal{F}_x^d}{\partial w}\right) \frac{\partial w}{\partial x} + \frac{\partial \lambda^d}{\partial x} \left(\frac{\partial \mathcal{F}_z^d}{\partial w}\right) \frac{\partial w}{\partial z} \right] = 0.$$

Then equation (2.10), yields

$$\sum_d \left[ -\frac{\partial \lambda^d}{\partial z} \left(\frac{\partial \mathcal{F}_z^d}{\partial w}\right) \frac{\partial w}{\partial x} + \frac{\partial \lambda^d}{\partial x} \left(\frac{\partial \mathcal{F}_z^d}{\partial w}\right) \frac{\partial w}{\partial z} \right] = 0,$$

which proves that the first component of equation (2.9) is zero. The corresponding proof for the second component is very similar.

Hence, the numerical behavior observed for the selected regular flow with regular drag-adjoint is not yet the asymptotic one.

---

[8] An erratum concerning figure 10 of [122] and a short addendum to this article is to be sent to the European Journal of Mechanics B/Fluids

Is then relation (2.9) useless ? Actually at fixed mesh size, when looking at plots of $(1/ds)\ \Lambda^T \dfrac{\partial R}{\partial W}$ and discrete Green-formula based evaluation of

$$\sum_{d=1}^{4} \left( \begin{array}{c} \dfrac{\partial \lambda^d}{\partial z} \dfrac{\partial \mathcal{F}_Z^d(W)}{\partial x} - \dfrac{\partial \lambda^d}{\partial x} \dfrac{\partial \mathcal{F}_Z^d(W)}{\partial z} \\ \\ -\dfrac{\partial \lambda^d}{\partial z} \dfrac{\partial \mathcal{F}_X^d(W)}{\partial x} + \dfrac{\partial \lambda^d}{\partial x} \dfrac{\partial \mathcal{F}_X^d(W)}{\partial z} \end{array} \right), \tag{2.11}$$

it appears that these plots are surprisingly similar. This is illustrated by a series of images presented in annex 3. Conversely, when refining the mesh, both fields have decreasing values. This similarity is also observed when the flow and/or the adjoint field involve discontinuities. Precisely, these discontinuities are then the marked zones in these similar plots and those marked zones get thinner and thinner when the mesh is refined.

## 2.3 Proposed mesh adaptation and mesh assessment method

### 2.3.1 Mesh adaptation based on $dJ/dX$. Intuitions. State of the art

The vector field of the total sensitivity of the goal $J$ w.r.t. the volume mesh node coordinates $X$, $dJ/dX$, seems to provide valuable information to carry out $J$-oriented mesh adaptation or to assess the suitability of mesh $X$ to compute accurately $J$. How it could be used in practice for these purposes is, however, not straightforward. For classical exercises of goal-oriented mesh adaptation, when plotting

$$\|dJ/dX\| \quad \text{or} \quad h\,\|dJ/dX\|$$

on a regular mesh, it appears that the iso-lines of these quantities roughly draw the node density map of $J$-oriented adapted meshes. This comparison is illustrated here in order to gain intuition about $dJ/dX$-based goal oriented mesh adaptation. It is not claimed this is a thorough argument as (1) unstructured goal-oriented adapted meshes published in classical articles are confronted with $dJ/dX$ fields calculated by me on structured meshes; (2) the schemes are not same; (3) we only make a start on the analysis of the critical zones for the accurate evaluation of the outputs of interest (this question is discussed in §2.6. See also [171] §5.4).

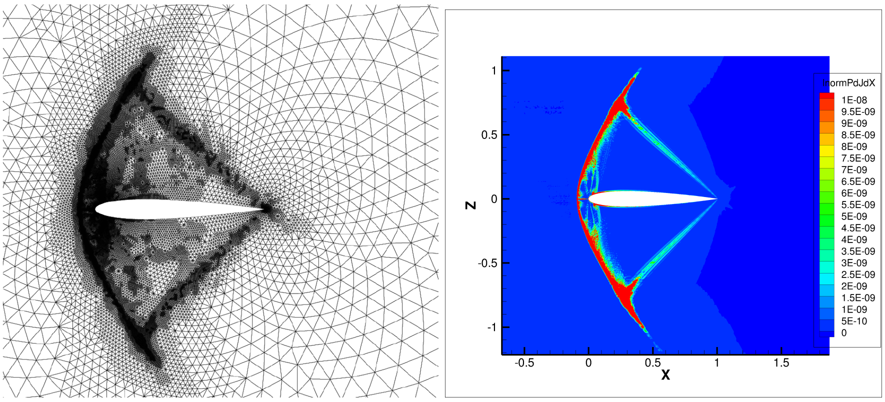


Figure 2.4: NACA0012 $M_\infty$=1.5, AoA=$1^o$. Left: mesh adapted for $CLp$ by Dwight's method ([45] figure 6. Reproduced with permission). Right: iso-($h\ \|dCLp/dX\|$) lines on a regular structured 1025×1025 mesh

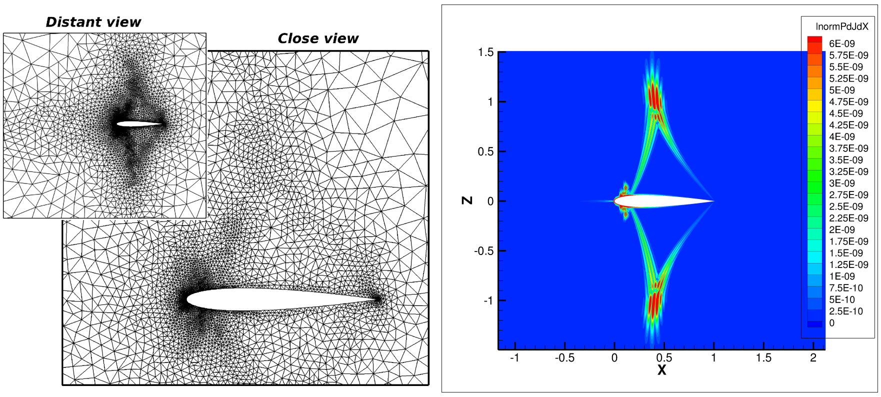


Figure 2.5: NACA0012 M$_\infty$=0.95, AoA=0$^o$. Left: mesh adapted for $CDp$ by Venditti and Darmofal's method ([171] figure 6); this plot is intented to be compared with figure 8 of reference [182]. Right: iso-($h\ \|dCDp/dX\|$) lines on a regular structured 1025×1025 mesh

For the NACA0012 at supersonic flow conditions M$_\infty$=1.5, AoA=1$^o$, the high values of iso-($h\ \|dCLp/dX\|$) (obtained with Roe's scheme on the 1025×1025 mesh of the classical study [180]) clearly draw a section of the detached shockwave and a section of two characteristic lines [9] from shockwave to trailing edge – see figure 2.4. These are precisely the dense mesh zones obtained by Dwight when proceeding to $CLp$-oriented mesh refinement with his method [46, 45, 47]. For the NACA0012 at flow conditions M$_\infty$=0.95, AoA=0$^o$, the high values of iso-($h\ \|dCDp/dX\|$) (obtained with Jameson-Schmidt-Turkel scheme on the same 1025×1025 mesh) draw a diamond shape plus small lines close to the leading edge as shown in figure 2.5. This is very consistent with the dense mesh zones obtained by me when proceeding to $CDp$-oriented mesh refinement with the method of Venditti and Darmofal. A last confrontation is possible for the same profile, at transonic flow condition M$_\infty$=0.85, AoA=2$^o$, for lift, looking at figures 2.1 (middle-down sub-plot) and 2.25. Again the dense zones of the mesh adapted for $CLp$ calculation (figure 2.1) match the isolines of high ($h\ \|dCLp/dX\|$) values.
As regular meshes have been used at this step, it can not be infered at this stage whether $\|dJ/dX\|$ or $h\ \|dJ/dX\|$ could be used for $J$-oriented mesh refinement. This matter is discussed in later subsections.

Somehow surprisingly, outside of ONERA, it seems only NIA has tried to use $dJ/dX$ for $J$-oriented mesh adaptation. In [187], Yamaleev et al. considered a nozzle and 2D Poisson problem for which quasi-exact solutions were available. They used the mesh sensitivity of the square of an output error and a descent method for the mesh to lower the error (and its sensitivity to the mesh). For the nozzle flow, the output was the nozzle pressure integral and the successful demonstration was made using the steady and then the unsteady adjoint of the FUN3D code. In [38], the method was extended to the case where the exact value is not known. The estimated reference value was then based on current estimation and a classical correction (dual-weighted residual after interpolation of flow and adjoint to a finer grid – see §2.1.1). The method was again successfully demonstrated for a nozzle flow.

### 2.3.2 First order variation of the functional output when displacing the mesh

After an hesitation that will briefly appear at the beginning of section 2.4, Nguyen-Dinh, Trontin and I proposed a goal-oriented mesh adaptation method that is based on the first order variation of the output of interest that admissible nodes displacement can cause. This first order variation of $J$ due to a mesh

---

[9] the flow is supersonic everywhere but in a small bubble dowstream the shockwave and close to the leading edge

displacement $dX$ is obviously

$$J(X+dX)-J(X)\simeq \frac{dJ}{dX}.dX$$

The only a priori restrictions in $dX$ being that the nodes displacement dX (1) does not distort the solid shape ; (2) does not distort the support of $J$ ; (3) does not lead to an inconsistent mesh. The simplest way to ensure requirement (3) is to bound the displacement of node $X_m$ by half the lowest distance to its neighbors, denoted $0.5 * h_m$. With this restriction for $dX$ the first-order change in the function of interest is obviously bounded by

$$0.5\sum_{m=1}^{n_X} ||dJ/dX_m||h_m$$

($m$ being the node index). This individual contribution $0.5||dJ/dX_m||h_m$ requires a slight modification to take into account conditions (1) and (2). In particular, when plotting the $dJ/dX$ vector field for simple configurations and classical functions (like nearfield pressure drag $CD_p$ of an airfoil or stagnation pressure integrated over the airfoil – see [185, 144, 143, 121]) it often exhibits long vectors at the solid wall with a significant component normal to the wall. These components, of course, can not guide a node displacement for mesh adaptation as they would change the shape of the solid item of interest. A projected field, denoted $\mathcal{P}(dJ/dX)$, has hence been introduced. It retains all components suitable for mesh adaptation, and cancels the other ones :

$$\begin{aligned}
\mathcal{P}(dJ/dX) &= dJ/dX && \text{Outside the support of} J \text{ and solid walls contour}\\
\mathcal{P}(dJ/dX) &= dJ/dX-(dJ/dX\cdot\vec{n})\vec{n} && \text{Inside the support of } J\text{, along the walls, at the outer border (normal } \vec{n})\\
\mathcal{P}(dJ/dX) &= 0 && \text{At a corner of the support of } J \text{ or at a trailing edge}
\end{aligned}$$

During a mesh adaptation, a node $w$ of the wall surface can only move tangently to the solid surface (before being projected on the exact surface) so that $dX_w = \epsilon\mathbf{t}$, $\mathbf{t}$ being tangent to the wall. The corresponding first order change in $J$ value is

$$J(X+dX_w)-J(X)\simeq dJ/dX_w.(\epsilon\mathbf{t}) = \epsilon\, dJ/dX_w.\mathbf{t} = \epsilon\, \mathcal{P}(dJ/dX_w).\mathbf{t}$$

More generally for any admissible mesh displacement field $dX$

$$J(X+dX)-J(X)\simeq (dJ/dX).dX = \mathcal{P}(dJ/dX).dX. \tag{2.12}$$

Hence, a sharper upper bound for the first-order change of $J$, due do the restriction to an admissible mesh displacement $dX$, is

$$|J(X+dX)-J(X)|\simeq |\frac{dJ}{dX}.dX| \le 0.5\sum_m ||\mathcal{P}(dJ/dX_m)||h_m. \tag{2.13}$$

This point is now discussed based on equation (2.13) and figure 2.6 representing three possible $\mathcal{P}(dJ/dX)$ fields. Figure (2.6)(a) is a typical case where equation (2.13) proves that a local mesh refinement is needed, as moving down the three mesh lines with high $||dJ/dX||$ values would cause a significant increase of $J$ value. Figure (2.6)(b) is a case of high sensitivity of $J$ value to the position of some nodes but, as these nodes can not be significantly moved, the evaluation of $J$ does not appear to be sensitive to a simple feasible mesh deformation – figure (2.6)(b) is actually typical of $\mathcal{P}(dJ/dX)$ fields for Euler flows, a near field function and a classical mesh with stretching of mesh lines/planes close to the wall. Figure (2.6)(c) is a case of high sensibility of $J$ value to the position of some nodes that can be significantly moved but the contribution of the different nodes in (2.13) tend to cancel if they are moved coherently.

Hence, it is not easy to decide whether this zone should be refined for a stable evaluation of the output of interest.

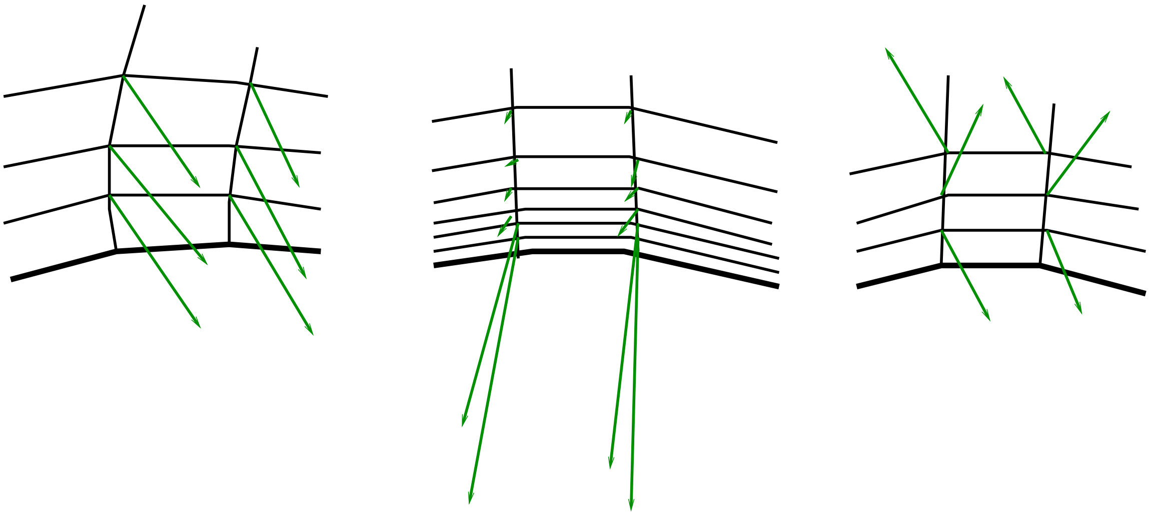


Figure 2.6: Three types of $\mathcal{P}(dJ/dX)$ fields

This third case and the wish to characterise the "reasonable" mesh displacements $dX$ when doing mesh adaptation by nodes displacement, initiated an effort to define a relevant spatial mean of $\mathcal{P}(dJ/dX)$ field. In a framework where the mesh displacement fields $dX$ corresponds to the values at the nodes of a $C^1$ regular function assumed to be well approximated by its first-order Taylor expansion on all circles of radius $L$, a spatial mean $\overline{\mathcal{P}(dJ/dX)}$ is exhibited such that

$$\mathcal{P}(dJ/dX).dX = \overline{\mathcal{P}(dJ/dX)}.dX$$

The construction of this $\overline{\mathcal{P}(dJ/dX)}$ mean first identifies $dX$ at any node with a spatial mean of $dX_k$ over a circle of radius $L$. Then the sum of the dot product and the sum of the mean are reversed to build $\overline{\mathcal{P}(dJ/dX)}$. Detailed definition is given in Annex 4.

### 2.3.3 Criterion for goal-oriented mesh assessment

A demand of Airbus during the PhD thesis of Nguyen-Dinh [121] was to propose a characterization of the meshes suitable for the calculation of a specific output functional. Of course, the error estimators of the classical goal-oriented mesh-refinement methods presented in section §2.1 provide such an information – Venditti and Darmofal method [182, 183] based on interpolation to a finer grid, Dwight's method [47] based on artificial dissipation of Jameson-Schmidt-Turkel scheme [85]. In the framework defined in the previous subsection, the relevance of $\theta$,

$$\theta = \frac{1}{n_X} \sum_{m=1}^{n_X} \theta_m \qquad\qquad \theta = 0.5\ ||\mathcal{P}(dJ/dX_m)||\ h_m \tag{2.14}$$

and $\overline{\theta}$,

$$\overline{\theta} = \frac{1}{n_X} \sum_{m=1}^{n_X} \overline{\theta_m} \qquad\qquad \overline{\theta}_m = 0.5\ ||\overline{\mathcal{P}(dJ/dX)}_m\ ||h_m,$$

for this purpose was studied. (NB: When considering several aerodynamic functions, $[J]$ is added to the previous notations in order to indicate the function to which the criterion is related.) A mesh assessment exercise, that is detailed in reference [122] section 6, is summarized here for one test flow condition and one function.

### 2.3.4 A simple example of $\theta[J]$-based mesh-assessment

A subsonic inviscid flow around a slightly modified NACA0012 is considered ($M_\infty = 0.5$ $AoA$=0°). The flow is calculated with JST scheme on hierarchy of regular meshes defined by Vassberg for a well-known mesh convergence study [180]. The functions of interest, $CDp$ and $P_a$ , are defined by:

$$CD_p = \oint_\Gamma \frac{2}{\gamma M_\infty^2} \left( \frac{p}{p_\infty} - 1 \right) \vec{n} \cdot \vec{e_\infty} \, dl \tag{2.15}$$

$$P_a = \frac{1}{L(\Gamma) p_{a\infty}} \oint_\Gamma p_a \, dl \tag{2.16}$$

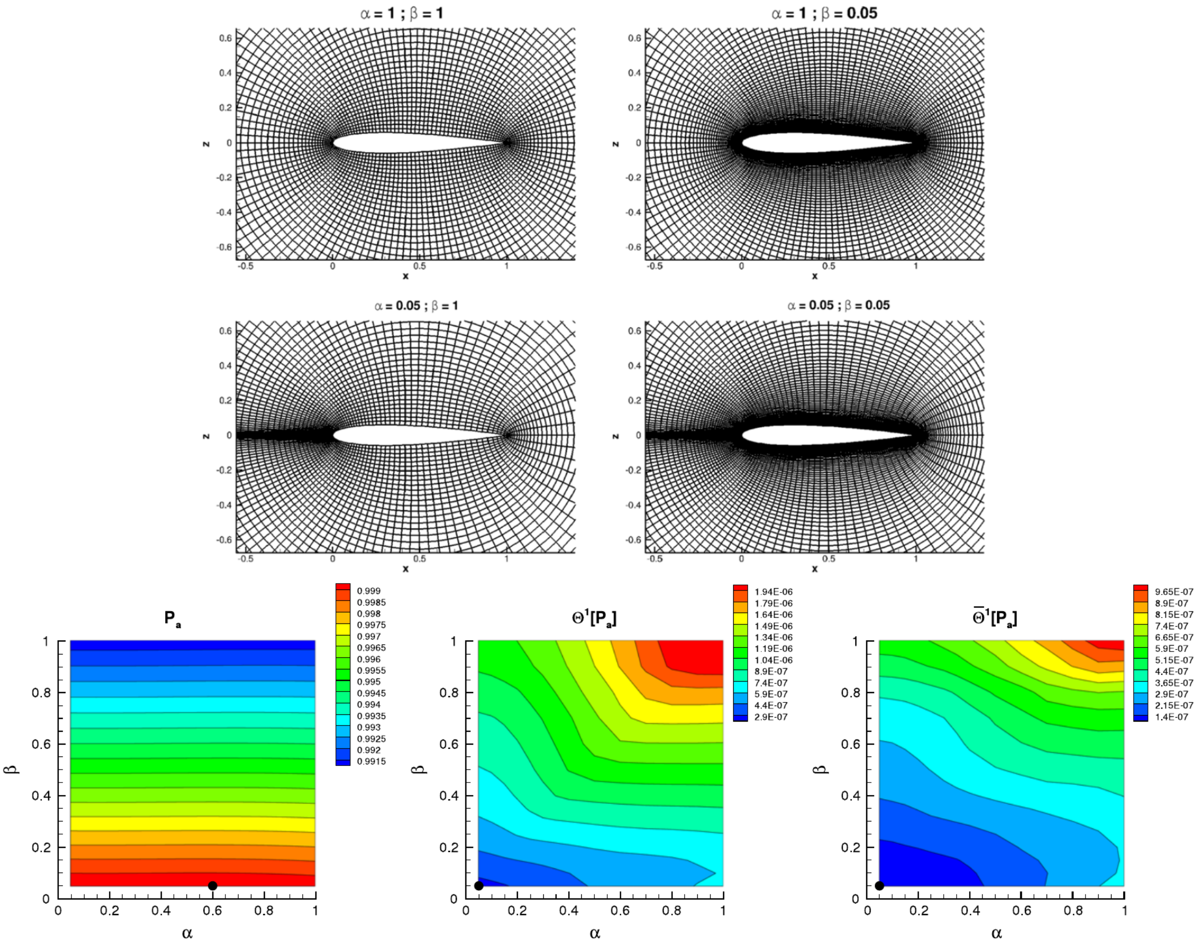
where $p_\infty$ and $p_{a\infty}$ are respectively the upstream static pressure and stagnation pressure. The output functional values are estimated on the series of regular grids in order to provide accuracy references for the future results on adapted meshes ([122] Tables 1 and 2, subcritical part). Of course the ideal values of $CD_p$ and $P_a$ for this inviscid subcritical flow are respectively 0. and 1. which allows exact error evaluation.

A two-parameter family of (129 × 129) meshes is then built by linear interpolation in the (2049 × 2049) mesh defined by Vassberg [180]. It is mathematically defined in reference [122] section §6.1.2 and presented here visually in the upper part of figure 2.7. The $\alpha$ parameter of the family drives a stretching of $I$-mesh lines upwind the profile and the $\beta$ parameter drives the stretching of $J$-mesh lines close to the wall.

Figure 2.7(a) presents $Pa$ values in terms of mesh parameters ($\alpha$, $\beta$) (196 meshes are used, 14 for each parameter). The theoretical $P_a$ value is 1. In this case numerical dissipation acts essentially monotonically on $Pa$ by reducing its value. We notice that meshes that provide the better $P_a$ estimations are those which are generated with the smallest value of $\beta$ ($\beta$ = 0.05 meshes such that J-lines are closely gathered around the airfoil).

The question to be answered is whether small values of criteria $\theta[P_a]$, and $\overline{\theta}[P_a]$ corresponds to accurate values of $P_a$.

Actually $\theta[P_a]$, and $\overline{\theta}[P_a]$ achieve their lower values for the small values of $\beta$ and $\alpha$ (see figure 2.7 (b) and (c)). On the lower border $\beta$ = 0.05, the variations of the function of interest $P_a$ are very small (from 0.99938 to 0.99948). It is at ($\alpha$, $\beta$) = (0.05, 0.05) that the criteria $\theta$ and $\overline{\theta}$ achieve their lower values. At this point of the parameter space $P_a$ is equal to 0.99946 whereas the closer value to the limiting one is obtained at ($\alpha$, $\beta$) = (0.6, 0.05) ($P_a$ = 0.99948). The first-order variation criteria, $\theta$ and $\overline{\theta}$, applied to $P_a$ point out good meshes for the computation of $Pa$ whereas its absolute minimum does not correspond to the best mesh but to one for which the estimation of $P_a$ is close to best.



(a) (b) (c)

Figure 2.7: Up : Family of parametrized structured (129 × 129) meshes about the NACA0012 – Down: 2D Subsonic flow about NACA0012 (a) $P_a$ as function of $\alpha$ (I-lines) and $\beta$ (J-lines) parametrization (b) $\Theta[Pa]$ in the parameter domain (c) $\overline{\Theta}[Pa]$ values in the parameter domain

### 2.3.5 Criterion for goal-oriented mesh refinement

The retained criterion, denoted $\theta[J]$ is hence

$$\theta[J]_m = 0.5||\mathcal{P}(dJ/dX_m)||\, h_m,$$

$h_m$ being the minimal distance of node $m$ to its neighbors. (When dealing with a unique $J$ function the $[J]$ mention is omitted as in the remaining part of this section).

For structured mesh adaptation, the criterion can be summed over the nodes of a mesh line (2D) / mesh plane (3D) and the highest values of this sum define the locations where mesh lines (2D) or mesh planes (3D) should be added or drawn closer.

In the more classical framework of unstructured mesh adaptation, $\theta_m$ is the local indicator for refinement. In case an asymptotic property is known for $\mathcal{P}(dJ/dX)$ or, at least, for $||\mathcal{P}(dJ/dX)||$, like

$$||\mathcal{P}(dJ/dX)|| = Ch^p + o(h^k)$$

(where the constant $C$ depends on the location but not on $h$ of course), the calculation of the desirable cell-size is straightforward. If $T$ is the threshold for $\theta$ values, that is locally exceeded, the reduced local mesh size compatible with $T$ limit for $\theta$ is such that

$$0.5\, C\, h_{new}^{p}\, h_{new} = \frac{T}{\theta_m}\theta_m = \frac{T}{\theta_m}\, 0.5\, C\, h_{cur}^{p}\, h_{cur}$$

so that

$$h_{new} = \left(\frac{T}{\theta_m}\right)^{1/(p+1)} h_{cur}.$$

## 2.4 Goal-oriented adaptation of structured meshes

### 2.4.1 Mesh adaptation for Euler flows with heuristic mesh transformation

Nguyen-Dinh, Trontin and I first carried out mesh adaptations for inviscid external flows by nodes displacement and node addition based on the higher values of

$$\mu_m = 0.5||\mathcal{P}(dJ/dX_m)||$$

instead of $\theta_m$ defined by equation (2.14). In these first adaptations, the functions of interest $J$ were the pressure drag or the stagnation pressure integrated over the wall (2.15) (2.16). During the mesh adaptation process, the global values

$$\theta = \frac{1}{n_X}\sum_{m=1}^{n_X} 0.5||\mathcal{P}(dJ/dX_m)||h_m \quad \text{and} \quad \mu = \frac{1}{n_X}\sum_{m=1}^{n_X} 0.5||\mathcal{P}(dJ/dX_m)||$$

were monitored. These trials where actually quite successful as the adaptations started from regular structured meshes for which adaptation based on $\mu_m$ lead to the required refinement close to the walls as would also have been obtained with $\theta_m$ criterion. Nevertheless at the end of these researches [147, 144, 143], in the conclusion of reference [143], it appeared that $\theta$ would be a better global indicator than $\mu$ for the suitability of the mesh to calculate $J$. Interested reader can have a look at $\mu[Pa]$ and $\theta[Pa]$ values in table 5 and table 6 of reference [143] where it appeared that the $\mu[Pa]$ values are not lower on stretched and specifically adapted grids, that are both well-suited for accurate near-field function calculation, than on less appropriate grids. Besides, a set of evidences indicated that $\theta_m$ would be a better local goal-oriented criterion that $\mu_m$ ; in particular, as the nodes of the very first mesh-lines adjacent to the wall are moved closer to the wall, their $\mu_m$ values get higher and higher.

These mesh adaptations were based on mono-block structured meshes about the NACA0012 aerofoil and the ONERA M6 wing. Very dense meshes were available thanks to Vassberg (NACA0012 airfoil [180]) and Destarac (ONERA M6 wing). They allowed simple addition or displacement of lines/planes by linear interpolation. The reader is referred to [143] for the description of the node addition and node displacement method and corresponding results (or, for node addition method, to §2.4.4 where its application to a 3D RANS flow is presented). Only aspects of $dJ/dX$ fields, $\theta$ and $\mu$ values are now discussed.

The NACA0012 test case with subsonic flow conditions M$_\infty$ $= 0.5$, AoA= $0^o$ is retained for this purpose. The meshes of Jameson and Vassberg's study [180] have O-topology. Each quadrilateral cell of these meshes has an aspect-ratio of one and the mesh-lines are orthogonal at each grid point. The sizes of those selected for the study are (129 × 129), (257 × 257), (513 × 513), (1025 × 1025), (2049 × 2049) discarding the finest and two coarsest meshes of the original study. These meshes are referred to as "quasi-regular". Besides, based on these meshes, a second family was built, with aspect ratio 1/8 at the wall and power law for the width of the cells in the direction from wall to far-field. Along the airfoil, the mesh size is minimum near $z = 0$ (upstream of leading edge, downstream of trailing edge) and $x = 0.5$ (shock location for transonic conditions $\mathrm{M}_\infty = 0.8$, AoA=$0^o$) at the upper and lower part of the airfoil. The ratio of the cell-width in this direction at these locations with the mean length in the other mesh direction is 1/3. The far-field boundary for both families is about 150 chord lengths away from the airfoil. These meshes are referred to as "stretched". For a discussion of the aspect of $\mathcal{P}(dJ/dX)$ vector fields, $\mathcal{P}(dP_a/dX)$ and $-\mathcal{P}(dCPp/dX)$ are presented in figure 2.8 and 2.9 for the (129×129) (257×257) and (513×513) meshes.

At the scale of the wider plots of these two sets of figures, $\mathcal{P}(dP_a/dX)$ and $-\mathcal{P}(dCPp/dX)$ can only be seen in the vicinity of the leading edge and for the point just behind the trailing edge. This is why they are plotted at the leading edge. When examining the plots from coarse to fine meshes, we note that :

(1) $\mathcal{P}(dJ/dX)$ are not much larger for the wall nodes although they involve a first order component in space (§2.2.1 – fig.4(b) in [144]).

(2) for the two coarsest meshes and at least for $P_a$, the $\mathcal{P}(dJ/dX)$ vectors for the nodes close to the wall are not shorter for the stretched meshes than for the quasi-regular ones. Besides more nodes have significant $\mathcal{P}(dJ/dX)$ vectors on the stretched (129 × 129) and (257 × 257) meshes than on the corresponding quasi-regular meshes. It is well-known for the long practice of Euler CFD that stretched close to wall meshes are better suited for near flow and aerodynamic function calculation than quasi-regular meshes [10], this gives an hint that minimizing the $\theta$ criterion may be more suitable than minimizing the $\mu$ criterion to adapt meshes for $J$ calculation.

(3) For external inviscid flows, the static pressure at the wall should be equal to its far-field value but is actually lowered by numerical dissipation so that displacing nodes to increase $P_a$ seems a good idea. When looking at the left column of figure 2.9, it appears that the $\mathcal{P}(dP_a/dX)$ fields "recommend" to move the first $J$-mesh lines closer to the wall, that is a sound advice for the reason recalled just above.

(4) When looking more closely at the $\mathcal{P}(dP_a/dX)$ field on the quasi-regular (513×513) mesh, it is observed that following the direction of the vectors to guide individual moves of J-mesh lines following the J-lines of the finest mesh [11], would bring closer to the wall lines J=2,3,5 but bring further from the wall line J=4. This displacement of these four J-lines would create an irregular spacing and would not be satisfactory. If adapting the mesh to increase $P_a$ is a sound idea, $P_a$ as a function of the node coordinates (or even the position of J mesh-lines) is very multi-modal function so that the maximization should be carried out with care enforcing standard regularity for the mesh.

Finally, table 2.1 confirms that the $\theta[J]$ values are lower on the stretched meshes than on the quasi-regular meshes. As concerning the $\mu[J]$ values, the $\mu[CD]$ values are higher for the five quasi-regular meshes than for the corresponding stretched meshes, whereas the ordering depends on the refinement level for $\mu[Pa]$. This confirms that $\theta[J]$ is better suited than $\mu[J]$ for assessing the capability of a mesh to lead to an accurate $J$ evaluation. [12]

### 2.4.2 Mesh adaptation for 2D (RANS) flows with a $\theta$-based heuristic

The work of Resmini, Lucor and I [157] extends to (RANS) flows the line adaptation methods used for Euler flows [143]. To get a very fine mesh allowing the definition of new meshes by linear interpolation Resmini mapped the NACA0012 (4097×4097) mesh of Vassberg and Jameson [180] about the RAE2822. The hierarchy of meshes allowing limiting value calculations for lift and drag was then derived by coarsening. The $\theta$ criterion was calculated on the successive meshes of the adaptation process. Its mean was computed along mesh-lines (2D) to define the zones to refine. $CDp-$oriented adaptations

---

[10] that are essentially tools for mesh convergence analysis

[11] J-mesh lines are those circling around the aerofoil. They are approximately located at constant distance to the wall

[12] with the only exception of $\theta[Pa]$ on the (2049 ×2049) meshes

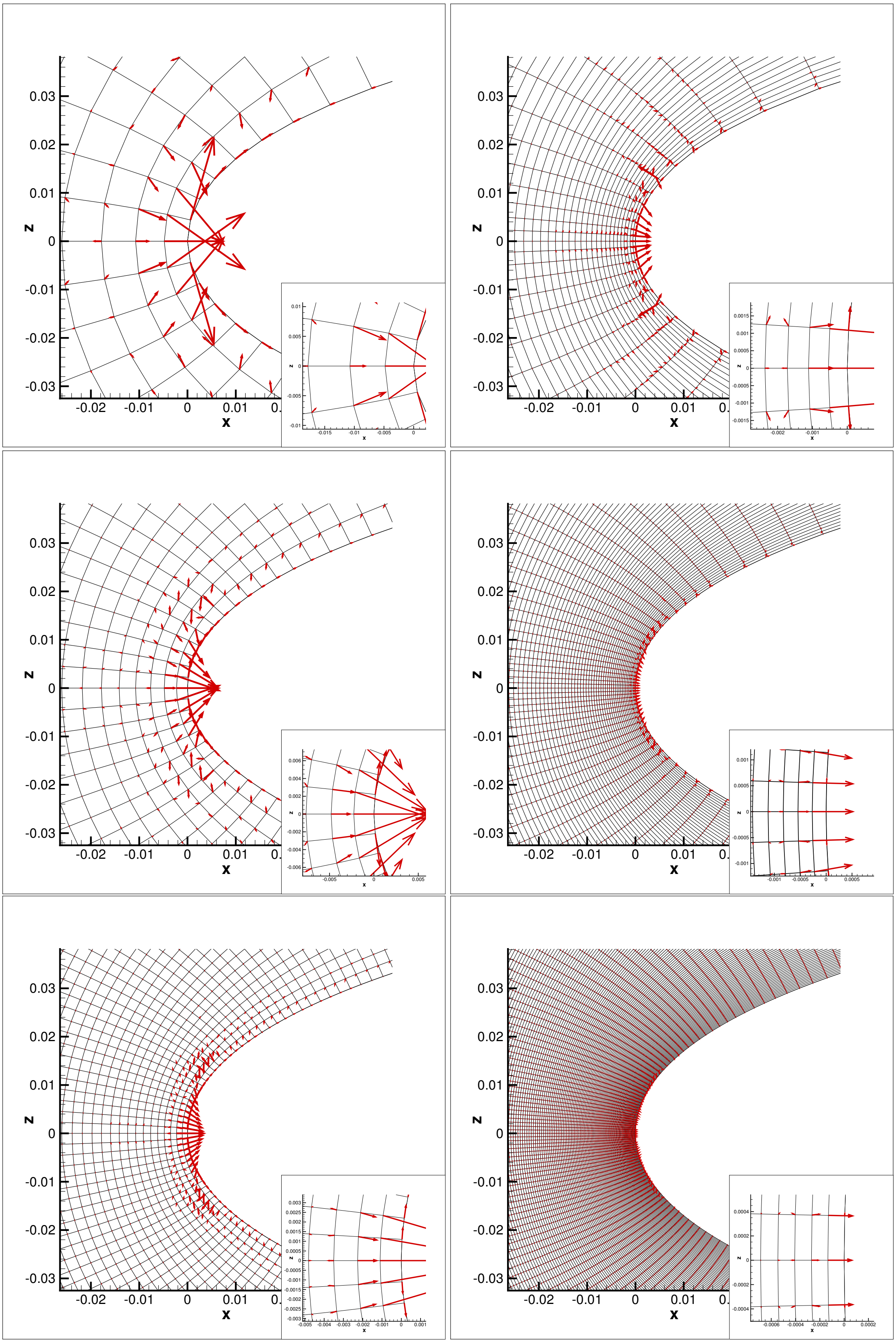


Figure 2.8: NACA0012, subcritical flow conditions. Examination of $-\mathcal{P}(dCDp/dX)$ for (129×129) (top) (257×257) (middle) (513×513) down for quasi-uniform (right) and stretched mesh (right) for Roe's scheme (same scale for all plots).

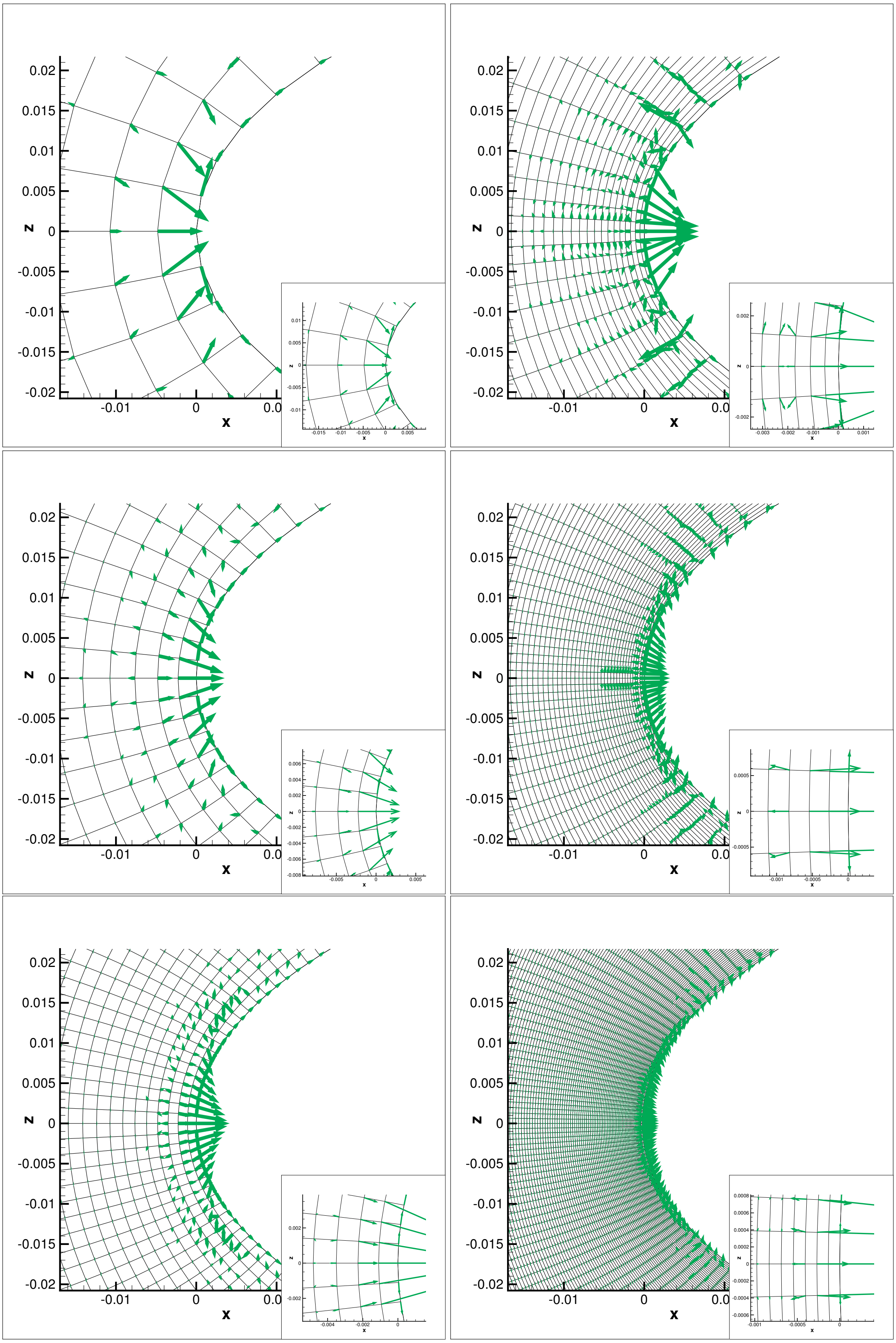


Figure 2.9: NACA0012, subcritical flow conditions. Examination of $\mathcal{P}(dPa/dX)$ for (129×129) (top) (257×257) (middle) (513×513) down for quasi-uniform (right) and stretched mesh (right) for Roe's scheme (same scale for all plots).

| | $CDp\,(\times 10^4)$ | $\mu[CDp]$ | $\theta[CDp]$ | $P_a$ | $\mu[P_a]$ | $\theta[P_a]$ |
|---|---|---|---|---|---|---|
| quasi-uniform grids | | | | | | |
| $lim.\ value$ | 0. | —— | —— | 1. | —— | —— |
| $2049 \times 2049$ | 0.169 | $3.95\ 10^{-8}$ | $1.22\ 10^{-11}$ | 0.99905 | $6.70\ 10^{-6}$ | $1.24\ 10^{-9}$ |
| $1025 \times 1025$ | 0.674 | $2.89\ 10^{-7}$ | $1.44\ 10^{-10}$ | 0.99812 | $2.96\ 10^{-5}$ | $3.75\ 10^{-8}$ |
| $513 \times 513$ | 2.634 | $1.73\ 10^{-6}$ | $1.30\ 10^{-9}$ | 0.99622 | $1.02\ 10^{-4}$ | $7.55\ 10^{-7}$ |
| $257 \times 257$ | 10.331 | $9.03\ 10^{-6}$ | $1.66\ 10^{-8}$ | 0.99217 | $3.19\ 10^{-4}$ | $1.04\ 10^{-5}$ |
| $129 \times 129$ | 40.986 | $4.38\ 10^{-5}$ | $3.99\ 10^{-7}$ | 0.98396 | $1.05\ 10^{-3}$ | $9.75\ 10^{-5}$ |
| stretched grids | | | | | | |
| $2049 \times 2049$ | 0.005 | $1.52\ 10^{-8}$ | $3.16\ 10^{-13}$ | 0.99983 | $1.60\ 10^{-5}$ | $1.65\ 10^{-9}$ |
| $1025 \times 1025$ | 0.018 | $6.20\ 10^{-8}$ | $3.17\ 10^{-12}$ | 0.99973 | $2.33\ 10^{-5}$ | $1.02\ 10^{-8}$ |
| $513 \times 513$ | 0.119 | $3.68\ 10^{-7}$ | $5.45\ 10^{-10}$ | 0.99950 | $8.39\ 10^{-5}$ | $4.36\ 10^{-8}$ |
| $257 \times 257$ | 1.079 | $2.72\ 10^{-6}$ | $1.15\ 10^{-9}$ | 0.99898 | $4.35\ 10^{-4}$ | $1.12\ 10^{-7}$ |
| $129 \times 129$ | 7.849 | $2.47\ 10^{-5}$ | $1.65\ 10^{-8}$ | 0.99784 | $2.34\ 10^{-3}$ | $3.45\ 10^{-6}$ |

Table 2.1: $CDp$ and $P_a$ for baseline and stretched grids. $M_\infty = 0.5$.

were carried out for the classical flow conditions about the RAE2822 leading satisfactory enhancement of $CDp$ convergence towards its limiting value w.r.t. calculations on regularly refined meshes.
Besides, an attempt to use non-matching joins was done for the RAE2822 case[13]. The fluid domain was split in three zones delimited by two edges that follow specific $J-$lines of the very fine mesh. It was actually possible the avoid unnecessary refinements up to far-field but the balance of the $\theta$ values between the three domains was not easy to manage and the exercise was eventually conducted with a fixed mesh about the profile, refining only the two other domains.
The reader is referred to figures 2 and 3 of reference [157] for the synthesis of this study.

### 2.4.3 Mesh adaptation for a 3D (RANS) flow with a $\theta$-based heuristic

The corresponding study for the ONERA M6 wing ([157] §5) is described here with some more details. The flow is at Re = 11.72 $10^6$ based on the mean aerodynamic chord of 0.64607m and Mach number at 0.8395 with AoA = $3.06^o$ and null angle of sideslip [14]. Under these conditions, the classical $\lambda$-shock structure is observed on the wing surface. (RANS) equations completed by Spalart-Allmaras turbulence model (SA) are retained. The selected discretization for mean flow and turbulence equations are detailed in [157] §2.2 §3.1. The accuracy of elsA's adjoint in $dJ/dX$ mode for (RANS)&(SA) discrete equations is also discussed and found satisfactory ([157] §3.2).

A 513 × 513 × 257 reference fine mesh was provided by Destarac and Dumont. Its O-O topology is presented by figure 2.10 and corresponding caption. The hierarchy of meshes allowing limiting value calculations for lift and drag is derived by coarsening the $513 \times 513 \times 257$ mesh. The calculated limiting values are $CD_{lim}$=0.01870 for drag and $CL_{lim}$=0.274 for lift. A mesh adaptation by plane addition is conducted up to a drag-error (w.r.t. $CD_{lim}$) lower than 0.0005.

Following the idea proposed by Nguyen-Din and me in [143], all involved $N_i \times N_j \times N_k$ meshes are defined by three increasing sequences of real numbers denoted $\varphi_i$, $\varphi_j$, $\varphi_k$. Sequence $\varphi_i$ (resp. $\varphi_j$, $\varphi_k$) defines the position of the $N_i$ $i$-planes (resp. $N_j$ $j$-planes, $N_k$ $k$-planes) within the 513 (resp. 513, 257) ones of the reference fine mesh. Those sequences fully define the indices and coefficients of the bilinear

---

[13] The adjoint module of the $elsA$ code can manage non-matching conditions

[14] the reference surface, 0.7532, is obtained as the trapezoidal part of the wing

interpolation $\mathcal{B}$ that calculates the current mesh from the reference one following

$$\begin{array}{ccccc} & \varphi & & \mathcal{B} & \\ \{1,N_i\}\{1,N_j\}\{1,N_k\} & \longrightarrow & [1,513]\,[1,513]\,[1,257] & \longrightarrow & \mathbb{R}^3 \\ (i,j,k) & & (\varphi_i,\varphi_j,\varphi_k) & & X(x,y,z) \end{array}$$

The starting mesh prior to adaptation is the same for the standard base refinement (i.e. by doubling the total number of nodes at each step) and for the $CD$-based mono-block adaptation: namely, with $N_i$ = $N_j$ = 65 and $N_k$ = 33, uniformly distributed in $i$ and $k$ directions and 20 $j$-mesh planes in the boundary-layer. At each adaptation step, the $dCDp/dX$ and then $\theta[CDp]$ fields are calculated. Means of $\theta[CDp]_m$ are then calculated over all $i$, $j$ and $k$ mesh-planes and then averaged two by two to attribute a mean to the $(N_i-1)$ $i$-cell-planes, the $(N_j-1)$-$j$ cell-planes and the $(N_k-1)$ $k$-cell-planes. The cell-planes with the higher values are then divided in two to five parallel planes depending on their mean $\theta$ value. In term of $\varphi$ sequences, this corresponds to enriched sequences. To ensure a smooth node location, the $\varphi_l$ sequences resulting from plane-addition are regularized with a third order dissipation applied to $(\varphi_{l+1}-\varphi_l)$ before using bilinear interpolation to get the next mesh.

$CD$ convergence is plotted in Figure 2.12 for regularly refined and $\theta$-based adapted grids. The asset given by the proposed method is patent. The $CD$ estimation error in the $CD$-based adapted mesh is under five drag counts w.r.t. both the estimation from the finest standard grid and from the Richardson method. Concerning the $CL$, the estimation error is under one lift count.

In the context of monoblock 3D structured meshes, in order to deal with meshes of different sizes and easily identify the refinement zones, the $\varphi_i$ sequences ($\{1,N_i\}\to[1,513]$) are transformed, by straightforward affine mapping, to discrete values of increasing ($[0,1]\to[0,1]$) functions denoted $\Phi_i$ (and just the same for $\varphi_j$ and $\varphi_k$). Figure 2.11 is precisely the plot of $\Phi_i$, $\Phi_j$ and $\Phi_k$ for the starting base mesh and for the $CD$-based adapted one [15]. The low slope areas in the distribution of the $CD$-based adapted mesh correspond to refinement areas. For instance, in Figure 2.11(a), a refinement is undergone around $i_{min}$ and $i_{max}$ (trailing edge) as well as around $(i_{min}+i_{max})/2$ (leading edge). Similarly, in 2.11(c), a refinement is identified around $k_{min}$ (wing tip) and $k_{max}$ (symmetry plane).

### 2.4.4 Euler and (RANS) adaptation based on an elliptic pde. Applications to 2D flows.

Nguyen-Dinh could benefit of Airbus-F strong experience in structured mesh adaptation by nodes displacement. One method used at that time by the CFD groups of this company relies on the elliptic equation

$$\sum_{i,j=1}^{3} g^{ij}\, x_{\xi^i\xi^j} + \sum_{k=1}^{3} g^{kk} P_k\, x_{\xi^k} = 0. \tag{2.17}$$

The unknown vector is the position vector $\mathbf{x}=(x_1,x_2,x_3)$ whereas $\xi=(\xi_1,\xi_2,\xi_3)$ are the curvilinear coordinates (structured planes/lines indices are discrete values of these curvilinear coordinates). After the contravariant and covariant base vectors of the $(\mathbf{x},\xi)$ mapping have been defined,

$$g_i = \frac{\partial \mathbf{x}}{\partial \xi_i} \quad \text{and} \quad g^i = \frac{\partial \xi}{\partial x_i},$$

the contravariant metric tensor is simply

$$g^{ij} = (g^i, g^j)$$

---

[15] Please note the uniform distribution in $i$ and $k$ while the exponential distribution in $j$ (due to the BL) for the starting base mesh

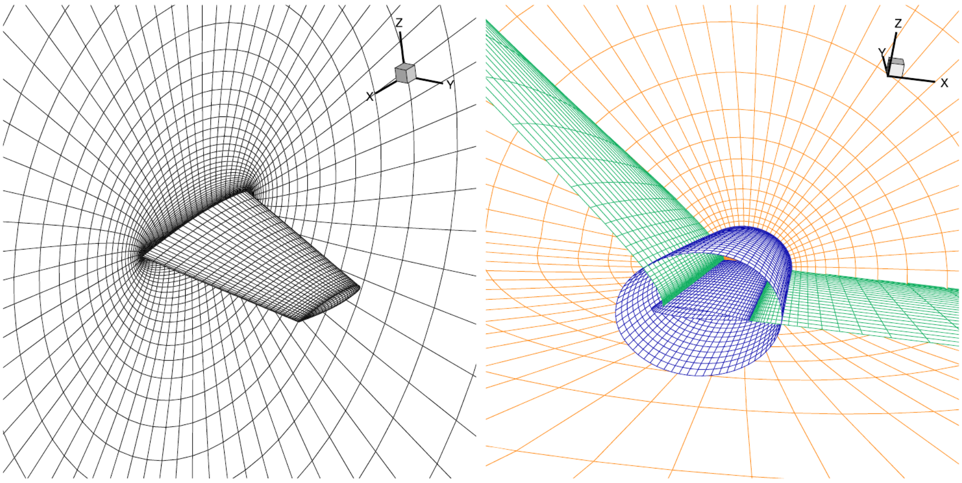
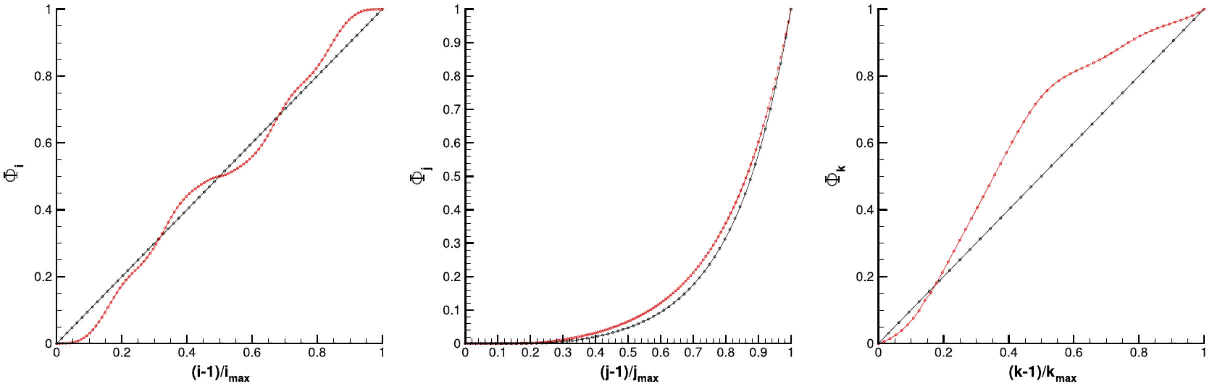


Figure 2.10: ONERA M6 wing, (RANS) flow. Starting regular structured 65×65×33 mesh prior to adaptation. Left : view of the symmetry plane and the wing surface. Right : view of the mesh topology, $i$-mesh planes in green run clockwise along the wing section from the trailing edge along the pressure side and back to the trailing edge along the suction side, $j$-mesh planes in blue go from the wing surface to the farfield boundary, $k$-mesh planes in orange run from the tip to the root of the wing (they are symmetric w.r.t. $z = 0$). The hemispheric farfield boundary is placed at 145 chords from the leading edge.



(a) $i$-mesh plane (b) $j$-mesh plane (c) $k$-mesh plane

Figure 2.11: ONERA M6 wing, (RANS) flow. Functions defining the position of the $i$- $j$- and $k$-planes w.r.t. those of reference fine mesh

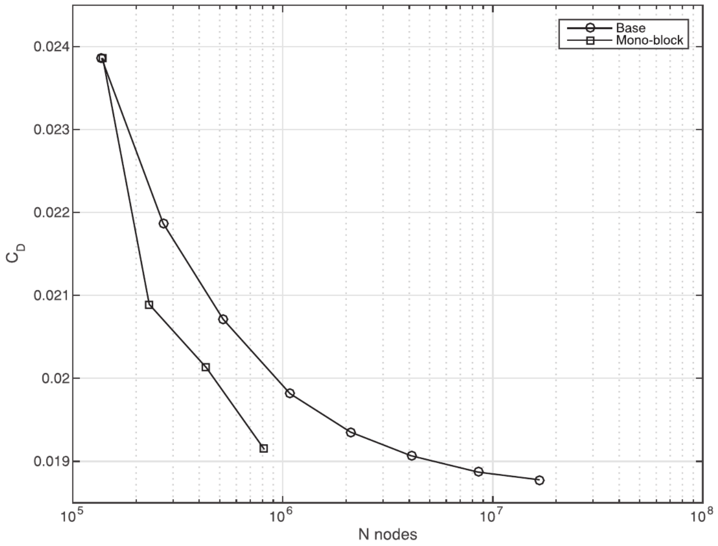


Figure 2.12: ONERA M6 wing, (RANS) flow. Convergence of $CD$ for the regular and adapted meshes

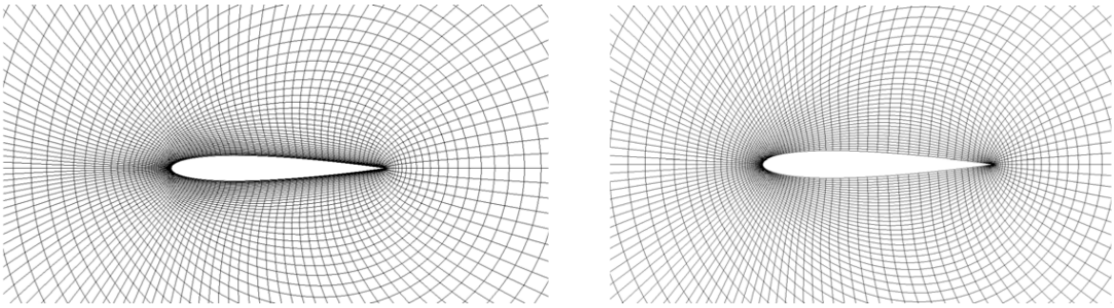
All terms of equation (2.17) are now defined but the so-called control functions, $P_k$. These $P_k$ fields entirely define the structured mesh. They may be calculated *a posteriori* even if the structured mesh has not been generated using this theoretical framework. For mesh adaptation, the $P_k$ can be modified according to

$$P_k = P_k^{initial} + \epsilon P_k^{adapt}$$

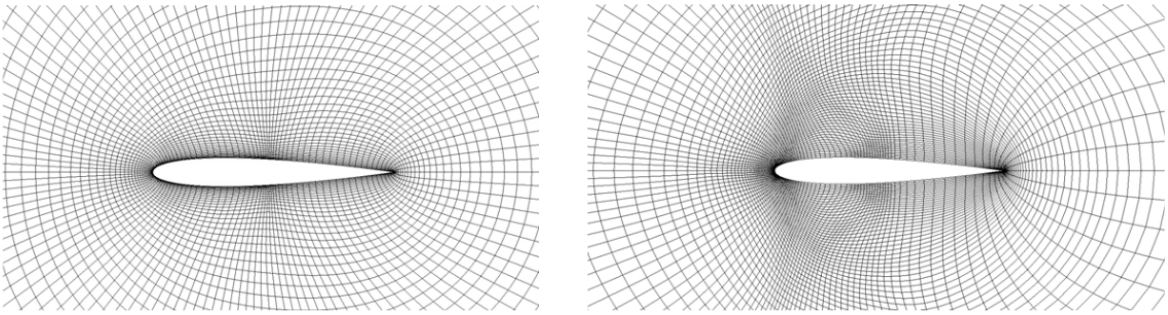
where $\epsilon$ is a constant factor, $P_k^{initial}$ is the $k$-th control function of the initial mesh and $P_k^{adapt}$ is built according to the user's criterion. The calculation of $P_k^{adapt}$ has been based on intermediate fields defined as the norm of $\mathcal{P}(dJ/dX)$ times the characteristic lengths in the two (in 2D) respectively three (in 3D) mesh directions.
As I was not closely associated to this part of the work of Nguyen-Dinh done at Airbus-Saint-Martin-du-Touch and the reader is referred to [122] §4, [121] §3.2 for the details of the $\theta$-based calculation of $P_k^{adapt}$ and to references 23 to 27 of [122] for the elliptic equation method.

Two mesh adaptations for 2D inviscid flows, carried out by Nguyen-Dinh, Sauvage and Meaux, are briefly described here to point out some assets of the method [122]. Starting from Vassberg and Jameson's (129 × 129) regular mesh about the NACA0012 [180], the elliptic equation method has been applied for one subcritical ($M_\infty$=0.5 and AoA=$0^o$) and one transonic ($M_\infty$=0.8 and AoA=$0^o$) flow, for the same two functions as before, $CDp$ and $P_a$. The reader is refered to [122]§7 for precise discussion of the satisfactory function value improvements, but the final adapted meshes are presented in figure 2.13 and their features are discussed hereafter.
The adaptations required 5 to 9 steps depending on the case. The ability of the method to manage large node displacements, while keeping a certain regularity in the mesh, is first noted. Besides, the method is able manage the refinement of specific zones like supersonic area ($CD_p$-oriented transonic), vicinity of the wall ($Pa$-oriented), vicinity of leading and trailing edge ($CD_p$-oriented), zone upwind the trailing-edge ($Pa$-oriented, subcritical flow). This versatility is clearly higher than the one of the techniques of mesh parametrization used before [147, 144, 143]. The clear differences close to the wall between the meshes adapted for $CD_p$ and for $Pa$ is finally underlined.

The PhD thesis of Nguyen-Dinh [121] then presents interesting (unpublished) mesh adaptations for (RANS) flows, retaining components of drag as goal, checking (possible) improvements of the all classical functions of external aerodynamics when adapting the mesh for the calculation of one of them. I only extract here from the 2D part of his work two plots of a $CD$-oriented mesh-adaptation for the RAE2822 aerofoil ($\mathrm{M}_{\infty}$=0.725 and AoA=2.466$^o$, Re/m = $6.5\ 10^6$). A hierarchy of $C$-meshes from (257×65) to (4097×1025) leads limiting values for four outputs of interest $CLp$, $CD$, $CDp$ and $CDf$. A $\overline{\theta}$-based $CD$-oriented mesh adaptation is then performed for the (513×129) mesh. Its nodes are displaced in three steps using the elliptic equation method. The improvement in the output $CD$ is very satisfactory (Initial mesh 123.93 $10^{-4}$ ; adapted mesh 119.41 $10^{-4}$ ; limiting value 118,60 $10^{-4}$ ) with simultaneous improvement of $CLp$, $CDp$, and $CDf$ ([121]). The final mesh and the local $\overline{\theta}[CD]$ are presented in figure 2.14. It is noted that bringing closer the mesh lines [16], in the zones of high values of $\overline{\theta}[CD]$ has lowered the local criterion on the adapted mesh.



Subcritical test case. (a) Mesh adapted for $P_a$ and (b) mesh adapted for $CD_p$.



Transonic test case. (a) Mesh adapted for $P_a$ and (b) mesh adapted for $CD_p$.

Figure 2.13: NACA0012 inviscid flow. Final meshes obtained by elliptic-equation-$\overline{\theta}$-based. $Pa-$oriented (left) and $CDp-$oriented (right) adaptations

---

[16] essentially the $I$-mesh lines

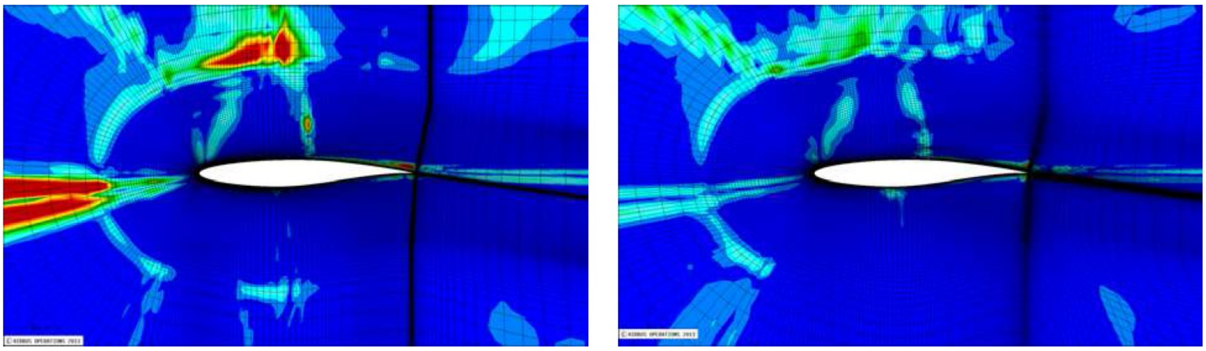


Figure 2.14: RAE2822 (RANS) flow. $\overline{\theta}[CD]$-criterion on initial mesh and $CD$-adapted elliptic-equation-$\overline{\theta}$-based mesh. ($513 \times 129$) meshes

### 2.4.5 3D RANS adaptation based on an elliptic pde.

The elliptic pde based method has also been assessed in 3D ([121] §4.1 to 4.3). The test case is the XRF1 Airbus configuration. Without specifically optimized geometrical preprocessing, it is considered that $\overline{\mathcal{P}(dJ/dX)}$ is too expensive to compute for structured multidomain meshes of several million points. This mean of $\mathcal{P}(dJ/dX)$ is replaced by a simpler formula that only involves the first neighbors. The corresponding local value of indicator is denoted $\tilde{\theta}_m$ (local value). Its mean over the mesh is denoted $\tilde{\theta}$.

The XRF1 Wing-Fuselage is retained for the mesh adaptation based on $\tilde{\theta}_m$. The flow conditions are $M_\infty$=0.83 and AoA=2.607$^o$, Re/m = 7.8 $10^6$. A hierarchy of five meshes with 3.2, 10, 13.5, 74 and 100M points is defined to obtain reference values for lift and drag (very fine grid values being considered as limiting values). These near-field forces values are gathered in table 2.2. The $CLp$-oriented adaptation of the medium mesh (13.5M nodes) is summarized here and the reader is refered to [121] §4.3 for more details. The adaptation steps reduce by 3.3% the error in $CL_p$ and all near-field functions exhibit improved values (see table 2.2). It is observed that $\theta[CL_p]$ is slightly increased (from 3.29 $10^{-12}$ to 3.38 $10^{-12}$) during the optimization and the same behavior is observed for $\tilde{\theta}[CL_p]$ (from 2.10 $10^{-12}$ to 2.28 $10^{-12}$). It is understood from these values that the 13.5 M mesh may probably not be significantly further improved for $CL_p$ evaluation. Finally the $\tilde{\theta}_m$ field is plotted is three vertical mesh planes for the initial and the adapted mesh (figure 2.15). It is observed that the zones of highest sensitivity have been decreased but near the wall for the first vertical cut.

A similar mesh-adaptation has been conducted for $CD_p$. It leads to a stronger improvement of goal value (12.8 % error reduction) but to a more complex discussion of the influence of the mesh-adaptation on other near-field functions.

| Mesh size | $CL/CL^{ref}$ | $CL_p/CL_p^{ref}$ | $CD/CD^{ref}$ | $CD_p/CD_p^{ref}$ | $CD_f/CD_f^{ref}$ |
|---|---|---|---|---|---|
| 100 M | 1. | 1. | 1. | 1. | 1. |
| 74 M | .99645 | .99666 | .99390 | 0.99608 | .98990 |
| 13.5 M (baseline) | .98476 | .98498 | 1.04371 | 1.05849 | 1.01656 |
| 13.5 M (adapted) | .98526 | .98529 | 1.03727 | 1.05008 | 1.01374 |
| 10 M | .98407 | .98429 | 1.04790 | 1.06562 | 1.01550 |
| 3.2 M | .96470 | .96491 | 1.19002 | 1.29034 | 1.00658 |

Table 2.2: XRF1 at $M_\infty$=0.83, AoA=2.607$^o$, Re/m = 7.8 $10^6$. $CL$, $CL_p$, $CD$, $CD_p$, $CD_f$ values for baseline meshes and medium adapted mesh.

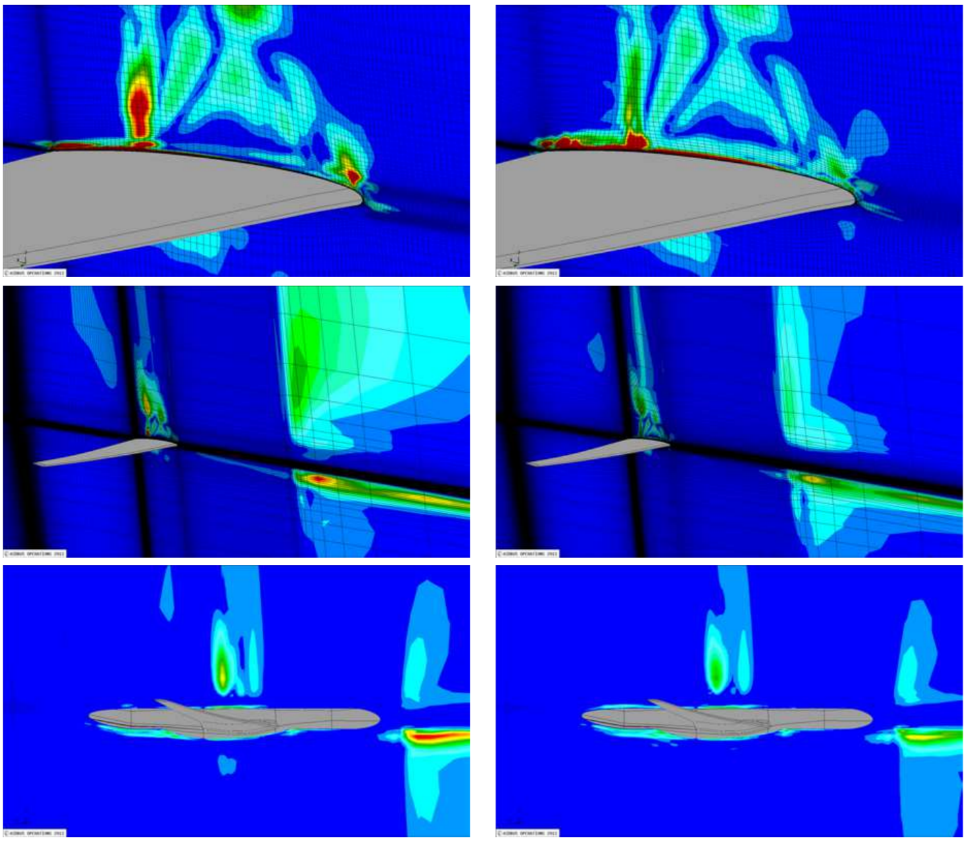


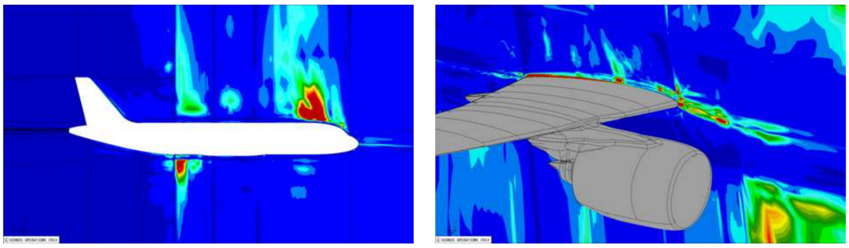
Figure 2.15: XRF1. Criterion $\tilde{\theta}[CDp]$. Left: initial mesh – Right: adapted mesh

### 2.4.6 Criterion examination for complex 3D (RANS) flows.

Finally, the last sections of [121] include the examination of the basic $\theta$ criterion on a "generic modern aircraft". The wing-body-pylon-nacelle-vertical-tail configuration is retained. The structured non-matching mesh is composed of 81 million points and 1394 mesh blocks involving non-matching joins between blocks. The $\theta$ fields of lift, drag and integrated wall-temperature are looked at [17]. The zones marked by high values of the $\theta$ criterion for the three functions exhibited significant differences. This is illustrated by figures 2.16, 2.17 and 2.18. These zones were found to be relevant by Airbus experts and

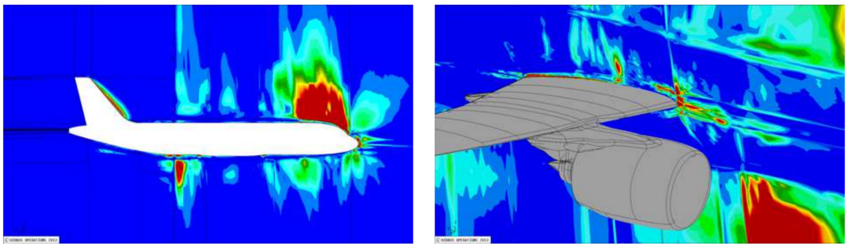
---

[17] whereas $\overline{\theta}$ and $\tilde{\theta}$ were not calculated for this configuration



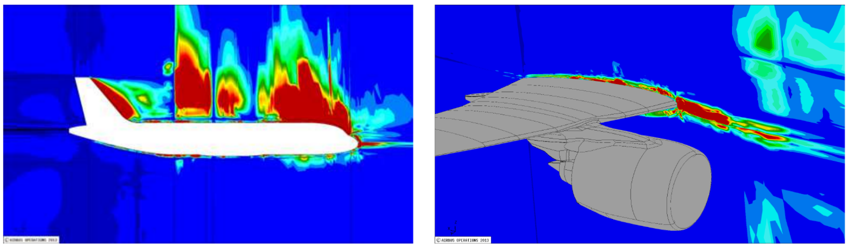
(a) (b)

Figure 2.16: Modern green aircraft. Criterion $\theta[CL]$. Left : symetry plane. Right : plane between the engine and the fuselage.



(a) (b)

Figure 2.17: Modern green aircraft. Criterion $\theta[CD]$. Left : symetry plane. Right : plane between the engine and the fuselage.



(a) (b)

Figure 2.18: Modern green aircraft. Criterion $\theta[Temp]$. Left : symetry plane. Right : plane between the engine and the fuselage.

Joël Brezillon recently confirmed that the local $\theta$ criterion is currently used by Airbus and subcontractors for non-matching structured mesh improvement by automatic addition of mesh planes in zones of high $\theta$ values [18].

## 2.5 Goal oriented adaption of unstructured meshes

The goal-oriented adaptation of unstructured meshes was started at ONERA in 2011 by Bourasseau in the framework of his PhD thesis [17]. He first extended the adjoint module of the $elsA$ code to unstructured meshes, for inviscid flows, under my supervision [18] then developed a goal-oriented mesh adaptation chain based on INRIA's MMG series of codes [39]. He eventually presented satisfactory mesh adaptations and goal convergence. As an illustration of Bourasseau's work, figure 2.19 is the plot the initial and fourth-adapted mesh for a 2D inviscid flow about the LS89 turbine blade. The classical subsonic inflow

---

[18] Todarello contributed, coding the direct and adjoint linearization of Roe-MUSCL scheme for unstructured meshes [170]

and outflow conditions have been used [19]. The function of interest, $Pa$, is the integral of total pressure at the outlet. The refinement criterion is $\theta[Pa]$. The final mesh exhibits strong refinement at the oblique shock-wave starting from trailing edge, the outflow and the acceleration zone close to leading edge which are precisely the regions where accuracy is expected to be needed for a satisfactory estimation of the output. We refer to [17] for more details.

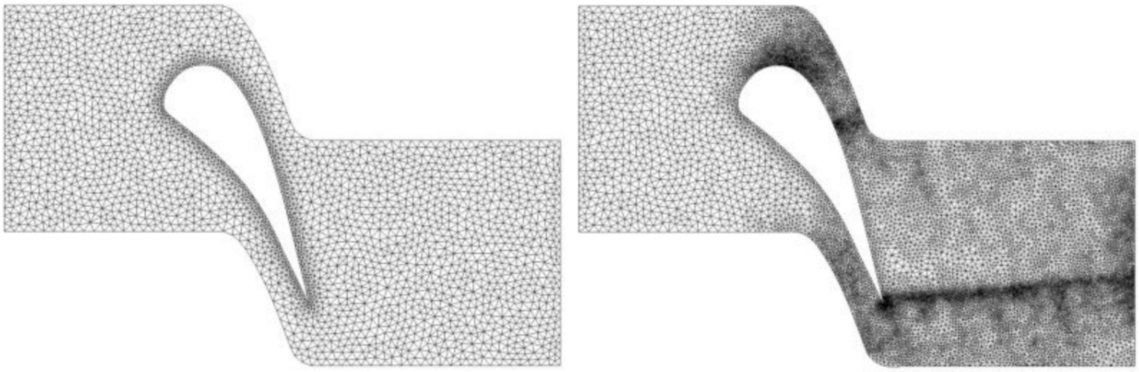


Figure 2.19: LS89 initial regular mesh (2705 nodes) and final $Pa$-oriented adapted mesh (11338 nodes)

Later on, based essentialy on extensions of the same adjoint [170] and adaptation code, mesh adaptations for external flows were performed by Vonck and me. The corresponding activity led to the most demonstrative results among those presented as (1) a complete comparison of the proposed method with Venditti and Darmofal's one [181, 182, 183] [20] was performed ; (2) accurate limting values could be estimated for the functions of interest allowing error calculations for the outputs. The corresponding research is presented in reference [171] where it was illustrated by mesh adaptations for three inviscid flows about the NACA0012 aerofoil at two transonic flow conditions ($M_\infty$=0.85, AoA=$2^0$) ($M_\infty$=0.95, AoA=$0^0$) and one supersonic flow condition ($M_\infty$=1.5, AoA=$1^0$). The derivation of the method is recalled hereafter ; its efficiency is illustrated by a lift-oriented mesh adaptation for the first transonic test case.

### 2.5.1 Validity of linear expansion of $J(X)$

The proposed refinement indicator relies on an upper bound of the first order change in $J$ when moving one node inside a circular vicinity included in the polygon defined by its neighbors. The validity of the first order Taylor expansion of $J$,

$$J(X + dX_i) \simeq J(X) + \frac{dJ}{dX_i} dX_i \ , \tag{2.18}$$

is hence first to be studied.

Six nodes are selected in a rather coarse unstructured mesh about the NACA0012 airfoil (see figure 2.20 left) for which we expect stronger non-linear behavior of $J(X)$ compared to finer meshes. For the transonic flow conditions ($M_\infty$=0.85, AoA=$0^0$ ), after the accuracy of $dCL_p/dX$ has been exhaustively

---

[19] Inflow: fixed total pressure and total temperature fixed flow direction parallel to machine axis. Outflow: fixed static pressure. its ratio w.r.t inlet total pressure being .7587

[20] which can be viewed as the reference for finite volume goal-oriented mesh adaptation

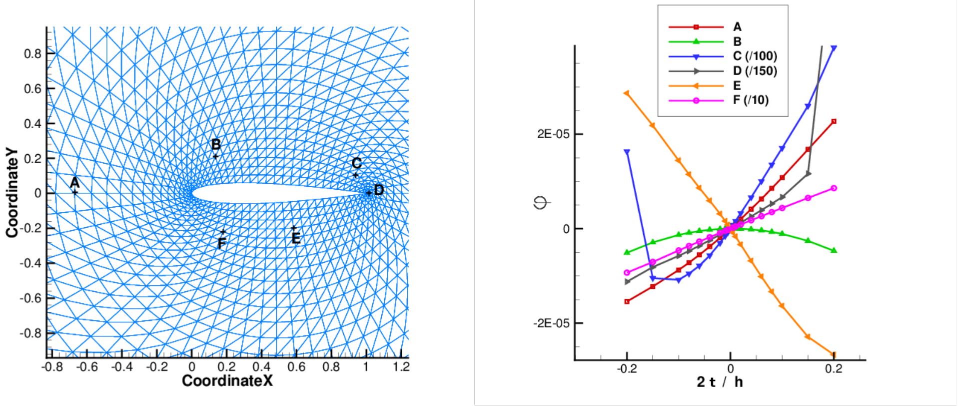


Figure 2.20: Left: location of points for check of total derivatives of outputs w.r.t. mesh coordinates. Right: evaluation of $CLp$ deviation, $\varphi$, w.r.t. nominal value at points A, B, C, D, E, F (scaled as indicated for points C, D and F)

checked, the domain of validity of the Taylor expansion (2.18) with respect to the coordinates of these six nodes is assessed. The difference term

$$\varphi(\delta X_m) = CLp(X + \delta X_m) - CLp(X),$$

$$\text{where} \quad \delta X_m = \frac{t}{0.5\ h_m}.n_m \qquad n_m = \frac{dCLp/dX_m}{||dCLp/dX_m||},$$

is calculated for the selected nodes and different values of the relative displacement $t$. (The Taylor expansion in $X_k$ is hence checked in the direction of the corresponding sensitivity $dCLp/dX_k$.) For each node, 18 shifted meshes are defined and the corresponding flow and lift are computed. The variation $\varphi$ is finally plotted as a function of the displacement relative to half the local meshes size, $0.5h_m$, in the direction $n_m$. The zone where the variation $\varphi$ can be considered as linear depends on the point but includes [-0.05 $h$,0.05 $h$] for most of the points – see figure (2.20 right).

### 2.5.2 Asymptotic behavior of $\mathcal{P}(dJ/dX)$

When dealing with structured meshes, theoretical efforts have been undergone to analyze the asymptotic behavior of $dJ/dX$ – see section §2.2 or references [121, 122]. When moving to unstructured meshes, the counterpart developments would have been much more tedious if even doable as (1) even a basic generic unstructured mesh is more geometrically complex than a structured Cartesian mesh ; (2) the coding of an inviscid second order scheme on an unstructured mesh involves more geometrical terms compared to its counterpart for structured meshes (Typically, gradients of primitive variables are needed for the calculation of MUSCL left and right states on unstructured grids but are not required on structured grids). A more practical approach has hence been adopted here.

The first step was to check on a hierarchy of embedded meshes [21] and for simple 2D inviscid flows, that

$$\mu[J] = \frac{1}{n_X} \sum_{m=1}^{n_X} ||\mathcal{P}(dJ/dX_i)||$$

is second order in the characteristic mesh size ($J$ stands for lift, $CLp$, or drag, $CDp$). More details can be be found in reference [171], in the discussion of figure 1.

---

[21] These meshes were derived from those of Vassberg and Jameson's study [180] by splitting every cell in two. See left column of figure 2.21

The second step was to discuss whether this global property reflects a local property. $ds_i$, denoting the area associated to the node of index $m$ in the classical construction of a dual mesh, $||\mathcal{P}(dCDp/dX_m)||/ds_m$ and $||\mathcal{P}(dCLp/dX_m)||/ds_m$ have been calculated and plotted for the hierarchy of meshes mentioned before. The pattern of isolines of these scalar fields is roughly the same as the mesh refines (see right column of figure 2.21 for drag). When an approximate asymptotic behavior of $||\mathcal{P}(dJ/dX)||$ will be needed to construct a mesh adaptation strategy, according to the observation above, $||\mathcal{P}(dJ/dX_m)||$ will be assumed to scale with the square of the local mesh size $h_m$ for 2D Eulerian flows and for the selected scheme.

### 2.5.3 Adaptation procedure

As discussed in section (2.2) the proposed goal oriented mesh adaptation method aims at regularizing the straightforward upper bound of the first order change that a displacement of a node can cause to the considered output. More explicitly, the retained criterion is

$$\theta_m = \left\| \mathcal{P}\left(\frac{dJ}{dX_m}\right) \right\| \frac{h_m}{2}. \tag{2.19}$$

whereby $h_m$ is defined as the distance of node $X_m$ to the closest neighboring node.
The corresponding global indicator is

$$\theta[J] = \frac{1}{n_X} \sum_{m=1}^{n_X} \left\| \mathcal{P}\left(\frac{dJ}{dX_m}\right) \right\| \frac{h_m}{2}.$$

The adaptation procedure is defined by using a threshold value, $T$, comparing the node criterion values $\theta_m$ to the threshold and refining the zones where it is larger. In order to get an approximation of the local change in criterion $\theta$ caused by a local change in characteristic mesh size $h$, it is necessary to make an assumption on the decay of $dJ/dX$ when $h$ is decreased: It is simply assumed that the asymptotic behavior observed for the family of regular meshes of the previous section is valid in the case of a local refinement – in other words, that $dJ/dX$ decays like $h^2$. The new local mesh size $h_m^{new}$, where the threshold is exceeded is then (as the mesh is adapted by refinement only),

$$h_m^{new} = h_m^{cur} \min\left( \left(\frac{T}{\theta_m}\right)^{1/3}, 1 \right). \tag{2.20}$$

In accordance to a classical reference on goal oriented mesh refinement [182], the threshold value for $\theta$ is set in such a way that not more than half the number of nodes are flagged for refinement. The remeshing is performed using MMG2D, from INRIA [39]. This tool uses the classical metric matrix formalism to define the goal shape of the cells in the vicinity of a node. In this case where isotropic mesh refinement is performed, the input provided to MMG2D is a simple diagonal 2x2 matrix with two equal coefficients equal to the inverse of the square of the desired local mesh size.

### 2.5.4 Lift-oriented mesh adaptation at ($M_\infty$=0.85, AoA=$2^0$). Comparison with reference method

The flow about the NACA0012 at this transonic condition involves a strong suction-side shock-wave, a pressure-side shock-wave and a sliding line behind the trailing edge. The iso-Mach number lines obtained on a fine regular grid are presented subsequently in figure 2.24 of section (2.6).
Limiting values and GCI confidence intervals for lift and drag result from series of calculations on fine structured meshes up to 4097×4097 [180]. A limiting value of 824.152 d.c. and a confidence interval of [823.839,826.963] d.c. based on the finest grid estimation ($CD_1 = 825.401 \times 10^{-4}$) are obtained.

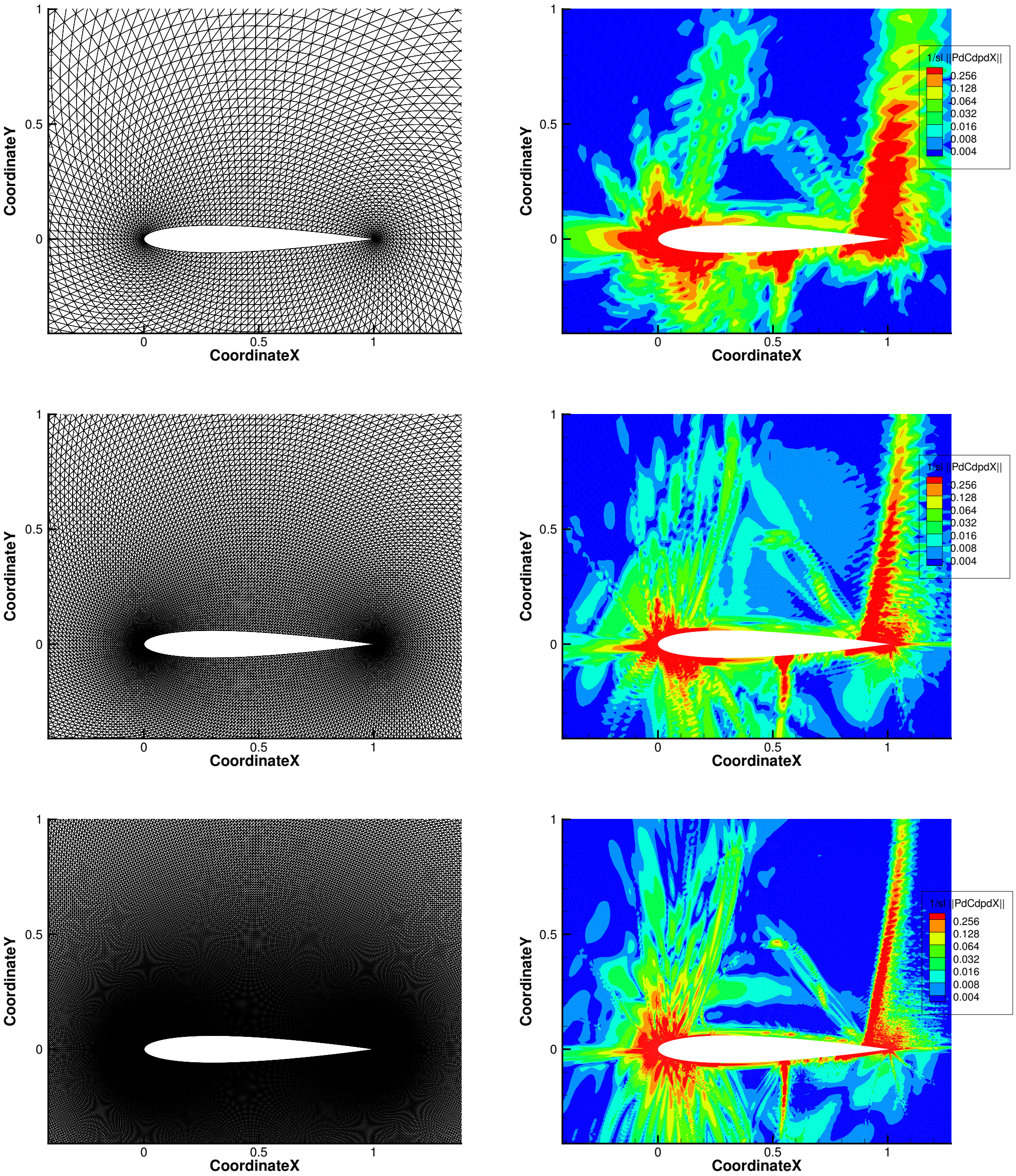


Figure 2.21: Meshes and corresponding iso-lines of $1/ds_i\|\mathcal{P}(dCDp/dX_i)\|$ about the NACA0012 (M$_\infty$=0.85 AoA = $2^0$)

Concerning lift, the corresponding quantities are 62.588 l.c. and interval [62.547,62.952] l.c. based on $CL_1 = 0.62750$ and two coarser grid values.

$CLp$-oriented mesh adaptation procedures are run with a 0.001 (0.1 l.c.) tolerance w.r.t. the rounded limiting value 62.59 l.c.. The threshold $T$ – equation (2.20) – of the proposed method and the error bound of the reference method are adjusted so that the size of the successive adapted meshes is about 3500, 5800, 8800 and 12800 nodes.

The lift value on the initial 1352-node mesh is $61.637 \times 10^{-2}$. It is lowered at the first adaptation step then increased towards the limiting value by the next three adaptation steps (see figure 2.22 for detailed convergence). Three adaptation steps are required for the $\theta$-indicator method until $CLp$ is included in the goal interval for lift. Three adaptation steps are also needed for Venditti and Darmofal's method with its correction until $CLp$ estimation is included in the interval [62.49,62.69] l.c. (respectively, five adaptation steps without this correction).

The adapted meshes at step five (fourth adapted meshes) are compared in figure 2.23 (reference method; 12973 nodes ; proposed method; 12502 nodes). They actually present very similar dense zones close to the upper side and lower side shock waves, upwind the profile and along a hat shape above the airfoil. The reason for this will be discussed in next section.

The plots of compared analysis of refinement criteria are the upper and bottom part of figure 2.23. They exhibit closely similar trends: the two methods initially select for refinement a large area upstream and above the profile. On the second to fifth steps, they alternatively detect a thin area upwind the profile, the upper-side shock-wave and the already mentioned hat shape above the airfoil. When looking closely to pictures 2.23, the only clear difference between the error estimator of the reference method and the sensitivity indicator of the proposed method is that the latter one is relatively stronger close to the wall and this leads to a higher mesh density in this zone.

Finally, the relevance of the projection operation involved in the criterion definition – equation (2.19) – is assessed. As expected from the experience gained with structured meshes, the indicator obtained without the projection operation (denoted no-projection $\theta$-indicator in the legend of figure 2.22) leads to over-refinement close to the wall and stall of $CLp$ value below the limiting value.

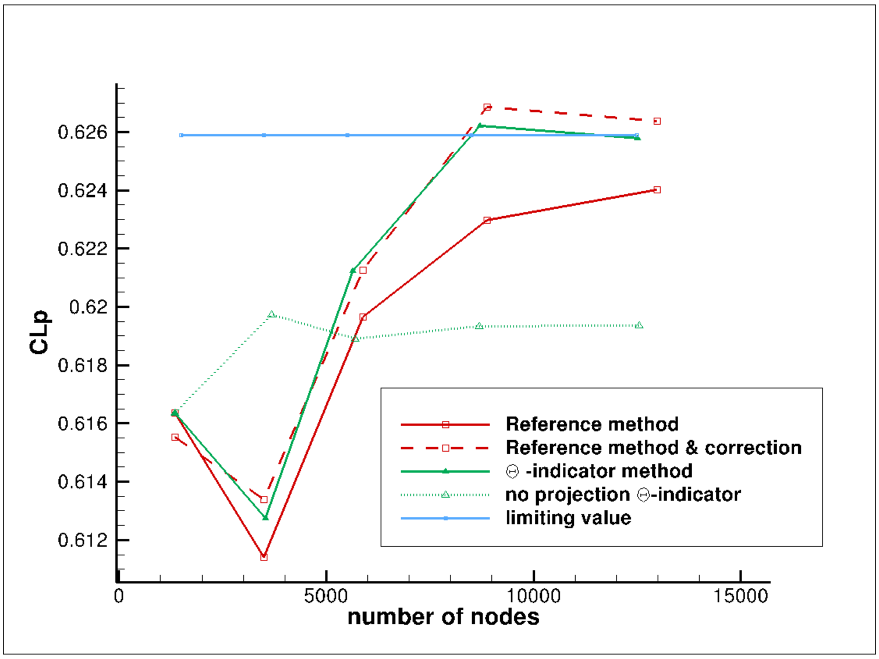


Figure 2.22: Convergence of $CL_p$ (right) for $\theta$-based and reference method ($M_\infty$ = 0.85 and AoA = $2^o$)

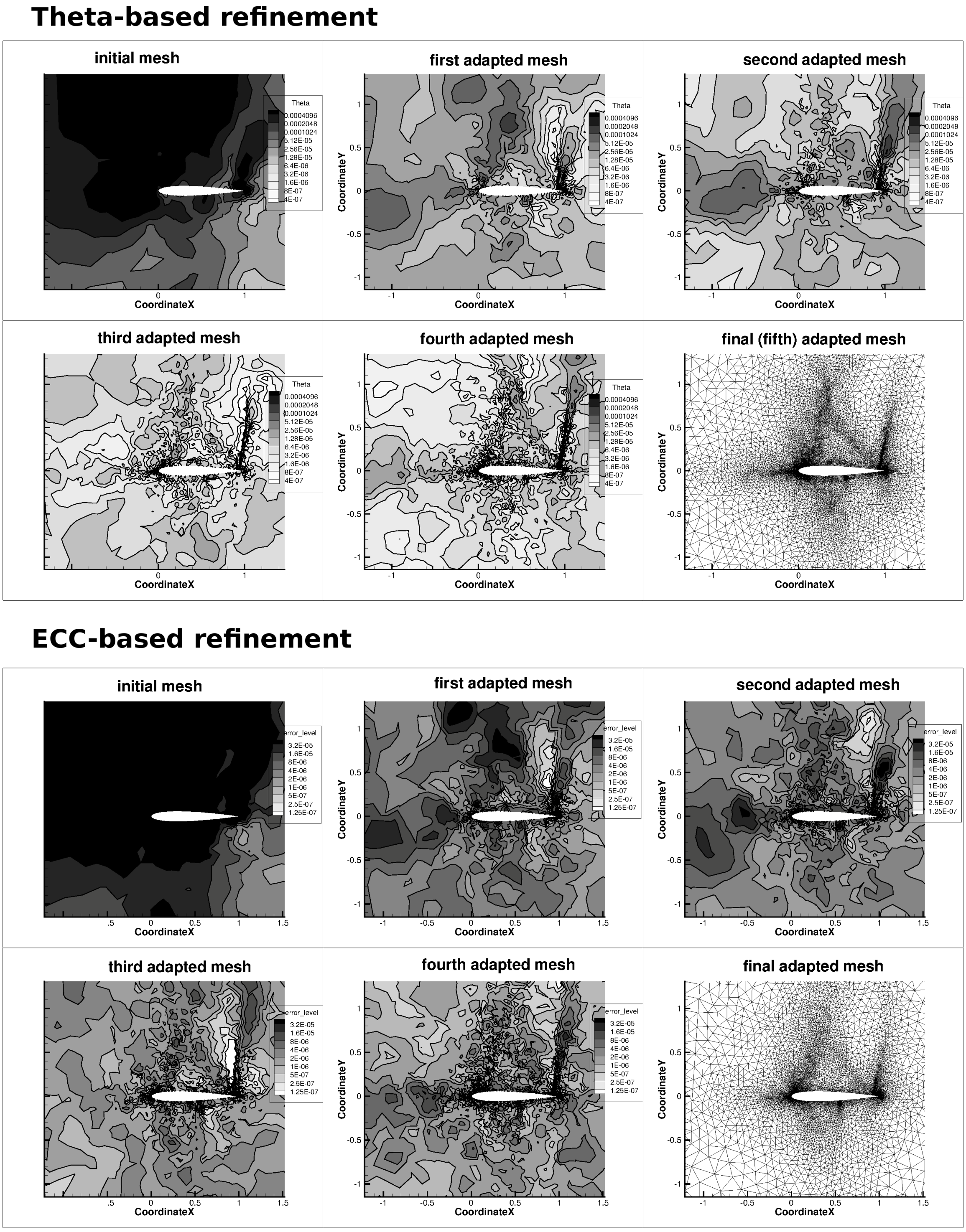


Figure 2.23: NACA0012 $M_\infty$=0.85, AoA=$2^o$ inviscid flow, $CLp$-oriented adaptation – Up: Fields of $\theta$ indicator at the successive steps. Down: Fields of absolute value of local contribution to error in computable correction of reference method at the successive steps.

### 2.5.5 Attempt to control the error in the output

As a path towards an heuristic error control, the global value of the indicator $\theta_{CL_p}$ was plotted as a function of the number of nodes and the error error in $CL_p$ (w.r.t. the limiting value calculated besides) has been plotted as a function of $\theta_{CL_p}$.

For this application, $\theta_{CL_p}$ scales with $1/n_X^{3/2}$ (see [171] fig 8(a)) $n_X$ being the total number of nodes. This is not very different from was is expected and observed on regular meshes. Other mesh adaptations have exhibited a decrease of $\theta$ as the inverse the square of $n_X$.
When looking at the error in $CL_p$ as a function of $\theta$, unfortunately, no simple correlation could be established so that no heuristic error control could be proposed.

### 2.5.6 Analysis of an adapted mesh and corresponding $\mathcal{P}(dJ/dX)$ field

As noted in the introduction of this chapter, goal-oriented adapted meshes are very different from those defined by feature-based criteria and are quite different from those that were defined by engineers experience before the introduction of goal-oriented mesh refinement. They actually gather nodes in zones that are not always easily connected with the flow and output functional of interest.
Désidéri and I contributed to the analysis of the map of the sensitive zones (for lift) for two inviscid 2D flows in [171] with an original method based on the characterization of the adjoint vector by perturbation of the residual (§1.1.5). This method is presented in the second subsection for a transonic flow and lift as selected goal. The corresponding flow, adjoint field and $dJ/dX$ fields are discussed in the first subsection.

The final lift-adapted meshes about the NACA0012 airfoil at ($M_\infty$,AoA) = (0.85, $2^o$) are presented in bottom-right part of the two sets of plots of figure 2.23. They exhibit refinement of shock waves, of the stagnation stream line and of large “circumflex accents” upwind both shock waves. These dense mesh zones are in fact the union of the strong gradient areas of the flowfield and zones of high values of the adjoint field. This can be easily checked looking at figure 2.24 that presents the iso-lines of Mach number and first component of lift-adjoint for the inviscid flow of interest (for the sake of simplicity all the plots of this section have been obtained on the regular (1025×1025) mesh of reference [180] with Roe-MUSCL scheme). These mesh-aspects are now classical ; they have been obtained not only with the classical method of Venditti and Darmofal [182] and the proposed method but also with Dwight’s method [47] for Jameson-Schmidt-Turkel scheme.

Besides figure 2.25 presents the iso-values of

$$\theta[CLp]_m = 0.5\ h_m\ ||\mathcal{P}(dCLp/dX_m)||$$

on the (1025×1025) regular mesh and hence shows the zones selected for refinement on this regular mesh by the $\theta$ method. The zones of high values of $h_m||\mathcal{P}(dJ/dX_m)||$ are definitely those listed above. In this specific case of a very regular mesh, they are the same as those of high $||\mathcal{P}(dCLp/dX_m)||$. The reason why $dCLp/dX$ “detects” strong gradients of flow and high values of adjoint stems from its definition, equation (1.9),

$$\frac{dJ_k}{dX} = \frac{\partial \mathrm{J}_k}{\partial X} + \Lambda_k^T \frac{\partial R}{\partial X}$$

and the structure of the lift-adjoint field which zones of high gradients correspond to narrow zones of high-values. Dealing with a near-field function, the field values of $dJ/dX$ are those of

$$\Lambda_k^T \frac{\partial R}{\partial X}$$

and it is easily understood that the $dJ/dX$ field gathers information of both direct and adjoint equations.
These observations do not fully explain the critical influence of the zones refined by all goal-oriented

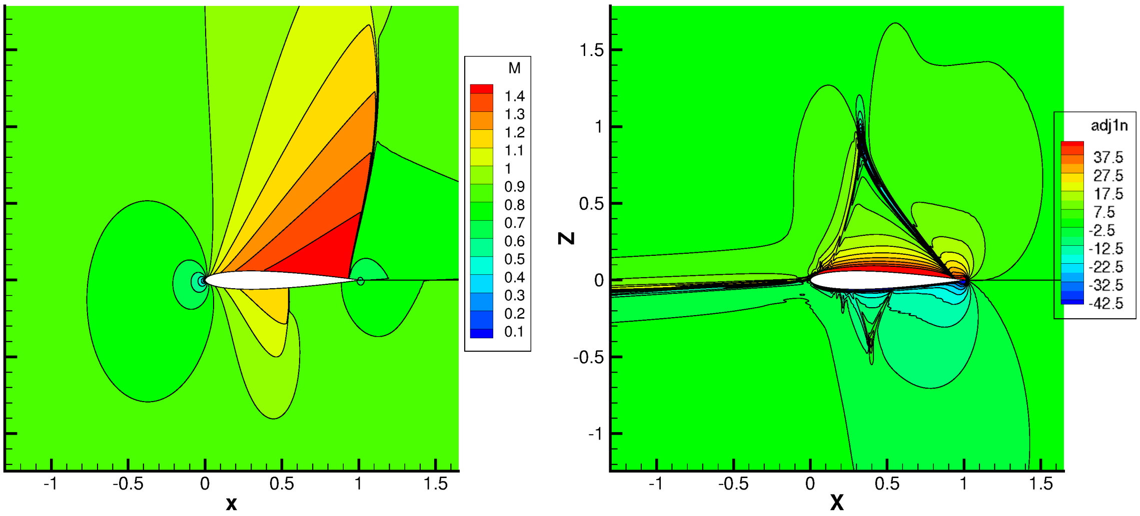


Figure 2.24: NACA0012 $M_\infty = 0.85$ and AoA = $2^o$ inviscid flow – iso-Mach number lines (left), iso-lines of first-component of lift adjoint vector (right)

methods for $CLp$ that are actually the same as those obtained for pressure drag. Of course accurate position and representation of shock waves is necessary for a satisfactory evaluation of lift or drag. The influence of the zone upwind the profile (that is the support of lift and near-field drag integration) is quite intuitive – it is necessary to convect a sound, non-dissipated flow up to the wall – and has been discussed by Giles and Pierce [60]. The downstream line of what we called "circumflex accent" upwind shocks has been well identified as a characteristic line (it is fully included in the supersonic zone) impinging the shock wave foot. As far as I know, the mechanism through which the flow in the upstream part of "circumflex accent" influences lift and drag has not been discussed ; however from a plot of a recent paper by Lozano [108], it appears that this line is also a characteristic line of the supersonic zone (the two lines of the circumflex accent belonging to the two different curve families)

### 2.5.7 Analysis of 2D inviscid adjoint fields

In order to gain insight in the influence mechanism on output that adjoint vector detects, let us first recall that this vector is the sensitivity of the function of interest (now specifically the lift coefficient $CLp$) to an infinitesimal perturbation of the residual (section §1.1.5). This observation led us to conduct a series of original tests in which the (density-related) first component of the residual of our CFD code was perturbed by a small amount ($\Delta R_1 = \pm 2 \times 10^{-4}$), and the resulting variation in the Mach number field was observed. These experiments were made at points 1-7 of Figure 2.26 which provides the isovalue contours of the corresponding first component of the adjoint vector $\lambda_1 = \partial CL_p / \partial R_1$

We first check that the sign of $\lambda_1$ given by this figure is indeed consistent with the variations in the Mach number field observed on figure 2.26 and analogous plots for the other five points omitted (for the sake of brevity). For points 1, 4, 6, 7, the upper-surface shock is pushed downstream thus enlarging the suction zone, and by this effect, increasing the lift. The effect on the lower-surface shock is opposite. Indeed, $\lambda_1$ is positive at these point. For points 2, 3 and 5, the upper-surface shock is pushed upstream and the lower-surface shock is shifted upstream. The main effect is caused by the upper-surface shock

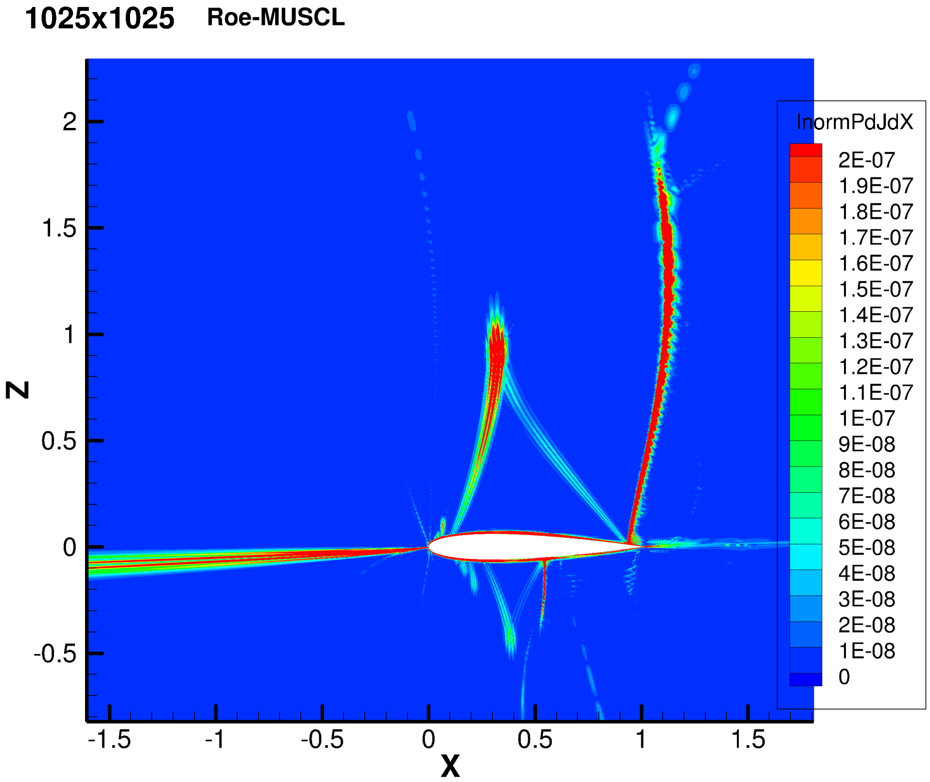


Figure 2.25: NACA0012 $M_\infty = 0.85$ and AoA $= 2^o$ inviscid flow – iso-lines of $\theta[CLp]_m = 0.5h_m\,||\mathcal{P}(dCLp/dX_m)||$

displacement and the lift is decreased.

Detailed plots of Mach number change due to residual perturbation is presented in Figures 2.26 for points 5 and 7. When now the perturbation is made at point 5, the Mach number locally altered along an the oblique characteristic line impinging the shockwave at the wall. Lastly, when the perturbation is made at point 7, the wave is reflected on the upper surface, and impinges the shock wave, creating a zone of strong influence.

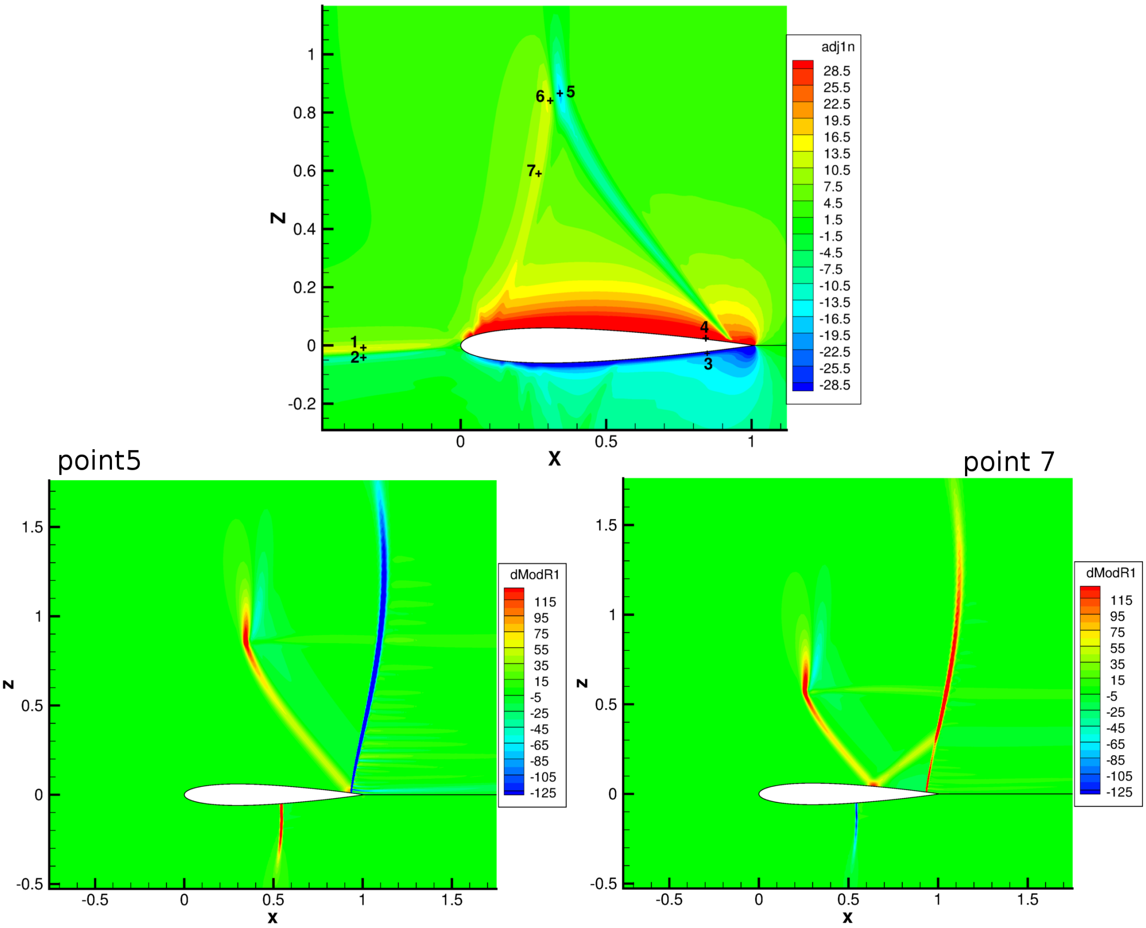


Figure 2.26: position of points for assessment of residual perturbation influence on CLp. Change of Mach number due to change of $R_1$ at point 5 (left) and 7 (right)

## 2.6 Maturity of (FV) goal-oriented mesh adaption methods. Perspective for proposed method

### 2.6.1 Technical maturity of classical FV goal-oriented mesh refinement methods

Five methods have been presented in section §2.1 from a theoretical point of view. Their technical maturity and actual usage for compressible flow simulations is discussed here before presenting the future developments of the proposed method. Of course, when dealing with possible applications, the robustness of the direct solver, the robustness of the adjoint solver, the remeshing process (possibly subsidiary tools for local mesh improvement), the projection of the state variables onto the newly generated meshes and the link between CAD and mesh are key-factors. This is the reason why I refer to CFD codes and (re)meshing tools in this sub-section.

• The adjoint-weighted-residual method, in the non-linear form introduced by Venditti and Darmofal [182, 183], has the assets to provide an error estimator for the goal and to be applicable to all FV schemes. Its drawback is to require a finer-than-current grid. (If the discrete adjoint is not perfect, a second drawback is that the correction term is most often wrong – see contribution of Tourrette in [172]). The method has raised a deep interest in the aeronautical CFD community and has been used not only with unstructured meshes as in the original reference but also with structured [94, 8, 141] and embedded-boundary Cartesian meshes [118, 119]. It is important to note that it has been fully developed in at least two industrial codes:
– Park and coworkers from the NASA Langley have implemented the method in the FUN3D suite of codes (the FUN3D code being a finite volume Navier-Stokes solver using dual mesh and flow-variables stored at the nodes of the original mesh). They have presented 3D Euler applications for sonic boom prediction [95, 89] and also 3D (RANS)&(SA) applications including the wing-body DLR-F6 DPW2 test case [128, 95]. The refinement of the current mesh is detailed in [128] for tetrahedral meshes (8 small tetrahedras are included in one tetrahedra of the original mesh) as well as the linear and higher order reconstruction (based on least-square fit of a quadratic function) and the combination of the goal oriented estimator with the Hessian of a flow variable [128] as in reference [183]. In 2008, Park proposed a single grid goal error estimate ([129] equation (4.22)) that mimics the original one but only uses higher and lower order reconstruction of discrete flow and adjoint on the current grid ; when this formula is used, no correction (and hence no error estimate) is available anymore. In recent publications, both the classical finer embbeded grid and single grid error indicators are used. They are actually compared in [131] where they are found to provide equivalently good results for the complex HighLiftPW configuration.
– In a recent publication, the adjoint-weighted-residual method was also demonstrated for 3D complex flows with HYDRA, the in-house CFD code of Rolls-Royce (HYDRA being a finite-volume edge-based / dual control volumes code whose discrete adjoint module has been derived by automatic differentiation) [4]. Actually, the goal-oriented error-estimation framework is not the focus of the article and no mesh-adaptation is carried out. The capability allows to select the most efficient structured blocking-meshing procedure for complex internal and external flows, in the sense that the flows computed on the corresponding meshes exhibit the lowest error for the functions of interest.
The interested reader is also refered to [152] that presents the coding of the method and mesh adaptation for 3D Euler flows using the DLR TAU code.

• In the method defined in 2003 by Venditti and Darmofal for anisotropic adaptation of viscous flows, the tightest element size was defined by the goal-oriented error analysis and the Hessian of Mach number was used to equidistribute the interpolation error for the flow variables. In comparison, the more recent analysis by Yano and Darmofal [189] is “strictly” goal-oriented as the search for changes in the local anisotropy are based on corresponding error reductions in the goal. It has also the advantage not to require a global finer grid [22] but the drawback not to provide an error estimator for the output functional of interest. The new approach has been applied to elliptic interface problems and conjugate heat transfer problems [77]. There does not seem to be many applications in aerodynamics up to now apart from [77] where the method is applied with Discontinuous Galerkin method to 2D turbulent flows.

• Dwight’s method has the assets to deal with the current grid only and to provide an error estimator. Its drawback is that Jameson Schmidt Turkel scheme [85] is the mandatory discretisation of inviscid fluxes. Although it is well-known, it seems only Albensoeder coded it [2] after it has been defined and assessed by Dwight for 2D Euler flows with the TAU code of DLR [46, 47].
Extension to (RANS) flow would clearly require a new discussion of the error attribuable to numerical, explicitely added, dissipation (counterpart of [47] §4 for Euler flows). This discussion would of course involve the discrete convective flux of turbulent variables whose dissipative part should probably be

---

[22] of course, with a specific (and surely tedious) coding, the correction and error of correction of [182, 183] could be computed without manipulating and storing at same time all the elements of the finer grid

cancelable monitoring a multiplicative parameter. Moreover, in many industrial codes, it is considered acceptable to discretize the convective flux of turbulent variables at first order which would probably make the discussion more complex and intricate with robustness issues.

• Fidkowski and Roe's entropy adjoint technique provides a free adjoint for adjoint-weighted-residual type methods. It naturally provides an error estimate for the goal and requires a finer than current grid [23]. Its drawback is to be limited to the entropy flux and, under specific assumptions, to Owatitsch drag. It has been demonstrated for Euler, laminar and turbulent flows about airfoils and also for an inviscid 3D flow about a wing [52, 50]. Although the articles describing the method are frequently quoted, it does not seem that it has been coded in other codes than the one(s) of Michigan University.

• The method developed by Alauzet, Dervieux, Loseille and Belme is based on an a priori error bound for the goal (involving linear approximation error) and a global minimization of that bound in a space of continuous meshes. It is hence very different from the previous ones. In particular, it is naturally anistropic and global. It does not require a finer grid but is limited to the numerical schemes for which the a priori error analysis leads to the suitable expression for the error-bound.
The theory has been published for Euler [104] and laminar flows [10], but also developed by Frazza and his supervisors for (RANS)&(SA) flows [55]. Complex mesh adaptations have been carried out and presented for Euler flows (about a F15 fighter and a Falcon business jet, in particular) [104]. Also a series of successfull mesh adaptations for 2D laminar flows is described in reference [10] [24]. Besides, Frazza has presented in his PhD thesis several pre-industrial and industrial applications (ONERA M6 wing, Rotor 37, High-lift CRM).
The goal-oriented mesh-adaptation chain of INRIA Gamma3 team is based on the mixed finite-volume finite-element solver Wolf ([55] §1) and the iterative mesh generator feflo.a. These tools are clearly able to deal with industrial configurations and [55] mentions support by SafranTech and Boeing Commercial Aircrafts (and feflo.a is also used at ONERA for research activities with CEDRE code). I am not aware of other laboratories having further developed the method.

### 2.6.2 Perspectives for $dJ/dX$ based method

The $dJ/dX$-based goal-oriented mesh refinement-method is free if the $dJ/dX$ vector field is calculated any-how (typically for shape optimization) and used without spatial mean ($\theta$ criterion). Otherwise, it requires an adjoint solve as all the methods discussed above but the one of Fidkowski and Roe. If a spatial mean of $dJ/dX$ is used ($\overline{\theta}$ criterion), the cost of this operation can be significant. The method has the advantages not to require a finer grid, not to be limited to a single output, not to be limited to a single scheme. It has the drawback not to provide an error estimator for the goal.

The capabilities of $dJ/dX$-based method for goal-oriented mesh adaptation have been extensively presented and discussed for structured meshes in the previous sections. Of course, unstructured meshes provide the most suitable framework for mesh adaptation and the 2D activity of isotropic refinement for 2D Euler flows should be followed by extension to 3D, (RANS) and, if possible, anistropic refinement. Although satisfactory, the 2D activity was not immediately followed by an effort towards 3D and then (RANS) flows due to the lack of adjoint code for these types of flow. Fortunately, a recent line of $elsA$ versions called $elsA$ *remotorisé* [25] benefits from this capability and the extension of $dJ/dX$-based method is currently being planned.

The first foreseen step is the detailed examination of the type of meshes that are compliant with the

---

[23] the proposed method summarized in [50] §VID not only involves a residual evaluation on the fine grid but a few relaxation iterations for direct or adjoint equations

[24] this paper presents in section §8.5 a 3D turbulent test case, *3D tubulent shock boundary layer interaction*, but it is not explicitely said how turbulence is taken into account

[25] unfortunately, there is today no public document about $elsA$ *remotorisé*

available numerical schemes. It is noticeable that authors working with well-known large codes have significantly different judgements on this topic. In [131], we read *the discretization of FUN3D is more accurate for semi-structured right-angle elements in the boundary layer*, whereas in [55] on the contrary, it is indicated and illustrated that the *finite-volume flow solver Wolf [performs satisfactory] RANS computations on fully unstructured meshes.* Following Park et al., the case where semi-structured stacks of prismatic elements near wall-bounded shear layers are beneficial to the solver accuracy is today the most frequent [130]. The constraint put on meshes for standard (RANS) simulation should also be applied to the new meshes generated during the adaptation process.

A second point to consider is the type of (RANS) mesh adaptation that is to be performed : full domain adaptation, adaptation with fixed mesh in the boundary layer [131], fixed number of structured layers in the boundary but possibly more elements [90], fixed wall mesh but adaptation of the number of mesh planes in the boundary layers (references in [31]) The projection operator was applied up to now to $dJ/dX$ to remove degrees of freedom that are not associated to mesh adaptation but to shape optimization. If some features of the mesh are fixed (like number of planes parallel to the wall) the operator could also be used to remove the components of $dJ/dX$ that are not to be exploited (like components normal to those planes). This of course, is to be tested and assessed.

Preserving geometric features of an initial anisotric mesh is a basic approach to produce anisotropic adapted meshes, as discussed just above for the boundary layer region. More generally, considering all the fluid domain, anisotropic goal-oriented mesh refinement methods are known to be more efficient than their isotropic counterpart whenever sharp flow structures influence the goal value. When considering only a local node vector $\mathcal{P}(dJ/dX_m)$ and taking its norm, all anisotropic information is lost. Hence a deeper (in pratice, spatially more global) extraction of information from $dJ/dX$ vector field should be considered to move towards general anisotropic mesh adaptation.

# Chapter 3

# Conclusion and perspectives

Discrete gradient computation started at ONERA beginning of 2002 in close cooperation with AIRBUS-F CFD team. The primal goal was then to provide an efficient means to calculate derivatives of forces and moments w.r.t. numerous design parameters for quick local optimization by descent methods. This target was progressively reached as illustrated by the optimization of the AVECA flying wing configuration by Méheut, Arntz and Carrier with more than a hundred design parameters – see references [117, 116] and also figure 3.2 right. These results demonstrate the current maturity of $elsA$ discrete adjoint module [145] and ONERA's techniques for parametrization, mesh deformation and advanced post-processing as well.

At the same time, ONERA participated in various groups and sessions devoted to adjoint method and local optimization (RTO-AVT-167 group, Dagstuhl 2014 workshop...), and contributed to lecture notes [14] and a bibliographic review article [139]. Significant effort was devoted to the adaptation of backward-Euler implicit stages [40, 138] to the solution of discrete adjoint equation. The $elsA$ adjoint solver was extended from fixed to rotating wings and one of the first (RANS) adjoint-based optimization of a hovering rotor could be performed [42]. Besides, efforts were dedicated to 2D local optimization based on the so-called shape gradient ; the corresponding original ideas proposed by Bompard and his supervisors have been summarized in section §1.3 of this document.

The ajoint module of the $elsA$ code is currently further improved: a time-spectral adjoint is under development ; GMRES with advanced restart is implemented ; the adjoint-mesh mode is extended to the relative frame-absolute velocity formulation. Besides, from an application point of view, it is foreseen that the module will be used in a near future, not only for shape optimization but also for meta-model based uncertainty quantification.

Concerning more fundamental issues about adjoint, the recent articles of Lozano about dual consistency of finite-volume discrete adjoint [107] and singular & discontinuous solutions of the adjoint Euler equations [109] have raised interest at ONERA. A research activity is ongoing about consistency of (extremely fine) discrete adjoint fields with the continuous equation, boundary conditions and jump conditions for adjoint derivatives at shock-waves [109]. As an illustration of this activity, the plot of an adjoint field with a strong normal-derivative discontinuity at shock is presented below (NACA0012 $M_\infty$=0.85, AoA = $2^o$).

In 2005 roughly speaking, as the maturity of the discrete adjoint codes of the large research centers and aircraft companies was growing, it appeared that the sensitivities of the volume-mesh with respect to design parameters would soon be the memory bottle-neck of adjoint-based gradient calculation for complex configurations. The necessity appeared to avoid finite differences for the geometrical sensitivity of the residual w.r.t. parameters. The residual was then differentiated w.r.t. mesh coordinates and all con-

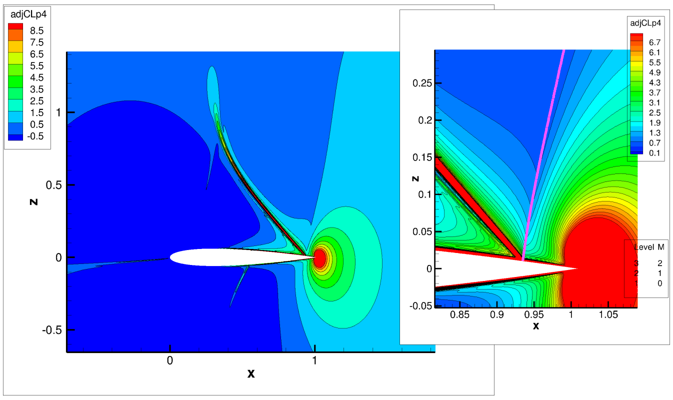


Figure 3.1: NACA0012 $M_\infty$=0.85 AoA=$2^o$ inviscid flow – Left: component of lift adjoint associated to z-axis momentum equation. Right: zoom at the suction-side shock-foot and location of shock (pink line)

tributions to the total derivative of the output(s) of interest w.r.t. volume mesh coordinates were gathered [1]. Nielsen and Park had not only a memory bootleneck in FUN3D adjoint-based optimization chain, but also a CPU-time burden as their volume-mesh sensitivities were the outputs of a linear-elasticity equation [2]. They proposed an elegant adjoint-mesh sensitivity method to circumvent this difficulty and increased the interest of the CFD community in the shape-gradient that is the output of their calculations [127].

After the total derivative of the functions of interest w.r.t volume-mesh nodes coordinate, $dJ_k/dX$, had become a possible output of the $elsA$ adjoint code, Destarac and I examined closely some plots of this newly available vector fields. The left part of figure 3.2 is one of these plots. It presents a NACA64A212 C-mesh with the wave drag integration contour defined by FFD72 [35] and the iso-lines of $dCDw/dz$ for the flow conditions ($M_\infty$=0.71, AoA=$2.5^o$) (inviscid flow). Zones of high sensitivity of $CDw$ clearly appear far from its contour raising the question whether the corresponding areas should be refined (leading edge, trailing edge and vicinity of wall).

A very significant research effort was then started with the aim of defining a $dJ/dX$-based goal-oriented refinement indicator and if possible, an error estimator as well. Considering existing methods for finite-volume CFD, the proposed one should not require a finer grid and not be specific to a single scheme or a single output functional [182, 183, 47, 52, 50].

A first effort [143] lead to the introduction of the upper-bound of the first-order variation of the output $J$ when performing an admissible displacement of one node, $\theta[J]_m$, and its mean value over all nodes, $\theta[J]$, where each vertex moves inside a circular neighborhood. These quantities are relevant for $J$-oriented mesh adaption as they are essentially decreasing when moving from unsuitable to suitable meshes for the calculation of $J$, which is not the case of $||dJ/dX_m||$ (local move) or the corresponding mean over the volume-mesh (global move). Besides, a projection removing all components of $dJ/dX$ altering the solid shape or the function support was also defined then and its relevance has been assessed thereafter in several mesh-adaptations ([171], fig 2.22 of this document).

For ordinary outputs, configurations, flow conditions and mesh densities, the $dJ/dX$ vector fields are actually rather irregular. When working with structured meshes, this raises the question of how to use a $dJ/dX$-based criterion without altering the regularity of the initial structured mesh. A first point in this respect is to avoid any mesh change somehow directly guided by $\mathcal{P}(dJ/dX)$, even if $J$ is monotonically altered by numerical dissipation, the reason being that all aerodynamic outputs are very multimodal

---

[1] Namely, avoid using equation (1.8), compute $\partial R/\partial X$, compute $dJ/dX$ according to equation (1.9)

[2] Namely, equation (1.13)

functions of the free volume-mesh node coordinates and that the gradient is short-sighted in this hill landscape. Figure 6 in [147] is an example of a bad mesh produced by direct $\mathcal{P}(dJ/dX)$-guidance and even more serious examples exist.

Fortunately, several methods may address this issue. The use of parametrized structured mesh-families, the definition of a spatial mean $\overline{\theta}$ associated to a length-scale, the solution of an elliptic structured-mesh deformation equation were satisfactory solutions for the definition of structured goal-oriented adapted meshes [144, 122, 121]. The global criterion $\theta[J]$ was proposed besides to assess the quality of existing meshes for the calculation of $J$. The local criterion $\theta[J]_m$ is still currently used by AIRBUS in the process of (RANS) structured mesh construction ([121]§4.4,[18]).

When dealing with adaptation of unstructured meshes by node addition, the irregularities of $dJ/dX_m$ or $\theta[J]_m$ fields are not such a serious issue. This was actually confirmed by successful 2D Euler mesh adaptations ([171]) which produced the expected refinement for classical 2D Euler flows and standard functions. Besides, the convergence speed towards the limiting value was found to be equivalent to the one of the error-based most used method [182, 183] that requires a finer-than-current auxiliary grid.

The $dJ/dX$ based method should be improved with respect to the two issues that were not solved up to now: (1) define a criteria to distiguish meshes for which nodes displacement can improve the $J$ evaluation and lower the global $\theta[J]$ criterion and meshes for which this is not possible anymore. Most probably, statistical analysis of the local $\theta[J]_m$ values would here be involved ; (2) associate a heuristic error estimator to the adaptation process.

Besides, the recent $elsA$ *remotorisé* versions will allow the extension of the method to 2D and 3D (RANS) for unstructured meshes along the lines discussed in §II.6 : determination of the type of suitable adapted meshes, adaptation of the criterion according to constraints imposed to the boundary-layer mesh and, if possible, extraction of relevant anisotropic information from $\mathcal{P}(dJ/dX)$ to move towards anisotropic refinement.

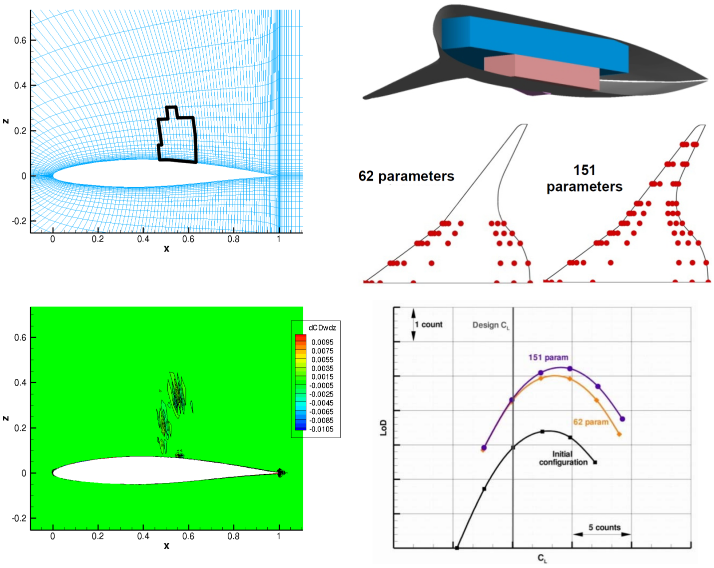


Figure 3.2: Left: NACA64A212 inviscid flow. M$\infty = 0.71$, AoA=2.5$^o$. Integration contour for $CDw$, iso-$dCDw/dz$ lines – Right : AVECA flying-wing configuration. Volumic constraints, design parameters, summary of multi-point optimization.

# Scientific and technical appendix

## Annex 1 – Continuous adjoint method

For the sake of completeness of the presentation adjoint method, this annex is devoted to a short presentation of continuous adjoint (although the method has not been implemented in the ONERA codes).
In this approach, the adjoint of the continuous governing equations with respect to a given objective function is derived, before being discretized. The first appearance was due to Pironneau [151], with Jameson providing the first treatment for compressible flows [78].
It is no longer possible to present the theory independently of the particular equations considered, therefore we first consider the 2d Euler equations in body-fitted coordinates. The following subsections are an extended presentation derived from an article by Giles and Pierce ([62, 63] section 3.5).

### 2d Euler equations in body-fitted coordinates

It is assumed that the problem in physical space with a body-fitted structured grid, can be transformed into computational coordinates $(\xi, \eta)$, see Figure 3.3, in such a way that the transformation of $D_{xy}$ to $D_{\xi\eta}$ is direct, that $D_{\xi\eta}$ is a rectangular domain $[\xi_{\min}, \xi_{\max}] \times [\eta_{\min}, \eta_{\max}]$ and that $\xi_{\min}$ corresponds to the surfaceof an airfoil. Note that, when performing shape optimization, whereas the coordinate change operator depends on $\alpha$, $D_{\xi\eta}$ itself does not.

Let $H$ and $K$ be the determinants of the Jacobian of the coordinate transformations

$$H(x,y) = \det \begin{pmatrix} \frac{\partial \xi}{\partial x} & \frac{\partial \xi}{\partial y} \\ \frac{\partial \eta}{\partial x} & \frac{\partial \eta}{\partial y} \end{pmatrix} \qquad K(\xi,\eta) = \det \begin{pmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} \end{pmatrix}$$

representing the change in size of a volume element under the transformation. At cooresponding points of $D_{xy}$ to $D_{\xi\eta}$, of course the two numbers are inverse one of each other. The 2D Euler equation in the

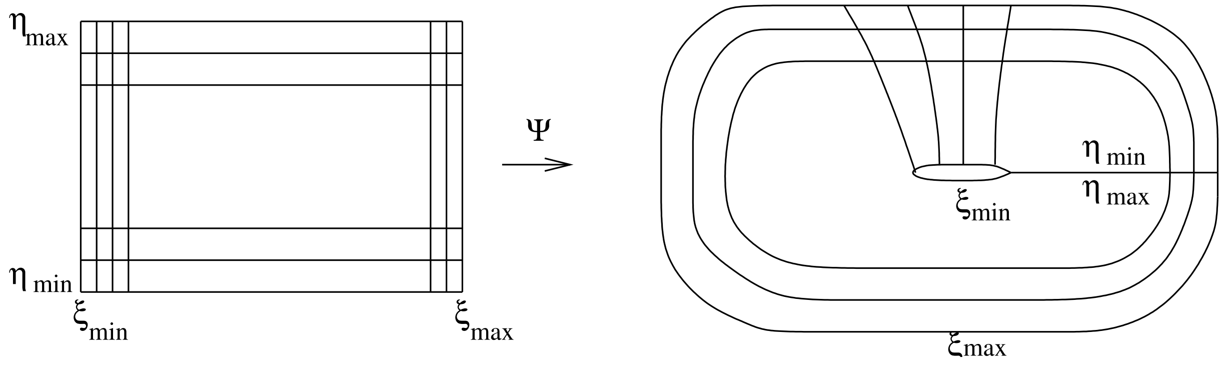


Figure 3.3: Body-fitted to computational coordinate domain transformation

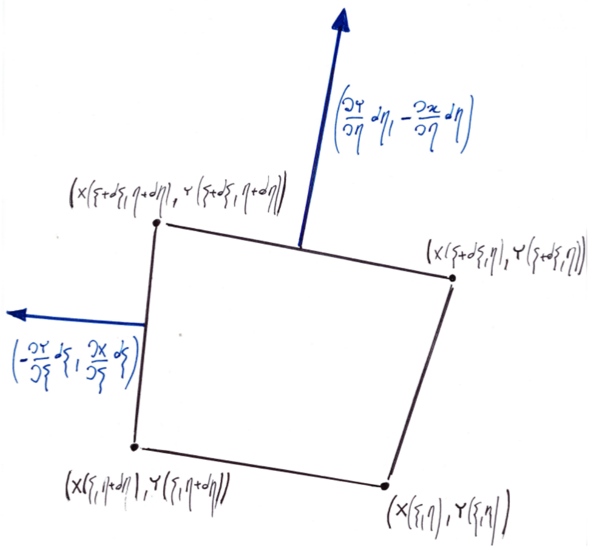


Figure 3.4: Link between generalised coordinates and Finite Volume CFD – the surface of the elementary quadrangle is $K(\xi, \eta) d\xi d\eta$

Cartesian $(x, y)$ coordinates read

$$\frac{\partial w}{\partial t} + \frac{\partial \mathcal{F}_x(w)}{\partial x} + \frac{\partial \mathcal{F}_y(w)}{\partial y} = 0$$

where

$$w = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \end{pmatrix} \quad \mathcal{F}_x(w) = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho u v \\ \rho u H \end{pmatrix} \quad \mathcal{F}_y(w) = \begin{pmatrix} \rho v \\ \rho v u \\ \rho v^2 + p \\ \rho V H \end{pmatrix}.$$

Then the Euler equations in the computational coordinates are

$$\frac{\partial W}{\partial t} + \frac{\partial F(W)}{\partial \xi} + \frac{\partial G(W)}{\partial \eta} = 0, \tag{3.1}$$

where

$$W = Kw \qquad F(W) = \frac{\partial y}{\partial \eta} \mathcal{F}_x(w) - \frac{\partial x}{\partial \eta} \mathcal{F}_y(w) \qquad G(W) = -\frac{\partial y}{\partial \xi} \mathcal{F}_x(w) + \frac{\partial x}{\partial \xi} \mathcal{F}_z(w).$$

The link of this formulation with finite volume structured CFD is well known. It is illustrated by figure 3.4 where the area and normal vectors of an elementary domain $[\xi, \xi + d\xi][\eta, \eta + d\eta]$ are recalled. Using the link between two systems of coordinates

$$\begin{pmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} \end{pmatrix} = \begin{pmatrix} \frac{\partial \xi}{\partial x} & \frac{\partial \xi}{\partial y} \\ \frac{\partial \eta}{\partial x} & \frac{\partial \eta}{\partial y} \end{pmatrix}^{-1} = \frac{1}{H(x,y)} \begin{pmatrix} \frac{\partial \eta}{\partial y} & -\frac{\partial \xi}{\partial y} \\ -\frac{\partial \eta}{\partial x} & \frac{\partial \xi}{\partial x} \end{pmatrix} = K(\xi, \eta) \begin{pmatrix} \frac{\partial \eta}{\partial y} & -\frac{\partial \xi}{\partial y} \\ -\frac{\partial \eta}{\partial x} & \frac{\partial \xi}{\partial x} \end{pmatrix}$$

the 2D Euler fluxes in generalized coordinates may be rewritten in their most classical form

$$F(W) = K \begin{pmatrix} \rho U \\ \rho U u + p \frac{\partial \xi}{\partial x} \\ \rho U v + p \frac{\partial \xi}{\partial y} \\ \rho U H \end{pmatrix} \quad G(W) = K \begin{pmatrix} \rho V \\ \rho V u + p \frac{\partial \eta}{\partial x} \\ \rho V v + p \frac{\partial \eta}{\partial y} \\ \rho V H \end{pmatrix}.$$

with

$$U = \frac{\partial \xi}{\partial x} u + \frac{\partial \xi}{\partial y} v \qquad V = \frac{\partial \eta}{\partial x} u + \frac{\partial \eta}{\partial y} v$$

In the specific case of the 2D profile (figure 3.3), the slip-wall boundary condition is $U = 0$ on $\xi = \xi_{\min}$, and a farfield condition is applied to the $\xi_{\max}$ boundary.

### Continuous adjoint equations approach for this set of equations

In the specific case of the 2D profile (figure 3.3), the slip-wall boundary condition is $U = 0$ on $\xi = \xi_{\min}$, and a farfield condition is applied to the $\xi_{\max}$ boundary. The objective function formulated in the new coordinate system is

$$\mathcal{J}(\alpha) = \int_{\xi_{\min}} J_1(w) \frac{\partial y}{\partial \eta} \,\mathrm{d}\eta + \int_{D_{\xi\eta}} J_2(w) K(\xi, \eta) \,\mathrm{d}\xi \,\mathrm{d}\eta, \tag{3.2}$$

where the domain of integration is now independent of $\alpha$. $\mathcal{J}$ corresponds in body-fitted coordinates to the integral over the profile of the $x$ component of a vector field (typically the force on the profile) plus a volume integral. The continuous adjoint equations can now be derived as follows: first write the first variation of the flow equations with respect to the design parameter $\alpha$. Referring to (3.1) there are two distinct types of variation: the flux terms $f(w)$ and $g(w)$ vary with $\alpha$, because the flow changes in the transformed coordinate space when the shape changes, and independently all metric terms also depend on $\alpha$:

$$f(w) \to f(w) + \frac{\partial f}{\partial w} \frac{\mathrm{d}w}{\mathrm{d}\alpha} \delta\alpha, \quad \frac{\partial x}{\partial \eta} \to \frac{\partial x}{\partial \eta} + \frac{\partial^2 x}{\partial \eta \partial \alpha} \delta\alpha.$$

Introduce $a(w) = \mathrm{d}f(w)/\mathrm{d}w$ and $b(w) = \mathrm{d}g(w)/\mathrm{d}w$ the flux Jacobians, then the linearized equation corresponding to the steady state form of (3.1) is

$$\begin{aligned} &\frac{\partial}{\partial \xi} \left\{ \left( a(w) \frac{\partial y}{\partial \eta} - b(w) \frac{\partial x}{\partial \eta} \right) \frac{\mathrm{d}w}{\mathrm{d}\alpha} \right\} + \frac{\partial}{\partial \eta} \left\{ \left( -a(w) \frac{\partial y}{\partial \xi} + b(w) \frac{\partial x}{\partial \xi} \right) \frac{\mathrm{d}w}{\mathrm{d}\alpha} \right\} \\ &+ \frac{\partial}{\partial \xi} \left\{ f(w) \frac{\partial^2 y}{\partial \eta \partial \alpha} - g(w) \frac{\partial^2 x}{\partial \eta \partial \alpha} \right\} + \frac{\partial}{\partial \eta} \left\{ -f(w) \frac{\partial^2 y}{\partial \xi \partial \alpha} + g(w) \frac{\partial^2 x}{\partial \xi \partial \alpha} \right\} = 0, \end{aligned} \tag{3.3}$$

and similarly for (3.2):

$$\begin{aligned} \frac{\mathrm{d}\mathcal{J}(\alpha)}{\mathrm{d}\alpha} &= \int_{\xi_{\min}} \left( \frac{dJ_1(w)}{dw} \frac{\mathrm{d}w}{\mathrm{d}\alpha} \frac{\partial y}{\partial \eta} + J_1(w) \frac{\partial^2 y}{\partial \eta \partial \alpha} \right) \mathrm{d}\eta \\ &+ \int_{D_{\xi\eta}} \left( \frac{\mathrm{d}J_2(w)}{\mathrm{d}w} \frac{\mathrm{d}w}{\mathrm{d}\alpha} K(\xi, \eta) + J_2(w) \frac{\partial K(\xi, \eta)}{\partial \alpha} \right) \mathrm{d}\xi \,\mathrm{d}\eta. \end{aligned} \tag{3.4}$$

Before continuing it is convenient to introduce notation for the flux Jacobian in the computational mesh directions $\xi$ and $\eta$,

$$a_1(w, \xi, \eta) = \left( a(w) \frac{\partial y}{\partial \eta} - b(w) \frac{\partial x}{\partial \eta} \right), \quad a_2(w, \xi, \eta) = \left( -a(w) \frac{\partial y}{\partial \xi} + b(w) \frac{\partial x}{\partial \xi} \right). \tag{3.5}$$

The idea behind the following procedure is to add to (3.4) the inner product of the linearized governing equations with an arbitrary four-component Lagrange multiplier $\lambda$, analogously to the discrete case in Section 1.1.4. Then we search for a condition on $\lambda$ for the gradient to be independent of the $\mathrm{d}w/\mathrm{d}\alpha$

terms. In this case we assume that the flow and adjoint solutions, $w$ and $\lambda$, are once continuously differentiable with respect to the computational coordinates, i.e. $w, \lambda \in C^1(D_{\xi\eta})^4$. Note that this is a very different restriction to that necessary in the discrete case. Proceeding from (3.3) we have that

$$\int_{D_{\xi\eta}} \lambda^T \left\{ \frac{\partial}{\partial \xi} \left( a_1(w,\xi,\eta) \frac{\mathrm{d}w}{\mathrm{d}\alpha} \right) + \frac{\partial}{\partial \eta} \left( a_2(w,\xi,\eta) \frac{\mathrm{d}w}{\mathrm{d}\alpha} \right) \right\} \mathrm{d}\xi \, \mathrm{d}\eta +$$
$$\int_{D_{\xi\eta}} \lambda^T \left\{ \frac{\partial}{\partial \xi} \left( f(w) \frac{\partial^2 y}{\partial \eta \partial \alpha} - g(w) \frac{\partial^2 x}{\partial \eta \partial \alpha} \right) + \frac{\partial}{\partial \eta} \left( -f(w) \frac{\partial^2 y}{\partial \xi \partial \alpha} + g(w) \frac{\partial^2 x}{\partial \xi \partial \alpha} \right) \right\} \mathrm{d}\xi \, \mathrm{d}\eta = 0.$$

Using integration by parts, and the fact that the flow sensitivity and coordinate derivatives such as $\partial^2 y / \partial \xi \partial \alpha$ are taken to be zero at the farfield we have

$$- \int_{D_{\xi\eta}} \frac{\partial \lambda^T}{\partial \xi} a_1(w,\xi,\eta) \frac{\mathrm{d}w}{\mathrm{d}\alpha} \, \mathrm{d}\xi \, \mathrm{d}\eta - \int_{D_{\xi\eta}} \frac{\partial \lambda^T}{\partial \eta} a_2(w,\xi,\eta) \frac{\mathrm{d}w}{\mathrm{d}\alpha} \, \mathrm{d}\xi \, \mathrm{d}\eta$$
$$- \int_{D_{\xi\eta}} \frac{\partial \lambda^T}{\partial \xi} \left( f(w) \frac{\partial^2 y}{\partial \eta \partial \alpha} - g(w) \frac{\partial^2 x}{\partial \eta \partial \alpha} \right) + \frac{\partial \lambda^T}{\partial \eta} \left( -f(w) \frac{\partial^2 y}{\partial \xi \partial \alpha} + g(w) \frac{\partial^2 x}{\partial \xi \partial \alpha} \right) \mathrm{d}\xi \, \mathrm{d}\eta$$
$$- \int_{\xi_{\min}} \lambda^T a_1(w,\xi,\eta) \frac{\mathrm{d}w}{\mathrm{d}\alpha} \, \mathrm{d}\eta - \int_{\xi_{\min}} \lambda^T \left( f(w) \frac{\partial^2 y}{\partial \eta \partial \alpha} - g(w) \frac{\partial^2 x}{\partial \eta \partial \alpha} \right) \mathrm{d}\eta = 0.$$

Adding this expression to (3.4) and extracting terms multiplying $\mathrm{d}w/\mathrm{d}\alpha$ we obtain, from the volume and surface integrals respectively:

$$\begin{cases} \dfrac{\mathrm{d}J_2(w)}{\mathrm{d}w} K(\xi,\eta) - \dfrac{\partial \lambda^T}{\partial \xi} a_1(w,\xi,\eta) - \dfrac{\partial \lambda^T}{\partial \eta} a_2(w,\xi,\eta) = 0, & \text{on } D_{\xi,\eta}, \\ -\lambda^T a_1(w,\xi,\eta) + \dfrac{\mathrm{d}J_1(w)}{\mathrm{d}w} \dfrac{\partial y}{\partial \eta} = 0, & \text{on } \xi_{\min}, \end{cases} \tag{3.6}$$

the *continuous adjoint* equations and boundary conditions.

One very significant feature of this problem is that not all objective functions $J_1$ lead to a well posed adjoint boundary condition, because the flux Jacobian $a_1$ is singular at a slip wall [78]. For the compressible Euler equations described here functions of pressure are admissible, which is fortunate given that integral forces on a profile are thereby allowed. On the other hand for the Navier-Stokes equations there is no clear way of accounting for wall shear-stresses, and hence viscous drag [7]. Given a solution of (3.6) the gradients of $\mathcal{J}$ may be written

$$\begin{aligned} \frac{\mathrm{d}\mathcal{J}(\alpha)}{\mathrm{d}\alpha} = & \int_{\xi_{\min}} J_1(w) \frac{\partial^2 y}{\partial \eta \partial \alpha} \, \mathrm{d}\eta - \int_{\xi_{\min}} \lambda^T \left( f(w) \frac{\partial^2 y}{\partial \eta \partial \alpha} - g(w) \frac{\partial^2 x}{\partial \eta \partial \alpha} \right) \mathrm{d}\eta \\ & - \int_{D_{\xi\eta}} \frac{\partial \lambda^T}{\partial \xi} \left( f(w) \frac{\partial^2 y}{\partial \eta \partial \alpha} - g(w) \frac{\partial^2 x}{\partial \eta \partial \alpha} \right) \mathrm{d}\xi \, \mathrm{d}\eta \\ & - \int_{D_{\xi\eta}} \frac{\partial \lambda^T}{\partial \eta} \left( -f(w) \frac{\partial^2 y}{\partial \xi \partial \alpha} + g(w) \frac{\partial^2 x}{\partial \xi \partial \alpha} \right) \mathrm{d}\xi \, \mathrm{d}\eta \\ & + \int_{D_{\xi\eta}} J_2(w) \frac{\partial K(\xi,\eta)}{\partial \alpha} \, \mathrm{d}\xi \, \mathrm{d}\eta. \end{aligned} \tag{3.7}$$

For the extension to the Navier-Stokes equations, also derived in curvilinear coordinates, see the seminary articles of Jameson, Martinelli and Pierce of 1997 [84], while a more discursive treatment is given in [81], which also includes considerations related to the use of the thus obtained gradients in shape optimization.

For finite volumes on unstructured meshes such coordinate transforms as described above are not used, and a formulation in physical coordinates is necessary. One approach was first published by Anderson and Venkatakrishnan in 1998 [7], and one year later by Hiernaux and Essers [74, 75]. Also of interest is an early adjoint formulation of the thin shear-layer equations in physical coordinates [60].

Finally we note that the continuous direct formulation, embodied by (3.3), rarely occurs in the gradient evaluation literature. Pelletier et al. [133, 132] applied it to incompressible flows, and the one of few compressible references is of Sharp and Sirovitch for hypersonic profile optimization [163]. At the end, of course, the continuous equations are discretized (using most often classical CFD schemes) to define are computational problem of finite dimension.

## Annex 2 – Curvature of an extruded curve

Classical notations are used for 2D geometry (see figure 3.5) : $s$ is the curve length, $(\mathbf{t}(s), \mathbf{n}(s))$ is the local frame of tangent and normal vectors (counter-clockwise oriented). The curvature $k(s)$ is positive or negative, and it can be defined as the derivative of the angle of the local tangent w.r.t. a fixed axis (for example $Ox$ axis) Its absolute value is the inverse of the radius of curvature. The formulas of Frenet state that

$$\frac{d\mathbf{t}(s)}{ds} = k(s)\mathbf{n}(s) \qquad \frac{d\mathbf{n}(s)}{ds} = -k(s)\mathbf{t}(s)$$

The local optimization process of interest defines curves from equation (1.29) (see figure 3.6) and uses a criterion based on the total variation of the targeted shape to validate a field of normal component deformation. Hence the curvature of an extruded curve is first estimated. (For the sake of simplicity, the continuous curve of interest is denoted $S^\tau(s)$ as its discrete counterpart, and $d(s)$ is preferred to a notation accounting for the smoothing like $\mathcal{S}(d)(s)$.)

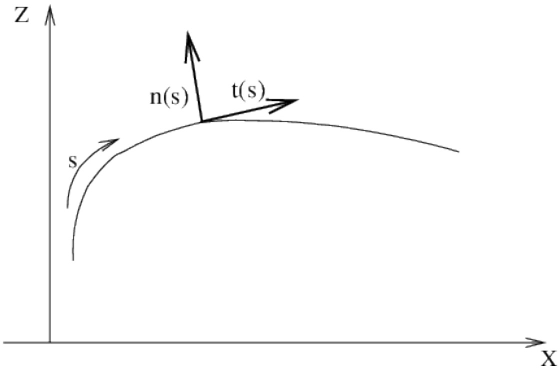


Figure 3.5: Notations for curvature calculation

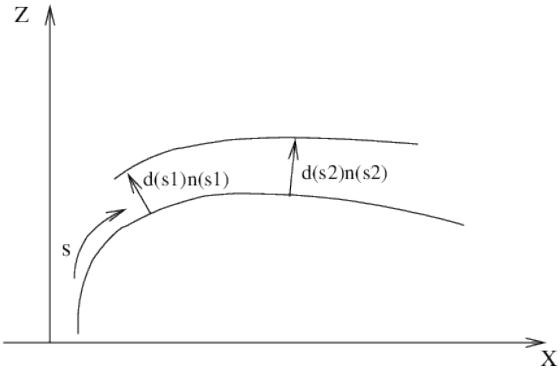


Figure 3.6: Extruded curve

The targeted curve is denoted here $S^\tau(s) = S(s) + \tau d(s)\mathbf{n}(s)$. The first two derivatives of the formula are given by

$$\begin{aligned}
\frac{dS^\tau(s)}{ds} &= (1 - \tau d(s)k(s))\,\mathbf{t}(s) + \tau d'(s)\mathbf{n}(s) \\
\frac{d^2S^\tau(s)}{ds^2} &= -\tau\left(2d'(s)k(s) + d(s)k'(s)\right)\mathbf{t}(s) + \Big(k(s) + \tau(d''(s) - d(s)k(s)^2)\Big)\,\mathbf{n}(s)
\end{aligned}$$

where we have used the fact that $s$ is the arc length along $S$ so that $dS(s)/ds = \mathbf{t}(s)$. Of course, $s$ is not the actual arc length along $S^\tau$ but from a classical formula for parametrized curves, one can easily derive the expression of $k^\tau(s)$, the curvature along $S^\tau$ :

$$k^\tau(s) = \frac{(k(s) + \tau(d''(s) - d(s)k(s)^2))(1 - \tau d(s)k(s)) + \tau^2 d'(s)(2d'(s)k(s) + d(s)k'(s))}{((1 - \tau d(s)k(s))^2 + \tau^2 d'(s)^2)^{3/2}} \tag{3.8}$$

As stated before, it is highly desirable that the curvature along $S^\tau$ satisfies a bounding condition - equation (1.30). We now try to derive from previous equations the corresponding condition on $d(s)$ :
– based on a Taylor expansion on parameter $\tau$. Obviously

$$k^\tau(s) = k(s)\left[\frac{(1 + \tau(d''(s)/k(s) - d(s)k(s)))(1 - \tau d(s)k(s))}{(1 - 2\tau d(s)k(s))^{3/2}} + O(\tau^2)\right]$$

Or

$$k^\tau(s) = k(s)\left[1 + \tau(d''(s)/k(s) + d(s)k(s)) + O(\tau^2)\right]$$

The curvature of the current airfoil is much more regular than $d(s)$ distributions that appear during the optimization and smoothing process. Hence, in first approximation, a strong increase in total variation of curvature between $S$ and $S^\tau$ would be caused by a high total variation of $d(s)$ or $d''(s)$. Actually, the total variation of $d(s)$ is decreased by all smoothing operators, hence the desirable property appears to be a limited total variation of $d''(s)$; or in other words, a smooth $d'''(s)$.
Additionally, the following limits can be shown without small $\tau$ assumption from formula (3.8) $k^\tau(s) \to 0$ as $d'(s) \to \infty$ and $|k^\tau(s)| \to \infty$ as $d''(s) \to \infty$.

# Annex 3 – Visualization of $(1/ds)\Lambda^T_{CD_p}(\partial R/\partial X)$ for a 2D Euler subcritical flow

The series of plots 3.7 illustrates the good consistency of left-hand-side and numerically estimated right-hand-side of equation (2.9) (based on the same flow and adjoint fields). This property is also observed if flow and/or adjoint involve discontinuities. For the visualisation, both fields are divided by the surface attached to the node in the dual mesh ($(1/ds)\Lambda^T_{CD_p}(\partial R/\partial X)$ versus discrete evaluation of term (2.11) ).

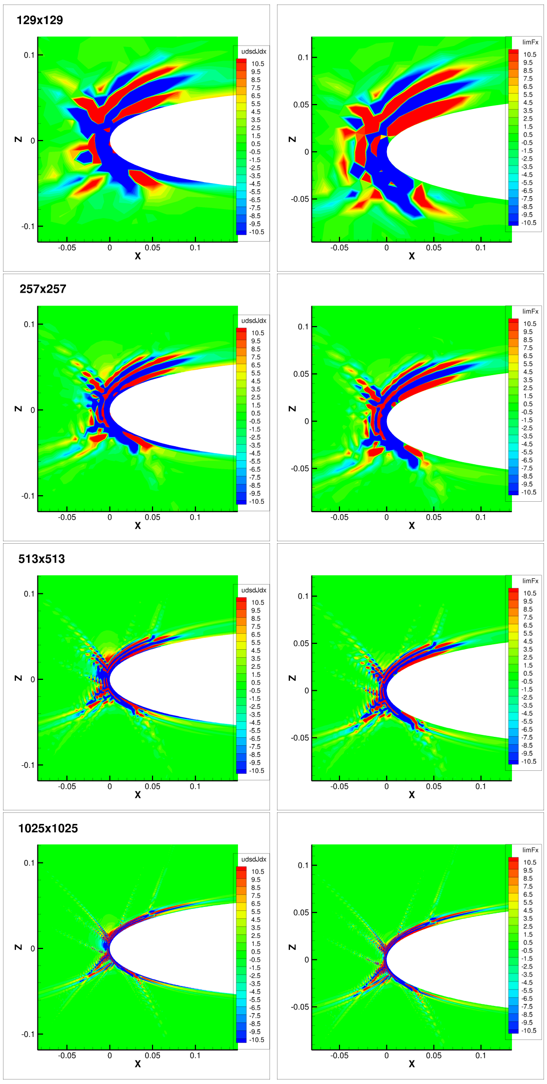


Figure 3.7: NACA0012 inviscid flow $M_\infty$=0.4 AoA =$5^o$. Roe-MUSCL (van Albada) scheme – Left: (1/ds) $\Lambda^T(\partial R/\partial x)$ – Right: discretization of term (2.11) using Green formulas

# Annex 4 – Definition of $\overline{\mathcal{P}(dJ/dX)}$, the spatial mean of $\mathcal{P}(dJ/dX)$ field

A discrete convolution based spatial mean $\overline{\mathcal{P}(dJ/dX)}$ is built. It is defined for a 2D problem and a structured mesh but extensions to 3D and unstructured meshes are straightforward. It is assumed that the discrete values of a realistic mesh displacement field $dX$ are the values of a $C^1$ regular function noted $XC$ taken at the nodes of the mesh. It is also assumed that $XC$ is well approximated by its first-order Taylor expansion on all circles of radius $L$. Finally, $\mathcal{D}_{(C,L)}$ denotes the disk centered in $C$ with radius $L$ and $\Psi_L$ denotes a radial a radial function of support $\mathcal{D}(O,L)$ and integral 1 on this disk. Using all these properties, it is easily checked that the displacement of the node $(i,j)$ (denoted $dX_{ij}$) can be estimated by the following relation:

$$\int_{\mathcal{D}_{(X_{ij},L)}} XC(u)\Psi_L(u-X_{ij})du \simeq XC(X_{ij})\int_{\mathcal{D}_{(X_{ij},L)}} \Psi_L(u-X_{ij})du = dX_{ij} \tag{3.9}$$

Although this property is available for all radial function $\Psi_L(r)$ of support $\mathcal{D}(O,L)$ and integral 1, it is desirable to use a decreasing function of $r$. In practice, the following kind of functions is used

$$\begin{array}{rcll}\Psi_L^{\nu}(u) & = & \dfrac{(L^2-r^2)^{\nu}}{2\pi\displaystyle\int_0^L r(L^2-r^2)^{\nu}dr} & \text{for } \|u\| \leq L \\ \Psi_L^{\nu}(u) & = & 0 & \text{for } \|u\| > L\end{array}$$

The simplest discretization on the mesh of equation (3.9) is

$$dX_{ij} = \sum_{(i'j')/X_{i'j'}\in\mathcal{D}_{(X_{ij},L)}} dX_{i'j'}\Psi_L(X_{i'j'}-X_{ij})ds_{i'j'}$$

where $ds_{ij}$ is the surface of the mesh element associated with the point $X_{ij}$ (plotted on figure 3.8(a)). However this relation is not exact on a constant field and inconsistent for coarse mesh zones. So we prefer the barycentric discretization

$$dX_{ij} = \frac{\sum_{(i'j')/X_{i'j'}\in\mathcal{D}_{(X_{ij},L)}} \Psi_L(X_{i'j'}-X_{ij})ds_{i'j'}\, dX_{i'j'}}{\sum_{(i'j')/X_{i'j'}\in\mathcal{D}_{(X_{ij},L)}} \Psi_L(X_{i'j'}-X_{ij})ds_{i'j'}} \tag{3.10}$$

Henceforth we note $\gamma_{ijL}$ the denominator of the right hand side of equation (3.10)

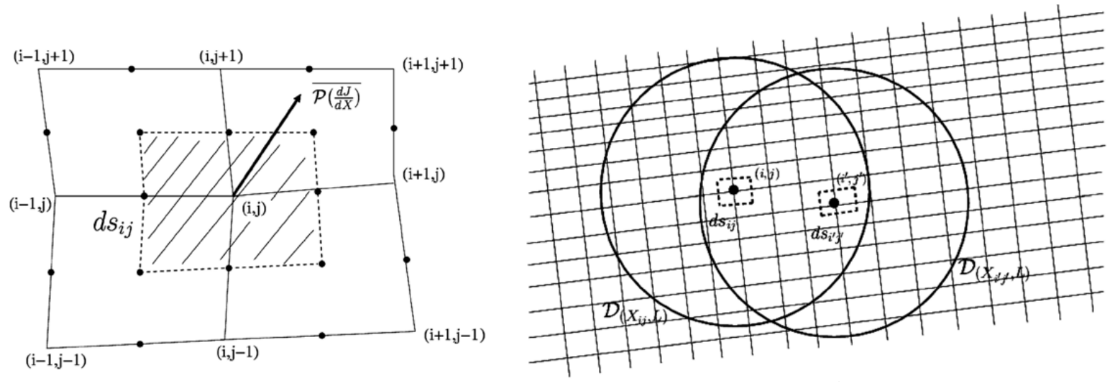
$$\gamma_{ijL} = \sum_{(i'j')/X_{i'j'}\in\mathcal{D}_{(X_{ij},L)}} \Psi_L(X_{i'j'}-X_{ij})ds_{i'j'}.$$

The dot product of interest can then be rewritten:

$$\mathcal{P}(dJ/dX).dX = \sum_{(ij)}\mathcal{P}\left(dJ/dX_{ij}\right)dX_{ij} = \sum_{(ij)}\frac{1}{\gamma_{ijL}}\mathcal{P}(dJ/dX_{ij}) \sum_{(i'j')/X_{i'j'}\in\mathcal{D}_{(X_{ij},L)}} \Psi_L(X_{i'j'}-X_{ij})ds_{i'j'}dX_{i'j'}$$

It is interesting to switch the sums over the indices $(i,j)$ and $(i'j')$ noticing that $X_{i'j'} \in \mathcal{D}_{(X_{ij},L)}$ is equivalent to $X_{ij} \in \mathcal{D}_{(X_{i'j'},L)}$ (see figure 3.8(b))

$$\mathcal{P}(dJ/dX).dX = \sum_{(i'j')}\left(ds_{i'j'} \sum_{(ij)/X_{ij}\in\mathcal{D}_{(X_{i'j'},L)}} \left(\frac{1}{\gamma_{ijL}}\mathcal{P}(dJ/dX_{ij})\Psi_L(X_{i'j'}-X_{ij})\right)\right)dX_{i'j'}$$



(a) (b)

Figure 3.8: (a) Surface $ds_{ij}$ associated to node $(i,j)$ in the definition of $\overline{\mathcal{P}(dJ/dX)}$ (b) disks $\mathcal{D}_{(X_{ij},L)}$ and $\mathcal{D}_{(X_{i'j'},L)}$

Our purpose was to build a mean such that for any displacement field $dX$ well approximated by a linear function at scale $L$

$$\mathcal{P}(dJ/dX).dX = \overline{\mathcal{P}(dJ/dX)}.dX$$

By simple identification with previous equation (and switching the indices with and without prime)

$$\overline{\mathcal{P}(dJ/dX)}_{ij} = ds_{ij} \sum_{(i'j')/{X'}_{ij'}\in\mathcal{D}_{(X_{ij},L)}} \frac{1}{\gamma_{i'j'L}}\mathcal{P}(dJ/dX_{i'j'}))\Psi_L(X_{i'j'}-X_{ij}) \qquad (3.11)$$

which defines the mean field of $\mathcal{P}(dJ/dX)$ as an explicit convolution-like mean.
Unfortunately, the equations (3.9),(3.10) are inaccurate at all nodes $(i,j)$ such that $\mathcal{D}_{(X_{ij},L)}$ is not entirely included in the fluid domain. Therefore the previous definition of the mean field $\overline{\mathcal{P}(dJ/dX)}$ is used only for the nodes $(i,j)$ such that $\mathcal{D}_{(X_{ij},L)}$ is entirely included in the fluid domain. For the other nodes, the definition has been extended by changing the shape of the integration domain.

# Bibliography

[1] H. Akima. A new method of interpolation and smooth curve fitting based on local procedures. *Journal of the Association for Computing Machinery*, 4:589–602, 1970.

[2] S. Albensoeder. *Application of Mesh Modifications and Adjoint Error Estimates*, volume 42 of *Notes on Numerical Fluid Mechanics and Multidisciplinary Design*. Springer, 2013.

[3] T. Albring, M. Sagebaum, and N. Gauger. Efficient aerodynamic design using the discrete adjoint method in su2. In *AIAA Paper Series, Paper 2016-3518*. 2016.

[4] Z. Ali, P. Tucker, and S. Shahpar. Optimal mesh topology for CFD. *Computer Methods in Applied Mechanics and Engineering*, 317:431–457, 2017.

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

[6] W. Anderson, J. Newman, D. Whitfield, and E. Nielsen. Sensitivity analysis for the Navier-Stokes equations on unstructured meshes using complex variables. *AIAA Journal*, 39(1):56–63, 2001.

[7] W. Anderson and V. Venkatakrishnan. Aerodynamic design optimization on unstructured grids with a continuous adjoint formulation. *Computers and Fluids*, 28:443–480, 1999.

[8] A. Barthet, C. Airiau, M. Braza, and L. Tourrette. Adjoint-based error correction applied to far-field drag breakdown on structured grid. In *AIAA Paper Series, Paper 2006-3315*. 2006.

[9] O. Baysal and M. Eleshaky. Aerodynamic design sensitivity analysis methods for the compressible Euler equations. *Journal of Fluids Engineering*, 113(4):681–688, 1991.

[10] A. Belme, F. Alauzet, and A. Dervieux. An a priori anisotropic goal-oriented error estimate for viscous compressible flow and application to mesh adaptation. *Journal of Computational Physics*, 376:1051–1088, 2019.

[11] R. Biedron, J. Carlson, J. Derlaga, P. Gnoffo, D. Hammond, W. Jones, B. Kleb, E. Lee-Rausch, E. Nielsen, M. Park, C. Rumsey, J. Thomas, and W. Wood. FUN 3D Manual 13.3. Technical Report TM 219808, NASA, 2018.

[12] M. Bompard. *Modèles de substitution pour l'optimisation globale de forme en aérodynamique et méthode locale sans paramétrisation*. PhD thesis, Université Nice-Sophia Antipolis, December 2011.

[13] M. Bompard, J.-A. Désidéri, and J. Peter. Best Hermitian interpolation in presence of uncertainties. Technical Report 7422, INRIA, 2010.

[14] M. Bompard and J. Peter. *Strategies for Optimization and Automated Design of Gas Turbine Engines – Chapter: Local Search Methods for Design in Aeronautics*. RTO-AVT, Bruxelles, 2010.

[15] M. Bompard, J. Peter, G. Carrier, and J.-A. Désidéri. Two-dimensional aerodynamic optimization with or without parametrization. In *AIAA Paper Series, Paper 2011-3073*. 2011.

[16] J.-C. Boniface. *Calculs d'écoulements compressibles autour des rotor d'hélicoptères et vol stationnire ou en vol d'avancement par résolution des équations d'Euler*. PhD thesis, Ecole Nationale des Arts et Métiers, December 1995.

[17] S. Bourasseau. *Contribution à une méthode de raffinement de maillage basée sur le vecteur adjoint pour le calcul de fonctions aerodynamiques*. PhD thesis, Université Nice-Sophia Antipolis, December 2015.

[18] J. Brezillon. Private Communication, 2018.

[19] W. Briley and H. McDonald. An overview and generalization of implicit Navier-Stokes algorithms and approximate factorization. *Computers and Fluids*, 30:827–828, 2001.

[20] D. Bristow and J. Hawk. Subsonic panel method for the efficient analysis of multiple geometry perturbations. Technical Report CR 3528, NASA, 1982.

[21] D. Bristow and J. Hawk. Subsonic panel method for designing wing surface from pressure distribution. Technical Report CR 3713, NASA, 1983.

[22] W. Bryley and H. McDonald. An overview and generalization of implicit Navier-Stokes algorithms and approximate factorization. *Computers and Fluids*, 30:807–727, 2001.

[23] L. Cambier and M. Gazaix. $elsA$: an efficient object-oriented solution to CFD complexity. In *AIAA Paper Series, Paper 2002-0108*. 2002.

[24] L. Cambier, S. Heib, and S. Plot. The elsA CFD software: input from research and feedback from industry. *Mechanics & Industry*, 14(3):159–174, 2013.

[25] L. Cambier and J.-P. Veuillot. Status of the $elsA$ CFD software for flow simulation and and multidisciplinary applications. In *AIAA Paper Series, Paper 2008-0664*. 2008.

[26] D. Campbell. Efficient viscous design of realistic aircraft configurations. In *AIAA Paper Series, Paper 98-2539*. 1998.

[27] G. Candler and R. MacCormack. Hypersonic flow past 3D configurations. In *AIAA Paper Series, Paper 1987-0480*. 1987.

[28] G. Carrier, D. Destarac, A. Dumont, M. Méheut, I. Salah el Din, J. Peter, S. Ben Khelil, J. Brezillon, and M. Pestana. Gradient-based optimization with the $elsA$ software. In *AIAA Paper Series, Paper 2014-0568*. 2014.

[29] P. Castonguay and S. Nadarajah. Effect of shape parametrization on aerodynamic shape optimization. In *AIAA Paper Series, Paper 2007-59*. 2007.

[30] M. Castro-Diaz, F. Hecht, B. Mohammadi, and O. Pironneau. Anisotropic unstructured mesh adaptation for flow simulations. *International Journal for Numerical Methods in Fluids*, 25(4):475–491, 1997.

[31] K. Chitale, O. Sahni, M. Shephard, S. Tendulkar, and K. Jansen. Anisotropic adaptation for transonic flows with turbulent boundary layers. *AIAA Journal*, 53:367–378, 2015.

[32] P. Cook, M. McDonald, and M. Firmin. Aerofoil RAE 2822 - pressure distributions, and boundary layer and wake measurements – experimental data base for computer program assessment. Technical Report AR 138, AGARD, 1979.

[33] A. Costes. Lissage du gradient de forme pour l'optimisation aérodynamique des profils. Technical report, ONERA Internship, 2015.

[34] J. Délery. *Traité d'aérodynamique compressible*. Lavoisier. Hermès, 2008.

[35] D. Destarac. Far-field / near-field drag balance. In *VKI Course on Aircraft drag prediction and reduction*. 2003.

[36] D. Destarac and J. Reneaux. Application de l'optimisation numérique à l'aérodynamique des avions de transport. *La Recherche Aérospatiale*, 2:39–55, 1993.

[37] P. Dierckx. An algorithm for surface-fitting with spline functions. *Journal of Numerical Analysis*, 1:267–283, 1981.

[38] B. Diskin and N. Yamaleev. Grid adaptation using adjoint-based error minimization. In *AIAA Paper Series, Paper 2011-3986*. 2011.

[39] C. Dobrzynski. MMG3D user guide. Technical Report 422, INRIA, 2012.

[40] F. Drullion. *Définition et études de sytèmes linéaires pour la simulation et l'optimisation de formes aérodynamiques par méthode de gradient.* PhD thesis, Université Bordeaux I, November 2004.

[41] A. Dumont. *Calcul de gradient pour l'optimisation des performances aérodynamiques d'un rotor d'hélicoptère en vol stationnaire*. PhD thesis, Université de Poitiers, April 2010.

[42] A. Dumont, A. Le Pape, J. Peter, and S. Huberson. Sensitivity analysis using direct differentiation and adjoint vector methods for helicopter rotor flow in hover. In *AIAA Paper Series, Paper 2008-5811*. 2008.

[43] A. Dumont, A. Le Pape, J. Peter, and S. Huberson. Aerodynamic shape optimization of hovering rotors using a discrete adjoint of the RANS equations. In *Proceedings of 65th American Helicopter Society annual forum, Grapevine, Texas USA*, 2009.

[44] A. Dumont, A. Le Pape, J. Peter, and S. Huberson. Aerodynamic shape optimization of hovering rotors using a discrete adjoint of the Reynolds-averaged Navier-Stokes equations. *Journal of the American Helicopter Society*, 56(032002):1–11, 2011.

[45] R. Dwight. Efficient a posteriori error estimation for finite volume methods. In *Proceedings of the RTO-AVT-147 Symposium*, 2007.

[46] R. Dwight. Goal-oriented mesh adaptation using a dissipation based error indicator. *International Journal for Numerical Methods in Fluids*, 56(8):1193–2000, 2007.

[47] R. Dwight. Heuristic *a posteriori* estimation of error due to dissipation in finite volume schemes and application to mesh adaptation. *Journal of Computational Physics*, 227:2845–2863, 2008.

[48] T. Economon, J. Alonso, T. Albring, and N. Gauger. Adjoint formulation investigations of benchmark aerodynamic design cases in SU2. In *AIAA Paper Series, Paper 2017-4363*. 2017.

[49] H. Elbanna and L. Carlson. Determination of aerodynamic sensitivity coefficients in the transonic and supersonic regimes. In *AIAA Paper Series, Paper 89-0532*. 1989.

[50] K. Fidkowski, M. Ceze, and P. Roe. Entropy-based drag error estimation and mesh-adaptation in two dimensions. *Journal of Aircraft*, 45(9):1485–1496, 2012.

[51] K. Fidkowski and D. Darmofal. Review of output-based error estimation and mesh adaptation in computational fluid dynamics. *AIAA Journal*, 49(4):673–694, 2011.

[52] K. Fidkowski and P. Roe. An entropy approach to mesh refinement. *SIAM Journal of Scientific Computing*, 32(3):1261–1287, 2010.

[53] R. Fletcher and C. Reeves. Function minimization by conjugate grandients. *Computer Journal*, 7:149–154, 1964.

[54] P. Frank and G. Shubin. A comparison of optimisation-based approaches for a model computational aerodynamics design problem. *Journal of Computational Physics*, 98:74–89, 1992.

[55] L. Frazza. *3D anisotropic mesh adaptation for Reynolds Averaged Navier-Stokes simulations*. PhD thesis, Paris Sorbonne Université, December 2018.

[56] P. Frey and F. Alauzet. Anisotropic mesh adaptation for CFD computations. *Computer Methods in Applied Mechanics and Engineering*, 194:5068–5082, 2005.

[57] M. Giles. On the iterative solution of the adjoint equations. In *Automatic Differentiation: From Simulation to Optimization*, pages 145–152. 2001.

[58] M. Giles, M. Duta, and J.-D. Müller. Adjoint code developments using the exact discrete approach. In *AIAA Paper Series, Paper 2001-2596*. 2001.

[59] M. Giles, M. Duta, J.-D. Müller, and N. Pierce. Algorithm developments for discrete adjoint methods. *AIAA Journal*, 41(2), 2003.

[60] M. Giles and N. Pierce. Adjoint equations in CFD: Duality, boundary conditions and solution behaviour. In *AIAA Paper Series, Paper 97-1850*. 1997.

[61] M. Giles and N. Pierce. Improved lift and drag estimates using adjoint Euler equations. In *AIAA Paper Series, Paper 1999-3293*. 1999.

[62] M. Giles and N. Pierce. An introduction to the adjoint approach to design. In *Proceedings of ERCOFTAC Workshop on Adjoint Methods*, 1999.

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

[64] M. Giles and N. Pierce. Adjoint and defect error bounding and correction for funvtional estimates. In *AIAA Paper Series, Paper 2003-3846*. 2003.

[65] M. Giles and N. Pierce. *Adjoint error correction for integral ouptputs*. Lecture Notes in Computational Science and Engineering. Springer, 2003.

[66] A. Griewank. *Evaluating Derivatives, Principles and Techniques of Algorithmic Differentiation*. No. 19 in Frontiers in Applied Mathematics, SIAM, Philadelphia, 2000. ISBN 08-987-1451-6.

[67] W. Haase, F. Bradsma, E. Elsholz, M. Leschziner, and D. Schwamborn. *EUROVAL - an European initiative on validation of CFD codes*, volume 42 of *Notes on numerical fluid mechanics*. VIEWEG, 1993.

[68] R. Hartmann, J. Held, and T. Leicht. Adjoint-based error estimation and adaptative mesh refinement for the RANS and k-$\omega$ turbulence model equations. *Journal of Computational Physics*, 230:4268–4284, 2010.

[69] R. Hartmann and P. Houston. Adaptative Discontinuous Galerkin finite element methods for the compressible Euler equations. *Journal of Computational Physics*, 183:508–532, 2002.

[70] L. Hascoët and V. Pascual. The Tapenade Automatic Differentiation tool: Principles, Model, and Specification. *ACM Transactions On Mathematical Software*, 39(3), 2013.

[71] R. Hicks and P. Henne. Wing design by numerical optimization. In *AIAA Paper Series, Paper 77-1247*. 1977.

[72] R. Hicks and P. Henne. Wing design by numerical optimization. *Journal of Aircraft*, 15(7):407–412, 1978.

[73] R. Hicks, E. Murman, and G. Vanderplaats. An assessment of airfoil design by numerical optimization. Technical Report TMX 3092, NASA, 1974.

[74] S. Hiernaux and J.-A. Essers. An optimal control theory based algorithm to solve 2D aerodynamic shape optimisation problems for inviscid and viscous flows. In *Proceedings of the RTO-AVT Symposium on Aerodynamic Design and Optimisation of Flight Vehicles*, 1999.

[75] S. Hiernaux and J.-A. Hessers. Aerodynamic optimization using Navier-Stokes equations and optimal control theory. In *AIAA Paper Series, Paper 99-3297*. 1999.

[76] M. Hojjat, E. Stavropoulou, and K.-U. Bletzinger. The vertex morphing method for node-based shape optimization. *Computer Methods in Applied Mechanics and Engineering*, 268:494–513, 2014.

[77] Y. Hu, C. Wagner, S. Allmaras, M. Galbraith, and D. Darmofal. Application of higher-order adaptive method to Reynolds-Averaged Navier-Stokes test cases. *AIAA Journal*, 54(9):2626–2644, 2016.

[78] A. Jameson. Aerodynamic design via control theory. *Journal of Scientific Computing*, 3(3):233–260, 1988.

[79] A. Jameson. Optimum aerodynamic design via boundary control. In *AGARD-FDP-VKI Special Course*. 1994.

[80] A. Jameson. Optimum aerodynamic design using CFD and control theory. In *AIAA Paper Series, Paper 95-1729*. 1995.

[81] A. Jameson. Aerodynamic shape optimization using the adjoint method. In *VKI Course in Optimization*. 2003.

[82] A. Jameson, J. Alonso, J. Reuther, L. Martinelli, and J. Vassberg. Aerodynamics shape optimization techniques based on control theory. In *Proceedings of Control Theory, CIME (International Mathematical Summer) and AIAA Paper 1998-2538*, 1998.

[83] A. Jameson and T. Baker. Solution of the Euler equations for complex configurations. In *AIAA Paper Series, Paper 1983-1929*. 1983.

[84] A. Jameson, L. Martinelli, and N. Pierce. Optimum aerodynamic design using the Navier-Stokes equations. *Theoretical and Computational Fluid Dynamics*, 10(1):213–237, 1998.

[85] A. Jameson, W. Schmidt, and E. Turkel. Numerical solutions of the Euler equations by finite volume methods using Runge-Kutta time-stepping schemes. In *AIAA Paper Series, Paper 1981-1259*. 1981.

[86] A. Jameson and J. Vassberg. Studies of alternative numerical optimization methods applied to the brachistochrone problem. *Computational Fluid Dynamics Journal*, 9(3):281–296, 2000.

[87] A. Jameson and J. Vassberg. Computational fluid dynamics for aerodynamic design: Its current and future impact. In *AIAA Paper Series, Paper 01-0538*. 2001.

[88] A. Jaworski and J.-D. Müller. An implicit finite volume method for solving the Euler equations. *Lecture Notes in Computational Science and Engineering*, 64:281–291, 2008.

[89] W. Jones, E. Nielsen, and M. Park. Validation of 3d adjoint based error estimation and mesh adaptation for sonic boom prediction. In *AIAA Paper Series, Paper 2006-1150*. 2006.

[90] C. Kavouklis and Y. Kallinderis. Parallel adaptation of general three-dimensionnal hybrid meshes. *Journal of Computational Physics*, 229:3454–3473, 2010.

[91] S. Kim, J. Alonso, and A. Jameson. Two-dimensional high-lift aerodynamic optimization using the continuous adjoint method. In *AIAA Paper Series, Paper 2000-4741*. 2000.

[92] S. Kim, H. Hosseini, K. Leoviriyakit, and A. Jameson. Enhancement of adjoint design methods via optimization of adjoint parameters. In *AIAA Paper Series, Paper-2005-448*. 2005.

[93] J. Kok. Resolving the dependance of free stream value for $k-\omega$ turbulence model. *AIAA Journal*, 38(7):1292–1295, 2000.

[94] N. Kroll, N. Gauger, J. Brezillon, K. Becker, and V. Schulz. Ongoing activities in shape optimization within the german project megadesign. In *Proceedings of ECCOMAS, Jyvaskyla*, 2004.

[95] E. Lee-Rausch, M. Park, W. Jones, D. Hammond, and E. Nielsen. Application of parallel adjoint-based error estimation and anisotropic grid adaptation for three-dimensional aerospace configurations. In *AIAA Paper Series, Paper 2005-4842*. 2005.

[96] S. Lee-Rausch and O. Kwon. Aerodynamic shape optimization of rotor blades in hover using unstructured meshes. In *Proceedings of 60th American Helicopter Society annual forum, Baltimore, Maryland USA*, 2004.

[97] T. Leicht and R. Hartmann. Error estimation and anisotropic mesh refinement for 3d laminar aerodynamic flow simulations. *Journal of Computational Physics*, 229:7344–7360, 2010.

[98] A. Lerat, J. Sidès, and V. Daru. An implicit finite volume method for solving the Euler equations. *Lecture Notes in Physics*, 170:343–345, 1982.

[99] W. Li and S. Krist. Spline-based airfoil curvature smoothing and its applications. *Journal of Aircraft*, 42(4):1065–1074, 2005.

[100] W. Li, S. Krist, and R. Campbell. Transonic airfoil shape optimization in preliminary design environment. In *AIAA Paper Series, Paper 2004-4629*. 2004.

[101] W. Li and S. Padula. Using high resolution design spaces for aerodynamic shape optimization under uncertainty. Technical Report TP-2004-213004, NASA, 2004.

[102] A. Loseille and F. Alauzet. Continuous mesh framework. part I: well-posed continuous interpolation error. *SIAM Journal Numerical Analysis*, 49(1):38–60, 2011.

[103] A. Loseille and F. Alauzet. Continuous mesh framework. part II: validation and applications. *SIAM Journal Numerical Analysis*, 49(1):61–86, 2011.

[104] A. Loseille, A. Dervieux, and F. Alauzet. Fully anisotropic mesh adaptation for 3D steady Euler equations. *Journal of Computational Physics*, 229:2866–2897, 2010.

[105] C. Lozano. Adjoint viscous derivatives with a reduced gradient formulation. *AIAA Journal*, 50(1):203–214, 2012.

[106] C. Lozano. Discrete surprises in the computation of sensitivities from boundary integrals in the continupus approach to inviscid aerodynamic shape optimization. *Computers and Fluids*, 56:118–127, 2012.

[107] C. Lozano. A note on the dual consistency of the discrete adjoint quasi-one dimensional Euler equations with cell-centred and cell-vertex central discretization. *Computers and Fluids*, 134-135:51–60, 2016.

[108] C. Lozano. On the properties of the solutions of the 2D adjoint Euler equations. In *Proceedings of EUROGEN 2017, Madrid*, 2017.

[109] C. Lozano. Singular and discontinuous solutions of the adjoint Euler equations. *AIAA Journal*, 56(11):4437–4451, 2018.

[110] L. Martinelli. *Calculation of viscous flows with a multigrid method*. PhD thesis, University of Princeton, January 1987.

[111] M. Meaux, M. Cormery, and G. Voizard. Viscous aerodynamic shape optimization based on the discrete adjoint state for 3d industrial configurations. In *Proceedings of ECCOMAS, Jyvaskyla*, 2004.

[112] B. Mohammadi and O. Pironneau. *Applied shape Design for Fluids*. Oxford Univ. Press, 2001.

[113] A. Mousavi, P. Castonguay, and S. Nadarajah. Survey of shape parametrization techniques and its effects on three-dimensional aerodynamic shape optimization. In *AIAA Paper Series, Paper 2007-3837*. 2007.

[114] J.-D. Müller and P. Cusdin. On the performance of discrete CFD codes using automatic differentiation. *International Journal for Numerical Methods in Fluids*, 47:939–945, 2005.

[115] J.-D. Müller and M. Giles. Solution adaptative mesh refinement using adjoint error analysis. In *AIAA Paper Series, Paper 2001-2550*. 2001.

[116] M. Méheut, A. Arntz, and C. Carrier. Aerodynamic shape optimizations of a blended wing body configuration for several wing planforms. In *AIAA Paper Series, Paper 12-3122*. 2012.

[117] M. Méheut and C. Carrier. Optimization of a blending wing body using the adjoint method. In *Evolutionary and Deterministic Methods for Design Optimization and Control, EUROGEN, Capua*, 2011.

[118] M. Nemec and M. Aftosmis. Adjoint error estimation and adaptative refinement for embedded-boundary cartesian mesh. In *AIAA Paper Series, Paper 2007-4187*. 2007.

[119] M. Nemec, M. Aftosmis, and M. Wintzer. Adjoint-based adaptative mesh refinement for complex geometries. In *AIAA Paper Series, Paper 2008-725*. 2008.

[120] J. Newman III, A. Taylor III, and G. Burgreen. An unstructured grid approach to sensitivity analysis and shape optimization using the Euler equations. In *AIAA Paper Series, Paper 95-1646*. 1995.

[121] M. Nguyen-Dinh. *Qualification des simulations numériques par adaptation anisotropique de maillages*. PhD thesis, Université de Nice-Sophia Antipolis, March 2014.

[122] M. Nguyen-Dinh, J. Peter, R. Sauvage, M. Meaux, and J.-A. Désidéri. Mesh quality assessment based on aerodynamic functional output total derivative. *European Journal of Mechanics B/Fluids*, 45:51–71, 2014.

[123] E. Nielsen and W. Anderson. Aerodynamic design optimization on unstructured meshes using the Navier-Stokes equations. *AIAA Journal*, 37(11):185–191, 1999.

[124] E. Nielsen, B. Diskin, and N. Yamaleev. Discrete adjoint-based design optimization of unsteady turbulent flow on dynamic unstructured grids. In *AIAA Paper Series, Paper 09-3802*. 2009.

[125] E. Nielsen, E. Lee-Rausch, and W. Jones. Adjoint-based design of rotors using the Navier-Stokes equations in a noninertial reference frame. In *Proceedings of 65th American Helicopter Society annual forum, Grapevine, Texas USA*, 2009.

[126] E. Nielsen, J. Lu, M. Park, and D. Darmofal. An implicit exact dual adjoint solution method for turbulent flows on unstructured grids. In *AIAA Paper Series, Paper 2003-0272*. 2003.

[127] E. Nielsen and M. Park. Using an adjoint approach to eliminate mesh sensitivities in aerodynamic design. *AIAA Journal*, 44(5):948–953, 2006.

[128] M. Park. Three-dimensional turbulent RANS adjoint-based error correction. In *AIAA Paper Series, Paper 2003-3849*. 2003.

[129] M. Park. *Anisotropic output-based adaptation with tetrahdral cut cells for compressible flows*. PhD thesis, Massachusetts Intitute of Technology, September 2008.

[130] M. Park, J. Krakos, T. Michal, A. Loseille, and J. Alonso. Unstructured grid adaptation: status, potential, impacts and recommended investments toward CFD Vision 2030. In *AIAA Paper Series, Paper 2016-3323*. 2016.

[131] M. Park, E. Lee-Rausch, and C. Rumsey. FUN3D and CFL3D computations for the first high-lift prediction workshop. In *AIAA Paper Series, Paper 2011-936*. 2011.

[132] D. Pelletier, E. Turgeon, S. Etienne, and J. Borggaard. Reliable sensitivity analysis via an adaptive sensitivity equation method. In *AIAA Paper Series, Paper 2002-2578*. 2002.

[133] D. Pelletier, E. Turgeon, D. Lacasse, and J. Borggaard. Adaptivity, sensitivity and uncertainty: Towards standards in CFD. In *AIAA Paper Series, Paper 2001-0192*. 2001.

[134] J. Peter. DTP CAÏMEN phase 2B, année 3, tâche C4. Méthodes d'accélération de convergence multigrilles et implicites. Technical Report RT ONERA 2/03715, ONERA, 2001.

[135] J. Peter. Discrete adjoint method in elsA (part I): method/theory. In *Proceedings of the ONERA-DLR Aerospace Symposium (ODAS), Toulouse*, 2006.

[136] J. Peter. Calcul de la dérivée totale d'une fonction aérodynamique par rapport aux coordonnées de maillage. mode "pleinement adjoint" pour le calcul des gradients pour l'optimisation de forme. Technical Report RT 1/13624, ONERA, 2008.

[137] J. Peter and V. Brunet. DTP CAÏMEN phase 2B, année 3, tâche C1. Contrôle de la dissipation numérique. Technical Report RT ONERA 3/03715, ONERA, 2001.

[138] J. Peter and F. Drullion. Large stencil viscous flux linearization for the simulation of 3D turbulent compressible flows with backward-Euler schemes. *Computers and Fluids*, 36:1005–1027, 2007.

[139] J. Peter and R. Dwight. Numerical sensitivity analysis for aerodynamic optimization: a survey of approaches. *Computers and Fluids*, 39:373–391, 2010.

[140] J. Peter and J.-A. Désidéri. Unstructured mesh adaptation for functional outputs. With application to two-dimensionnal inviscid flows. In *AIAA Paper Series, Paper-2016-1930*. 2016.

[141] J. Peter, M. Lazareff, and V. Couaillier. Verification, validation and error estimation in cfd. *International Journal of Engineering Systems Modelling and Simulation*, 2:75–86, 2010.

[142] J. Peter, M. Marcelet, and S. Burguburu. Introduction to shape design in aerodynamics and some applications. Technical Report Publication ONERA 2006-01, ONERA, 2006.

[143] J. Peter, M. Nguyen-Dinh, and P. Trontin. Goal oriented mesh adaptation using total derivative of aerodynamic functions with respect to mesh coordinates. In *AIAA Paper Series, Paper 2012-0158*. 2012.

[144] J. Peter, M. Nguyen-Dinh, and P. Trontin. Goal-oriented mesh adaptation using total derivative of aerodynamic functions with respect to mesh coordinates – with application to Euler flows. *Computers and Fluids*, 66:194–214, 2012.

[145] J. Peter, F. Renac, A. Dumont, and M. Méheut. Discrete adjoint method for shape optimization and mesh adaptation in the $elsA$ code. status and challenges. In *Proceedings of 50th 3AF Symposium on Applied Aerodynamics, Toulouse*, 2015.

[146] J. Peter, F. Renac, and P. Trontin. Développement du logiciel elsA en mode pleinement adjoint. deuxième partie. Technical Report RT 1/14611, ONERA, 2009.

[147] J. Peter, P. Trontin, and M. Nguyen-Dinh. Goal-oriented mesh adaptation using total derivative of aerodynamic functions with respect to mesh coordinates. In *AIAA Paper Series, Paper-2011-30*. 2011.

[148] C.-T. Pham. *Linéarisation du Flux Visqueux des Equations de Navier-Stokes et de Modèles de Turbulence pour l'Optimisation Aérodynamique en Turbomachines*. PhD thesis, L'Ecole Nationale Supérieure d'Arts et Métiers, September 2006.

[149] N. Pierce and M. Giles. Adjoint recovery of superconvergent functionals from PDE approximations. *SIAM Review*, 42(2):247–264, 2000.

[150] X. Pinel and M. Montagnac. Block Krylov methods to solve adjoint problems in aerodynamic design optimization. *AIAA Journal*, 51(9):2183–2191, 2013.

[151] O. Pironneau. On optimum profiles in Stokes flow. *Journal of Fluid Mechanics*, 59(1):117–128, 1973.

[152] J. Ponsin, A. Coloto, E. Andrès, P. Bitrian, and C. Lozano. Implementation of an adjoint-based error estimation and grid adaptation module in the DLR TAU code. In *Proceedings of ICAS 2010*, 2010.

[153] P. Prieur and W. Splettstoesser. ERATO an ONERA-DLR cooperative program on aeroacoustic optimization. In *Proceedings of 25th European Rotorcraft Forum, september 1999, Roma, Italy*, 1999.

[154] F. Renac, C.-T. Pham, and J. Peter. Sensitivity analysis for the RANS equations coupled with a linearized turbulence model. In *AIAA Paper Series, Paper 2007-3948*. 2007.

[155] J. Reneaux. Méthode de définition de profils par optimisation numérique. *La Recherche Aérospatiale*, 5:303–321, 1984.

[156] J. Reneaux and J. Thibert. The use of numerical optimization for airfoil design. In *AIAA Paper Series, Paper 85-5026*. 1985.

[157] A. Resmini, J. Peter, and D. Lucor. Mono-block and non-matching multi-block structured mesh adaptation based on aerodynamic functional derivatives for RANS flows. *International Journal for Numerical Methods in Fluids*, 83:866–884, 2017.

[158] J. Reuther, A. Jameson, M. Remlinger, and D. Saunders. Constrained multi-point aerodynamic shape optimization using an adjoint formulation and parallel computers, Part I. *Journal of Aircraft*, 36(1):51–60, 1999.

[159] J. Reuther, A. Jameson, M. Remlinger, and D. Saunders. Constrained multi-point aerodynamic shape optimization using an adjoint formulation and parallel computers, Part II. *Journal of Aircraft*, 36(1):61–74, 1999.

[160] P. Roe. Approximate Riemann solvers, parameters vectors, and difference schemes. *Journal of Computational Physics*, 43:292–306, 1983.

[161] G. Rogé and L. Martin. Goal-oriented anisotropic grid adaptation. *Comptes rendus à l'Académie des Sciences – Mathématique*, 346(I):1109–1112, 2008.

[162] S. Schmidt, C. Ilic, V. Schultz, and N. Gauger. Three dimensional large scale aerodynamic shape optimization based on shape calculus. *AIAA Journal*, 51(11):2615–2627, 2013.

[163] H. Sharp and L. Sirovitch. Constructing a continuous parameter range of computational flows. *AIAA Journal*, 27(10):1326–1331, 1989.

[164] G. Shubin. Obtaining cheap optimization gradients from computational aerodynamics codes. Technical Report AMS-TR-164, Boeing Computer Service, Applied Mathematics and Statistics, June 1991.

[165] G. Shubin and P. Frank. A comparison of implicit gradient approach and the variational approach to aerodynamic design optimization. Technical Report AMS-TR-163, Boeing Computer Service, Applied Mathematics and Statistics, June 1991.

[166] W. Splettstoesser, van der Wall B., B. Junker, K. Schultz, Y. Delrieux, and P. Crozier. Wind tunnel test results and proof of design for an aeroacoustically optimized rotor. In *Proceedings of 25th European Rotorcraft Forum, september 1999, Roma, Italy*, 1999.

[167] A. Stück and T. Rung. Adjoint RANS with filtered shape derivatives for hydrodynamic optimization. *Computers and Fluids*, 47:22–32, 2011.

[168] J. Swanson and E. Turkel. On central-difference and upwind schemes. *Journal of Computational Physics*, 101:292–306, 1992.

[169] A. Taylor III, V. Korivi, and G. Hou. Sensitivity analysis applied to the Euler equations: A feasibility study with emphasis on variation of geometric shape. In *AIAA Paper Series, Paper 91-0173*. 1991.

[170] G. Todarello. Goal oriented adaptation of unstructured meshes. application to finite volume methods. Technical report, TU-Delft. MSc Thesis, october 2014.

[171] G. Todarello, F. Vonck, S. Bourasseau, J. Peter, and J.-A. Désidéri. Finite-volume goal-oriented mesh-adaptation using functional derivative with respect to nodal coordinates. *Journal of Computational Physics*, 313:799–819, 2016.

[172] L. Tourrette. *ADIGMA. A European initiative on the development of adaptative higher-order methods variational methods for aerospace applications*. Springer, 2010.

[173] G. van Albada, B. van Leer, and W. Robers Jr. A comparative study of computational methods in cosmis gas dynamics. *Astronomy and Astrophysics*, 108:76–84, 1982.

[174] van Leer B. Towards the ultimate conservative difference scheme. V – a second order sequel to godunov's method. *Journal of Computational Physics*, 32:101–136, 1979.

[175] G. Vanderplaats. CONMIN - a fortran program for constrained function minimization - user's manual. Technical Report TM-X-62282, NASA, 1973.

[176] G. Vanderplaats. *Numerical optimization techniques for engineering design: with applications*. McGraw Hill, 1984.

[177] G. Vanderplaats and R. Hicks. Numerical airfoil optimization using a reduced number of design coordinates. Technical Report TM-X-73151, NASA, 1976.

[178] M. Vasquez, A. Dervieux, and B. Koobus. Aerodynamical and sonic boom optimization of a supersonic aircraft. Technical Report RR 4520, INRIA, 2002.

[179] J. Vassberg, N. Harrison, D. Roman, and A. Jameson. A systematic study on the impact of dimensionality for a two-dimensional aerodynamic optimization model problem. In *AIAA Paper Series, Paper 2011-3176*. 2011.

[180] J. Vassberg and A. Jameson. In pursuit of grid convergence for two-dimensional Euler solutions. *Journal of Aircraft*, 47(4):1152–1166, 2010.

[181] D. Venditti and D. Darmofal. Adjoint error estimation and grid adaptation for functional outputs: Application to quasi-one-dimensional flow. *Journal of Computational Physics*, 164:204–227, 2000.

[182] D. Venditti and D. Darmofal. Grid adaptation for functional outputs: Application to two-dimensional inviscid flows. *Journal of Computational Physics*, 176:40–69, 2002.

[183] D. Venditti and D. Darmofal. Anisotropic grid adaptation for functional outputs: Application to two-dimensional viscous flows. *Journal of Computational Physics*, 187:22–46, 2003.

[184] G. Warren, W. Anderson, J. Thomas, and S. Krist. Grid convergence for adaptative methods. In *AIAA Paper Series, Paper 1991-1952*. 1991.

[185] M. Widhalm, J. Brezillon, and T. Leicht. Investigation of adjoint based gradient computation for realistic 3D aero-optimization. In *AIAA Paper Series, Paper 2010-9129*. 2010.

[186] S. Xu, D. Radford, M. Meyer, and J.-D. Müller. Stabilisation of discrete steady adjoint solvers. *Journal of Computational Physics*, 299:175–195, 2015.

[187] N. Yamaleev, B. Diskin, and K. Pathak. Error minimization via adjoint-based anistropic grid adaptation. In *AIAA Paper Series, Paper 2010-4436*. 2010.

[188] W. Yamazaki, S. Mouton, and G. Carrier. Geometry parameterization and computational mesh deformation by physics-based direct manipulation approaches. *AIAA Journal*, 48(8):1817–1834, 2010.

[189] L. Yano and D. Darmofal. An optimization-based framework for anisotropic simplex mesh adaptation. *Journal of Computational Physics*, 231:7626–7649, 2012.

[190] S. Yoon and A. Jameson. A lower-upper symmetric Gauss-Seidel method for the Euler and Navier-Stokes equations. *AIAA Journal*, 26(9):1025–1026, 1988.

[191] B. Zhou, T. Albring, N. Gauger, T. Economon, F. Palacios, and J. Alonso. A discrete adjoint framework for unsteady aerodynamic and aeroacoustic optimization. In *AIAA Paper Series, Paper 2015-3355*. 2015.

[192] G. Zoutendijk. *Methods of feasible direction*. Elsevier, 1960.

# Appendix

## Annex A – Publications

### Rank A journal articles

J. Peter. Non linear implicit scheme using Newton's method for the numerical solution of the Navier-Stokes equations. *Aerospace Science and Technology* 3:157-166 (1998)

J. Peter and F. Drullion. Large stencil viscous flux linearization for the simulation of 3D turbulent compressible flows with backward-Euler schemes. *Computers & Fluids* 36 (2007)

J. Peter and R.P. Dwight. Numerical sensitivity analysis for aerodynamic optimization: a survey of approaches. *Computers and Fluids* 39 (2010)

A. Dumont, A. Le Pape, J. Peter, S. Huberson. Aerodynamic shape optimization of hovering rotors using a discrete adjoint of the Reynolds-Averaged Navier-Stokes equations. *Journal of the American Helicotper Society* 56 (2011)

J. Peter, M. Nguyen-Dinh, P. Trontin. Goal-oriented mesh adaptation using total derivative of aerodynamic functions with respect to mesh coordinates – With application to Euler flows. *Computers and Fluids* 66 (2012)

M. Nguyen-Dinh, J. Peter, R. Sauvage, M. Meaux, J.-A. Désidéri. Mesh quality assessment based on aerodynamic functional output total derivative. *European Journal of Mechanics B/Fluids* 45 (2014)

A. Resmini, J. Peter, D. Lucor. Sparse grids-based stochastic approximations with applications to aerodynamics sensitivity analysis. *International Journal for Numerical Methods in Engineering* 106:32-57 (2016)

G. Todarello, F. Vonck, S. Bourasseau, J. Peter, J.-A. Désidéri. Finite-volume goal-oriented mesh-adaptation using functional derivative with respect to nodal coordinates. *Journal of Computational Physics* 313 (2016)

A. Resmini, J. Peter, D. Lucor. Mono-block and non-matching multi-block structured mesh adaptation based on aerodynamic functional derivatives for RANS flows. *International Journal for Numerical Methods in Fluids* 83 (2017)

### Other journal articles

F. Drullion, J. Peter. Efficient relaxation techniques using alternation of relaxation operator for the three dimensional Reynolds-Averaged Navier-Stokes equations. *Advances and Applications in Fluid Mechanics* 2(2):195-212 (2007)

M. Marcelet, J. Peter, G. Carrier. Sensitivity analysis of a strongly coupled aero-structural system using the discrete direct and adjoint methods *Revue Européene de Mécanique Numérique* Volume 17(8):1077-1106 (2008)

J. Peter, M. Marcelet. Comparison of surrogate models for turbomachinery design. *WSEAS Transactions on Fluid Mechanics* 3(1) (2008)

J. Peter, C.-T. Phâm, F. Drullion, Recent CFD activities at ONERA : Implicit stages for complex flow simulations and gradient computation for shape optimization. *Advances and Applications in Fluid Mechanics* 5(1):1-25 (2009)

J. Peter, M. Lazareff, V. Couaillier. Verification, validation and error estimation in CFD for compressible flows. *International Journal of Engineering Systems Modelling and Simulation*, 2(1/2):75-86 (2010)

J. Peter, G. Carrier, D. Bailly, P. Klotz, M. Marcelet, F. Renac. Local and global search methods for design in aeronautics. *Aerospace Lab* Issue 2 "CFD platforms and coupling" (2011)

A. Chkifa, A. Cohen, P.-Y. Passagia, J.Peter. A comparative study between Kriging and adaptative sparse tensor product methods for high-dimensional approximation and optimization problems in aeronautics design. *ESAIM:proceedings* 48:248-261 (2015)

### Recent communications

J. Dumont–Le Brazidec, J. Peter. *Strongly nested 1D interpolatory quadrature and extension to nD via Smolyak method.* ECCOMAS 2018 (Glasgow)

J. Peter, S. Goertz, R. Graves *Three-parameter uncertainty quantification for generic missile FG5.* 55th AIAA Aerospace Sciences 2017 (Dallas)

J. Peter. J.-A. Désidéri. *Unstructured mesh adaptation for functional outputs. With application to two dimensionnal inviscid flows.* 54th AIAA Aerospace Sciences 2016 (San Diego)

J. Peter, F. Renac, A. Dumont, M. Méheut. *Discrete adjoint method for shape optimization and mesh adaptation in the elsA code. Status and challenges.* Congrès 3AF 2015 (Toulouse)

J. Peter. *A fully nested interpolatory quadrature defined from Fejér's second rule.* ECCOMAS CFD 2014 (Barcelone)

## Annex B – Teaching and supervising activities

### Teaching

*Generalized polynomial chaos and stochastic collocation methods for uncertainty quantification in aerodynamics.* J. Peter, E. Savin.
VKI lecture series STO-AVT-326 organized in VKI by Th. Magin (2018) and at Stanford University by Catherine Gorle (2019).

*Vérification des simulations numériques, raffinement de maillage ciblé.*
*Propagation d'incertitudes en simulation aérodynamique.*
FMA03 professionnal training organized by V. Couaillier (2013, 2014, 2016, 2017, 2018)

*Non-intrusive uncertainty quantification methods. With application to turbomachinery and aircraft flows.* ERCOFTAC Course on Uncertainty quantification (three sessions. 2011 and 2012)

*Vérification des simulations numériques en mécanique des milieux continus. Notions de validation.* Professional training co-organized with J.-A. Désidéri for Collège Polytechnique (2010, 2011 et 2012)

*Local search methods for design in aeronautics.* M. Bompard, J. Peter.
Chapter of the RTO-AVT-167 course on *Strategies for optimization and automated design of gas turbine engines.* ISBN 978-92-837-0124-8. Lecture given at Concordia University Montreal (October 2009) and

Berlin University (September 2010)

*Introduction à l'optimisation de formes aérodynamiques et quelques exemples d'application.* J. Peter. Lecuture for MFN summer school. Fréjus 1st to 7th June 2003.
Extended lecture notes *Introduction à l'optimisation de formes aérodynamiques et quelques exemples d'application.* J. Peter, M. Marcelet, S. Burguburu. Publication ONERA 2006-01.

*Introduction à la Mécanique des fluides.* D. Sipp, J. Peter.
Fondation EPF, école d'ingénieurs. Third year course. (2003/2004 to 2007/2008)

### Supervising

- ONERA Weekly/daily technical support & assistance to scientific supervision of PhD students

Frédérique Drullion. (PhD supervisor Rémi Abgrall). Définition et étude de systèmes linéaires pour la simulation et l'optimisation de formes aérodynamiques. (defended November 2005)

Chi-Tuân Pham. (PhD supervisor Alain Lerat). Linéarisation du flux visqueux des équations de Navier-Stokes et de modèles de turbulence pour l'optimisation aérodynamique en turbomachines. (defended Septembre 2006)

Meryem Marcelet. (PhD supervisor Alain Lerat. ONERA-supervision by G. Carrier and J. Peter) Calcul de gradient direct et adjoint en aéroélasticité pour l'optimisation de forme. (defended December 2008)

Antoine Dumont (PhD supervisor Serge Huberson. ONERA-supervision by J. Peter and A. Le Pape). Optimisation de pales d'hélicoptère en vol stationnaire par méthode de gradient. (March 2010)

Manuel Bompard. (PhD supervisor Jean-Antoine Désidéri). Meta-modèles utilisant l'information de gradient, exacte ou imprécise, pour l'optimisation globale de formes aérodynamiques. (defended December 2011)

- Co-supervision of PhD students

Maxime Nguyen-Dinh. (PhD co-supervisor J.-A. Désidéri) Qualification de simulations numériques par adaptation anisotropique de maillages. Université de Nice-Sophia Antipolis. (defended March 2014)

Sébastien Bourasseau. (PhD co-supervisor J.-A. Désidéri) Contribution à une méthode de raffinement de maillage basée sur le vecteur adjoint pour le calcul de fonctions aérodynamiques. Université de Nice-Sophia Antipolis. (defended December 2015)

Andrea Resmini. (PhD co-supervisor D. Lucor) Sensivitiy analysis for numerical simulation of compressible flows in external aerodynamics. Université Pierre et Marie Curie. (defended December 2015)

- Member of PhD jurys

Member of the PhD jury of the students (co-)supervised at ONERA, Frédérique Drullion, Chi-Tuân Pham, Meryem Marcelet, Antoine Dumont, Manuel Bompard, Maxime Nguyen-Dinh, Sébastien Bourasseau, Andrea Resmini.

Member of the PhD jury of Jan Van Langenhove. *Adaptive control of deterministic and stochastic approximation errors in simulations of compressible flow*, defended November 25th 2017.

## Annex C – Contribution to EU projects and S&T groups

- NODESIM-CFD. Non-deterministic simulation for CFD-Based Design methodologies (November 2006 – February 2010). Contributor.

- UMRIDA Uncertainty Management for Robust Industrial Design in Aeronautics (October 2013 – September 2016). Contributor

- ITN ANADE Advances in Numerical and Analytical tools for DEtached flow prediction (January 2012 – December 2015). Supervisor of one student. Contributor to one of the teaching sessions.

- S& T-AVT 167. Strategies for Optimization and Automated Design of Gas Turbine Engines (2007–2010). Responsible for the teaching of local search methods.

- S& T-AVT 191 Sensitivity Analysis and Uncertainty Quantification Methods for Military Vehicle Design (2011 – 2015). Responsible of the external aero group.

## Annex D – Organization of scientific events

- ONERA's Symposium on approximation and uncertainty in flow simulation
  ONERA Châtillon. December, 3rd, 2010
  Approximately 40 participants
  Organizer and scientific coordinator

- ERCOFTAC course on Mathematical methods and tools in uncertainty management and quantification
  ONERA Châtillon. October, 24&25, 2013
  Approximately 30 participants
  Local coordinator

- ERCOFTAC course on LBM for Indusstrial Applications. Overview, guidance and examples
  ONERA Châtillon. June, 13&14, 2016
  Approximately 30 participants
  Local coordinator

## Annex E – Curriculum Vitae

| | |
|---|---|
| 1989-1992 | Ecole Polytechnique (X89) |
| 1992-1993 | Master in Numerical Analysis (Paris 6 University). Magna cum laude distinction (mention Bien) |
| 1993-1996 | PhD thesis at ONERA (SNECMA CIFRE funding) |
| 1997 | Dassault Systemes. Maths & Geometry group. Engineer |
| 1998- | ONERA / Departement of Aerodynamics, Aeroelasticity & Aeroacoustics<br>Engineer. Senior research fellow (Maître de Recherche)<br>Departement's scientific coordinator for multifidelity optimization |

# Contributions to discrete adjoint method in aerodynamics for shape optimization and goal-oriented mesh-adaptation

This document sums up the researches conducted in Computational Fluid Dynamics, based on discrete adjoint of finite-volume schemes, for shape optimization and for goal-oriented mesh adaptation.

Concerning shape optimization, the researches were devoted to the adaptation of backward-Euler implicit stages to the FPI resolution of the discrete adjoint equation, to the adaptation of a basic external flow adjoint module to the mechanical formulation suitable for the simulation of rotors in hovering flight and to the optimization of airfoils using a smoothed shape gradient.

Concerning goal-oriented mesh adaptation, an adaptation criterion based on the total derivative of the goal w.r.t. volume mesh node coordinates has been proposed. This criterion includes a projection of this vector field and the local characteristic size of the current mesh. Successfull mesh adaptation based this criterion were conducted for various types of meshes (structured/unstructured) and flows (Euler and (RANS)), in 2D and 3D.

**Keywords :** AERODYNAMICS ; NUMERICAL SIMULATION ; MESH ADAPTATION ; DISCRETE ADJOINT ; SHAPE OPTIMISATION

# Contributions à la méthode adjointe discrète pour l'optimisation de forme et le raffinement de maillage ciblé

Le manuscrit résume les travaux de recherche s'appuyant sur la méthode adjointe discrète en simulation aérodynamique pour l'optimisation de forme et pour le raffinement de maillage ciblé.

Dans le domaine de l'optimisation de forme, les travaux ont porté sur l'adaptation de phases implicites Euler-rétrograde pour la résolution de l'équation adjointe discrète par méthode de point fixe, sur l'adaptation des discrétisations et formulations usuelles à la formulation mécanique utilisée pour la simulation de rotors en vol stationnaire et sur l'optimisation de profils par gradient de forme lissé.

Dans le domaine de l'adaptation de maillage ciblée, on a proposé un critère d'adaptation basé sur la dérivée totale de la fonction d'intérêt par rapport aux coordonnées du maillage volumique. Ce critère fait intervenir une projection de ce champ vectoriel et la taille locale du maillage courant. Des adaptations de maillage ont été conduites avec succès selon ce critère pour différents types de maillages (structuré/non structuré) et d'écoulements (régi par les équations d'Euler ou RANS), en 2D et en 3D.

**Mots clés :** AERODYNAMIQUE ; SIMULATION NUMERIQUE ; MAILLAGE ADAPTATIF ; ADJOINT ; OPTIMISATION FORME

ONERA
THE FRENCH AEROSPACE LAB

BP 80100 - Chemin de la Hunière FR-91123 PALAISEAU CEDEX - Tél. : +33 1 80 38 60 60 - Fax : +33 1 80 38 65 10

www.onera.fr