Sensitivity analysis for numerical simulation of compressible flows in external aerodynamics.

Andrea Resmini

COLLÈGE DOCTORAL DE SORBONNE UNIVERSITÉS
École doctorale sciences mécaniques, acoustique, électronique & robotique de Paris

THÈSE

pour obtenir le grade de
Docteur en Sciences
de l'Université Pierre et Marie Curie
Spécialité : Mécanique de fluides

Présentée par

Andrea RESMINI

Sujet de la thèse :

Analyse de sensibilité pour la simulation numérique des écoulements compressibles en aérodynamique externe

soutenue le 11 Décembre 2015 devant le jury composé de :

M.	G. Biau	Examinateur	Professeur, UPMC
M.	P.-M. Congedo	Rapporteur	Chercheur-HDR, INRIA Bordeaux
M.	P. Larrieu	Examinateur	Chargé méthodes numériques, AIRBUS
M.	D. Lucor	Directeur de thèse	Directeur de recherche CNRS, LIMSI
M.	J. Peter	Encadrant	Ingénieur de recherche, ONERA
Mme.	M.-V. Salvetti	Examinateur	Professeur, Università di Pisa
M.	E. Valero	Examinateur	Professeur, Univ. Politécnica de Madrid
M.	M. Visonneau	Rapporteur	Directeur de recherche CNRS, ÉCN

Résumé : Analyse de sensibilité pour la simulation numérique des écoulements compressibles en aérodynamique externe.

L'analyse de sensibilité pour la simulation numérique des écoulements compressibles en aérodynamique externe par rapport à la discretization de maillage et aux incertitudes liées à des paramètres d'entrées du modéle a été traitée 1- par le moyen des méthodes adjointes pour le calcul de gradient et 2- par approximations stochastiques non-intrusives basées sur des grilles creuses. 1- Une méthode d'adaptation de maillages *goal-oriented* basée sur les dérivées totales des fonctions aérodynamiques d'interêt par rapport aux nœuds du maillage a été introduite sous une forme améliorée. La méthode s'applique au cadre de volumes finis pour des écoulements RANS pour des maillages mono-bloc et multi-bloc structurés. Des applications 2D pour des écoulements transsoniques ainsi que subsonique détaché atour d'un profil pour l'estimation du coefficient de traînée sont présentées. L'apport de la méthode proposée est vérifié. Les maillages anisotropes obtenus arrivent à bien capturer l'écoulement et à estimer les fonctions aérodynamiques globales. Les contraintes imposées par les maillages structurés sont assouplies par l'usage de raccords non-coïncidents qui limite la propagation d'un raffinement local sur tout le domaine. 2- Les méthodes du polynôme de chaos généralisé sous forme pseudospectrale creuse et de la collocation stochastique construite sur des grilles creuses isotropes et anisotropes sont examinées. Les maillages anisotropes sont obtenus par le biais d'une méthode adaptive basée sur l'analyse de sensibilité globale. L'efficacité des ces approximations est testée avec des fonctions test et des écoulements aérodynami-ques visqueux autour d'un profil en présence d'incertitudes géométriques et opéra-tionnelles. Pour des grilles creuses isotropes, les deux méthodes montrent des niveaux de performance similaires. Les atouts de la méthode proposée de collocation stochastique avec adaptivité basée sur les indices de Sobol' sont vérifiés mais restent dépendent du problème étudié. L'integration des méthodes et aboutissements 1- et 2- dans une approche couplée permettrait de contrôler de façon équilibrée l'erreur déterministe/stochastique *goal-oriented*.

Mots-clés:

analyse de sensibilité, maillage structuré, adaptation, adjoint, RANS, non-matching, multi-bloc, profil, quantification d'incertitude, grilles creuses, approximations stochastiques pseudospectral, collocation stocastique, adaptivité, aérodynamique, écoulement compressible.

Abstract: Sensitivity analysis for numerical simulation of compressible flows in external aerodynamics.

Sensitivity analysis for the numerical simulation of external aerodynamics compressible flows with respect to the mesh discretization and to the model input parametric uncertainty has been addressed respectively 1- through adjoint-based gradient computation techniques and 2- through non-intrusive stochastic approximation methods based on sparse grids. 1- An enhanced goal-oriented mesh adaptation method based on aerodynamic functional total derivatives with respect to mesh coordinates in a RANS finite-volume mono-block and non-matching multi-block structured grid framework is introduced. Applications to 2D RANS flow about an airfoil in transonic and detached subsonic conditions for the drag coefficient estimation are presented. The asset of the proposed method is patent. The obtained anisotropic meshes well capture flow features as well as global aerodynamic functionals. The constraints imposed by structured grid are relaxed by the use of non-matching multi-block approach which limits the convection of local mesh refinement through all the computational domain. 2- The generalized Polynomial Chaos in its sparse pseudospectral form and stochastic collocation methods based on both isotropic and dimension-adapted sparse grids obtained through an improved dimension-adaptivity method driven by global sensitivity analysis are considered. The stochastic approximations efficiency is assessed on multi-variate test functions and airfoil viscous aerodynamics simulation in the presence of geometrical and operational uncertainties. Sparse pseudospectral and collocation approximations exhibit similar level of performance for isotropic sparse simulation ensembles. Computational savings and accuracy gain of the proposed adaptive stochastic collocation driven by Sobol' indices are patent but remain problem-dependent. Integration of achievements 1- and 2- into a coupled approach in future work will pave the way for a well-balanced goal-oriented deterministic/stochastic error control.

Keywords:

sensitivity analysis, structured mesh, adaptation, adjoint, RANS, non-matching, multi-block, airfoil, uncertainty quantification, sparse grids, pseudospectral stochastic approximations, stochastic collocation, adaptivity, aerodynamics, compressible flow.

Contents

Acknowledgements

I would like to address my gratitude to Dr. Lucor and Dr. Peter for carefully guiding me all along these three years of PhD. Dr. Lucor has been able to introduce me to the Uncertainty Quantification domain and his guide has been really precious during my research. I have to thank equally Dr. Peter for helping me on a daily basis at ONERA and especially for his experience both in adjoint methods and in the *elsA* code. Moreover, they have both been really pleasant and I am sure we will keep a very good personal relationship in the future.

I thank Dr. Couaillier for letting me be a part of the Numerical Methods for Fluid Dynamics team in the CFD department at ONERA as well as the European project ITN ANADE, which has been carefully coordinated by Prof. Valero. In this framework, I had the chance to be seconded to the University of Cambridge and I would to like express my gratitude to Dr. Juniper for welcoming me in his team for a few months.

The working environment has been extremely pleasant thanks to the other PhD students at ONERA, especially Khalil and Jean-Baptiste. Also, I have to mention my ITN ANADE colleagues which have made this research experience really valuable and special thoughts go to Andrea, Maria Chiara, Bastien, Dirk, David and Christophe.

Finally, i would like to express my gratitude to my family, which has always supported me and without whom I would have not reached this goal. And, of course, to my closest friends in Italy, France and all over Europe: Federica, Sara, Roberto, *Marcushof family*, both Giuseppe, Matteo, Alice, Nicolas and many more.

This PhD thesis research has been fully supported by the Marie-Curie ITN ANADE project funded through the European Commission FP7 under grant agreement N. 289428.

Funded by
the European Union

Overview

The fast development and availability of computational models and resources in various fields of science and engineering, such as Computational Fluid Dynamics (CFD), has led to a growing awareness of the researchers towards the reliability and limitations of their simulations. Several factors influence the quality of the CFD solution especially the numerical and space discretization, the choice of physical model and the model parametric data handling. This PhD thesis has been performed in the CFD department at ONERA - The French aerospace laboratory. Consistently with the ONERA missions, namely to bridge the gap between academia and aeronautical industries, it has been chosen to focus on the numerical side of an industrial finite volume compressible CFD solver (elsA) in an external aerodynamics framework. The sensitivity of the flow with respect to the mesh and to the model input values uncertainty has been addressed respectively through adjoint-based gradient computation techniques and through non-intrusive stochastic approximation methods based on sparse solution sampling.

Chapter 1 introduces some key CFD elements for external aerodynamics applications with a focus on airfoils. The exploited flow governing equations are introduced as well as the global aerodynamic functionals. A brief overview of Validation and Verification (V&V) is given through definitions of V&V processes as well as a short historical review of global aerodynamic functionals accuracy in experiments and numerical simulations is given. The interactions between V&V, Uncertainty Quantification (UQ) and sensitivity analysis have been equally introduced.

Gradient computations methods for local search methods in aeronautical design are introduced in Chapter 2. The current implementation in the *elsA* code is given including the new code development linked with this PhD thesis. Namely, the full linearization of the mean-flow viscous flow fluxes and the Spalart-Allmaras turbulence model. Examples given verify the implemented code modifications as well as their assets with respect to the gradient computation precision.

Chapter 3 presents the enhanced goal-oriented mesh adaptation method based on aerodynamic functional total derivatives with respect to mesh nodes in a RANS

finite-volume mono-block and non-matching multi-block structured grid framework. This method falls under the category of methods involving the adjoint vector of the function of interest. The contribution of Spalart-Allmaras turbulence model is taken into account through its linearisation. Meshes are adapted accordingly to the proposed indicator. Applications to 2D RANS flow about a RAE2822 airfoil in transonic and detached subsonic conditions for the drag coefficient estimation are presented.

Uncertainty Quantification methods are introduced in Chapter 4. Especially, the generalized Polynomial Chaos in its sparse pseudospectral form and stochastic collocation methods based on both isotropic and dimension-adapted sparse grids are considered. An improved version of a stochastic collocation with dimension-adaptivity driven by global sensitivity analysis is proposed. The UQ study compares sample-based polynomial surrogates, well-suited for moderately high- dimensional stochastic problems. The stochastic approximations efficiency is assessed on multi-variate test functions and airfoil aerodynamics simulations. The latter study addresses the probabilistic characterization of global aerodynamic coefficients derived from viscous subsonic steady flow about a NACA0015 airfoil in the presence of geometrical and operational uncertainties with both simplified aerodynamics model and RANS simulation.

Finally, conclusions and perspectives are given. The work done in this PhD study allows a better understanding and handling of the numerical issues linked to space discretization and input values uncertainty for external aerodynamic applications in an engineering framework. The main perspective for future work is a well-balanced goal-oriented deterministic/stochastic error control for CFD simulations.

Chapter 1

Introduction: elements of CFD and V&V

Computational engineering has successfully entered in both scientific research and engineering R&D. In particularly, for the prediction of the behaviour of fluids, Computational Fluid Dynamics (CFD) has undergone a significant expansion and improvement since the early 1970's. From that moment, several physical approximation and numerical methods have been developed giving birth to numerous CFD solvers. The evolution of computer capacity has eased both the daily use of CFD in the research community as well as the reduction of the overall costs and computational time. Often, in the aeronautical domain, experiments are costly and demanding. The possibility to obtain similar results through a fast and cheap CFD simulation is clearly attractive but it may lead to some reflections. Indeed, the *reliability* of these simulations need to be addressed and assessed. Verification and Validation (V&V) process consecrates to this objective.

The present chapter introduces some key CFD elements for external aerodynamics applications with a focus on airfoils. In Section 1.1, the exploited flow governing equations are introduced as well as the global aerodynamic functionals. A brief overview of Validation and Verification (V&V) is given in Section 1.2 through definitions of V&V processes as well as a short historical review of global aerodynamic functionals accuracy in experiments and numerical simulations is given. The interactions between V&V and Uncertainty Quantification (UQ) are highlighted in Section 1.3. In Section 1.4, the two main topics of this PhD thesis are also introduced: the flow sensitivity with respect to the mesh and to the input values.

Nomenclature

Abbreviations

AoA	Angle of attack
DES	Detached Eddy Simulation
DNS	Direct Numerical Simulation
CFD	Computational Fluid Dynamics
FV	Finite Volume
LES	Large Eddy Simulation
KW	$k-\omega$ turbulence model
QoI	Quantity of Interest
RANS	Reynolds-Averaged Navier-Stokes
SA	Spalart-Allmaras turbulence model
UQ	Uncertainty Quantification
V&V	Verification and Validation
WT	Wind Tunnel

Greek letters

δ	Error
Δ	Difference / scatter
μ	Dynamic viscosity
$\tilde{\mu}$	Mean value
$\tilde{\nu}$	Turbulent kinematic viscosity
ρ	Density
σ	Standard deviation (std)
$\boldsymbol{\tau}$	Viscous stress tensor

Latin letters

C	Global aerodynamic coefficient
C_D	Drag coefficient
C_L	Lift coefficient
C_m	Momentum coefficient
E	Total energy
E_C	Coefficient C error
$\boldsymbol{F}$	FV flux
k	Turbulent kinetic energy
M	Mach number
$\boldsymbol{n}$	Unit normal vector
p	Static pressure
$\boldsymbol{q}$	Heat flux
Re	Reynolds number
S	Contour of V
$\boldsymbol{\mathcal{S}}$	Turbulent source term
u	Uncertainty
$\boldsymbol{U}$	Velocity vector
V	Mesh cell volume
$\boldsymbol{W}$	Conservative variables vector

Superscripts

T	Transpose

Subscripts

$\boldsymbol{c}$	Convective
r	Reynolds component
T	Turbulent component
$\boldsymbol{v}$	Viscous / diffusive
∞	Farfield condition

1.1 RANS equations and aerodynamic functions estimation

The Reynolds-Averaged Navier-Stokes (RANS) model is generally not suited for the simulation of complex flow, nevertheless it can predicts the general flow features as well as forces and moments for particular configuration such as isolated airfoils. The corresponding solution is particularly sensible to the space discretization of the flow domain and to the choice of the turbulence model. The RANS equations where the turbulence closure is ensured by the Spalart-Allmaras (SA) turbulence model [1] are extensively used both for code development and numerical applications in this PhD thesis. For this reason, the RANS equations system is given hereafter already coupled with the SA model.

1.1.1 RANS + SA equations

In this thesis framework, this equations system is discretized with the Finite-Volume (FV) scheme. Given a bounded domain $\Omega \subset \mathbb{R}^3$ and a mesh cell $V \subset \Omega$ with contour S, the equations read as follow

$$\frac{\partial}{\partial t}\int_V \boldsymbol{W}\mathrm{d}V + \oint_{\partial V}(\boldsymbol{F_c}+\boldsymbol{F_v})\cdot\boldsymbol{n}\mathrm{d}S - \int_V \boldsymbol{\mathcal{S}}\mathrm{d}V = 0 \tag{1.1}$$

where $\boldsymbol{W} = (\rho, \rho\boldsymbol{U}, \rho E, \rho k, \rho\tilde{\nu})^T$ is the conservative variables vector[1] with ρ the density, $\boldsymbol{U} = (u, v, w)^T$ the velocity vector, $E = p/(\gamma-1)\rho + \boldsymbol{U}^2/2$ the total energy, p the static pressure and $\tilde{\nu}$ the kinematic turbulent viscosity which is defined by the turbulence model, in this case SA. The unit normal vector to ∂V is denoted $\boldsymbol{n}$. The convection $\boldsymbol{F_c}$ and viscous $\boldsymbol{F_v}$ fluxes are hereafter defined:

$$\boldsymbol{F_c} = \begin{pmatrix} \rho\boldsymbol{U}^T \\ \rho\boldsymbol{U}\boldsymbol{U}^T + p\boldsymbol{I} \\ (\rho E + p)\boldsymbol{U}^T \\ \rho\tilde{\nu}\boldsymbol{U}^T \end{pmatrix}, \quad \boldsymbol{F_v} = \begin{pmatrix} 0 \\ -\boldsymbol{\tau} - \boldsymbol{\tau_r} \\ -(\boldsymbol{\tau}+\boldsymbol{\tau_r})\boldsymbol{U} + \boldsymbol{q}^T + \boldsymbol{q_T}^T \\ -\frac{1}{\sigma}(\mu + \rho\tilde{\nu})\nabla\tilde{\nu}^T \end{pmatrix}. \tag{1.2}$$

The state equation is given by the perfect gas law $p = \rho RT$ where T is the temperature and R is the gas constant. The viscous stress tensor and the heat flux read as follow

$$\begin{aligned} \boldsymbol{\tau} &= -\frac{2}{3}\mu(\nabla\cdot\boldsymbol{U})\boldsymbol{I} + \mu(\nabla\boldsymbol{U} + \nabla\boldsymbol{U}^T), \\ \boldsymbol{q} &= -\frac{C_p\mu}{Pr}\nabla T, \end{aligned}$$

where μ is the dynamic viscosity defined by the Sutherland law and $Pr = 0.72$ is the Prandtl number. The specific heat constant under constant pressure is $C_p = \gamma R/(\gamma-1)$ with γ as the specific heat ratio. The Reynolds stress tensor and the turbulent heat flux are obtained through the Boussinesq hypothesis:

$$\begin{aligned} \boldsymbol{\tau_r} &= -\frac{2}{3}\mu_T(\nabla\cdot\boldsymbol{U})\boldsymbol{I} + \mu_T(\nabla\boldsymbol{U} + \nabla\boldsymbol{U}^T), \\ \boldsymbol{q_T} &= -\frac{C_p\mu_T}{Pr_T}\nabla T, \end{aligned}$$

with $Pr_T = 0.72$. The turbulent dynamic viscosity μ_T is given by the following relation of the SA model:

$$\mu_T = \mu\chi f^{v1}. \tag{1.3}$$

For details concerning the last relation and on the definition of the source term $\boldsymbol{\mathcal{S}}$ of equation (1.1), the reader is suggested to refer to Section 2.5 where the SA

[1] The superscript T defined the transpose operator while the subscript T the turbulent component.

model is detailed.

As mentioned earlier, the equations are discretized through FV cell-centred second order schemes on multi-block structured mesh in the ONERA *elsA* code [2]. The main scheme used in this work is the Roe's one [3]. For the convection term in particular, a second-order MUSCL scheme [4] with van Albada limiter [5] is used. Nevertheless, it is beyond the objectives of this thesis to review and recall all the schemes deployed in the *elsA* code. The reader may refer to [2] for more details.

1.1.2 Aerodynamic functions

CFD in an external aerodynamics framework needs not only to compute the flow but also to predict the so-called global aerodynamic functions. These functionals are important in applied context in order to estimate the performance of the studied solid object configuration, *i.e.* the effectiveness of the designed product. Typically, for aeronautical applications, information on the lift and drag forces is essential for this kind of assessment. It is then common to attach to airfoils (and even to entire aircraft configurations) coefficients which describes their aerodynamic efficiency, *e.g.* mainly the lift C_L and drag C_D coefficients but many other kind of coefficients exist.

These coefficients may be computed using a near-field or far-field approach [6, 7] where the first one reflects the body's perspective while the second one the flowfield perspective. The near-field approach needs the integration of forces over the solid shape of the aerodynamic object ($\mathcal{O}$ with surface S) into consideration. The lift and drag coefficients may be defined as follow:

$$C_L = \underbrace{\frac{\oint_{\partial\mathcal{O}}(p-p_\infty)n_\zeta \mathrm{d}S}{\frac{1}{2}\rho_\infty||\boldsymbol{U}_\infty||^2 S_{ref}}}_{\text{pressure component } (C_{L_p})} \underbrace{-\frac{\oint_{\partial\mathcal{O}}(\boldsymbol{\tau}_\zeta \cdot \boldsymbol{n})\mathrm{d}S}{\frac{1}{2}\rho_\infty||\boldsymbol{U}_\infty||^2 S_{ref}}}_{\text{friction component } (C_{L_f})}, \tag{1.4}$$

$$C_D = \underbrace{\frac{\oint_{\partial\mathcal{O}}(p-p_\infty)n_\chi \mathrm{d}S}{\frac{1}{2}\rho_\infty||\boldsymbol{U}_\infty||^2 S_{ref}}}_{\text{pressure component } (C_{D_p})} \underbrace{-\frac{\oint_{\partial\mathcal{O}}(\boldsymbol{\tau}_\chi \cdot \boldsymbol{n})\mathrm{d}S}{\frac{1}{2}\rho_\infty||\boldsymbol{U}_\infty||^2 S_{ref}}}_{\text{friction component } (C_{D_f})}, \tag{1.5}$$

where, χ and ζ are respectively the directions tangent and perpendicular to the farfield flow and S_{ref} is a reference surface .

The far-field approach allows the breakdown of the drag coefficient into many

other components:

$$C_D = \underbrace{C_{D_p} + C_{D_f}}_{\text{near-field}} = \underbrace{C_{D_w} + C_{D_i} + \overbrace{C_{D_{vp}} + C_{D_f}}^{C_{D_v}} + C_{D_{sp}}}_{\text{far-field}}, \tag{1.6}$$

where C_{D_w} is the wave drag component (from shock waves), C_{D_i} is the induced drag component[2] (from trailing vortex wake), C_{D_v} is the viscous drag component which is composed by the viscous pressure $C_{D_{vp}}$ and friction C_{D_f} parts and finally $C_{D_{sp}}$ is the spurious drag component due to numerical dissipation. For details on the computation of these coefficients, the reader is suggested to refer to Destarac [6].

The near-field coefficients are available directly from an *elsA* computation while the far-field ones need to be obtained from the ONERA *FFD72* code by using a converged *elsA* flow solution. For some applications of this thesis (*e.g.* in Section 4.4.3), the aerodynamic coefficients have been obtained through potential flow computations with the XFOIL code [8] where only a near-field analysis is available.

1.2 Verification & Validation

Definitions of Verification and Validation (V&V) are given in quality process documents which differ from one domain to another. Generally speaking, the *verification* consists in controlling the compliance of a product with respect to the specifications given before its development, while the *validation* consists in verifying the product capability to deliver services for which it has been designed for. Sections 1.2.1 and 1.2.2 will detail definitions and common practices in a CFD framework [9].

1.2.1 Verification of a CFD solver

Verification of a CFD solver consists in checking that the code correctly discretizes and solves the equations of the chosen mathematical model. In particularly, for a given configuration, it is controlled that the numerical solution obtained on a mesh hierarchy converges with the expected order to the solution given by the continuous differential problem while the discretization step tends to zero. Roache [10] has distinguished the 1- *code verification* and 2- *solution verification* in the verification process. This approach has been largely accepted and it may be found in the Roy's methods review [11].

[2]The induced drag component is assimilated to a spurious drag $C_{D_{sp}}$ for 2D analysis.

Code verification

The code verification process gathers all elementary checks of the CFD solver at a fixed discretization. Namely :

1. Unit tests.

2. Comparison of the physical properties of the discretized solutions, *i.e.* in CFD for a compressible perfect flow with uniform far-field condition, checks of the stagnation quantities and of the entropy in the flow field.

3. Comparison of the numerical solution with the continuous one if the latter is known. In this framework, the more the exact solution is complex, the more this verification technique is reliable[3]. The order precision of the numerical scheme may also be verified by comparing the exact solution with the numerical one obtained from different mesh refinement levels following Richardson's theory [12, 13].

4. If the continuous analytical solution is unknown, it is necessary to generate a *manufactured solution*. Namely, the numerical solution is interpolated and injected in the equations of the mathematical model for computing a source term. This manufactured solution is the solution of the mathematical problem defined by the original model where on the right-hand-side the source term appears. It is then possible to compare the numerical solution with the theoretical solution of the manufactured problem[4].

Solution verification

The solution verification process analyses the discretization error through order and mesh convergences. Roy [11] considered also the rounding and convergence errors. This checks may be performed based on:

1. Indicators and estimators available for finite element schemes.

2. Mesh hierarchy (minimum three meshes, four in most cases) in order to control the state variables convergence. The oldest method has been introduced by Richardson [12, 13]. If the convergence is not monotonic, Richardson theory fails. Nevertheless, it is possible to define the error bounds through heuristic formulae, *e.g.* CGI [10]. A complete verification study must include also the order convergence analysis based on regular and irregular meshes.

[3] In incompressible laminar flow simulation, check on the Poiseuille flow does ensure a reliable verification neither on time discretization nor on space disretization of the momentum equation along the ax perpendicular to the wall. The Lamb-Oseen flow ensures a more straightforward verification.

[4] For incompressible flow simulation, the reader can refer to Eça *et al.* [14]. Examples are given also in Roy [11].

3. Schemes with different orders. In this situation, it is necessary to check the behaviour of the solutions sequence for growing order, *e.g.* the rise of high-order methods (for instance Discontinuous Galerkin) in CFD makes this check quite common.

It is necessary to underline the need of both of these kind of verifications. Indeed, the solution verification is not able to detect errors given by wrong operation on physical constant (*e.g.* division by 2). On the other hand, it is possible to have errors detectable only on very well refined mesh used only during a solution verification as mentioned here above.

1.2.2 Validation with a CFD solver

The validation with a CFD solver consists in quantify to which extent the chosen model is a good representation of the real physical problem. The most used definition for validation reads as follows:

> *The process of determining the degree to which the model (and its associated data) is in accurate representation of the real world from the perspective of the intended uses of the model.*

Roache's contribution in [14] details this definition. Namely, the analysis is deepen for the following expressions:

1. *degree*: the requested solution quality is really domain-dependent;
2. *real world*: this turns to coincide with the experiments for most of V&V experts;
3. *intended use*: this expression rises some doubts. In particularly, may a code, which predicts accurately integral values for a specific application but not the overall variables field, considered as validated?

In order to deepen the discussion concerning the validation processes, it is necessary to define and distinguish between *error* and *uncertainty*. In this context, several meetings and documents have been done in order to enlighten these issues (*e.g.* AIAA Guide G-077-1998, ASME PTC 60 - V&V10, ASME PTC PTC 61 - V&V20). By referring to Coleman review in [14] and Roache [15]:

1. the *error* δ_i is a positive (or negative) real quantity equals to the difference between a computed (or measured) quantity and its value obtained from the corresponding physical problem.
2. the *uncertainty* u_i is a majoration of $|\delta_i|$, *i.e.* $\delta_i \in [-u_i, +u_i]$.

Other definitions of these two quantities are available. For instance, AIAA (AIAA-G-077-1998 [16]) terminology later completed by Oberkampf *et al.* (2002)[17] and Trucano *et al.* (2006)[18] defines:

1. the *error* as a recognizable deficiency in any phase or activity of the modeling process that is not due to the lack of knowledge.
2. the *uncertainty* as a potential deficiency in any phase or activity of the modeling process that is due to the lack of knowledge. It is also possible to distinguish between *aleatory* and *reducible or epistemic* uncertainty. The first one is a physical variability present in the system or its environment not strictly due to a lack of knowledge and cannot be reduced. The second one is a potential deficiency that is due to a lack of knowledge.

A validation approach

The following approach follows the guidelines in [14, 15]. The methodology is applicable for all mechanical problems where an experimental and a numerical results of the same problem are available. The approach is shown for the estimation of aerodynamic global effort through wind-tunnel and CFD simulations.

It is considered the ideal flight of an aircraft in a air-flow domain at a fixed Mach number, Reynolds number, incidence and sideslip angles. The objective of this study is the estimation of global aerodynamic coefficients, C, such as the lift or drag coefficients. This coefficient C may be obtained through wind-tunnel experiments (from hereafter noted as exp) and CFD simulations (noted as sim). The scatter between these two evaluations is $E_C = C_{exp} - C_{sim}$.

Introducing the unknown solution of the mechanical problem C_{ideal}, the error is given by

$$E_C = C_{sim} - C_{exp} = (C_{sim} - C_{ideal}) - (C_{exp} - C_{ideal}) = \delta_{sim} - \delta_{exp}. \tag{1.7}$$

Analyzing the term coming from the experimental side, it is possible to express δ_{exp} in the following fashion:

$$\delta_{exp} = C_{exp} - C_{ideal}, \tag{1.8}$$

$$\delta_{exp} = \delta_{mes} + \delta_{cond}, \tag{1.9}$$

$$|\delta_{exp}| \leq |\delta_{mes}| + |\delta_{cond}| \leq u_{mes} + u_{cond} = u_{exp}. \tag{1.10}$$

The term δ_{exp} represents then the limited capability of the measurement devices (δ_{mes}) and the disparity between ideal and experimental flow conditions (δ_{cond}, *e.g.* inability to produce desired flow conditions, devices altering the flow w.r.t. free flight). Concerning the numerical side, δ_{sim} may be rewritten as follows:

$$\delta_{sim} = C_{sim} - C_{ideal}, \tag{1.11}$$

$$\delta_{sim} = \delta_{num} + \delta_{fp} + \delta_{input} + \delta_{model}, \tag{1.12}$$

$$|\delta_{sim}| \leq u_{num} + u_{fp} + u_{input} + u_{model} = u_{sim}. \tag{1.13}$$

The error δ_{sim} gathers the following terms.

1. The numerical approximation error δ_{num}. It may be evaluated through mesh refinement and it should vanish for very small space step size while, for fixed space step size, δ_{num} should decreases for increasing scheme order (if such numerical schemes are available).

2. The finite precision algebra error δ_{fp}. It may estimated by specific dedicated tool (*e.g.* CADNA[19]) and it should cancel at the limit of long mantissa algebra.

3. The physical modeling error δ_{model}. It is strictly linked with the chosen fluid dynamics model. This error decreases for increasing model complexity: RANS (good for attached flows, access only to averaged flow variables), DES (accurate except for phenomena at boundary layer scale), LES (accurate but highly mesh-dependent) and DNS (the most accurate, no modeling error).

4. The input value error δ_{input}. It may disappear if more information is obtained (*e.g.* accurate value for wall roughness). This kind of error is estimated through uncertainty quantification (UQ) methods if the distribution law of the input parameters is known.

All these errors, δ_*, may be estimated except for δ_{model}. In this context, the validation approach leads to the following relations:

$$\delta_{model} = \delta_{sim} - \delta_{num} - \delta_{fp} - \delta_{input}, \tag{1.14}$$

$$= E_C + \delta_{exp} - \delta_{num} - \delta_{fp} - \delta_{input}, \tag{1.15}$$

$$\delta_{model} - E_C = \delta_{exp} - \delta_{num} - \delta_{fp} - \delta_{input}, \tag{1.16}$$

thus

$$E_C - u_{exp} - u_{num} - u_{fp} - u_{input} \leq \delta_{model} \leq E_C + u_{exp} + u_{num} + u_{fp} + u_{input}. \tag{1.17}$$

It is clear that some of these terms may vanish, namely for initial non-simplified model $u_{model} = 0$, or for perfectly defined problem, $u_{input} = 0$.

It is important to underline that although it is possible to speak about the verification *of* a code, it is more correct to speak about the *use* of a code in a validation process. The code, together with the experimental evidence, may take part in u_{exp}, u_{input} and u_{model} evaluations but ultimately, it is in the estimation of u_{num} and u_{fp} where the code is the only operator in the validation approach.

The study of the influence of δ_{input} and u_{input} on CFD simulations has lately caught significant attention in the CFD community. Uncertainty quantification methods are able to assess this problem. Section 1.3 explores same similarities and differences between V&V and UQ while Chapter 4 is dedicated to UQ methods with applications to aerodynamic simulations.

1.2.3 Aerodynamic global coefficients validation

In Section 1.2.2 some details concerning the validation process have been addressed. In particularly, it has been pointed out the need to bound the range of applicability of a certain code (*i.e.* the intended use). It is interesting to overview the precision sought for global aerodynamic functions [9].

The reference precisions requested by aircraft manufacturers (clients) for WT tests are unchanged since the 80ies [20]. Table 1.1 shows the precision sought for the lift, drag and momentum coefficients reported in AGARD 1982 [21] and 1988 [22]. The standard deviation is denoted as σ and if $\tilde{\mu}$ is the mean, the interval $[\tilde{\mu}-2\sigma, \tilde{\mu}+2\sigma]$ includes 95% of results[5]. The increments values stand for the scatter between two different configurations, *e.g.* two candidate flaps for a project.

Table 1.1: Typical objectives incertitudes for global aerodynamic coefficients in a WT framework.

	Absolute values		Increments	
	Confidence interval ($\pm 2\sigma$)	std (σ)	Confidence interval ($\pm 2\sigma$)	std (σ)
C_L	$\pm 1 \cdot 10^{-2}$ (*lift-count*)	$0.5 \cdot 10^{-2}$	$\pm 0.5 \cdot 10^{-2}$	$0.25 \cdot 10^{-2}$
C_D	$\pm 1 \cdot 10^{-4}$ (*drag-count*)	$0.5 \cdot 10^{-4}$	$\pm 0.5 \cdot 10^{-4}$	$0.25 \cdot 10^{-4}$
C_m	$\pm 1 \cdot 10^{-3}$	$0.5 \cdot 10^{-3}$	$\pm 0.5 \cdot 10^{-3}$	$0.25 \cdot 10^{-3}$

Once the desired precision has been defined, it is interesting to see whether the precision is attained in both WT and CFD applications. First of all, the standard deviation is analyzed. During the first drag prediction workshop in 2001 [20], 35 WT tests have been compared obtained from three different WTs. Table 1.2 reports the obtained results. It is clear that even experimental results do not attain the desired precision. Since the long-term objective is to replace WT experiments with CFD simulations, CFD simulations should aim to obtain the *same* reference precision [23] given previously in Table 1.1. The GARTEUR documents allow us to compare the numerical drag estimation incertitudes from 1988, AG05 [24], to 2007, AG39 [25]. The values are reported in Table 1.3. In GARTEUR AG05 the simulations are for fully potential and Euler flows while in GARTEUR AG39, RANS flows have been assed. From this table it is possible to highlight that in 20 years the standard deviation has been just halved and it is still far above the sought precision.

The analysis may be now focused on the mean of the aerodynamic functions. For wing/fuselage 3D configurations, the documents DPW-1 2001 [20], DPW-2 2003 [26] and GARTEUR AG39 2007 [25] allow the comparison given in Table 1.4

[5]For normal distributions.

for the gap between the mean of the computed values and the experimental ones. The objective precision is not reached even fixing laminar/turbulent transition.

Table 1.2: Uncertainty level for WT experimental global aerodynamic coefficients.

	std (σ)		
	Test F4	w.r.t. Ref ?	
C_L	$0.24 \cdot 10^{-2}$	✓	$[< 0.5 \cdot 10^{-2}]$
C_D	$4 \cdot 10^{-4}$	×	$[> 0.5 \cdot 10^{-4}]$
C_m	$5 \cdot 10^{-3}$	×	$[> 0.5 \cdot 10^{-3}]$

Table 1.3: Uncertainty level for the numerical drag coefficient.

	std (σ)			
	GARTEUR AG05 (1988)	GARTEUR AG39 (2007)	w.r.t. Ref ?	
C_D	$10 \cdot 10^{-4}$	$5 \cdot 10^{-4}$	×	$[> 0.5 \cdot 10^{-4}]$

Table 1.4: Comparison of the mean values for experimental and numerical global aerodynamic coefficients.

	$\Delta[\tilde{\mu}_{CFD} - \tilde{\mu}_{WT}]$				
	DPW-1 (2001)	DPW-2 (2003)	GARTEUR AG39 (2007)	w.r.t. Ref ?	
C_L	-	-	$-0.7 \cdot 10^{-2}$	✓	$[< 1 \cdot 10^{-2}]$
C_D	$+7 \cdot 10^{-4}$	$-3 \cdot 10^{-4}$	$-7 \cdot 10^{-4}$	×	$[> 1 \cdot 10^{-4}]$
C_m	$-29 \cdot 10^{-3}$	$-17 \cdot 10^{-3}$	$+9 \cdot 10^{-3}$	×	$[> 1 \cdot 10^{-3}]$

These kind of analysis bring us to some conclusions. Within those, the sought precision of $1 \cdot 10^{-4}$ for C_D is really ambitious and quite far from the capability of commercial CFD code with a large spectrum of applications [23]. It has been shown that even the WT tests do not reach the required precision given by the WT clients. The precision of the numerical simulations seem to get closer to the precision obtained in WT applications. In this context, two *provocative* conclusions may be reported from Destarac [6] and Tinoco [27] which may show how it can still be considered tricky the validation process in an engineering framework:

> *If your computation predicts drag with an error of 2 to 5 drag counts, it is a good computation; if the prediction is perfect, something must be wrong with the computation; if the error is of 20 drag counts, something may be wrong with the experiment.* Destarac

It is difficult, if not impossible, to put a precise numerical definition on what is CFD validation and when CFD is "good enough"; but I know it when I see it. Tinoco

1.2.4 Example: 2D turbulent flow about airfoil NACA0015

An example of the δ_{sim} error of Section 1.2.2 is performed here. In this exercise, it has been chosen to keep a standard[6] approach on the validation process by excluding δ_{input} and u_{input} in the analysis. It is then supposed that the problem is perfectly defined.

The present study addresses the prediction of 2D subsonic turbulent flow about a NACA0015 airfoil. The computations are carried out by means of RANS simulations. The turbulence closure is assured by a Spalart-Allmaras turbulence model [1] or the Kok $\kappa - \omega$ model [28]. The ONERA finite-volume compressible CFD *elsA* [2] code has been used. The analysis of lift C_L and drag C_D coefficients has been carried out. The flow is at $M_\infty = 0.291$, $Re = 1.9 \cdot 10^6$ and $AoA = 5°$. The experimental values are $\boldsymbol{C_L} = 0.53$, $\boldsymbol{C_D} = 0.0086$ [29]. The coloured digits represents the desired precision for lift (red) and drag (blue) coefficients given in Table 1.1 for the confidence interval for absolute values. While the underlined digits is the desidered precision for the standard deviation σ.

Table 1.5 shows the obtained results for three different meshes: M1, M3 and M5 with $\mathcal{O}(0.1\text{M})$, $\mathcal{O}(0.4\text{M})$ and $\mathcal{O}(1.7\text{M})$ points respectively. From this three results, it is possible to have an insight on the δ_{num}. Two turbulence models have used as mentioned before, this leads to different coefficients estimations, *i.e.* it gives an idea δ_{model} in the context of RANS simulations. The *elsA* code has been compiled in single (*sp*) and double (*dp*) precision, *i.e.* δ_{fp}. All these results are at converged solution. On the other hand, Table 1.6 shows the δ_{conv} for the M3 mesh at different convergence level for the SA model compiled in double precision.

1.3 V&V and UQ

The importance of the assessment of CFD simulations reliability has been approached in several works, *e.g.* [16, 30, 31, 32], but it is only more recently that the CFD community started to question about *how to quantify* the trustworthiness of CFD simulations. Early works such as [33, 34] meet this issue but it is only with the establishment of UQ that this problem have been properly addressed. UQ is considered nowadays as a valuable complement to V&V and always more frequently it is common to talk about Verification & Validation and Uncertainty

[6] *deterministic, i.e.* the introduction of the input values uncertainty would have forced us to use a *probabilistic* approach by using the distribution laws of input values as well as stochastic approximation methods introduced in Chapter 4

Table 1.5: Drag and lift coefficients comparisons for the NACA0015 turbulent RANS + SA flow. Results obtained at converged solution for different meshes, turbulence models and code precisions.

	C_L			C_D			
	SA *sp*	SA *dp*	KW *dp*	SA *sp*	SA *dp*	KW *dp*	
M1	0.5282	0.5282	0.5196	0.013195	0.013195	0.012142	↑
M3	0.5270	0.5270	0.5161	0.012639	0.012639	0.012372	δ_{num}
M5	0.5263	0.5262	0.5148	0.012549	0.012537	0.012691	↓
	← δ_{fp} →			← δ_{fp} →			
	← δ_{model} →			← δ_{model} →			

Table 1.6: Drag and lift coefficients comparisons for the NACA0015 turbulent RANS + Sa flow at different densitu ρ explicit convergence levels. Computations with SA in double precision.

M3	$R_\rho = 10^{-2}$	$R_\rho = 10^{-3}$	$R_\rho = 10^{-4}$	...	$R_\rho = 10^{-9}$
C_L	0.522872	0.527097	0.527060	...	0.527055
C_D	0.012850	0.012646	0.012639	...	0.012639
		← δ_{conv} →			

Quantification (V&V&UQ, VV&UQ or VVUQ), *e.g.* [35, 36]. In which extent UQ is complementary to V&V, differs from the latter and, more specifically, from the validation process?

Uncertainty Quantification considers the fluid flow problem as *stochastic* and not anymore as deterministic. Reality, including experiments, exhibits fluctuation around nominal values and typically a confidence level or a probabilistic nature is attached to experimental values. The latter were then compared to traditional CFD simulations performed at nominal values. UQ allows to obtain numerical results in a probabilistic framework which make the comparison between observations and simulations more sound. Unfortunately, experimental statistics are often poor and only error bars are available. It is common then to check whether the experimental values are captured by the numerical probability density function envelope of the QoI. An UQ process includes 1- the *definition* and *modelisation* of all sources of uncertainty, 2- their *propagation* through the computational model and 3- finally the *quantification* of the uncertainty on the QoI. Figure 1.1 shows a conceptual diagram of VV&UQ and the interactions between V&V and UQ. UQ plays manly two important roles.

a) The first one consists in transferring valuable information from observations to the mathematical model which represents the studied physical phenomenon.

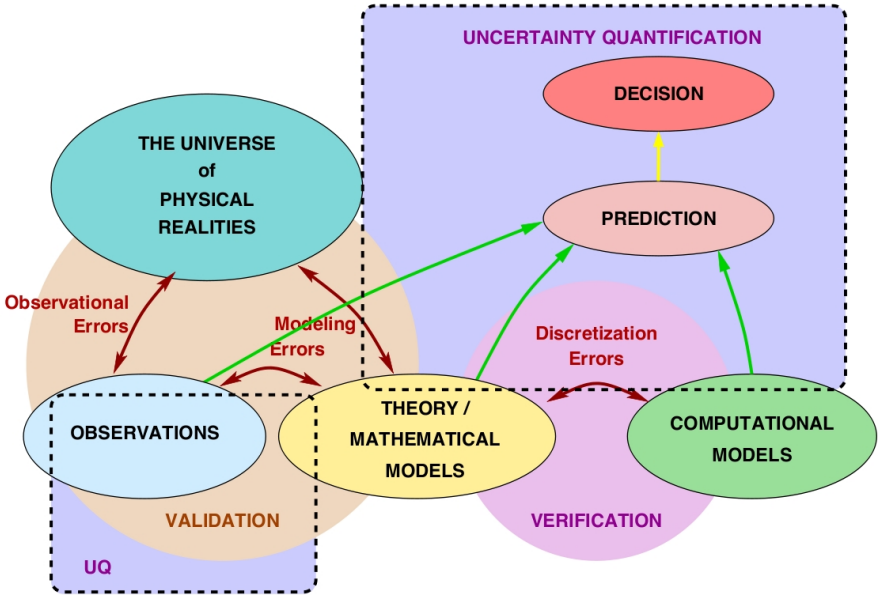


Figure 1.1: Conceptual diagram for Verification and Validation and Uncertainty Quantification [37] (derived from [38]).

These is often done through a Bayesian approach in a data assimilation framework. For instance, in a fluid dynamics contest, the turbulence model constants may be modified in order to better fit experimental results and simulations thanks to experimental evaluations [39].

b) The second use of UQ is as a tool for assessing the reliability of simulations through the quantification of the uncertainty on the QoI due to the considered sources of uncertainty. The probability that a certain value of the QoI occurs may 1- help the *decision-maker* in the judgement of the trustworthiness of the CFD simulations, 2- verify whether certain threshold of the QoI may be attained or not, 3- highlight how the QoI is sensible to the considered uncertainties.

The second use of UQ (*i.e.* b) is the one exploited in this PhD thesis. In the following section this choice is equally presented.

UQ may be considered then as a valuable tool in the validation process. Nevertheless, in Figure 1.1, UQ influences also the verification side of V&V. UQ methods may significantly modify the mathematical model equations and in turn the numerical schemes thus, they influence the verification process. In this case, the deployed stochastic approximation method is called to be *intrusive* since the fluid flow equations have now a stochastic nature. This may be avoided by using *non-intrusive* methods which consider the CFD model as a black-box. The system of equations however must be well-posed over all the parametric uncertainty range thus, with the non-intrusive approach, the CFD model is exploited in an oper-

ating regime close to the one for which the code has been designed. Chapter 4 is dedicated to UQ methods and describes some of the available stochastic approximations. Notwithstanding, new approaches have been developed in order to couple goal-oriented discretization error control and stochastic problems. A solution verification approach for the adaptation of both the mesh and the way random variables are approximated have been presented by Almeida *et al.* [40] for linear stochastic advection-diffusion models. Palacios *et al.* [41] have introduced a new robust grid adaptation technique which minimize the discretization error over the variations range of the uncertain parameters. The deployment of these coupled methods is generally expensive and at the moment they seem not affordable for aeronautical engineering applications involving RANS flows with an high number of uncertain parameters. In this PhD thesis, the control of the discretization error and of the input values uncertainty on the QoI has been kept separated.

1.4 From the VV&UQ to the sensitivity analysis in a fixed-model approach

In Section 1.2.2, the four main errors that compose the δ_{sim} have been introduced. From an engineering point of view, it is reasonable to consider $\delta_{fp} << \delta_{num,model,input}$. It is common practice to compile and run CFD code in double precision. The example in Section 1.2.4 has shown for instance that the impact of the simple versus double precision compilation may negligible. The fluid dynamics model is undoubtedly the *elephant in the room* of all CFD applications and studies (also in the example given in Section 1.2.4). The results obtained from one model to another may differ significantly as the complexity of the model itself (RANS, DES, LES, DNS). Within the four before mentioned errors, δ_{model} is the closest to physical consideration and evaluations.

The main physical approximations exploited in this manuscript are linked with the derivation of the RANS equations coupled with an eddy viscosity turbulence model: the Reynolds decomposition and the Boussinesq hypothesis. These assumptions, as long as with other turbulence hypothesis, simplify the physical complexity and allows an easier and cheaper approach to CFD while decreasing the physical prediction accuracy. A typical example in external aerodynamics is the flow detachment prediction. Considering an upward stroke in a pitching airfoil, it is well known that *all* turbulence models fail in correctly predict the flow behavior at high angles of attack [42]. These models indeed overestimate the Reynolds stresses, overestimate the lift, fail in correctly compute the drag and show a delay in the pitching moment inversion thus an overestimation of the stall angle [43]. These models do not take into account the laminar and transitional character of the flow, *i.e.* the boundary layer is considered fully turbulent. Many turbulence model modifications exist which sometimes improve the flow prediction. In this

context, it is also useful to recall the importance of the δ_{num} influence on the assessment of δ_{model}. It is known indeed that the space resolution (mesh size) has a great impact on the detachment phenomenon prediction. A low space resolution induces an underestimation of the lift which erroneously leads to a better agreement with the experimental evidence [43]. It is then essential to carefully scrutinize the published numerical studies concerning turbulence modeling in order to rule out misleading conclusions.

The objective of this PhD thesis is to keep the discussion closer to the numerical side of CFD codes. In this context, if the objective is to assess the sensitivity of a code where the latter is considered as a black-box including the physical model, it is then necessary to evaluate δ_{num} and δ_{input} by ignoring δ_{model} in the analysis. The obtained discussions are valuable to get some insight on the sensitivity of a *given* particular model with respect to the selected uncertainties and sources of errors. The sensitivity with respect to the mesh and to the input values is assessed.

The extensive use of Computational Fluid Dynamics (CFD) models and resources in the aeronautical community has required the need of accurate predictions. Often, only some quantities of interest (QoI), *e.g.* forces and moments, rather than the overall flow-field, are considered in aerodynamic simulations. In this framework, δ_{num} is addressed through the development of goal-oriented mesh adaptation techniques which has seen a rapid growth since the mid 90's. Most of the time, these methods involve the adjoint vector of the QoI. Chapters 2 introduces gradient computations methods for local search approaches in aeronautical design. These gradient methods are exploited in the mesh adaptation techniques presented in Chapter 3.

On the other hand, δ_{input} is associated with input parameters discretized with random variables from a probability space, the system becomes stochastic and response sampling for infinite values of the parameters becomes intractable. The numerical challenge is then to represent the parameter-dependent solution or related QoI of the system with a continuous approximation (*i.e. surrogate*) over the entire range by relying only on discrete system evaluations (or *realizations*) for (possibly few) different values of the parameters. These non-intrusive approaches, despite relying on samples, may therefore be seen as functional approximations of the QoI. Several uncertainty quantification methods exist. Chapters 4 addresses this analysis.

Chapter 2

Local search methods for design in aeronautics and gradient computation methods

Numerical optimization aims at locating the minima of a regular function (called objective function) on a finite-dimensional design space, while satisfying a certain number of constraints (expressed as inequality verified by the so-called constraint functions). More precisely, local optimization aims at finding a local optimum in the neighbourhood of an initial guess, whereas global optimization aims at finding the global optimum on the whole design space. These problems are, of course, the mathematical counterparts of mechanical optimization problems - like drag minimization of an aircraft. Section 2.1 introduces the optimization problem as well as some of the most exploited methods in aeronautics. These methods are often based on gradients of forces and moments with respect to design design parameters and their computation is the topic of Section 2.2. The implementation of the discrete adjoint in the *elsA* code is introduced in Section 2.3 and the new code developments of this PhD thesis framework are explained in Sections 2.4 and 2.5. The gradient precision improvement is shown through examples.

Nomenclature

Abbreviations

5p-cor	5 points corrected
Adj	Discrete adjoint method
AoA	Angle of Attack
CFD	Computational Fluid Dynamics
FD	Finite Differences
FV	Finite Volume
LE	Leading Edge
Lin	Discrete direct differentiation method
KKT	The Karush-Kuhn-Tucker condition
KW	Wilcox $k-\omega$ turbulence model
QoI	Quantity of Interest
RANS	Reynolds-Averaged Navier-Stokes
SA	Spalart-Allmaras turbulence model
TE	Trailing Edge
THL	THin Layer approximation

V&V	Verification and Validation

Greek letters

α, α_l	Shape parameter
δ	Increment
λ_i	Adjoint vector of $\mathcal{G}_i$
$\boldsymbol{\Lambda}$, $\boldsymbol{\Lambda}_m$	Adjoint vector of $\mathcal{J}$, $\mathcal{J}_m$
μ	Dynamic viscosity
$\bar{\mu}_T$	Frozen turbulent dynamic viscosity
$\tilde{\nu}$	Turbulent kinematic viscosity
ρ	Density
$\boldsymbol{\tau}$	Viscous stress tensor
$\boldsymbol{\omega}$	Vorticity vector

Latin letters

$\boldsymbol{\mathcal{C}}$	Turbulent cross-diffusion term
C_L	Lift coefficient
d	Descent direction
D_α	Design space
$\boldsymbol{\mathcal{D}}$	Turbulent destruction term
E	Total energy
$\boldsymbol{F}$	FV flux
$\mathcal{G}$, $\mathcal{G}_i$	Constraint function
H	Hessian matrix of $\mathcal{J}$
$\mathcal{J}$, $\mathcal{J}_m$	Aerodynamic function as a function of α_l
J, J$_m$	Aerodynamic function as a function of $\boldsymbol{W}$ and $\boldsymbol{X}$
J, J_m	Aerodynamic function as a function of $\boldsymbol{X}$ only
$\mathcal{L}$	Lagrangian functional
M	Mach number
$\boldsymbol{n}$	Unit normal vector
N_G	Number of constraint functions
N_J	Number of objective functions
N_l	Number of shape parameters
N_W	Size of vectors $\boldsymbol{W}$ and $\boldsymbol{R}$
$\boldsymbol{\mathcal{P}}$	Turbulent production term
$\boldsymbol{R}$	Discrete residual vector
Re	Reynolds number
S	Surface vector
$\boldsymbol{\mathcal{S}}$	Turbulent source term
$\boldsymbol{S}$	Surface mesh
T	Temperature
$\boldsymbol{u}$, $\boldsymbol{U}$	Velocity vector
V	Mesh cell volume
$\boldsymbol{W}$	Conservative variables vector / flow solution
$\boldsymbol{X}$	Mesh coordinates

Superscripts

T	Transpose
lin	Linearized

Subscripts

c	Cell centre
$\boldsymbol{c}$	Convective
L	Lower bound
i	Cell interface
T	Turbulent component
U	Upper bound
$\boldsymbol{v}$	Viscous / diffusive

2.1 Local optimization through gradients

Numerical optimization for airplane design was used almost as soon as simulation codes appeared. The aerodynamic optimizations carried out by G.N. van der Plaats at NASA in the mid 70's illustrate this early interest in optimization [44]. At that time, non gradient-based (*e.g.* simplex method) as well as gradient-based (*e.g.* descent methods) approaches were used for 2D and simple 3D configurations. The estimation of the gradients for descent methods was computed through finite-differences. Since then, the framework of aerospace optimization has known at least three drastic extensions.

1. Several global optimization methods have been defined and intensively used, *e.g.* evolutionary algorithm, particle swarm, ant colony, simulated annealing

et cet.

2. Surrogate functions have been used for a part of the evaluation of the global optimization methods leading to significant cost reductions, *e.g.* neural network, Kriging, polynomial regression, support vector machine *et cet.*

3. Adjoint and direct differentiation method have been defined and studied in order to compute the needed gradients for descent algorithms.

Given an aerodynamic object where the geometry is subject to N_l shape parameters α_l, $l \in [1, N_l]$, it is denoted $\mathcal{J}(\alpha_l)$ the objective function while it is denoted $\mathcal{G}_i$, $i \in [1, N_G]$, the N_G constraint functions. The vector of shape parameters α_l is supposed to vary in the design space D_α, a parallelepiped of $\mathbb{R}^{N_l}$. It is supposed that the constraint functions are negative at admissible design points.

The local optimization problem reads as follows: seek for α^* in D_α such that $\mathcal{J}(\alpha^*) = \min \mathcal{J}(\alpha)$ on D_α where $\mathcal{G}_i(\alpha^*) \leq 0$, $\forall i \in [1, N_G]$.

2.1.1 The Karush-Kuhn-Tucker condition

For the unconstrained optimization of a C^2 function of $\mathbb{R}^{N_l}$, classical conditions of existence for minima read as follow.

1. Local optimum located in α^*, $\nabla\mathcal{J}(\alpha^*) = 0$ is a necessary condition. $\nabla\mathcal{J}(\alpha^*) = 0$ and $H(\alpha^*)$ positive definite is a sufficient condition, where H is the Hessian matrix of $\mathcal{J}$.

2. Global optimum located in α^*, $\nabla\mathcal{J}(\alpha^*) = 0$ is a necessary condition. $\nabla\mathcal{J}(\alpha^*) = 0$ and $H(\alpha)$ positive definite on $\mathbb{R}^{N_l}$ is a sufficient condition.

In order to define the necessary condition for optimality in a constrained problem, it is necessary to define the design space D_α. The latter is supposed to be a parallelepiped for the sake of simplicity:

$$D_\alpha = [\alpha_{1,L}, \alpha_{1,U}] \times [\alpha_{2,L}, \alpha_{2,U}] \times [\alpha_{3,L}, \alpha_{3,U}] \times ... \times [\alpha_{N_l,L}, \alpha_{N_l,U}], \tag{2.1}$$

where L and U define the lower and upper bounds. The domain bounds are rewritten as $2N_l$ additional constraints:

$$\begin{cases} \mathcal{G}_{N_G+1}(\alpha) = \alpha_{1,L} - \alpha_1 \\ \mathcal{G}_{N_G+2}(\alpha) = \alpha_1 - \alpha_{1,U} \\ \mathcal{G}_{N_G+3}(\alpha) = \alpha_{2,L} - \alpha_2 \\ ... \\ \mathcal{G}_{N_G+2N_l}(\alpha) = \alpha_{N_l} - \alpha_{N_l,U} \end{cases} \tag{2.2}$$

The Karush-Kuhn-Tucker (KKT) necessary conditions for an optimization problem under inequality constraints are defined as follow:

$$\begin{array}{l} \alpha^* \text{is an admissible state} \\ \nabla\mathcal{J}(\alpha^*) + \Sigma\lambda_i^*\nabla\mathcal{G}_i(\alpha^*) = 0 \\ \lambda_i^*\mathcal{G}_i(\alpha^*) = 0 \qquad \lambda_i^* \geq 0. \end{array} \tag{2.3}$$

Where the third line states that only constraints attaining the limit value zero may have their gradient included in the linear combination in the second line. By introducing a Lagrangian $\mathcal{L}(\alpha, \lambda_1, \cdots, \lambda_{N_G}) = \mathcal{J}(\alpha) + \Sigma\lambda_i\mathcal{G}_i(\alpha)$, the KKT conditions can be rewritten:

$$\begin{array}{l} \alpha^* \text{is an admissible state} \\ \nabla_\alpha\mathcal{L}(\alpha^*, \lambda_1^*, \cdots, \lambda_{N_G}^*) = 0 \\ \lambda_i^*\mathcal{G}_i(\alpha^*) = 0 \qquad \lambda_i^* \geq 0. \end{array} \tag{2.4}$$

The KKT condition is a necessary condition for optimality. This condition is the counterpart for constrained problems of the necessary condition $\nabla\mathcal{J} = 0$ for unconstrained ones. It is a sufficient condition only when objective and constraint functions are convex.

2.1.2 Descent methods

The aim of this section is not to review all the available descent methods but rather to introduce two of the most still exploited ones: the conjugate gradient and feasible direction methods. In many algorithms of multi-dimensionnal optimization, after a descent direction d^k has been defined, a one dimensional minimization along d^k is performed. This means that a step t is sought to diminish significantly $q(t) = \mathcal{J}(\alpha^k + td^k)$. A good line-search is obviously desirable, but the number of exact evaluations of $\mathcal{J}$ must remain as low as possible : the goal is not to find the optimal $t > 0$ at an intermediate step but to reduce the objective function $\mathcal{J}$ efficiently at each iteration. Conversely a bad line-search can slow down the global algorithm and a compromise must be found between the performance of the line-search and the number of evaluations of $\mathcal{J}$. This minimization along a descent direction during multi-dimensionnal optimization is one of the reasons for the interest in 1D-minimization.

These methods are used in current aerodynamic design optimizations at ONERA. Méheut *et al.* describe some gradient-based single and multi-point applications using the *elsA* code for airfoil and wing optimization for inviscid flow [45] as well as wing planform optimization for RANS flow [46]. An efficient application of these approaches is given by Minelli *et al.* [47] for the reduction of both sonic boom and drag for supersonic jet. Aeroelastic applications are also worth considering for the optimization of a flexible transport aircraft wing [48].

Conjugate gradient method

The idea behind the conjugate gradient methods comes from the analysis of the behaviour of the steepest descent for the specific case of positive definite quadratic forms ($\mathcal{J}(\alpha) = \frac{1}{2}\alpha^T H\alpha + b^T\alpha$). In this case, the conditioning of positive definite matrix H strongly affects the robustness and convergence speed of the steepest descent. To improve robustness, conjugate gradient methods use a descent direction d^k orthogonal to d^{k-1} in the sense of H, *i.e.* $\left(d^{k-1}\right)^T Hd^k = 0$. Thus

$$d^k = -\nabla\mathcal{J}(\alpha^k) + \beta^k d^{k-1} \qquad \beta^k = \frac{\left(\nabla\mathcal{J}(\alpha^k)\right)^T Hd^{k-1}}{\left(d^{k-1}\right)^T Hd^{k-1}}. \tag{2.5}$$

In this simple case (positive definite and constant matrix H), the search of the unique minimum of $\mathcal{J}$ is equivalent to the resolution of the linear system $H\alpha + b = 0$ (search for the unique point where gradient of $\mathcal{J}$ is zero) and it can be proven that the algorithm converges to the unique solution in N_l or less steps.

For non quadratic cases, β^k can be computed through the Fletcher-Reeves or Polak-Ribiere formulae. The first one is based on an other formula of β^k in the quadratic positive definite case $\beta_k = \frac{\|\nabla\mathcal{J}(\alpha^k)\|^2}{\|\nabla\mathcal{J}(\alpha^{k-1})\|^2}$ which can be directly applied to a non-quadratic function as it does not refer anymore to the matrix H:

$$\beta^k = \frac{\|\nabla\mathcal{J}(\alpha^k)\|^2}{\|\nabla\mathcal{J}(\alpha^{k-1})\|^2} \tag{2.6}$$

The second extension, proposed by Polak and Ribiere in 1969, reduces also to the same algorithm in the quadratic positive definite case. It is known to lead to a more efficient algorithm for specific applications. The formula is:

$$\beta^k = \frac{\|\nabla\mathcal{J}(\alpha^k)\|^2}{\|\nabla\mathcal{J}(\alpha^{k-1})\|^2} - \frac{\left(\nabla\mathcal{J}(\alpha^k)\right)^T \nabla\mathcal{J}(\alpha^{k-1})}{\|\nabla\mathcal{J}(\alpha^{k-1})\|^2}. \tag{2.7}$$

Once the β^k may be computed, the conjugate gradient method follows the Algorithm 1.

Feasible direction method

The feasible direction method works directly on the non-linear equations of the problem. Its goal is to build a sequence of points $\alpha^{(p)}$ such that $\alpha^{(p)} = \alpha^{(p-1)} + ld^{(p)}$ where the displacement along direction d leads to lower values of both the objective and active constraint [1] functions. Once d has been defined, the factor l

[1] all constraints satisfying $\mathcal{G}_i(\alpha^{(p-1)}) = 0$

Data: We set $k = 0$ and $d^{-1} = 0$; an initial iterate α^0 and a stopping tolerance ϵ is given
Result: -
while $\|\nabla \mathcal{J}(\alpha^k)\| \geq \epsilon$ **do**
  Compute $d^k = -\nabla \mathcal{J}(\alpha^k) + \beta^k d^{k-1}$
  Find t^* by line-search on $q(t) = J(\alpha^k + t d^k)$
  Update current iterate: $\alpha^{k+1} = \alpha^k + t^* d^k$
  Set $k = k + 1$
end

Algorithm 1: Conjugate gradient algorithm

is determined by a monodimensional optimization. The vector d must satisfy

$$\nabla \mathcal{J}(\alpha^0) \cdot d \leq 0, \tag{2.8}$$

$$\nabla \mathcal{G}_i(\alpha^0) \cdot d \leq 0 \quad \text{where} \quad \forall i \quad \mathcal{G}_i(\alpha^0) = 0. \tag{2.9}$$

Figure 2.1 shows the vector d and how these requirements are satisfied in the case where $N_l = 2$.

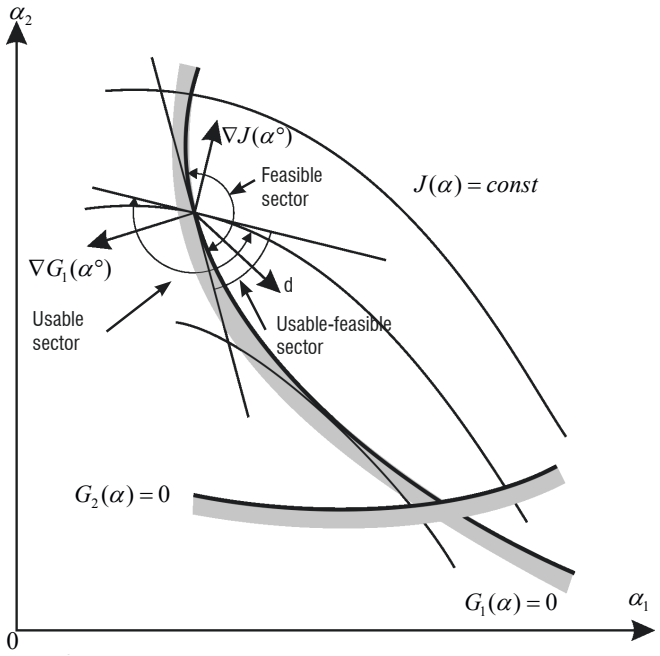


Figure 2.1: Principle of feasible direction search.

The tricky point is the determination of the vector d ensuring the best descent. For a simple two dimensional problem ($N_l = 2$) with one active convex constraint $\mathcal{G}_1$, it is easy to check that the minimisation $\nabla J(\alpha^0) \cdot d$ with $\nabla \mathcal{G}_1(\alpha^0) \cdot d \leq 0$ will result in a vector d with a non-admissible state α_1. To solve this issue, scalar

factors θ_i are included in the problem:

$$\nabla \mathcal{G}_i(\alpha^0) \cdot d + \theta_i \leq 0 \quad \text{where} \quad \theta_i > 0 \quad \text{and} \quad \forall i \quad \mathcal{G}_i(\alpha^0) = 0. \tag{2.10}$$

The value of θ_i is eventually linked with the one of $\nabla \mathcal{J}(\alpha^0) \cdot d$. Algorithm 2 shows the research of the best descent direction. Obviously, if $\mathcal{G}_i$ is a linear constraint then $\theta_i = 0$. For non-linear constraints, the simplest choice is $\theta_i = 1$. More complex choices exist [49].

Data: Set $k = 0$; an initial iterate α^0 and a stopping tolerance ϵ are given.
Result: -
while *KKT-conditions not satisfied* **do**
  Find d^k which maximizes β;
  subject to $\nabla \mathcal{J}(\alpha^k) \cdot d^k + \beta \leq 0$;
  $\nabla \mathcal{G}_i(\alpha^k) \cdot d^k + \theta_i \beta \leq 0$ where $\forall i \quad \mathcal{G}_i(\alpha^k) = 0$ and d^k bounded;
  Compute optimal step t minimizing $q(t) = \mathcal{J}(\alpha^k + t d^k)$;
  Update $\alpha^{k+1} = \alpha^k + t d^k$;
  Set $k = k + 1$.
end

Algorithm 2: Feasible direction method

2.2 Gradient computation methods

The previous section has shown how the gradients of the objective function with respect to the shape parameters are needed for classical local search methods for design in aeronautics. In this Section, the main discrete gradient computation methods are presented.

The notation follows the one introduced in Section 2.1. As concerning the numerical scheme, the discretization of the Navier-Stokes equations leads to a nonlinear equations system defined by

$$\boldsymbol{R}(\boldsymbol{W}, \boldsymbol{X}) = 0, \tag{2.11}$$

where $\boldsymbol{R}$ represents the discrete residual vector on each cell, $\boldsymbol{W}$ is the steady flow solution (size N_W) and $\boldsymbol{X}$ defines the mesh coordinates. In order to compute first order derivatives, it is assumed that:

1. $\boldsymbol{X}$ is C^1 regular, function of α_l;
2. $\boldsymbol{R}$ is C^1 regular w.r.t. its two vector arguments;
3. $\det[\partial \boldsymbol{R}/\partial \boldsymbol{W}](\boldsymbol{W}, \boldsymbol{X}) \neq 0$ where $\boldsymbol{R}(\boldsymbol{W}, \boldsymbol{X}) = 0$.

In this framework, the implicit function theorem ensures that $\boldsymbol{W}(\alpha_l)$ is a C^1 regular function of α_l. The objective function read as follow

$$\mathcal{J}(\alpha_l) = \mathrm{J}(\boldsymbol{W}(\alpha_l), \boldsymbol{X}(\alpha_l)). \tag{2.12}$$

Given a descent algorithm, only one objective function is considered under several constraints. Thus for $\mathcal{J}_m$ with $m \in [1, N_J]$, it is necessary to run N_J-times the chosen descent algorithm.

There exists mainly three methods for the discrete gradient computations:

1. Finite differences (FD): it implies at least N_l non-linear computations of size N_W;
2. The discrete direct differentiation method (Lin): it needs the resolution of N_l linear systems of size N_W;
3. The discrete adjoint method (Adj): it requires the resolution of N_J linear systems of size N_W.

In the aeronautical community, it is common to have few objectives and/or constraints (N_J) while having several dozens (or even hundreds) of design parameters (N_l). It is reasonable then to consider the discrete adjoint method more attractive.

2.2.1 Finite differences - FD

The finite differences method is the oldest one and does not require any specific coding in the CFD solver. A finite difference step size is chosen ($\delta\alpha_l$) and the shifted meshes ($\boldsymbol{X}(\alpha_l \pm \delta\alpha_l)$) need to be constructed. The corresponding flows $\boldsymbol{W}(\boldsymbol{X}(\alpha_l \pm \delta\alpha_l))$ are computed. The derivative of the objective functions is then given by

$$\frac{d\mathcal{J}_m}{d\alpha_l} = \frac{\mathrm{J}_m(\boldsymbol{W}(\alpha_l + \delta\alpha_l), \boldsymbol{X}(\alpha_l + \delta\alpha_l)) - \mathrm{J}_m(\boldsymbol{W}(\alpha_l - \delta\alpha_l), \boldsymbol{X}(\alpha_l - \delta\alpha_l))}{2\delta\alpha_l}. \tag{2.13}$$

Special care needs to be taken while choosing $\delta\alpha_l$ in order to obtain valuable derivatives of the objective functions. Moreover, the computation of $\boldsymbol{X}(\alpha_l \pm \delta\alpha_l)$ is often replaced by $\boldsymbol{X}(\alpha_l) \pm \delta\alpha_l \frac{d\boldsymbol{X}}{d\alpha_l}$ for storage saving purposes.

2.2.2 The discrete direct differentiation method - Lin

The discrete direct differentiation method consists in resolving directly the following linear system obtained from the derivation of equation (2.11) w.r.t. α_l:

$$\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{W}} \frac{d\boldsymbol{W}}{d\alpha_l} = -\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{X}} \frac{d\boldsymbol{X}}{d\alpha_l}, \tag{2.14}$$

where the derivative $d\boldsymbol{W}/d\alpha_l$ is the flow sensitivity w.r.t. the shape parameter α_l. The right-hand side of (2.14) requires first the solution of the flow equations and it is computed through finite differences while the Jacobian matrix $\partial\boldsymbol{R}/\partial\boldsymbol{W}$ is evaluated at steady $\boldsymbol{W}(\alpha_l)$. The derivative of the objective functions is then given by

$$\frac{d\mathcal{J}_m}{d\alpha_l} = \frac{\partial \mathrm{J}_m}{\partial \boldsymbol{X}}\frac{d\boldsymbol{X}}{d\alpha_l} + \frac{\partial \mathrm{J}_m}{\partial \boldsymbol{W}}\frac{\boldsymbol{W}}{d\alpha_l}, \tag{2.15}$$

where $d\boldsymbol{X}/d\alpha_l$ is known and $\partial \mathrm{J}_m/\partial\boldsymbol{X}$ and $\partial \mathrm{J}_m/\partial\boldsymbol{W}$ are obtained from the ONERA *FFD72* code [6]. The flow sensitivity $d\boldsymbol{W}/d\alpha_l$ is implemented in the linearisation optimisation module of *elsA*.

2.2.3 The discrete adjoint method - Adj

In the discrete adjoint method scheme, in order to compute the derivative of the function $\mathcal{J}_m(\alpha_l)$ with respect to α_l, a Lagrangian functional is introduced:

$$\mathcal{L}_m(\boldsymbol{W}, \boldsymbol{X}, \boldsymbol{\Lambda}_m) = \mathrm{J}_m(\boldsymbol{W}, \boldsymbol{X}) + \boldsymbol{\Lambda}_m^T \boldsymbol{R}(\boldsymbol{W}, \boldsymbol{X}), \tag{2.16}$$

where $\boldsymbol{\Lambda}_m$ is the Lagrange multiplier (or adjoint vector) for the objective function $\mathcal{J}_m$. Given the before mentioned hypothesis, it is possible to define the aerodynamic function J_m as a function of the mesh only, *i.e.* $J_m(\boldsymbol{X}(\alpha_l)) = \mathrm{J}_m(\boldsymbol{W}(\boldsymbol{X}(\alpha_l)), \boldsymbol{X}(\alpha_l))$.

The Lagrange multiplier is then the solution of $d\mathcal{L}_m/d\boldsymbol{W} = 0$:

$$\boldsymbol{\Lambda}_m^T \frac{\partial \boldsymbol{R}}{\partial \boldsymbol{W}} = -\frac{\partial \mathrm{J}_m}{\partial \boldsymbol{W}}. \tag{2.17}$$

The right hand side of (2.17) requires first the solution of the flow equations and it is computed through analytical differentiation while the Jacobian matrix in the left hand side is built at steady solution. The derivative of objective functions is then given by

$$\frac{d\mathcal{J}_m}{d\alpha_l} = \frac{d\mathcal{L}_m}{d\alpha_l} = \frac{\partial \mathrm{J}_m}{\partial \boldsymbol{X}}\frac{d\boldsymbol{X}}{d\alpha_l} + \boldsymbol{\Lambda}_m^T \frac{\partial \boldsymbol{R}}{\partial \boldsymbol{X}}\frac{d\boldsymbol{X}}{d\alpha_l} = \left(\frac{\partial \mathrm{J}_m}{\partial \boldsymbol{X}} + \boldsymbol{\Lambda}_m^T \frac{\partial \boldsymbol{R}}{\partial \boldsymbol{X}}\right)\frac{d\boldsymbol{X}}{d\alpha_l}. \tag{2.18}$$

In brackets it is possible to identify:

$$\frac{dJ_m}{d\boldsymbol{X}} = \frac{\partial \mathrm{J}_m}{\partial \boldsymbol{X}} + \boldsymbol{\Lambda}_m^T \frac{\partial \boldsymbol{R}}{\partial \boldsymbol{X}}, \tag{2.19}$$

where the first term $(\partial \mathrm{J}_m/\partial\boldsymbol{X})$ represents the direct dependency of J_m to the mesh nodes location and the second term $(\boldsymbol{\Lambda}_m^T(\partial\boldsymbol{R}/\partial\boldsymbol{X}))$ corresponds to the flow field modification on the support of J_m due to a change of mesh nodes location at converged solution $(\boldsymbol{R} = 0)$.

In order to avoid the storage of the mesh sensitivity w.r.t. design variables, Nielsen and Park [50] used the *implicit*, $\boldsymbol{D}(\boldsymbol{X}, \boldsymbol{S}) = 0$, or *explicit*, $\boldsymbol{X} = \boldsymbol{X}(\boldsymbol{S})$, dependence between $\boldsymbol{X}$ and $\boldsymbol{S}$, where the latter is the surface mesh. In the first case the equations read as follow

$$\boldsymbol{\Lambda}_m^T \frac{\partial \boldsymbol{R}}{\partial \boldsymbol{W}} = -\frac{\partial \mathrm{J}_m}{\partial \boldsymbol{W}}, \tag{2.20}$$

$$\boldsymbol{\Gamma}^T \frac{\partial \boldsymbol{D}}{\partial \boldsymbol{X}} = \left(\frac{\partial \mathrm{J}_m}{\partial \boldsymbol{X}} + \boldsymbol{\Lambda}_m^T \frac{\partial \boldsymbol{R}}{\partial \boldsymbol{X}} \right) = \frac{dJ_m}{d\boldsymbol{X}}, \tag{2.21}$$

$$\frac{d\mathcal{J}_m}{d\alpha_l} = \left[\boldsymbol{\Gamma}^T \frac{\partial \boldsymbol{D}}{\partial \boldsymbol{S}} \right] \frac{d\boldsymbol{S}}{d\alpha_l}, \tag{2.22}$$

while for an explicit dependence, the following equations replace equation (2.18):

$$\boldsymbol{\Lambda}_m^T \frac{\partial \boldsymbol{R}}{\partial \boldsymbol{W}} = -\frac{\partial \mathrm{J}_m}{\partial \boldsymbol{W}}, \tag{2.23}$$

$$\frac{d\mathcal{J}_m}{d\alpha_l} = \left[\frac{dJ_m}{d\boldsymbol{X}} \frac{d\boldsymbol{X}}{d\boldsymbol{S}} \right] \frac{d\boldsymbol{S}}{d\alpha_l}. \tag{2.24}$$

Therefore, the $dJ_m/d\boldsymbol{X}$ terms are the standard outputs of a modular memory efficient adjoint solver where the complex parametrization (*i.e.* $d\boldsymbol{X}/d\alpha_l$) is not addressed anymore. The proposed goal-oriented method is based on this quantity. The mesh refinement indicator is described in Chapter 3. In Sections 2.4 and 2.5, the linearization of critical terms that where missing in the linearisation and in the adjoint optimisation module of *elsA* are reported.

Code verification

Finite differences is a good tool to verify the output of equation 2.18 while test based on shifted meshes for independent nodes displacements is an appropriate approach for verifying equation 2.19. A deeper verification of an adjoint code may be more tricky. If in the same gradient computation solver, there exists both the discrete direct differentiation method (Lin) and the discrete adjoint method (Adj), it is then possible to verify the adjoint one by duality test. Assuming that the direct differentiation method is verified[2], the term $\boldsymbol{\Lambda}_m^T \frac{\partial \boldsymbol{R}}{\partial \boldsymbol{W}}$ may be checked as follows

$$\forall (U, V) \in \mathbb{R}^{N_W} \quad \left(U^T \frac{\partial \boldsymbol{R}}{\partial \boldsymbol{W}} \right)_{\mathrm{Adj}} \cdot V = U^T \cdot \left(\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{W}} V \right)_{\mathrm{Lin}}. \tag{2.25}$$

This technique allows the verification even of single operators at a time (convective mean flow flux, diffusive mean flow flux *et cet.*). The adjoint implementations introduced in the following sections have been verified in this way.

[2]The verification is straightforward since the flow sensitivity $d\boldsymbol{W}/d\alpha_l$ may be directly compared w.r.t. FD computations.

2.3 Differentiated systems of equations

2.3.1 Solution of the direct and adjoint linear systems

The Jacobian matrix ($\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{W}}$) (as defined in the previous section) and its transpose are large, sparse, multi-banded matrices. Thus, their inverse can not be computed by a direct method, at least for large 2D and 3D problems. Some kind of iterative strategy (conjugate gradient method, relaxation) has to be implemented. A classical strategy consists in solving the linear system using a Newton-type or relaxation algorithm. An approximate Jacobian, noted $(\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{W}})^{(APP)}$ appears on the left hand-side of the algorithm equation. This matrix can be equal or very similar to the approximate Jacobian used as implicit matrix for steady state computations with backward-Euler schemes. On the right-hand side of the algorithm equation is the term that has to be driven to zero. The true Jacobian ($\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{W}}$) appears in that right-part of the equation. When considering a complicated set of equations (RANS equations for exemple), it can by replaced by an accurate approximation noted $(\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{W}})^{(ACC)}$. The equation for the adjoint method reads as follows

$$\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{W}}^{T(APP)} \left(\boldsymbol{\Lambda}_m^{(l+1)} - \boldsymbol{\Lambda}_m^{(l)} \right) = - \left(\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{W}}^{T(ACC)} \boldsymbol{\Lambda}_m^{(l)} + \left(\frac{\partial \mathrm{J}_m}{\partial \boldsymbol{W}} \right)^T \right) \tag{2.26}$$

while for the direct differentiation method reads

$$\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{W}}^{(APP)} \left(\left(\frac{d\boldsymbol{W}}{d\alpha_l} \right)^{(l+1)} - \left(\frac{d\boldsymbol{W}}{d\alpha_l} \right)^{(l)} \right) = - \left(\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{W}}^{(ACC)} \frac{d\boldsymbol{W}}{d\alpha_l}^{(l)} + \left(\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{X}} \frac{d\boldsymbol{X}}{d\alpha_l} \right)_{\mathrm{FD}} \right) \tag{2.27}$$

where (l) is the iteration index of the Newton-relaxation method.

2.3.2 State of the differentation in the *elsA* code

Equations systems

The Euler equations are fully linearised while concerning the RANS equations, it is possible either to choose the differentiation of only the five mean flow equations without differentiating the turbulent dynamic viscosity μ_T, *i.e.* equation (2.32) (frozen μ_T assumption, $\bar{\mu}_T$) or the fully differentiation of the equations system including the turbulence model (μ_T^{lin}). The available linearized turbulence models are Spalart-Allmaras [1] and Wilcox $k-\omega$ (KW) [51]. An example of how the linearization of the turbulence model affects the gradient computation precision is shown in Section 2.3.3.

Fluxes and source terms

For the mean-flow components, the differentiation w.r.t. the conservative variables (flowfield) and the geometry is available for the Roe flux [3] (first and second order with MUSCL approach [4] with van Albada limiter [5]), centred flux (either skew-symmetric or divergence form) and viscous flux with a gradient at cell-centres corrected at interfaces (5p-cor)[3]. Concerning the turbulent quantities, the differentiation w.r.t. the conservative variables (flowfield) and the geometry is available for the first order Roe flux, viscous flux with 5p-cor formula and the chosen turbulence model (SA[4] or KW[5]) source terms.

2.3.3 Verification on a RAE2822 airfoil

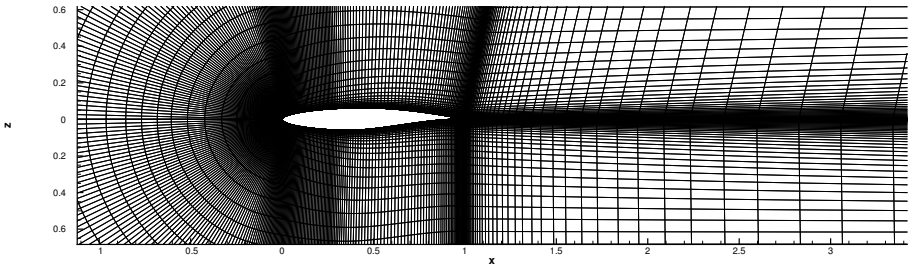


Figure 2.2: Structured C-mesh for the RAE2822 airfoil.

The fully turbulent flow over a RAE2822 airfoil is considered. The turbulence closure is ensured by the Spalart-Allmaras model [1]. The flow is at $Re = 7.52 \cdot 10^6$ based on the chord, $M_\infty = 0.73$ and $AoA = 2.3°$. The mesh is depicted in Figure 2.2. The considered shape parameter α is an angle of rigid rotation around the leading edge (LE). The influenced of the linearization of the turbulence model is visible in Table 2.1 where the $\frac{dC_{Lp}}{d\alpha}$ and $\frac{dC_{Dp}}{d\alpha}$ are reported. Namely, the precision highly increases with the μ_T^{lin} approach. Please note also that the results obtained from the discrete direct differentiation and the adjoint methods coincide.

2.4 Linearization of the viscous mean-flow 5p-cor fluxes w.r.t. conservative variables

The discussion will be perfomed only on the second to fourth components of the mean-flow viscous fluxes. The same applies for the fifth and sixth components,

[3]The derivation w.r.t. the flowfield is the new code development explained in Section 2.4.
[4]The derivation w.r.t. the geometry is the new code development explained in Section 2.5.
[5]The derivation w.r.t. the geometry is not available at the moment.

Table 2.1: RAE2822. $\frac{dC_{L,p}}{d\alpha}$ and $\frac{dC_{D,p}}{d\alpha}$ obtained from Lin and Adj computation with both $\bar{\mu}_T$ and μ_T^{lin}.

Method		$\frac{dC_{L,p}}{d\alpha}$ $[\cdot 10^{-1}]$	Rel. Err. %	$\frac{dC_{D,p}}{d\alpha}$ $[\cdot 10^{-3}]$	Rel. Err. %
FD	-	1.703	-	5.688	-
Lin	$\bar{\mu}_T$	1.124	34	4.749	17
Adj	$\bar{\mu}_T$	1.120	34	4.780	16
Lin	μ_T^{lin}	1.694	0.53	5.714	0.46
Adj	μ_T^{lin}	1.691	0.70	5.724	0.63

i.e. T and $\tilde{\nu}$. The viscous mean-flow 5p-cor fluxes $\boldsymbol{F}_{\boldsymbol{v}_i}$ is given by the following relations:

$$\boldsymbol{\nabla u}_i = \frac{1}{\frac{1}{2}(V_{cp}+V_{cm})}\frac{1}{2}\left((||S_i|| + ||S_{ip}||)\cdot \boldsymbol{n}_i \boldsymbol{u}_{cp} - (||S_i|| + ||S_{im}||)\cdot \boldsymbol{n}_i \boldsymbol{u}_{cm}\right) \tag{2.28}$$

$$\begin{aligned}\boldsymbol{\nabla u}_i^{\text{5p-cor}} &= \boldsymbol{\nabla u}_i + \\ &+ \frac{1}{\frac{1}{2}(V_{cp}+V_{cm})}\frac{1}{4}\left((\boldsymbol{u}_{cpp}+\boldsymbol{u}_{cp})||S_{ipp}||\cdot \boldsymbol{n}_{ipp} - (\boldsymbol{u}_{cpm}+\boldsymbol{u}_{cp})||S_{ipm}||\cdot \boldsymbol{n}_{ipm} + \right. \\ &+ \left.(\boldsymbol{u}_{cmp}+\boldsymbol{u}_{cm})||S_{imp}||\cdot \boldsymbol{n}_{imp} - (\boldsymbol{u}_{cmm}+\boldsymbol{u}_{cm})||S_{imm}||\cdot \boldsymbol{n}_{imm}\right), \end{aligned} \tag{2.29}$$

$$\boldsymbol{\tau} = -\frac{2}{3}\mu_c\left(\boldsymbol{\nabla u}_i^{\text{5p-cor}}\right)\boldsymbol{I} + \mu_c\left(\boldsymbol{\nabla u}_i^{\text{5p-cor}} + (\boldsymbol{\nabla u}_i^{\text{5p-cor}})^T\right), \tag{2.30}$$

$$\boldsymbol{F}_{\boldsymbol{v}_i}^{\boldsymbol{u}} = \boldsymbol{\tau}\cdot \boldsymbol{S}_i. \tag{2.31}$$

The derivation w.r.t. the conservative variables of $\boldsymbol{F}_{\boldsymbol{v}_i}^{\boldsymbol{u}}$ needs special care. While being on a interface i of Figure (2.3), it is necessary to have access to all conservative values stored in cells *cm*, *cp*, *cmm*, *cpm*, *cmp* and *cpp*. In the current *elsA* code implementation, given an interface i, it is not possible to know the *nature* (*i.e.* boundary, internal interface joins) of interfaces *imm*, *ipm*, *imp* and *ipp*. For this reason, the derivation is done in two consecutive steps:

1. the derivation of all the green flux formulae involved in the $\boldsymbol{\nabla u}_i^{\text{5p-cor}}$ computation, *e.g.* $(\boldsymbol{u}_{cpp}+\boldsymbol{u}_{cp})||S_{ipp}||\cdot \boldsymbol{n}_{ipp}$ of equation (2.29), is performed for all interfaces of the mesh;
2. then the derivation of $\boldsymbol{\nabla u}_i^{\text{5p-cor}}$ and $\boldsymbol{F}_{\boldsymbol{v}_i}^{\boldsymbol{u}}$ is addressed.

In the previous implementation of *elsA*, the viscous mean-flow 5p-cor fluxes were not fully linearized. The so-called thin-layer approximation (THL) was hold. Namely, the correction defined in equation (2.29) was not taken into consideration during the linearization thus only (2.28) was differentiated. This allowed a more straightforward derivation although introducing errors.

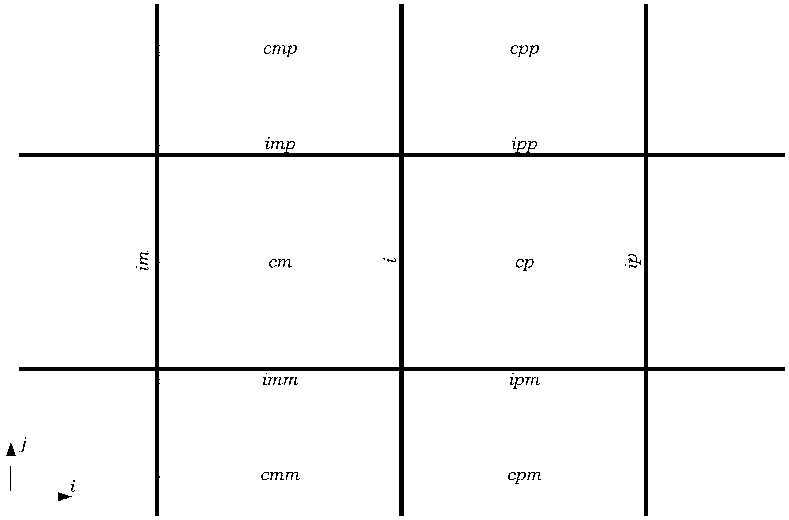


Figure 2.3: 2D structured mesh stencil for viscous 5p-cor flux at an interface i.

2.4.1 Verification on a NACA0015 airfoil

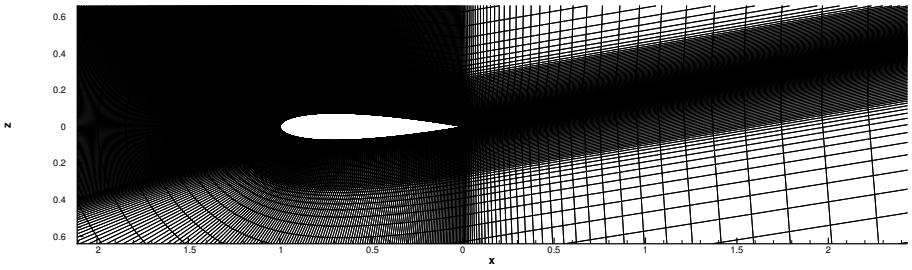


Figure 2.4: Structured C-mesh for the NACA0015 airfoil.

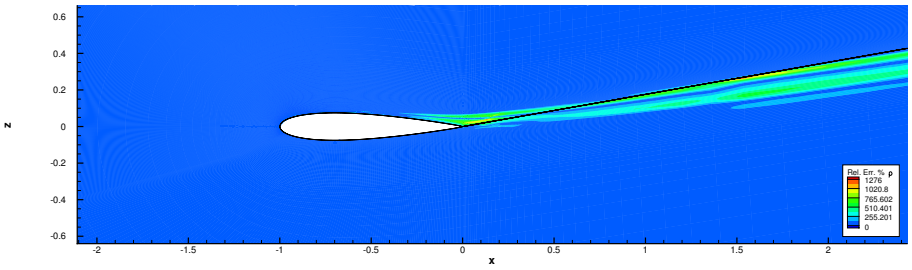
The proper linearization of the viscous mean-flow 5p-cor fluxes may be verified through the comparison of the flow sensitivity w.r.t. the shape parameter, denoted $d\boldsymbol{W}/d\alpha$, obtained from the linearized Navier-Stokes equations (Lin) and from finite differences (FD). The fully turbulent flow over a NACA0015 airfoil is considered. The turbulence closure is ensured by the Spalart-Allmaras model. The flow is at $Re = 1.955 \cdot 10^6$ based on the chord, $M_\infty = 0.291$ and $AoA = 10°$. The mesh is depicted in Figure 2.4. The considered shape parameter is an angle of rigid rotation around the trailing edge (TE). The FD results have been obtained through a 5-point formula.

Figures 2.5 and 2.6 show the relative percentage error between the computed flow sensitivity and the FD one w.r.t. the domain-averaged conservative variable

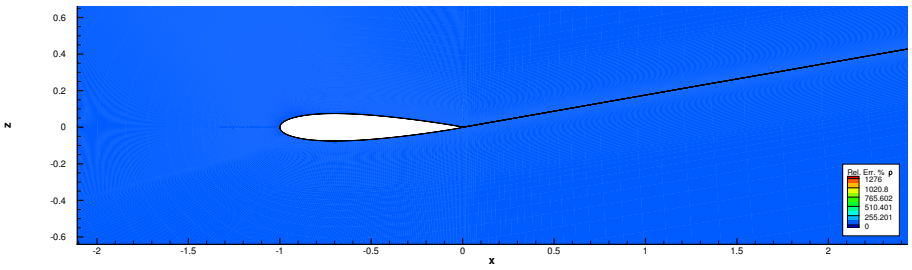
into consideration, *i.e.* $|d\rho/d\alpha_{\text{Lin}} - d\rho/d\alpha_{\text{FD}}|/|\bar{\rho}|$. In order to show the asset given by the full linearization of the viscous mean-flow 5p-cor fluxes, the error given by the previous implementation based on the thin-layer approximation (THL) is also given. It is possible to notice the sharp decrease of the relative error while using the full linearization for the ρ, ρu[6] and ρE sensitivities. Concerning $d\rho\tilde{\nu}/d\alpha$, the overall error is reduced as well but some issues have arisen at the interface join in the wake. The author believes that this behaviour is not linked to the linearization of fluxes itself, but on the numerical treatment of this interface.

The corresponding adjoint counterpart for this implementation has been equally coded and verified through duality tests.

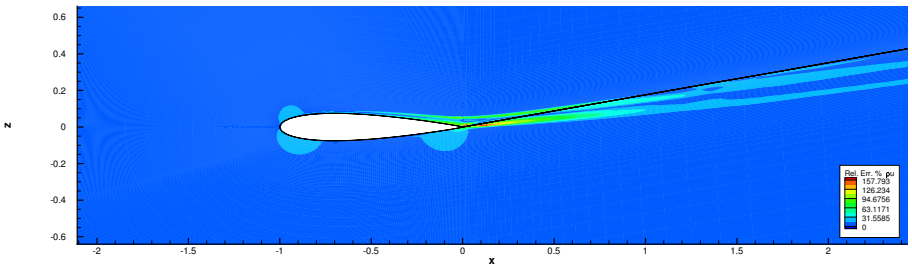
[6]The same applies for ρw



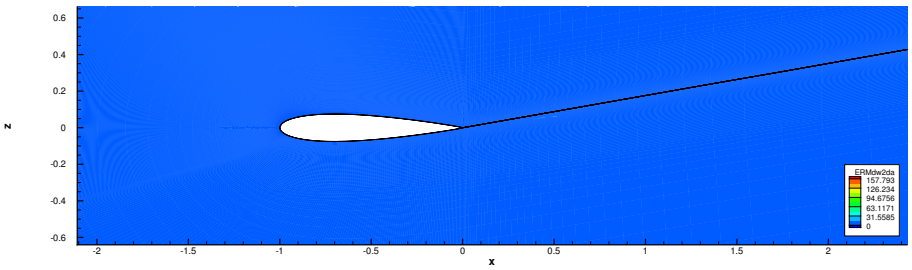
(a) ρ sensitivity. THL approximation.



(b) ρ sensitivity. Full linearization.

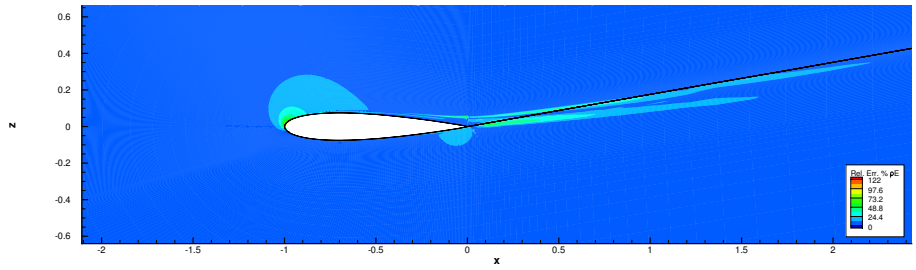


(c) ρu sensitivity. THL approximation.

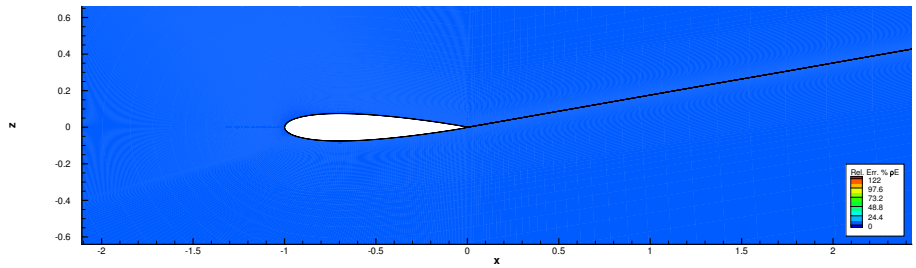


(d) ρu sensitivity. Full linearization.

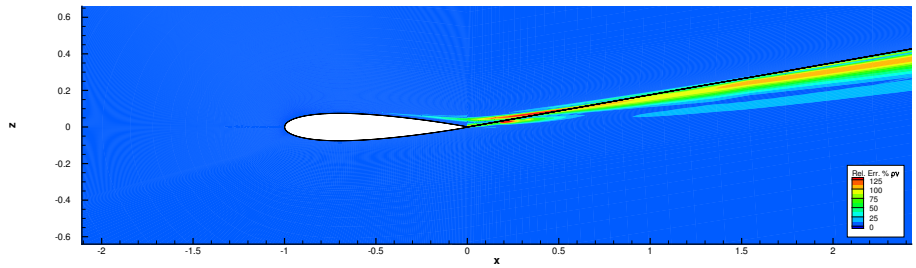
Figure 2.5: NACA0015. Relative percentage error of the computed flow sensitivity w.r.t. a TE rigid rotation. Comparison of THL approximation and full linearization.



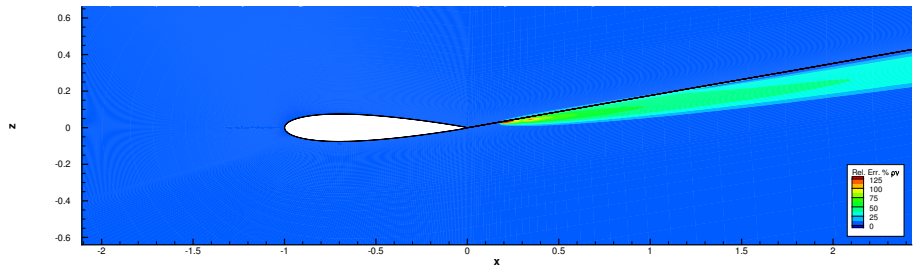
(a) ρE sensitivity. THL approximation.



(b) ρE sensitivity. Full linearization.



(c) $\rho\tilde{\nu}$ sensitivity. THL approximation.



(d) $\rho\tilde{\nu}$ sensitivity. Full linearization.

Figure 2.6: NACA0015. Relative percentage error of the computed flow sensitivity w.r.t. a TE rigid rotation. Comparison of THL approximation and full linearization.

2.5 Linearization of the SA turbulence model w.r.t. the mesh coordinates

In this thesis, the Spalart-Allmaras (SA) turbulence model [1] is considered. Recalling equations (1.1) and (1.2) of Chapter 1, the sixth steady RANS equations reads as follows:

$$\oint_{\partial V} \left(\rho\tilde{\nu}\boldsymbol{U}^T - \frac{1}{\sigma}(\mu + \rho\tilde{\nu})\nabla\tilde{\nu}^T \right) \cdot \boldsymbol{n} \mathrm{d}S - \int_V \boldsymbol{\mathcal{S}} \mathrm{d}V = 0 \tag{2.32}$$

where ρ is the density, $\boldsymbol{U} = (u, v, w)^T$ the velocity vector and $\tilde{\nu}$ the kinematic turbulent viscosity which is defined by the turbulence model, in this case SA. V is the cell volume and the unit normal vector to ∂V is denoted $\boldsymbol{n}$. $\boldsymbol{\mathcal{S}}$ is the source term composed by a production $\boldsymbol{\mathcal{P}}$, destruction $\boldsymbol{\mathcal{D}}$ and cross-diffusion $\boldsymbol{\mathcal{C}}$ terms. As mentioned earlier, the analytical differentiation of $\boldsymbol{R}$ w.r.t. flowfield $\boldsymbol{W}$ (in equation 2.17) and metric $\boldsymbol{X}$ (in equation 2.19) terms needs to be undergone. It is possible either to differentiate only the five mean flow equations without differentiating the turbulent dynamic viscosity μ_T, *i.e.* equation (2.32) (frozen μ_T assumption, $\bar{\mu}_T$) or to fully differentiate the equations system including the turbulence model (μ_T^{lin}).

The convection term is discretized using first-order Roe's flux. The discrete diffusion term is based on cell-centred gradient of $\tilde{\nu}$. The gradients at two adjacent cells are averaged at the common interface and corrected in the direction linking the two cells' centres, from $\tilde{\nu}$ cell values in order to have a short stencil in the considered direction. Discrete $\boldsymbol{\mathcal{P}}$, $\boldsymbol{\mathcal{D}}$ and $\boldsymbol{\mathcal{C}}$ terms are calculated from cell-centred values and gradients. It is then needed to derive these schemes w.r.t. mesh nodes coordinates.

2.5.1 Turbulent convection terms

The centred turbulent convection flux reads as follows

$$\boldsymbol{F}_{\boldsymbol{c}_i} = \frac{1}{2}\left(\rho\tilde{\nu}\boldsymbol{u}_{cp} + \rho\tilde{\nu}\boldsymbol{u}_{cm}\right)||S_i|| \cdot \boldsymbol{n} - \frac{1}{2}h_i\left(\rho\tilde{\nu}_{cp} - \rho\tilde{\nu}_{cm}\right) \tag{2.33}$$

where, in the *elsA* code, the second term is the so-called Roe correction with h_i as the Harten correction coefficient at the interface i. Given the following relations

$$\boldsymbol{u}_i = \frac{\rho \boldsymbol{u}_{cm} + \rho \boldsymbol{u}_{cp}}{\rho_{cm} + \rho_{cp}}, \tag{2.34}$$

$$q_i = ||\boldsymbol{u}_i||, \tag{2.35}$$

$$E_i = \frac{\rho E_{cm} + \rho E_{cp}}{\rho_{cm} + \rho_{cp}}, \tag{2.36}$$

$$a_i = \sqrt{\gamma(\gamma - 1)\left(E_i - \frac{1}{2}q_i^2\right)}, \tag{2.37}$$

$$\lambda_i = |\boldsymbol{u}_i \cdot ||S_i|| \cdot \boldsymbol{n}|, \tag{2.38}$$

$$\sigma_i = \texttt{tharten} \cdot \texttt{hartgeom} \cdot a_i \cdot ||S_i||, \tag{2.39}$$

where `tharten` is user-defined and `hartgeom` $= 1$, the Harten correction coefficient h_i is computed in the following fashion

$$h_i = \left[\frac{1}{2} + \text{sign}\left(\frac{1}{2}, \lambda_i - \sigma_i\right)\right]\lambda_i + \frac{\left[\frac{1}{2} - \text{sign}\left(\frac{1}{2}, \lambda_i - \sigma_i\right)\right]\frac{1}{2}\left(\lambda_i^2 + \sigma_i^2\right)}{\left[\frac{1}{2} + \text{sign}\left(\frac{1}{2}, -\sigma_i\right)\right]\epsilon + \left[\frac{1}{2} - \text{sign}\left(\frac{1}{2}, -\sigma_i\right)\right]\sigma_i}. \tag{2.40}$$

In the convection term, the derivatives with respect to the geometric terms are given by $\frac{\partial \boldsymbol{F}_{\boldsymbol{c}_i}}{\partial ||S_i||}$ and $\frac{\partial \boldsymbol{F}_{\boldsymbol{c}_i}}{\partial \boldsymbol{n}}$.

2.5.2 Turbulent diffusion terms

The turbulent diffusion term in a 5p-cor scheme is given by the following relations:

$$\boldsymbol{\nabla}\tilde{\nu}_i = \frac{1}{\frac{1}{2}(V_{cp} + V_{cm})}\frac{1}{2}\left((||S_i|| + ||S_{ip}||) \cdot \boldsymbol{n}_i\tilde{\nu}_{cp} - (||S_i|| + ||S_{im}||) \cdot \boldsymbol{n}_i\tilde{\nu}_{cm}\right) \tag{2.41}$$

$$\begin{aligned}\boldsymbol{\nabla}\tilde{\nu}_i^{\text{5p-cor}} &= \boldsymbol{\nabla}\tilde{\nu}_i + \\ &+ \frac{1}{\frac{1}{2}(V_{cp} + V_{cm})}\frac{1}{4}\left((\tilde{\nu}_{cpp} + \tilde{\nu}_{cp})||S_{ipp}|| \cdot \boldsymbol{n}_{ipp} - (\tilde{\nu}_{cpm} + \tilde{\nu}_{cp})||S_{ipm}|| \cdot \boldsymbol{n}_{ipm} + \right. \\ &+ \left.(\tilde{\nu}_{cmp} + \tilde{\nu}_{cm})||S_{imp}|| \cdot \boldsymbol{n}_{imp} - (\tilde{\nu}_{cmm} + \tilde{\nu}_{cm})||S_{imm}|| \cdot \boldsymbol{n}_{imm}\right),\end{aligned} \tag{2.42}$$

$$\boldsymbol{F}_{\boldsymbol{v}_i} = \frac{1}{2}(C_{cp} + C_{cm})\left(\boldsymbol{\nabla}\tilde{\nu}_i^{\text{5p-cor}} \cdot ||S_i|| \cdot \boldsymbol{n}_i\right), \tag{2.43}$$

where $C = \frac{1}{\sigma_{\tilde{\nu}}}(\mu + \rho\tilde{\nu})$. Whilst the $\boldsymbol{F}_{\boldsymbol{v}_i}$ is written in a 5p-cor formulation as for equation (2.31), the derivation w.r.t. the geometry is more straightforward since it is not needed to know the *nature* of the stencil' cells. The derivation is performed in one single step.

2.5.3 Turbulent source terms

As mentioned earlier, the turbulent source term consists in the three terms hereafter detailed.

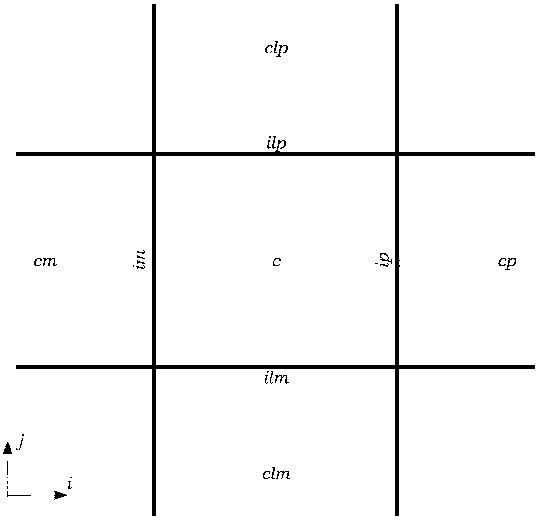


Figure 2.7: 2D structured mesh stencil for SA source term at a cell c.

Production term

$$\mathcal{P}_c = C_{b1}\rho\tilde{\nu}_c\tilde{S} \tag{2.44}$$

$$\tilde{S} = ||\boldsymbol{\omega}_c|| + \tilde{\nu}_c f_c^{v2}\frac{1}{\kappa^2\eta_c^2} \tag{2.45}$$

$$f_c^{v2} = 1 - \frac{\chi_c}{1+\chi_c f_c^{v1}} \tag{2.46}$$

$$f_c^{v1} = \frac{\chi_c^3}{\chi_c^3 + C_{v1}} \tag{2.47}$$

$$\chi_c = \frac{|\rho\tilde{\nu}_c|}{\mu_c}, \tag{2.48}$$

where η_c is the distance from the c cell centre to the wall and $\boldsymbol{\omega}_c$ is the vorticity. In the production term $\mathcal{P}_c$, the geometrical dependence comes from these two terms. The first component of $\boldsymbol{\omega}_c$ is indeed:

$$\begin{aligned}\frac{\partial u}{\partial y} = \frac{1}{V_c}\Bigg(&\frac{1}{2}(u_c+u_{cp})||S_{ip}||\cdot n_y - \frac{1}{2}(u_{cm}+u_c)||S_{im}||\cdot n_y \\ &+ \frac{1}{2}(u_c+u_{clp})||S_{ilp}||\cdot n_y - \frac{1}{2}(u_{clm}+u_c)||S_{ilm}||\cdot n_y\Bigg).\end{aligned} \tag{2.49}$$

Similarly for the remaining 3 components (5 in 3D[7]). The contribution of the geometrical dependence given by η_c has been omitted in this $elsA$ implementation. The reason for this choice resides in a current constraint given by the code. Indeed, given a volume cell c, it is not generally possible to know the coordinates of the centre of the wall interface from which the distance η_c has been computed. The derivation of η_c needs then to be skipped by treating this value as a geometry-free variable[8].

Cross-diffusion term

$$\mathcal{C}_c = \frac{C_{b2}}{\sigma}\rho_c \boldsymbol{\nabla}\tilde{\nu}_c \cdot \boldsymbol{\nabla}\tilde{\nu}_c, \tag{2.50}$$

$$\mathcal{C}_c = \min\left(\mathcal{C}_c, 20\rho\tilde{\nu}_c \max(0, C_{b1}\tilde{S})\right). \tag{2.51}$$

In the cross-diffusion term $\mathcal{C}_c$, the geometrical dependence comes from the terms $\boldsymbol{\nabla}\tilde{\nu}_c$ and $\tilde{S}$ as in $\mathcal{P}_c$. While deriving, it is important to distinguish the three cases of equation (2.51).

Destruction term

$$\mathcal{D}_c = C_{w1} f_c^w \frac{(\rho\tilde{\nu}_c)^2}{\rho_c \eta_c^2}, \tag{2.52}$$

$$f_c^w = g_c\left(\frac{1 + C_{w3}^6}{g_c^6 + C_{w3}^6}\right)^{\frac{1}{6}}, \tag{2.53}$$

$$g_c = \tilde{r}_c + C_{w2}\left(\tilde{r}_c^6 - \tilde{r}_c\right), \tag{2.54}$$

$$\tilde{r}_c = \max\left(0, \min\left(10, \frac{1}{\max\left(\tilde{S}, \texttt{cutoff}\right)}\frac{1}{\kappa^2\eta_c^2}\tilde{\nu}_c\right)\right). \tag{2.55}$$

Also in the destruction term $\mathcal{D}_c$, the geometrical dependence comes from the terms $\tilde{S}$ and η_c.

The model coefficients are listed in Table 2.2.

[7]It is recalled that the discussion is given in 2D while the implementation has been done in 3D thus the z component of equation (2.49) has been omitted

[8]For some simple cases, it has been possible to include the derivation of the distance η_c. Preliminary tests have shown that the influence of omitting this term negligible.

Table 2.2: The Spalart-Allmaras model coefficients.

Coefficient	Value	Coefficient	Value
C_{b1}	0.1355	C_{w1}	$\frac{C_{b1}}{\kappa^2} + \frac{1+C_{b2}}{\sigma}$
C_{b2}	0.622	C_{w2}	0.3
σ	$\frac{2}{3}$	C_{w3}	2
κ	0.41	C_{v1}	7.1

2.5.4 Verification on a AS28G wing

Recalling equation (2.18), in the *elsA* implementation, the $d\mathcal{J}_m/d\alpha_l$ may be computed in two fashions:

$$\left(\frac{d\mathcal{J}_m}{d\alpha_l}\right)_{\text{AdjParam}} = \frac{\partial \text{J}_m}{\partial \boldsymbol{X}}\frac{d\boldsymbol{X}}{d\alpha_l} + \boldsymbol{\Lambda}_m^T\left(\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{X}}\frac{d\boldsymbol{X}}{d\alpha_l}\right)_{\text{FD}}, \tag{2.56}$$

$$\left(\frac{d\mathcal{J}_m}{d\alpha_l}\right)_{\text{AdjMesh}} = \frac{dJ_m}{d\boldsymbol{X}}\frac{d\boldsymbol{X}}{d\alpha_l}. \tag{2.57}$$

In the previous implementation of the code, the frozen turbulence assumption was still the only option in the AdjMesh method. The derivation introduced in Section 2.5 has allowed the use of μ_T^{lin} also in AdjMesh. In order to check the proper implementation in the code, it is important to verify whether the AdjParam and AdjMesh (with the 5p-cor scheme) give the same results with the μ_T^{lin}.

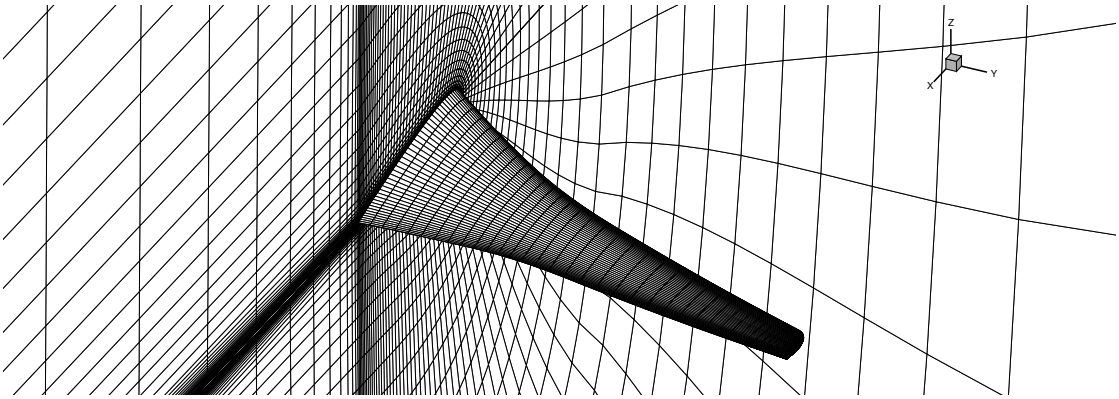


Figure 2.8: Footprint of the structured mesh on the AS28G wing surface and on the symmetry plane of the computational domain.

The AS28G wing is considered. The fully turbulent flow is at $Re = 1.49 \cdot 10^6$ based on the chord, $M_\infty = 0.8$ and $AoA = 2.2°$. The turbulence closure is ensured by the Spalart-Allmaras model. The mesh has 414050 nodes and it is depicted in

Figure 2.8. The shape parameter α is an increasing angular twisting from the root to the wing tip. The considered objective aerodynamic functional is the pressure lift coefficient, $C_{L,p}$. Table 2.3 shows the results for the $dC_{L,p}/d\alpha$ obtained from the *elsA* adjoint module and FD. As it is possible to see, the previous implementation of AdjMesh (denoted with a † in the Table) was producing a different value from the one given by AdjParam with the THL approximation. The reason was indeed in the different treatment of the turbulent closure where the term $\Lambda^6 \frac{\partial R^6}{\partial \boldsymbol{X}}$ in equation (2.19) was missing. The result obtained from the new implementation of AdjMesh taking into account the linearization of the turbulence model w.r.t. the geometry (denoted with a $*$ in the Table) matches the one of AdjParam (full 5p-cor)[9].

Table 2.3: AS28G. Results of the $dC_{L,p}/d\alpha$ obtained from AdjParam and AdjMesh methods with THL and 5p-cor approaches for the mean-flux viscous terms and with frozen or linearized μ_T. The previous *elsA* implementation is denoted with a † while the new one with a $*$.

Method	Mean-flow viscous term lin.	Tubulence closure lin.	$\frac{dC_{L,p}}{d\alpha}$ $[\cdot 10^{-2}]$	Rel. Err. %
FD	-	-	-3.06	-
AdjParam	THL	(μ_T^{lin})	-2.30	25
AdjParam	full 5p-cor	(μ_T^{lin})	-2.34	24
AdjMesh†	THL	$(\bar{\mu}_T)$	-1.97	36
AdjMesh*	full 5p-cor	(μ_T^{lin})	-2.34	24

[9]Please note that after the development explained in Section 2.4, the THL approximation option has been suppressed in the *elsA* code. For this reason, in Table 2.3, the result AdjMesh THL with μ_T^{lin} is missing. It is finally possible to notice the influence of the sole full linearization of the 5p-cor viscous fluxes detailed in Section 2.4 by comparing the results given by the two AdjParam. Namely, a reduction of 1% of the relative error.

Chapter 3

Adjoint-based mesh adaptation techniques. Application to 2D RANS flow.

Successful theory of *a posteriori* error and goal-oriented mesh adaptation has been developed in the finite-element framework since the second half of the 90s. Important contributions have been given by Johnson, Rannacher, Becker and co-workers [52, 53, 54], Giles, Pierce and co-workers [55], Prudhomme et al. [56], Larson and Barth [57], Machiels et al. [58], Hartman and co-workers [59, 60, 61] and Alauzet, Dervieux and co-workers [62]. Concerning finite-volume, the major reference is the Venditti and Darmofal method [63, 64, 65]. A list of main applications involving the latter method can be found in [66].

This chapter is focused on the discretization error and more precisely on the accuracy of the output functionals. Often, only some quantities of interest (QoI), *e.g.* forces and moments, rather than the overall flow-field, are scrutinized in aerodynamic simulations. In this framework, the development of goal-oriented mesh adaptation techniques has seen a rapid growth since the mid 90's. Most of the time, these methods involve the adjoint vector of the QoI. The present Chapter presents an enhanced goal-oriented mesh adaptation method based on aerodynamic functional total derivatives with respect to mesh nodes in a finite-volume structured grid framework for RANS flow. The method principles have been already introduced by Peter *et al.* [66] and Nguyen-Dinh *et al.* [67] with applications to Euler flows. In contrast to the Venditti and Darmofal method which requires two levels of meshes, the proposed method is based on a scalar indicator for one mesh level only. The use of only one mesh level is quite rare in literature, nevertheless it is possible to notice the contribution of Dwight [68, 69] where only on mesh level is needed but limited to the classical Jameson *et al.* numerical scheme [70]. Feasibility studies on goal-oriented error estimation for incompressible flows for grid adaptation purposes have been also addressed [71].

Using the proposed method, chapter's objectives are to properly adapt the mesh taking into account the precious information coming from the complete linearisation of the complete Navier-Stokes equations and to show the flexibility of the approach in engineering applications where non-traditional mesh configurations may be used. In this context, adaptations are carried out on standard mono-block structured grid as well as to non-matching[1] multi-block structured mesh for 2D RANS flows. The latter kind of adaptations are uncommon in literature while their use is quite usual for complex geometry configuration in the presence of structured mesh. Structured grid may be disadvantageous with respect to unstructured one in term of number of nodes (a local refinement spreads through the entire computational domain) and in term of mesh construction flexibility (a structured with well defined quad elements mesh is hardly achievable on complex configurations). Nevertheless, from a numerical point view, they turn out to be algorithmically efficient due to their simple addressing approach based on regular connectivity, *i.e.* through only 2 (or 3) indices for 2D case (or 3D case). The use of non-matching multi-block meshes alleviates the constraints given by structured mesh. An example of this is shown.

Section 3.1 reviews the main adjoint-based goal-oriented mesh adaptation methods available in literature while Section 3.2 presents the proposed mesh adaptation indicator based on aerodynamic functional total derivatives with respect to mesh nodes. The application to a 2D RANS flow about a RAE2822 airfoil is given in Section 3.3.

Nomenclature

Abbreviations

AoA	Angle of Attack
BL	Boundary Layer
QoI	Quantity of Interest
RANS	Reynolds-Averaged Navier-Stokes
SA	Spalart-Allmaras turbulence model

Greek letters

ϵ	Cell width
$\mathbf{\Lambda}$, $\mathbf{\Lambda}_m$	Adjoint vector of $\mathcal{J}$, $\mathcal{J}_m$
ϕ	Interpolation operator
θ	Mesh refinement indicator

Latin letters

$\mathcal{B}$	Interpolation operator
C	Airfoil chord
c_i	Mesh refinement criterion for mesh lines i
$\bar{c}_{i+\frac{1}{2}}$	Mesh refinement criterion for rows of cells i
C_D	Drag coefficient
C_L	Lift coefficient
h,h_0, h_j	Cell mesh hight
J	Aerodynamic function as a function of $\boldsymbol{X}$ only
M	Mach number
$\boldsymbol{n}$	Unit normal vector
N_i,N_j	Number of mesh lines

[1] In literature, the term *non-conforming* is also used to group non-conventional mesh types. In this chapter, the term *non-matching* is preferred in order to strengthen the point on the type of non-conforming mesh used.

N_{TOT}	Total number of mesh nodes
P	Projection operator
$r_{i,j}$	Admissible mesh displacement
$\boldsymbol{R}$	Discrete residual vector
Re	Reynolds number
$\boldsymbol{X}$	Mesh coordinates
y^+	Dimensionless wall distance

Subscripts

i	Direction along the airfoil surface
j	Direction normal to the airfoil surface

3.1 Literature review of adjoint-based mesh adaptation strategies for the reduction of the discretization error in a FV framework

In the following, three main mesh adaptation strategies are recalled, namely the one of Pierce & Giles, Venditti & Darmofall and Dwight. The review of Fidkowski and Darmofal [72] details the state of the art about output-based error estimation and mesh adaptation.

3.1.1 The method of Pierce and Giles

The following method has been introduced by Pierce and Giles [73, 74] and it consists in an adjoint-based error estimation for linear functions where the extension to non-linear functions is very complex and not tempted. Considering that an adjoint problem exists, in a Hilbert space $\mathcal{H}$, whose inner product is denoted $(.,.)$, it is supposed that:

1. it is possible to solve exactly (or approximately) the well-posed *direct* linear differential equation $Lw = f$ before the scalar product of the solution with another vector g of $\mathcal{H}$;
2. it is possible to solve exactly (or approximately) the well-defined corresponding adjoint problem $L^*\lambda = g$ before the dot product the solution with f.

The common goal is (g, w) which reads as follows

$$(g, w) = (L^*\lambda, w) = (\lambda, Lw) = (\lambda, f). \tag{3.1}$$

By denoting the approximate solution w_h where h represents the average mesh size, the error in the estimation of (g, w) is

$$(g, w) - (g, w_h) = (g, (w - w_h)) = (L^*\lambda, (w - w_h)) = (\lambda, L(w - w_h)) = (\lambda, f - Lw_h). \tag{3.2}$$

The latter error expression may be rewritten in the case where the adjoint problem has been equally approximately solved:

$$(g, w) - (g, w_h) = (\lambda_h, f - Lw_h) + (\lambda - \lambda_h, f - Lw_h). \tag{3.3}$$

The adjoint times the discrete residual $(\lambda_h, f - Lw_h)$ is the main error term and it is the computable correction of the output value.

3.1.2 The method of Venditti and Darmofal

For finite differences and finite-volume schemes, Venditti and Darmofal [63, 64, 65] have proposed a method for non-linear functions with applications to compressible flows. By denoting W the flow field, X the volume mesh and R the scheme's residual, the equation $R(W, X) = 0$ is satisfied at steady state. Two levels of meshes are needed, the coarse one has a characteristic mesh size H while the fine one has a characteristic mesh size h. It is considered that the computation on the h level mesh is prohibitively expensive. The coarse-grid flow-field and adjoint vector reconstructed on the fine grid *via* some consistent interpolation operator are denoted W_h^H and Λ_h^H respectively.

Given the functional output of interest J_h, the Taylor's expansion of J_h about the interpolated coarse-grid solution reads as follows

$$\mathrm{J}_h\left(W_h, X_h\right) = \mathrm{J}_h\left(W_h^H, X_h\right) + \frac{\partial \mathrm{J}}{\partial W}\Big|_{W_h^H}\left(W_h - W_h^H\right) + \mathcal{O}\left(||W_h - W_h^H||^2\right). \quad (3.4)$$

The following adjoint-like equation is defined

$$\left(\Lambda_h\big|_{W_h^H}\right)^T \frac{\partial R_h}{\partial W_h}\Big|_{W_h^H} = -\frac{\partial \mathrm{J}_h}{\partial W_h}\Big|_{W_h^H}. \quad (3.5)$$

Using equation (3.5), relation (3.4) may be rewritten in the following fashion

$$\begin{aligned}\mathrm{J}_h\left(W_h, X_h\right) &= \mathrm{J}_h\left(W_h^H, X_h\right) - \left(\Lambda_h\big|_{W_h^H}\right)^T \frac{\partial R_h}{\partial W_h}\Big|_{W_h^H}\left(W_h - W_h^H\right) + \mathcal{O}\left(||W_h - W_h^H||^2\right) \\ &= \mathrm{J}_h\left(W_h^H, X_h\right) + \left(\Lambda_h\big|_{W_h^H}\right)^T R_h\left(W_h^H, X_h\right) + \mathcal{O}\left(||W_h - W_h^H||^2\right). \end{aligned} \quad (3.6)$$

It is possible to compare equations (3.2) and (3.6) in order to establish comparisons between Pierce & Giles and Venditti & Darmofal methods, namely the product between the adjoint and the discrete residual. Relation (3.6) needs to be rewritten in the case where the adjoint solution on the fine grid is not available or not affordable. In this case indeed neither the solution of equation (3.5) is available. It is possible then to use the interpolated coarse-grid adjoint:

$$\mathrm{J}_h\left(W_h, X_h\right) \simeq \mathrm{J}_h\left(W_h^H, X_h\right) + \underbrace{\left(\Lambda_h^H\right)^T R_h\left(W_h^H, X_h\right)}_{\text{computable correction}} \quad (3.7)$$

$$+ \underbrace{\left(\left(\Lambda_h\big|_{W_h^H}\right)^T - \left(\Lambda_h^H\right)^T\right) R_h\left(W_h^H, X_h\right)}_{\text{error in computable correction}} \quad (3.8)$$

The value of $\mathrm{J}_h(W_h^H, X_h) + \Lambda_h^H R_h(W_h^H)$ is the functional estimator check by Venditti & Darmofal. In order to adapt the mesh, it is suggested to reduce uniformly the error in computable correction.

These formulae have been largely used in the CFD community. The reader is suggested to refer to Peter *et al.* [66] for a comprehensive review of articles with applications to unstructured, structured, embedded-boundary Cartesian meshes in Euler and RANS flows.

Concerning RANS flows, it is important to underline the poor conditioning of the Jacobian in the case where the turbulence model is linearized in complex configuration. This is one of the reason for which the *frozen turbulent viscosity* approximation is still largely adopted alongside with the complexity of the linearization of the models themselves.

3.1.3 The method of Dwight

The adjoint-based method proposed by Dwight [68, 69] is closely dependent on the discretization scheme. Namely, to the Jameson *et al.* scheme [70] for Euler flows. Considering the artificial dissipation coefficients $(k^{(2)}, k^{(4)})$, the functions of interest's error appeared to be mainly due to artificial dissipation. Dwight proposed the following estimator:

$$\eta = k^{(2)} \frac{d\mathrm{J}}{dk^{(2)}} + k^{(4)} \frac{d\mathrm{J}}{dk^{(4)}} \tag{3.9}$$

The adjoint method allows to compute the derivatives of the latter relation, $\frac{d\mathrm{J}}{dk^{(2),(4)}} = \Lambda^T \frac{dR}{dk^{(2),(4)}}$. By defining independently the dissipation coefficients in each control volume i, the local estimator reads as follows

$$\eta_i = \Lambda \left(k_i^{(2)} \frac{dR}{dk_i^{(2)}} + k_i^{(4)} \frac{dR}{dk_i^{(4)}} \right). \tag{3.10}$$

The local indicator η_i is used to drive the mesh refinement. The value $\mathrm{J} - k^{(2)} d\mathrm{J}/dk^{(2)} - k^{(4)} d\mathrm{J}/dk^{(4)}$ is considered as the corrected output value.

3.2 Use of dJ/dX for mesh adaptations

The proposed method is based on the $dJ_m/d\boldsymbol{X}$ introduced in equation (2.19) of Chapter 2. The scalar refinement indicator based on a single mesh level is presented in the following section. Section 3.3 reports its use on 2D airfoil applications.

3.2.1 The refinement indicator

The $dJ/d\boldsymbol{X}$[2] field provides valuable information about the sensitivity of J with respect to the mesh coordinates $\boldsymbol{X}$. Considering the following first order Taylor expansion

$$J(\boldsymbol{X} + d\boldsymbol{X}_{i,j}) \approx J(\boldsymbol{X}) + \frac{dJ}{d\boldsymbol{X}_{i,j}} d\boldsymbol{X}_{i,j}, \tag{3.11}$$

the effect on J due to an isolated move of a node $\boldsymbol{X}_{i,j}$ in the polygon defined by the neighboring nodes is linked to the sensitivity $dJ/d\boldsymbol{X}_{i,j}$. The latter is true for a displacement $d\boldsymbol{X}_{i,j}$ for which the Taylor expansion is valid. This criterion suggests that the objective function J is highly sensible to the $\boldsymbol{X}_{i,j}$ node if $dJ/d\boldsymbol{X}_{i,j}$ is large[3]. A refinement in that node location is then needed.

Some cautions need however to be taken into account on the obtained $dJ/d\boldsymbol{X}$ field. In particularly, it will be projected according to the mesh node location. Namely, $\mathsf{P}(dJ/d\boldsymbol{X}) = dJ/d\boldsymbol{X}$ if the node is outside the support[4] of J; $\mathsf{P}(dJ/d\boldsymbol{X}) = dJ/d\boldsymbol{X} - (dJ/d\boldsymbol{X} \cdot \boldsymbol{n}) \cdot \boldsymbol{n}$ if the node is on the walls, farfield (or mesh block) borders; while $\mathsf{P}(dJ/d\boldsymbol{X}) = 0$ if the node is on a corner of the support of J or on a corner of farfield (or mesh block) domain. Figure 3.1 shows three typical situations for the projected $dJ/d\boldsymbol{X}$. In particularly Figure 3.1(a) reppresents the case in which the objective functional J is highly depended on the mesh since a displacement of the nodes with an high $||dJ/d\boldsymbol{X}||$ would equally modify the J value. On the other hand, in Figure 3.1(b), the mesh may be considered already *good* since the displacement of nodes with high $||dJ/d\boldsymbol{X}||$ is not feasible. This situation occurs often in the first cells of the Boundary Layer (BL). In some cases, the situation is less clear as in Figure 3.1(c) in which one node displacement's effect on J value may be cancelled out by another node displacement.

In this thesis, the θ-criterion introduced by Peter *et al.* [66] and Nguyen-Dinh *et al.* [67] will be exploited. This indicator reads as follows:

$$\theta(i,j) = \left\| \mathsf{P}\left(\frac{dJ}{d\boldsymbol{X}}\right) \right\| r_{i,j} \tag{3.12}$$

where $r_{i,j}$ stands for the admissible mesh displacement, *i.e.* half of the distance to the nearest node. Please recall that, for structured grid, (i,j) indices uniquely identify a mesh node.

[2]The pedices m and l drop for sake of brevity.

[3]In this framework, there is no *quantitative* link between $\theta(i,j)$ and its effect on J. The term *large* is always w.r.t the ensemble of all $\theta(i,j)$ for a given $\boldsymbol{W}$ and $\boldsymbol{X}$. This justifies the use of the term *indicator* instead of *estimator*.

[4]In the current chapter applications, the *support* of J, where J is a global aerodynamic function, is typically the airfoil surface.

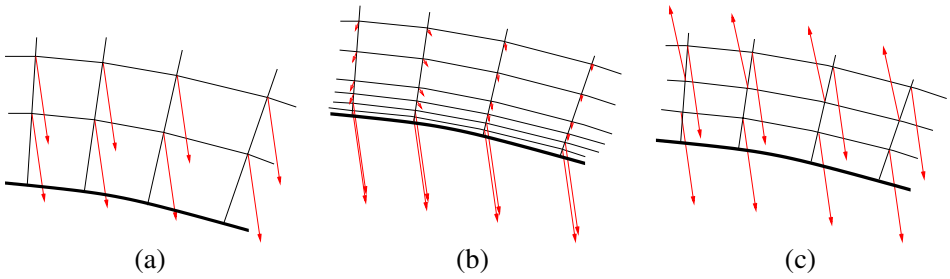


Figure 3.1: Typical situations for the projected $dJ/d\boldsymbol{X}$ from [67]. (a) Large regular $\mathsf{P}(dJ/d\boldsymbol{X})$ with large possible displacement of nodes; (b) large regular $\mathsf{P}(dJ/d\boldsymbol{X})$ without large possible displacement of nodes and (c) large non-regular $\mathsf{P}(dJ/d\boldsymbol{X})$ with large possible displacement of nodes.

Similarities can be found between the θ-criterion, in particularly $dJ_m/d\boldsymbol{X}$, and the error expression given by Venditti *et al.* in equation (12) of their article [64]. In both expressions, zones considered crucial for the computation of the QoI (through the adjoint vector $\boldsymbol{\Lambda}_m$ in equation (2.19) and R_h^{Ψ} in [64]) and zones where the sensitivity of the explicit residual is high (through the $\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{X}}$ operator in equation (2.19) and R_h in [64]) are considered as influential in the QoI estimation.

3.2.2 The *elsA* code

The ONERA Navier-Stokes finite-volume compressible *elsA* [2] solver has been used. The numerical scheme is based on Roe flux [3] using a second-order MUSCL scheme [4] (van Albada limiter [5]) for the convective term, while first-order Roe flux for the turbulent flow. The viscous fluxes of the averaged and turbulent flows are discretized through a centred scheme with cell-centred gradients corrected at the cell faces. The turbulence closure is assured by the Spalart-Allmaras turbulence model [1].

Non-matching mesh joins

The *elsA* code deals mainly with structured meshes. This kind of grids may pose issues both for complex geometry and for the generally high number of nodes needed with respect to an unstructured mesh. In order to alleviate the latter drawback, *elsA* can handle non-matching join between different structured mesh blocks which limit the spread through the entire computational domain following a local mesh refinement. This interface may be straight or also curved. In the first case the conservation property is ensured while in the second case, this approach is quasi-conservative [75]. In the examples given in Section 3.3, the non-matching multi-block strategy allows to limit the local refinement in i direction over all the domain by splitting the mesh in 3 concentric O-meshes.

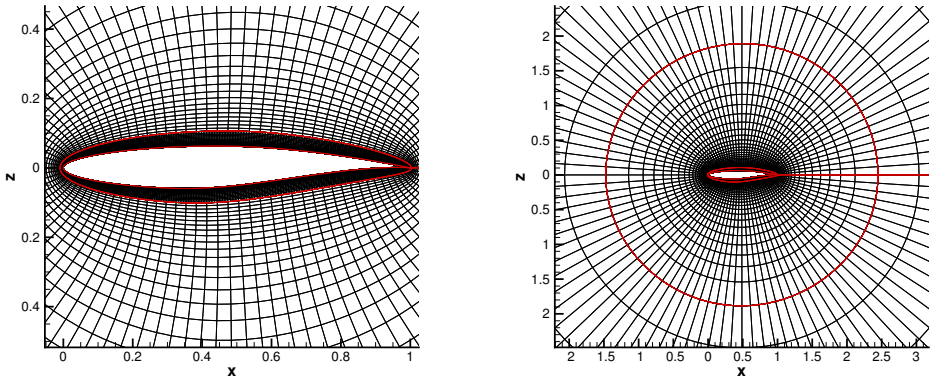
3.3 Application to a RAE2822 airfoil

The proposed mesh adaptation technique is applied to a 2D RANS flow about a RAE2822 airfoil at two different operational conditions: a transonic condition that corresponds to Case 6 of Cook *et al.* [76] equally used by Venditti *et al.* [65] and a detached subsonic condition derived from the latter by keeping the same Reynolds to Mach number ratio as in helicopter rotor operational condition. The efficiency of the proposed method is tested over a mono-block mesh as well as on a non-matching multi-block configuration. Refinement based on the functional C_D is considered which estimation is in general harder than the C_L one. For the C_D-based adapted mesh, the C_L value is also given for sake of completeness. The considered tolerance is equal to 0.0005, *i.e.* 5 drag counts. This value is in line with Li *et al.* [77]. Higher [78, 79] and lower [65] tolerances are also considered in literature.

3.3.1 Mesh characteristics

For all applications in this section, structured O-meshes has been used where the index i runs clockwise along the airfoil from the trailing edge while j goes from the airfoil wall to the farfield boundary. The circular farfield boundary is placed at 150 chords from the airfoil leading edge. The hierarchy of NACA0012 meshes from Vassberg *et al.* [80] has been modified for a RAE2822 airfoil. Moreover, they have been adapted for RANS flows by adding a boundary layer (BL) growth in j direction. The number of mesh nodes in the BL can be imposed where the BL height is computed by using the estimation of the turbulent BL over a flat plate through the one-seventh-power law. The first cell height h_0 is fixed but differs in the two operational conditions in order to have y^+ lower than 1 in both cases. The cell height h in j direction is given by $h_{j+1} = (h_j)p$ where p varies as $1 < p < 1.1$ based on the number of nodes in j, N_j.

Concerning the non-matching multi-block configuration, the original mono-block mesh has been split in three concentric O-meshes. In many engineering applications, the surface mesh refinement is constrained or needs special care. In this context, it has been decided to test the proposed method in the case where the first block mesh (which contains the airfoil) is frozen through all the adaptation procedure. This block is uniformly refined in i with 401 nodes with 40 nodes in the BL, see Figure 3.2 (a). For the two operational conditions, the first block dimension (in j direction) varies. In the subsonic case, it has been enlarged in order to ensure that the recirculation bubble is fully included in the first block. The second and third blocks are adapted following the θ-criterion. The cell height growth is conserved through the fixed block interfaces.



(a) Blocks 1 and 2

(b) Blocks 2 and 3

Figure 3.2: Starting non-matching multi-block mesh. In the special case of the starting mesh, the interface between blocks 2 and 3 is match. During the adaptation steps, mesh nodes are free to move in a non-matching framework, see for instance Figure 3.10 (b) and (c).

Refinement procedure

The refinement is carried out by adding lines in both directions i and j. The new mesh is interpolated in the reference fine grid with a number of nodes $N_i = N_j = 2049$. The nodes position is defined through the following interpolation operators ϕ and $\mathcal{B}$

$$\{1, N_i\}\{1, N_j\} \xrightarrow{\phi} [1, 2049][1, 2049] \xrightarrow{\mathcal{B}} \mathbb{R}^2. \tag{3.13}$$

In order to add lines in the current mesh, a criterion is introduced based on the θ-criterion (3.12). For the sake of brevity, only the i direction is considered. Introducing $\bar{c}$ as

$$\bar{c}_{i+\frac{1}{2}} = \frac{c_i + c_{i+1}}{2} \quad \text{where} \quad c_i = \frac{1}{N_j}\sqrt{\sum_{j=1}^{N_j} \theta_{i,j}^2}, \tag{3.14}$$

the method consists in adding a number of lines at $i + \frac{1}{2}$ which is proportional to $\bar{c}_{i+\frac{1}{2}}$ value. A maximum number equals to 4 is retained for each $i + \frac{1}{2}$ location. In the particular case of i direction, the mesh lines (cell widths) are regularized in order to ensure a smooth nodes location[5]. An explicit third-order dissipation flux is used. The cell widths ϵ are obtain through $n+1$ iterations through the relation $\epsilon_{i+\frac{1}{2}}^{n+1} = \epsilon_{i+\frac{1}{2}}^{n} + D_{i+1}^4 - D_i^4$ where $D_i^4 = -k^4(\epsilon_{i+\frac{3}{2}} - 3\epsilon_{i+\frac{1}{2}} + 3\epsilon_{i-\frac{1}{2}} - \epsilon_{i-\frac{3}{2}})$. These

[5]For j direction there is no need for a smoothing operator since the cell height h follows the before mentioned rule from the wall to the farfield boundary.

fluxes are set to zero at i equals 1 and N_i while $\epsilon_{\frac{1}{2}} = \epsilon_{\frac{3}{2}}$ and $\epsilon_{N_i+\frac{1}{2}} = \epsilon_{N_i-\frac{1}{2}}$.

3.3.2 Transonic flow

As mentioned earlier, a transonic condition is retained as in Section 6.2.1 of Venditti *et al.* [65] paper. The flow is at Reynolds number equals to $6.5{\cdot}10^6$ based on the chord, Mach number M_∞ at 0.725 with $AoA = 2.466°$. A shock wave appears at the pressure side at $x/C \approx 0.55$ where C is the chord.

The starting mesh is the same for the standard base refinement (*i.e.* by doubling $N_{i,j}$ at each step) and for the C_D-based mono-block adaptation. Namely, with $N_i = N_j = 51$ uniformly distributed in i direction and $N_{BL} = 10$. Concerning the multi-block adaptation, the first block is frozen and uniformly refined in i with 401 nodes with 40 nodes in the BL as mentioned in Section 3.3.1. This first block has been extracted from the second last base mesh of the black curve in Figure 3.3. The remaining blocks have $N_i = 101$. The C_D convergence is plotted in Figure 3.3 where the asset given by the proposed method is patent for the mono-block adaptation with respect to the standard refinement. The method equally managed to refine the non-matching multi-block case where the physical shock cross the non-matching interface between blocks 1 and 2. The limit output values may be found in Table 3.1. For both kind of meshes, the C_D estimation error is under 4 drag counts. A very good agreement also between numerical results and experiments is proved. The underestimation of C_D is visible also in Figure 8 (right) of Venditti *et al.* [65] paper through the Allmaras' computation. The C_L values show excellent agreement one to each other. The nodes saving for the mono-block case is of the order of 75% with respect to the second last base mesh. While concerning the multi-block case, please note that almost 19000 nodes out of 39500 are imposed by the frozen first block. The precision scatter between the results of the mono- and multi-block meshes may be explained by the quasi-conservative approach at high curvature interfaces (*e.g.* leading edge) between the two non-matching blocks, as mentioned in Section 3.2.2. The obtained mesh pattern is discussed in Section 3.3.4.

3.3.3 Subsonic detached flow

For the subsonic detached case, the operational condition has been derived from the transonic one by keeping the same Reynolds to Mach number ratio as in helicopter rotor operational condition. Namely, the flow is at Reynolds number equals to $2.7{\cdot}10^6$ based on the chord, farfield Mach number at 0.3 with $AoA = 11°$. The converged flow exhibits a trailing edge separation bubble at $x/C > 0.9$.

Figure 3.4 shows the C_D convergence. The starting mesh are equal to the one used for the transonic case except for the first cell height h_0 and the physical size of

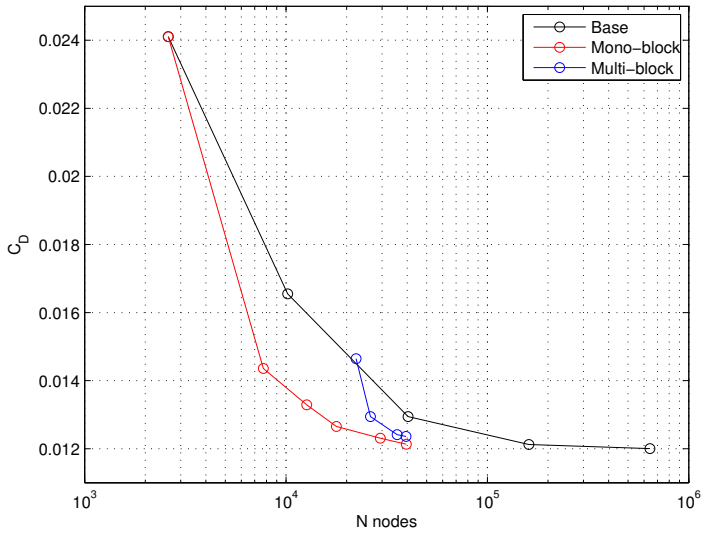


Figure 3.3: Convergence of C_D for the base solution, mono- and multi-block adaptation in the transonic flow condition.

Table 3.1: Transonic case. Comparison of C_D and C_L values obtained from the standard base refinement (finest grid) and from the C_D-based adapted mono- and multi-block meshes. Number of nodes in i and j direction are shown as well as the number of nodes in the BL and the total mesh size, N_{TOT}. The experimental values from Cook *et al.* [76] are equally reported.

	Block	N_i	N_j	N_{BL}	N_{TOT}	C_D	C_L
Experiment	-	-	-	-	-	0.0127	0.743
Base (finest)	-	801	801	80	641601	0.01200	0.744
Mono-block	-	221	179	77	39559	0.01213	0.738
Multi-block	1 (*frozen*)	401	47	40	39437	0.01235	0.743
	2	210	71	-			
	3	142	40	-			

the first block in the case of multi-block refinement as mentioned in Section 3.3.1. This justifies the different starting point in term of number of nodes of the blue curves in Figure 3.3 and 3.4. Also in this case, the proposed method well behaved w.r.t. the standard base refinement. In Table 3.2, the limit values are reported. The C_D estimation error is also in this case of maximum 3 drag counts while of 0 drag count for the mono-block case. The C_L values are also all coherent. The nodes saving for the mono-block case is of the order of 82% with respect to the

second last base mesh. For the multi-block mesh, almost 24000 nodes out of 45000 are imposed by the frozen first block. The obtained mesh pattern is discussed in the following section.

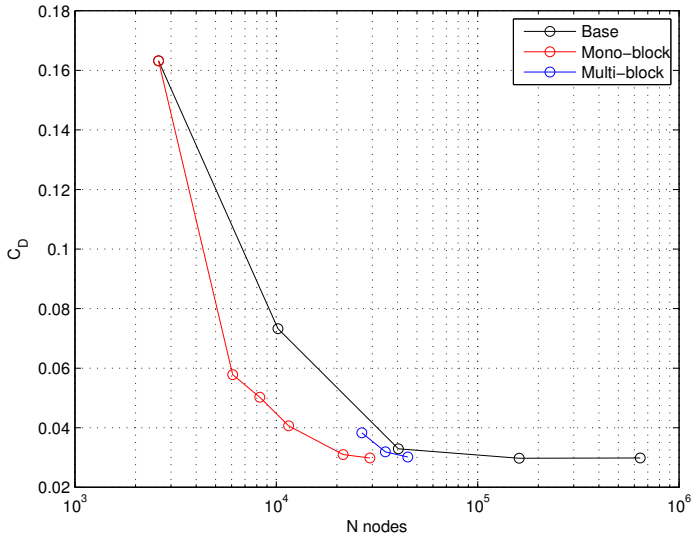


Figure 3.4: Convergence of C_D for the base solution, mono- and multi-block adaptation in the subsonic detached flow condition.

Table 3.2: Subsonic detached case. Comparison of C_D and C_L values obtained from the standard base refinement (finest grid) and from the C_D-based adapted mono- and multi-block meshes. Number of nodes in i and j direction are shown as well as the number of nodes in the BL and the total mesh size, N_{TOT}.

	Block	N_i	N_j	N_{BL}	N_{TOT}	C_D	C_L
Base (finest)	-	801	801	80	641601	0.02987	1.295
Mono-block	-	163	179	71	29177	0.02986	1.290
Multi-block	1 (*frozen*)	401	59	40	45027	0.03018	1.293
	2	178	80	-			
	3	132	54	-			

3.3.4 Analysis of the obtained mesh pattern

In order to deepen the analysis of the proposed method and of the results, it is interesting to get a closer look to the obtained anisotropic C_D-based adapted meshes as well as to the values of the θ-criterion. This analysis is carried out only for the

mono-block case for the sake of brevity. The same considerations apply to the multi-block case.

Figure 3.5 shows the starting mesh as well as the C_D-based adapted ones for both operational conditions. The image layout is similar to the one of Figure 9 of Venditti *et al.* [65] paper which allows a straightforward comparison between the obtained meshes. Taking into account the intrinsic construction differences between structured and unstructured grids, in the transonic case (b), similar areas are highlighted. Namely, the upstream flow region (typical for adjoint-based adaptation), the wake, the shock region at $x/C \approx 0.55$, leading and trailing edge. In structured grid is clear that these local refinements are convected through all the computational domain. For the subsonic case (c), refinement is performed in the upstream flow region, separation region at $x/C > 0.8$, wake, leading and trailing edge.

The Mach iso-contours are given in Figures 3.6-3.8 for the starting mesh as well as for the finest mesh of the standard base refinement and for the C_D-based adapted meshes for both operational conditions. Good agreement is proved between Figure 3.7 (a) with Figure 3.8 (a) and Figure 3.7 (b) with Figure 3.8 (b). Please note the poor initial estimation of the flow for the starting meshes, Figure 3.6.

The θ-criterion for the mono-block transonic case adaptation is shown in Figure 3.9 for all adaptation steps. It is possible to notice the progressive reduction of the θ values as well as the crucial zones for this operational condition, *e.g.* upstream flow region, the wake and the shock region at $x/C \approx 0.55$. Please note that the relation (3.12) which defines θ depends both on the local mesh size through $r_{i,j}$ but also on the $dJ/d\mathbf{X}$. For this reason, θ may vary also in zone where no mesh refinement has been performed, *e.g.* pressure side in Figure 3.9 (e)-(f).

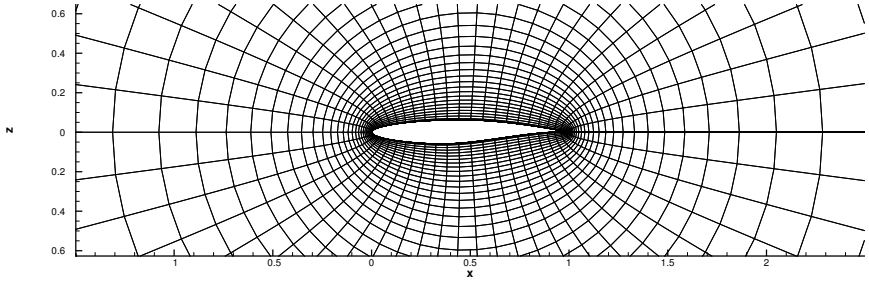
Finally, in Figure 3.10, the starting mesh as well as the C_D-based adapted ones for both operational conditions for the multi-block case are shown. The block interfaces are depicted in green. The same area highlighted in Figure 3.5 are visible here. In this case, the use of non-matching multi-block approach limits the convection of these refinements to the farfield boundaries. For the upstream flow region, the refinement crosses the interfaces and equally for the wake region but in a lightly fashion.

3.4 Some concluding remarks

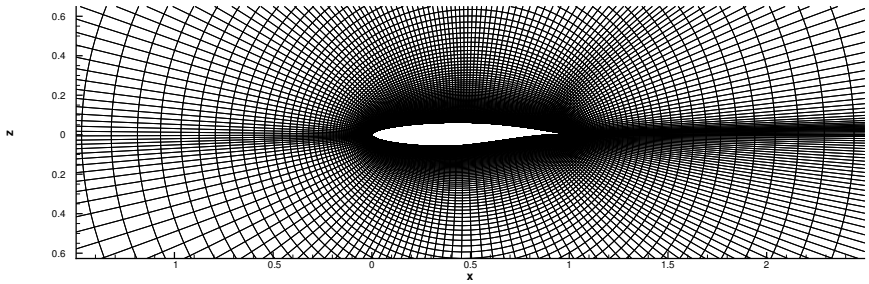
An enhanced goal-oriented mesh adaptation method has been presented based on aerodynamic functional total derivatives with respect to mesh nodes in a finite-volume mono-block and non-matching multi-block structured grid framework for

RANS flow. The linearisation of the Spalart-Allmaras turbulence model improves the precision of the estimated adjoint. The retained refinement indicator θ based on the $dJ/d\mathbf{X}$ and on the local mesh size has been proved to be a robust tool for identifying zones considered as influential in the QoI estimation in the sense of a first order Taylor expansion of J w.r.t. single node displacement, *i.e.* equation (3.11). Applications to 2D RANS flow about a RAE2822 airfoil have been presented. For the mono-block mesh case, for both aerodynamic conditions, the obtained C_D-based adapted meshes lead to a good estimation of forces. Although structured grid has been used, a clear anisotropy in the mesh pattern is visible. The constraints given by this kind of meshes have been relaxed thanks to non-matching multi-block technique which limits the spread of local mesh refinement through all the domain. In this context, the proposed method has proven to be efficient also to adapt the multi-block mesh in the presence of a more refined frozen wall mesh block. This recalls how also regions of the domain which are not close to the airfoil boundary need refinement. This is equally true for QoI (thus J) whose support is only the airfoil surface mesh (*e.g.* C_D) which is entirely included in the well refined frozen wall mesh block. The method has proven its efficiency and versatility in term of flow types (Euler [66] and RANS here), of aerodynamic operational conditions (transonic and detached subsonic) and of mesh types (mono- and multi-block under constraints). The method can be extended to unstructured grid [81] and to 3D case.

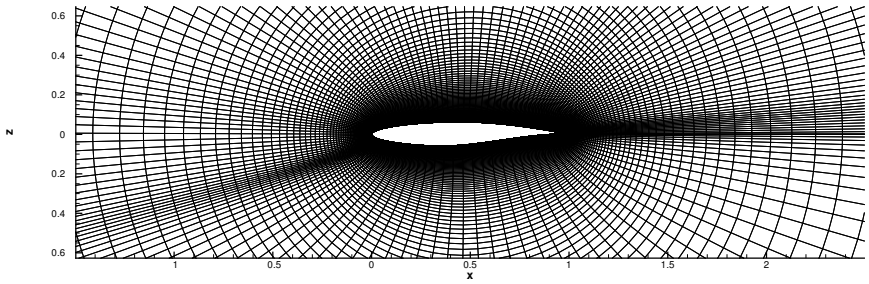
The following chapter will address the analysis of the impact of input values' uncertainty on the QoI for 2D aerodynamic simulations about an airfoil. The Uncertainty Quantification approach is a valuable tool to quantify the latter. Stochastic methods through which uncertainties (typically operational and geometrical) are taken into consideration in the physical problem are introduced. A method for improving the performance of one of this stochastic approximation is reported as well as tested on test functions.



(a) Starting mesh (2061 nodes)



(b) C_D-based adapted mesh, transonic flow (39559 nodes)



(c) C_D-based adapted mesh, subsonic flow (29117 nodes)

Figure 3.5: Comparison of mono-block meshes. The starting mesh is given in (a) while final adapted meshes in (b) and (c).

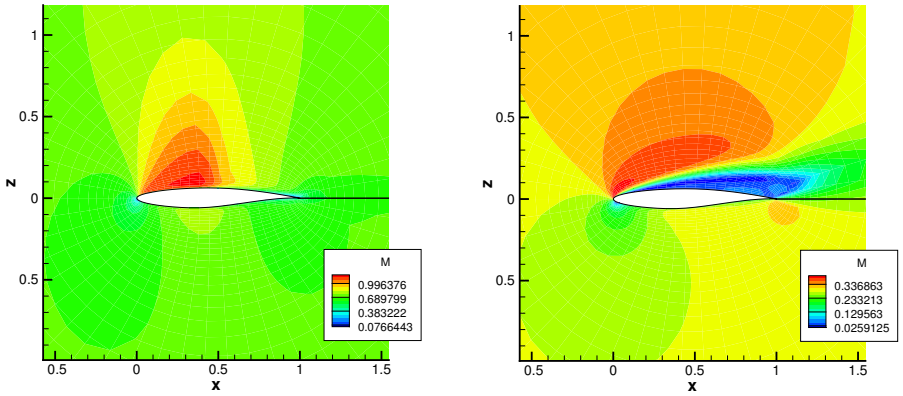


Figure 3.6: M isocontour. Mono-block starting mesh. Left: transonic, right: subsonic.

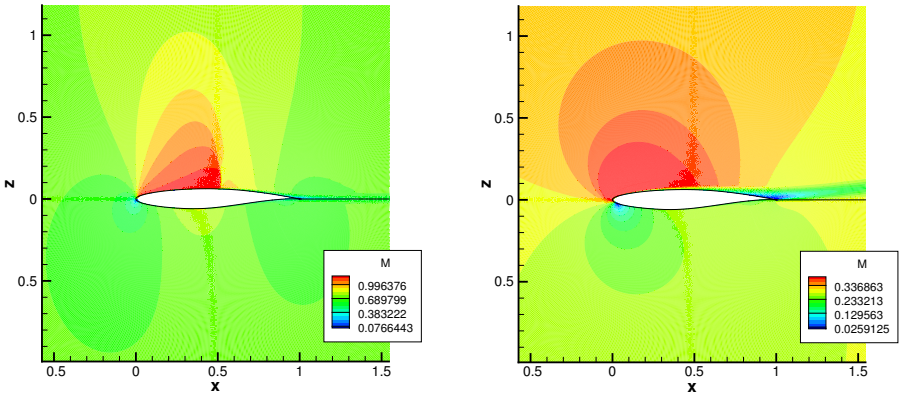


Figure 3.7: M isocontour. Mono-block finest mesh. Left: transonic, right: subsonic.

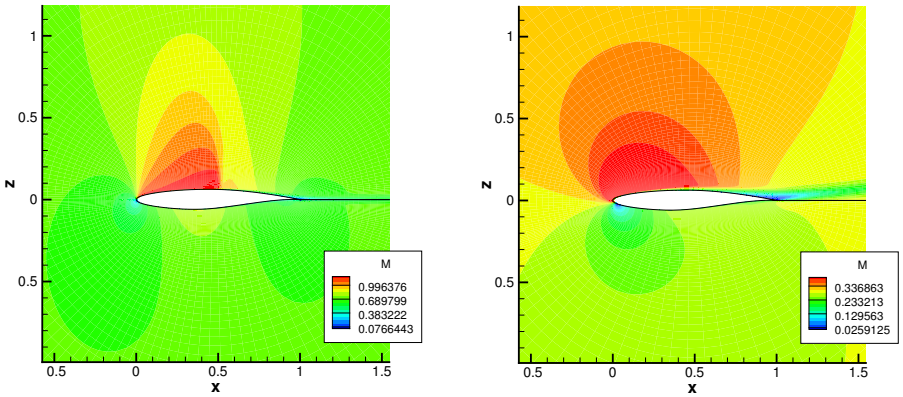
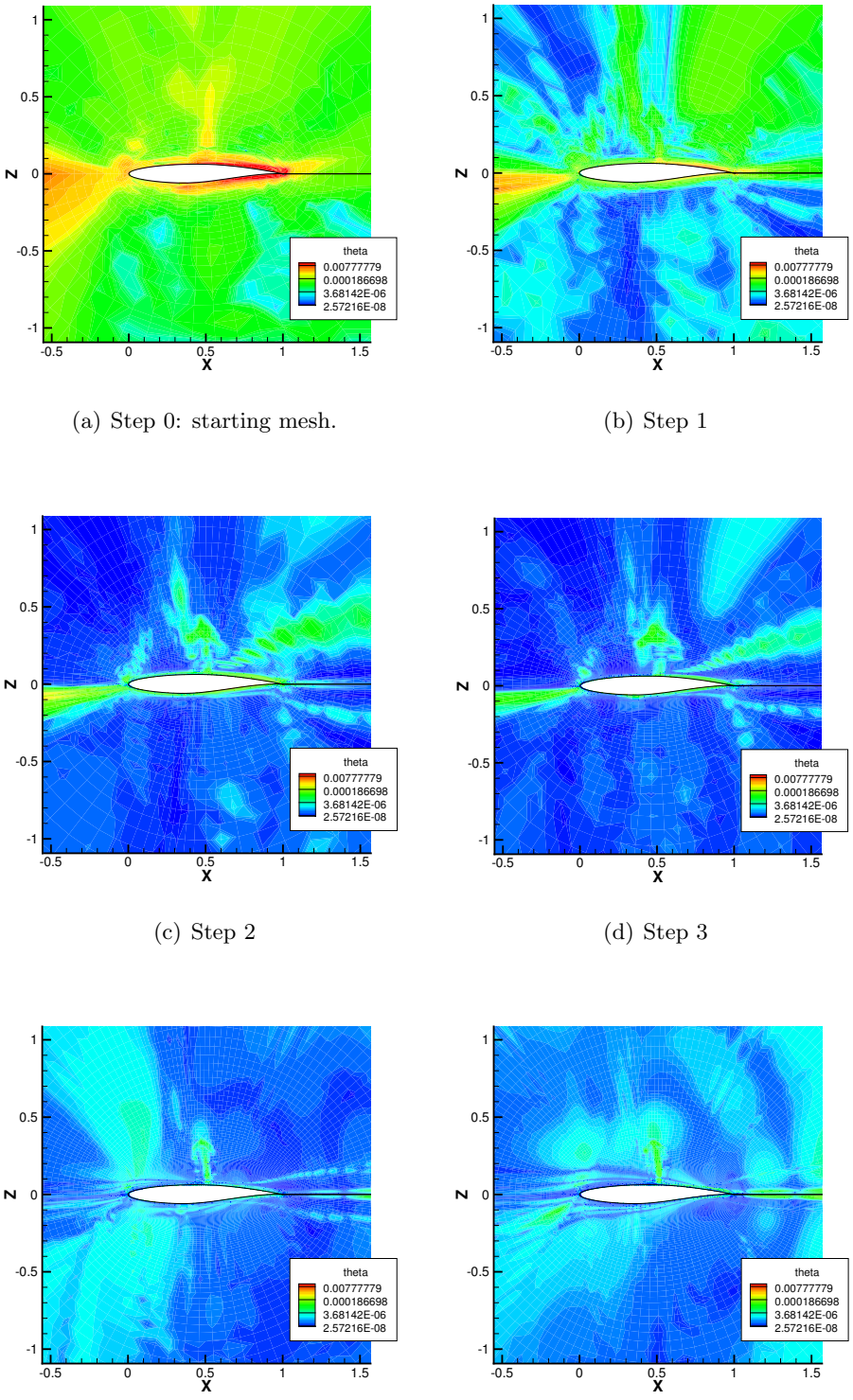


Figure 3.8: M isocontour. C_D-based adapted mono-block starting mesh. Left: transonic, right: subsonic.



(a) Step 0: starting mesh.

(b) Step 1

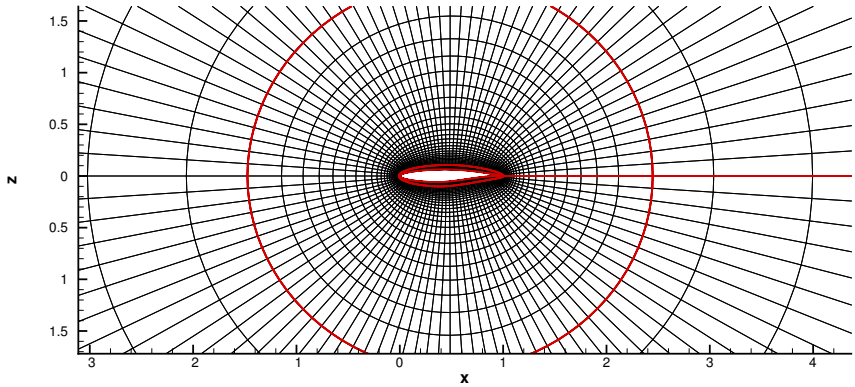
(c) Step 2

(d) Step 3

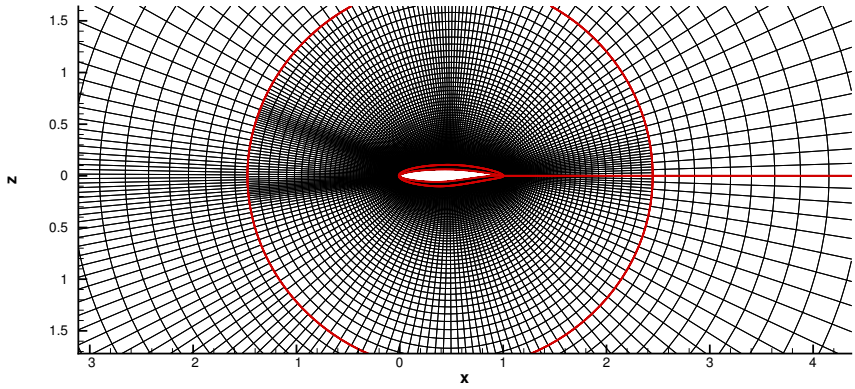
(e) Step 4

(f) Step 5: final mesh.

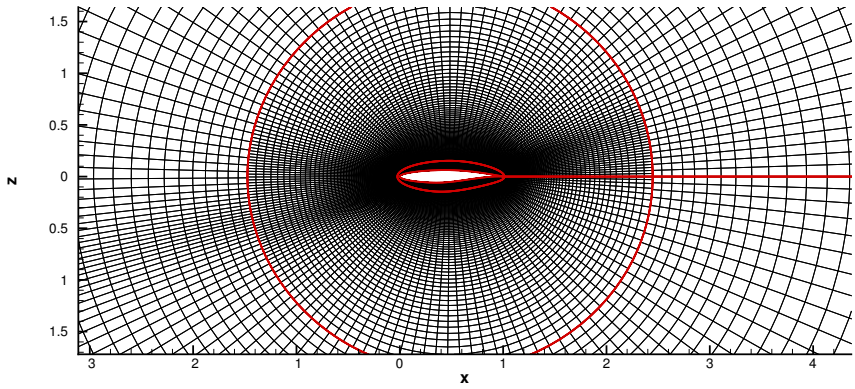
Figure 3.9: Comparison of the θ-criterion for the mono-block transonic case mesh adaptation.



(a) Starting mesh - transonic (22284 nodes) - subsonic (26603 nodes)



(b) C_D-based adapted mesh, transonic flow (40005 nodes)



(c) C_D-based adapted mesh, subsonic flow (45027 nodes)

Figure 3.10: Non-matching multi-block meshes. The starting one is given in (a) while final adapted ones in (b)-(c).

Chapter 4

Non-intrusive uncertainty quantification study for aerodynamic simulations

The fast development and availability of computational models and ressources in various fields of science and engineering, such as Computational Fluid Dynamics (CFD), has led to a growing awareness of the researchers towards the reliability and limitations of their simulations. In particular, the assessment of the range of applicability of these models to varying operational and environmental conditions is closely dependent on the definition and understanding of the model *parameters*. CFD is widely accepted as a key tool for aerodynamic design. Initial aerodynamic performance is explored by computational simulations and through wind tunnel tests. Unfortunately while CFD computations are in general performed for given fixed conditions, experiments and real flight conditions persistently show alterations of the observed quantities from nominal values. Evans *et al.* [82] have carefully itemized many possible disparities between calculations and experiments, including changes in flow conditions, manufacturing tolerances, design to manufacture issues, dynamic distortion, permanent or degrading factors. Hence, any validation of CFD simulations with respect to experimental data must include estimation of discretization error [83], modeling error and assessment of uncertain parameters influence [30]. A natural way of representing this dependability on parameters is by evaluating/sampling the solution of the system under investigation for different values of the parameters.

When these parameters are discretized with random variables from a probability space, the system becomes stochastic and response sampling for infinite values of the parameters becomes intractable. The numerical challenge is then to represent the parameter-dependent solution or related quantities of interest (QoI) of the system with a continuous approximation (*i.e.* *surrogate*) over the entire range by relying only on discrete system evaluations (or *realizations*) for (possibly

few) different values of the parameters. These non-intrusive approaches, despite relying on samples, may therefore be seen as functional approximations of the QoI. There exists in fact many computational methodologies – such as (robust) optimisation and control [84], uncertainty quantification (UQ) [85], parametric calibration via inference [86, 87] – that may benefit vastly from global and accurate computational model surrogates in order to quantify and reduce uncertainties. An important aspect of UQ, that is of great value for engineering applications, is the global sensitivity analysis (GSA) which describes how the variation in the output can be apportioned to different sources of variation in the input parameters of a model.

Aerodynamic modelings may be subject to numerous uncertainties [88, 89] in the geometry [90] and in the operational conditions [91] of an aircraft. For instance, even small parametric deviations from the planned geometry have a significant effect on the drag/lift coefficients, so that geometry uncertainties must be included in the aerodynamic *robust* design optimization problem. Depending on the scarcity of the available experimental data, the probability distributions of these uncertainties have to be inferred or modeled. Two types of situations are of interest when dealing with these uncertainties: wind-tunnel (WT) and real flight (RF) conditions. The nature and the magnitude of these uncertainties are very different in those two situations and will be described in the following.

Operational uncertainties

Operational parameters include streamflow velocity (Mach number), density, angle of incidence, temperature, pressure, altitude *i.e.* all the streamflow properties. The sources of uncertainties that may influence these parameters differ in WT and RF situations. Three main uncertainty sources may be identified in WT applications.

1. errors due to sensor uncertainty. It is given by the sensor constructor after calibration and later on may be adjusted comparing the signal with other kind of sensor. This error may be decomposed in a random and a bias components.

2. errors given by spatial and temporal non-homogeneity in the WT test section.

3. errors resulting from WT corrections in order to suppress walls and model support effects. These kinds of uncertainties are closely linked with the test case performed.

Uncertainty ranges and distributions also depend on the flow regime. In general, uncertainty ranges are quite small, especially for large Mach number values that are easier to obtain and control.

In RF conditions, the sources are linked with the spatial and temporal non-homogeneity of the streamflow, the complex flight kinematic and possible extreme

rare events, *e.g.* wind gust [92]. Most influent macroscopic flight conditions have been shown to be the angle of incidence and the Mach number. The uncertainty of these parameters mostly results from atmospheric and flow fluctuations that are higher than in the WT case. Incidence and Mach number may be modeled as independent [93] or dependent random parameters [91]. The literature is very scarce on the representation of these parameters as time-dependent random processes.

Geometrical uncertainties

Optimal shape of aircraft wings has been the subject of a large body of research literature. Impact of geometric variability is in general reduced thanks to robust design (with adjustment of nominal design parameters) and tolerancing (with ajustement of the local level of variability). But the result still remains very hazardous mainly due to the sequentiality of these approaches. Therefore, despite advances in manufacturing and maintenance engineering techniques, airfoils very often exhibit some deviation from their intended shape and size due to noisy manufacturing processes or wear. Stochastic representations are useful for estimating the impact of geometric variability on the aerodynamic performance.

WT facilities exhibit several types of imperfections that often requires a detailed calibration process in which flow features, *e.g.* airspeed, pressure/temperature variations, disturbances/turbulence levels, influence of measurement sensors, and related uncertainties are determined at various tunnel sections. After calibration, the control of these uncertainties remain flow-dependent (*e.g.* supersonic regime are less prone to spatial/temporal non-homogeneity). In this work, only the airfoil surface shape imperfections are considered, thus the scatter between the theoretical and measured profile. The range of accuracy depends mainly on the tolerance class and the bluff body material (composite structure accuracy is lower than the metal one) that have been chosen. For a full-scale aircraft in RF conditions, the sources are clearly linked with the manufacturing and assembly constrains *e.g.* surface imperfections, junctions or unknown deformations/wear during operation, *e.g.* ice aggregation during the flight.

Considering only surface shape imperfections in the characterization of geometric variability, several possible types of discretization may be considered. Geometrical uncertainties may be modeled by variations of characteristic quantities (*i.e.* random variables) globally controlling the shape, such as the camber and thickness [94, 89] or the position of some trailing-edge controls, such as the flap angle [95]. In general, this approach does not allow to properly model ad-hoc input uncertainties estimated by local measurements. Stochastic approaches describing the uncertainties by random fields are also feasible as long as they are paired with efficient approximation and discretization techniques in order to reduce the additional computational effort. With a large amount of geometry data, it becomes

then possible to find a statistically optimal basis for representing the scatter of the measured data. This is called the principal-component analysis (PCA) [96] or the Karhunen-Loève (KL) expansion [97]. Standard formulations often assume stationarity of the random process which indicates that correlations are constant along the profile. While this may be conceptually well suited to model rugosity that is mainly a property of the material, it is not appropriate for the representation of spatially varying manufacturing tolerance for instance. Indeed, near the leading edge ($x/c < 0.2$), airfoil geometrical accuracy is much higher with respect to the remaining part.

An Uncertainty Quantification study will allow to take into account these sources of uncertainty and quantify their effects on the QoI. In Section 4.1 the stochastic framework and computational approaches are introduced with a special focus on methods based on sparse grids introduced in Section 4.2. The pseudospectral polynomial chaos and the stochastic collocation representation with details on a new stochastic collocation algorithm and implementation of a dimension-adaptive refinement based on sensitivity analysis are described in Section 4.3. Applications of these UQ methods to test functions and 2D aerodynamic simulations through a simplified aerodynamics model as well as through RANS approach are shown in Section 4.4.

The main achievements of this chapter have published in paper [98].

Nomenclature

Abbreviations

ANOVA	ANalysis Of VAriance
AoA	Angle of Attack
BL	Boundary Layer
CC	Clenshaw-Curtis rule
CDF	Cumulative Density Function
CFD	Computational Fluid Dynamics
CoD	Course of Dimensionality
DES	Detached Eddy Simulation
DNS	Direct Numerical Simulation
FJ	Fejer rule
G	Gauss-type rule
GP	Gauss-Patterson-Kronrod rule
gPC	generalized Polynomial Chaos
GSA	Global Sensitivity Analysis
GSG	Generalized Sparse Grid
LES	Large Eddy Simulation
MC	Monte Carlo
PS	PeudoSpectral
PDF	Probability Density Function
QoI	Quantity of Interest
RANS	Reynolds-Averaged Navier-Stokes
RF	Real Flight
RSM	Reynolds Stress Model
SA	Spalart-Allmaras turbulence model
SC	Stochastic Collocation
SC-Full	Full tensor grid SC
SC-Sobol	Sobol-based SC
SgPC	Sparse generalized Polynomial Chaos
SI	Sobol' Indices
UQ	Uncertainty Quantification
V&V	Verification and Validation
WT	Wind Tunnel

Greek letters

Λ	Set of nodes of a quadrature
μ	Mean value
$\boldsymbol{\xi}$	Random variables
$\boldsymbol{\Xi}$	Random space
$\tilde{\boldsymbol{\Xi}}$	Nodal set
ϕ_j	Orthonormal polynomials

Φ_j	N-variate polynomial
$\boldsymbol{\omega}$	Realizations
σ	Standard deviation (std)

Latin letters

A	Bump amplitude
$\boldsymbol{\mathcal{A}}$	New active set gathering all $\boldsymbol{j}$
c	Airfoil chord
C_d	Drag coefficient
C_{dp}	Pressure drag coefficient
C_l	Lift coefficient
C_m	Momentum coefficient
d	Physical space dimension
$\mathbb{E}$	Expectation
$\boldsymbol{j}$	Index vector defining the refinement in each (i) dimension
$\boldsymbol{j}^{+1}_{1,\dots,N}$	Isotropic refinement
$\boldsymbol{j}^{+1}_{i}$	Anisotropic refinement in (i) dimension
$\boldsymbol{j}^{+1}_{i,j}$	Anisotropic refinement in cross-dimension
K	Number of MC samples
L_k	Lagrange interpolation polynomials
M	Mach number
n	Number of nodes of $\mathcal{Q}$
N	Stochastic dimension
N_{crit}	Exponent of the $e^{N_{crit}}$ method
$\mathcal{O}$	Order of magnitude
$\boldsymbol{\mathcal{O}}$	Old active set gathering all $\boldsymbol{j}$
$\mathcal{P}$	Projection approximation
q	Polynomial exactness of $\mathcal{Q}$
$\mathcal{Q}$	Quadrature rule
$\mathcal{R}$	Stochastic problem
Re	Reynolds number
$\mathcal{S}$	Pseudospectral approximation
$S^{(i)}$	Main Sobol' index
$S^{(i),(j)}$	Cross Sobol' index
$S_T^{(i)}$	Main total Sobol' index
u	Solution of problem $\mathcal{R}$
$\mathcal{U}$	Uniform distribution
$\mathbb{V}$	Variance
w	Weights of $\mathcal{Q}$
$\boldsymbol{x}$	Deterministic variables
X_{tr}	Transition point
y	Node of $\mathcal{Q}$
y^+	Dimensionless wall distance

Superscripts

TD	Total Degree
SD	Sparse Degree
(i)	Stochastic dimension i component

Subscripts

l	Level of the quadrature rule $\mathcal{Q}$
$l(j)$	j-th node of $\mathcal{Q}$ of level l

4.1 Stochastic framework and computational approaches

Powerful methods have been developed in which the regularity of the solution through polynomial approximation is put to use [99]. Projection and interpolation are two possible strategies to construct multivariate (global or piecewise) polynomial approximations. The generalized Polynomial Chaos (gPC) [100, 101, 102] in its pseudospectral (PS) form [85] and the Stochastic Collocation (SC) [103, 104] methods are among the most exploited ones. Both approaches rely on discrete and uncoupled model evaluations (as Monte-Carlo method) and have no conceptual difficulties in treating non-linear problems. However, they face the same curse of dimensionality (CoD), they lead to a number of function evaluations that scales exponentially with the number of random dimensions.

The use of sparse grids (based on sparse tensor product approximation spaces) [105] alleviates to some extent the CoD and is particularly effective for the approximation of sufficiently smooth functional in moderately high-dimensional parameter spaces. Nevertheless, quadrature grids pose strong limits on the way with which

the grid is enriched. This is due mainly to the orthogonality constraint and to the limited flexibility on the sampling procedure. Other options than sparse grids are also available. In a SC approach, for instance, the simplex method proposed by Witteveen *et al.* [106] is worth mentioning. The latter may be even improved through a goal-oriented uncertainty propagation using stochastic adjoints [107]. In a gPC framework, it is possible to use adjoint equations in order to identify random variables providing maximum impact on the QoI by using the gradient information [108]. It is also worth mentioning the least square method with l_1-regularization based on purely Random Sampling (RS) or RS on a quadrature grid. This method is also known as Compressed Sensing (CS) [109].

The impact of the combination of different interpolating polynomial spaces and corresponding sparse grids on the SC accuracy have been investigated and compared in great details to intrusive approaches, such as spectral Galerkin methods [110]. On the other hand, spectral projection methodology has been recently improved. Several groups have proposed (Smolyak–based) *sparse* pseudospectral approximations (SgPC) [111, 112]. These approximations are general and well adapted to a broad choice of sparse grids and quadrature rules and they guarantee no internal aliasing errors [113]. These recent advances advocate for a more fair comparison between SC and PS surrogate models, in particular when those are constructed on the same sparse input points. For a moderate number of parameters ($\mathcal{O}(10)$) with bounded spans, isotropic and nested sparse grids make possible a systematic exploration of the response of the computational model to parametric variations/uncertainties. In particular, variance-based GSA [114] is now available to both representations. While straightforward for hierarchical PS approximation, it is also possible to compute efficiently global sensitivity coefficients of Sobol' [115] from interpolating SC approximations based on Smolyak cubatures [116, 117].

In order to further reduce the number of function calls, *anisotropic* adaptivity may be used. Most efforts so far have been carried out in the SC community, *e.g.* [118, 119]. A standard approach is the one of Gerstner [120] who generalized the sparse grid construction for integration based on greedy selection of admissible grid index sets according to some predefined measure. More sophisticated adaptive techniques [121, 122] also take advantage of low effective dimensions well approximated by low-dimensional functions of the solution, *e.g.* sum of functions of increasing dimensions, cf. the analysis of variance (ANOVA) decomposition [123] and Sobol indices (SI) calculation [115]. With these approaches, an (anchored-) ANOVA decomposition of the QoI is constructed/truncated, either *a priori* or *a posteriori*, in order to reduce the dimensionality of the approximation with the consequence of introducing a modeling error. The recovered lower-order dimensional problems – involving at most second- to third-order interaction terms – are then integrated or interpolated thanks to anisotropic adaptive schemes based on sparse grids; this second step introducing this time a discretization error.

Inspired by the idea of taking advantage of the emerging low-order functional interactions of high-dimensional models, a simpler adaptive strategy is proposed here in the framework of SC. The dimension adaptivity of the interpolation is driven by a variance-based GSA where SI are iteratively calculated. Successive evaluations of first- and second-order indices provide guidance to the refinement procedure thanks to a greedy selection of admissible index sets that contributes the most to the solution global variability.

An important step is the choice of an appropriate stochastic approximation, handling both the description of the input uncertainties and the computation of the statistics and sensitivities of the CFD solution. Stochastic spectral methods have emerged as potential good candidates [99].

In the following, let us consider a general problem that may be written in the following fashion: let $u(\boldsymbol{x}, \boldsymbol{\xi}(\boldsymbol{\omega}))$ be the solution of equation

$$\mathcal{R}(\boldsymbol{x}, \boldsymbol{\xi}) \equiv \mathcal{G}\big(\boldsymbol{x}, \boldsymbol{\xi}(\boldsymbol{\omega}); u(\boldsymbol{x}, \boldsymbol{\xi}(\boldsymbol{\omega}))\big) - \mathcal{V}\big(\boldsymbol{x}, \boldsymbol{\xi}(\boldsymbol{\omega})\big) = 0, \tag{4.1}$$

where $\mathcal{G}$ and $\mathcal{V}$ are differential operators acting on $\mathcal{D} \times \Xi$, where $\boldsymbol{x} \in \mathcal{D} \subset \mathbb{R}^d$ with $d \in \{1, 2, 3\}$, $\boldsymbol{\xi}(\boldsymbol{\omega}) = \{\xi^{(1)}(\omega_1), \ldots, \xi^{(i)}(\omega_i), \ldots, \xi^{(N)}(\omega_N)\} \in \boldsymbol{\Xi} \subset \mathbb{R}^N$ are N independent random variables. On the probability space $(\Omega, \mathcal{F}, \mathcal{P})$, the realizations are $\boldsymbol{\omega} \in \Omega$ (set of outcomes), $\mathcal{F} \subset 2^\Omega$ the σ-algebra of sets and $\mathcal{P} : \mathcal{F} \to [0, 1]$ the probability measure. The probability density function of $\boldsymbol{\xi}$ is noted as $\rho : \boldsymbol{\xi} \mapsto \mathbb{R}_+$. It is assumed that the solution of our problem belongs to the space $L_2(\boldsymbol{\Xi}, \rho)$ of second-order random variables, with an appropriate inner product [99]. Within this framework two main classes of methods may be identified:

1. Galerkin (*Intrusive*) methods: probabilistic model is introduced within the partial differential equations describing the flow and therefore the CFD code needs to be modified in order to solve the obtained stochastic system of equations.

2. Direct (*Non-intrusive*) methods: the CFD deterministic code is seen as a black-box by the stochastic approximation. It is then only possible to operate by means of input and output data sampling.

In this study, one of the requirements was to use the ONERA finite-volume compressible CFD *elsA* solver [2] without modifications, therefore intrusive methods have been disregarded. Nevertheless, there exists *semi-intrusive* methods which are - strictly speaking - intrusive schemes but require only a limited number of modification to an existing numerical code [124].
The most well-known non-intrusive approach is the Monte-Carlo (MC) method [125]. This sampling method consists in randomly generating K input data realizations and performing a deterministic simulation for each of these realizations.

The approach is straightforward and robust and the convergence is dimension-independent, although slow, *i.e.* $\mathcal{O}(K^{-\frac{1}{2}})$. Several methods exist to improve the MC convergence rate, *e.g.* Quasi MC methods [126] with a computational complexity of order $\mathcal{O}\big((\log K)^N K^{-1}\big)$.
Other methods have been developed in which the regularity of the solution through polynomial approximation is put to use [99]. Projection and interpolation are two possible strategies to construct multivariate polynomial approximations. Within all the spectral methods, the generalized Polynomial Chaos (gPC) [100, 101, 102] in its pseudospectral form and the Stochastic Collocation (SC) [103, 104] methods are among the most exploited ones. Both approaches rely on discrete model evaluations and face the CoD: *i.e.* the number of function evaluations scales exponentially with the number of random dimensions. Sparse sampling/approximations – and adaptive variants [120] – of linear operators on tensor product spaces [127] somewhat alleviate the computational burden to $\mathcal{O}\big(K^{-r}(\log K)^{(N-1)(r+1)}\big)$[1] by taking advantage of higher smoothness and lower-effective dimensionality of the functional to represent.

4.2 Sparse sampling

There exists a large body of literature dedicated to the use of sparse grids/cubatures [105] for lowering the cost of high-dimensional integral computations [127, 128] and stochastic interpolations [129, 130]. Sparse grids that are structured in the form of quadratures/cubatures make error analysis more convenient because they integrate exactly functionals that are polynomials of a certain degree.
In one dimension (noted dimension $^{(i)}$), one may predict the polynomial exactness q_l obtained with a quadrature $\mathcal{Q}_l^{(i)}$ of a certain level $l \in \mathbb{N}_0$ and corresponding number of points $n_l^{(i)}$ and weights $w_l^{(i)}$. Sparse grids are always hierarchically and incrementally assembled from a sequence of one-dimensional quadrature formulas and their performance is strongly dependent on the growth of q_l vs. $n_l^{(i)}$ as l increases.
Many quadrature families exist with different (linear or exponential) growth of $n_l^{(i)}$ with respect to l, e.g. Table 6.1 in [112]. Noting $\Lambda_l^{(i)}$ the set of nodes for a quadrature level l

$$\Lambda_l^{(i)} = \left(y_{l(1)}^{(i)}, ..., y_{l(n_l^{(i)})}^{(i)} \right), \tag{4.2}$$

where $y_{l(j)}^{(i)}$ stands for the j-th node of quadrature $\mathcal{Q}_l^{(i)}$, one may distinguish two classes of grids: – if $\Lambda_l^{(i)} \subset \Lambda_{l+1}^{(i)}$ then the rule is said to be *nested*. This is for instance the case in Newton-Cotes, Clenshaw-Curtis (CC) [131], Fejér (FJ) and Gauss-Patterson-Kronrod (GP) [132] formulae, which are often preferred for

[1] For integrands which have bounded mixed partial derivatives of order r.

bounded measures. However, if – $\Lambda_l^{(i)} \not\subset \Lambda_{l+1}^{(i)}$, such as for all standard Gauss-type (G) formulae, the rule is said to be *non-nested*. In this work, we mainly rely on exponential growths such as CC rule: $n_l^{(i)} = 2^l + 1$, or G/GP rules: $n_l^{(i)} = 2^{l+1} - 1$.

Due to the extensive use of the CC rule, its formulation is given. The nodes are the extrema of the Chebyshev polynomials including the boundary of the support $\Gamma \in [-1, 1]$:

$$y_{l(j)}^{(i)} = -\cos\left(\frac{\pi(j-1)}{n_l^{(i)} - 1}\right), \quad j = 1, ..., n_l^{(i)}, \tag{4.3}$$

while for $n_1^{(i)} = 1$, $y_{1(1)}^{(i)} = 0$. The weights are given by the following relation

$$w_{l(n_l^{(i)}+1-j)}^{(i)} = \frac{2}{n_l^{(i)} - 1}\left(1 - \frac{\cos(\pi(j-1))}{n_l^{(i)}\left(n_l^{(i)} - 2\right)} - 2\sum_{k=1}^{\frac{n_l^{(i)}-3}{2}} \frac{1}{4k^2 - 1}\cos\left(\frac{2\pi k(j-1)}{n_l^{(i)} - 1}\right)\right) \tag{4.4}$$

for $j = 2, ..., n_l^{(i)} - 1$. For the first and last weights the value is $\frac{1}{n_l^{(i)}\left(n_l^{(i)}-2\right)}$.

The polynomial degree of exactness $q_l^{(i)} = \deg(\mathcal{Q}_l^{(i)})$ differs for different rules, for instance:

G:	$q_l^{(i)} = 2n_l^{(i)} - 1$,	$\forall u \in \mathbb{P}_{2n_l-1}^{(i)}$,
CC, FJ:	$q_l^{(i)} = n_l^{(i)} - 1$,	$\forall u \in \mathbb{P}_{n_l-1}^{(i)}$,
GP:	$q_l^{(i)} = 2n_l^{(i)} - n_{l-1}^{(i)}$,	$\forall u \in \mathbb{P}_{2n_l-n_{l-1}}^{(i)}$.

Although Gauss-type formulae are the most accurate (*e.g.* by a factor of two compared to CC formula), it was reported that the degree of accuracy is comparable for non-polynomial functional, i.e. $u \notin \mathbb{P}$ [133]. In the same paper, Trefethen found the error bound of a $n_l^{(i)}$-point CC formula to be approximately $\mathcal{O}\left(\frac{(2n_l^{(i)})^{-k}}{k}\right)$ for a k-times differentiable integral. CC and FJ rules are particularly flexible because there exists explicit formulae for their nodes and weights.

If $N > 1$, prescribing the multi-index $\boldsymbol{l} = (l_1, \ldots, l_N) \in \mathbb{N}_0^N$, the grid may be built through a simple full tensor product:

$$\mathcal{Q}_{\boldsymbol{l}}^{(\boldsymbol{N})} = \left(\mathcal{Q}_{l_1}^{(i)} \otimes ... \otimes \mathcal{Q}_{l_N}^{(i)}\right). \tag{4.5}$$

Figure 4.1 shows the index diagram refinement for a full tensor product grid with N=2. It is clear that for high N, the total number of nodes $n_{\boldsymbol{l}}^{(\boldsymbol{N})}$ grows exponentially. In order to limit this growth, sparse techniques are an efficient solution.

4.2.1 The isotropic Smolyak method

The Smolyak [127] algorithm with sparse tensorization may be used [128, 129]. The Smolyak algorithm is a linear combination of product formulae. The chosen linear combination preserves the one dimensional interpolation property for $N > 1$. The obtained grid has a significant less number of nodes w.r.t. to the tensor product one[2]. By defining quadrature differences in the following fashion

$$\Delta\mathcal{Q}_{l_k}^{(i)} \equiv \left(\mathcal{Q}_{l_k}^{(i)} - \mathcal{Q}_{l_{k-1}}^{(i)}\right) \quad k = 1, ..., N, \tag{4.6}$$

the sparse cubature formula in N dimension is:

$$\mathcal{Q}_{\boldsymbol{l}}^{(\boldsymbol{N})} = \sum_{\boldsymbol{k}\in\mathcal{L}} \left(\Delta\mathcal{Q}_{k_1}^{(1)} \otimes \ldots \otimes \Delta\mathcal{Q}_{k_N}^{(N)}\right). \tag{4.7}$$

The isotropic (*uniform*) sparse grids are then given by:

$$\Lambda_{\boldsymbol{l}}^{(\boldsymbol{N})} \equiv \mathcal{H}(\boldsymbol{l}, N) = \bigcup_{\boldsymbol{j}\in\mathcal{L}} \left(\Lambda_{j_1}^{(1)} \otimes \ldots \otimes \Lambda_{j_N}^{(N)}\right),$$

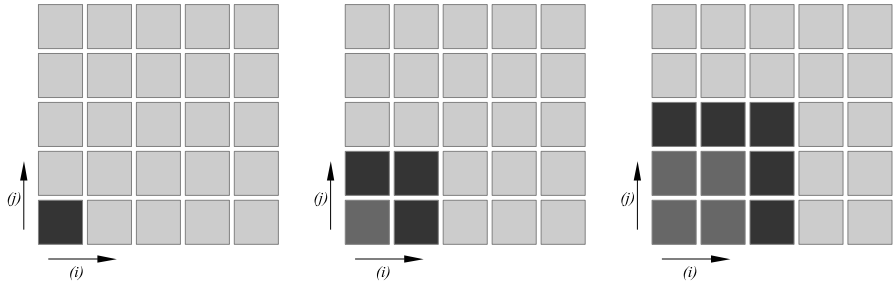
where the set of indices $\mathcal{L}$ must be admissible[3]: *i.e.* $\forall \boldsymbol{j} = (j_1 \ldots j_N) \in \mathcal{L} \mid j_{1\leq i\leq N} > 0$ then $(\boldsymbol{j} - \boldsymbol{e}_i) \in \mathcal{L}$ for all i, where $\{\boldsymbol{e}_{i=1\ldots N}\}$ is the canonical basis of $\mathbb{N}^N$, and $l_1 = l_2 = \ldots = l_N \equiv l$ (the same applies for index k). Any admissible set is valid to construct $\mathcal{L}$, even though Smolyak suggested truncating with a *total order* multi-index set: *i.e.* $\mathcal{L} := \{\boldsymbol{j} \in \mathbb{N}_0^N : \|\boldsymbol{j}\|_1 \leq l\}$, but other options exist [110].
As the sparse quadrature is enriched, the grid points used in the earlier levels are retained. Therefore, the sparse quadrature is always *embedded* regardless of the nature of the 1D basis quadrature rule used. If the one-dimensional quadrature rule is nested, *i.e.* if $\Lambda_l^{(i)} \subset \Lambda_{l+1}^{(i)}$, then:

$$\Lambda_{\boldsymbol{l}}^{(\boldsymbol{N})} = \bigcup_{\|\boldsymbol{j}\|_1=l} \left(\Lambda_{j_1}^{(1)} \otimes \ldots \otimes \Lambda_{j_N}^{(N)}\right),$$

resulting in a much smaller grid. Growth rate of points per level do not take analytical form and have to be tabulated [134]. Figure 4.2 shows the 2D index diagram for three consecutive quadrature level $\boldsymbol{l}$, with a isotropic Smolyak sparse grid (SSG) growth. Each square represents the index $\boldsymbol{j}_{1,2} := (j_1, j_2)$. This method significantly reduces the number points but the obtained grid is clearly symmetric in all directions. In this case, the representation is efficient if the influence of each parameter/dimension on the QoI is comparable. In order to reduce further $n_{\boldsymbol{l}}^{(\boldsymbol{N})}$ while keeping a good accuracy, *anisotropic* dimension-adaptive features may be introduced [120, 118].

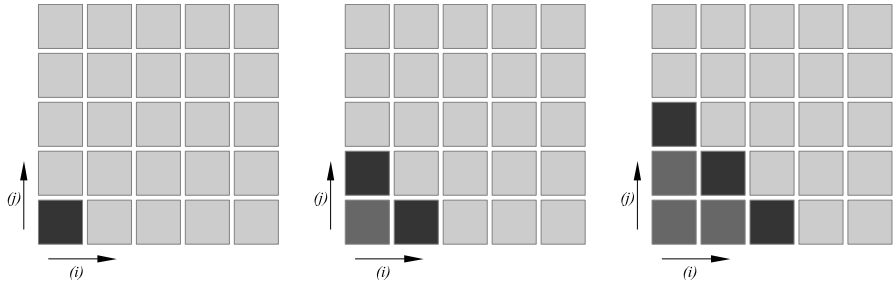
[2]It is important to notice that for non-nested $\mathcal{Q}_l^{(i)}$, the asset of the Smolyak algorithm is generally patent for $N \geq 4$. In the nested case, Smolyak sparse grids have less nodes for $N > 1$.

[3]This is a necessary condition for the sum of the sparse quadrature to telescope correctly.



(a) Initialization, $\boldsymbol{l}_0$ (b) $\boldsymbol{l}_0 + 1$ (c) $\boldsymbol{l}_0 + 2$

Figure 4.1: 2D index diagram for full tensor refinement.



(a) Initialization, $\boldsymbol{l}_0$ (b) $\boldsymbol{l}_0 + 1$ (c) $\boldsymbol{l}_0 + 2$

Figure 4.2: 2D index diagram for the isotropic Smolyak sparse grid algorithm.

4.3 Stochastic approximations

4.3.1 Pseudospectral polynomial approximation

Spectral expansions are in general efficient provided sufficient smoothness is present in the solution. In situations where the modification of legacy solvers is not tractable, a non-intrusive formulation is an alternative to the Galerkin formalism. In this case, a finite set of basis functions is considered, and the solution is projected onto the subspace spanned by these functions. Thanks to the probabilistic framework and approximation theory, the hierarchical nature of the spectral representation provides a functional format that is very convenient to perform subsequent tasks such as sensitivity analysis, optimization, etc... In the following, the standard and – a more recent Smolyak-based *sparse* – projection-based gPC approximations will be reviewed in their respective pseudospectral form.
Orthonormal polynomials $\phi_j^{(i)}$ with respect to $\rho^{(i)}$ along the direction (i) are considered. It is now possible to construct the N-variate polynomial in $\boldsymbol{\xi}$: $\Phi_{\boldsymbol{j}}(\boldsymbol{\xi}) = \prod_{k=1}^{N} \phi_{j_k}^{(k)}(\xi_k)$ that will be the base of the full tensor approximation of the solution. The notation $\{\Phi_{\boldsymbol{j}\in\mathbb{N}^N}\}$ stands for an orthonormal polynomial basis of $L_2(\boldsymbol{\Xi}, \rho)$. The projection approximation $\mathcal{P}_l^{(i)}(u)$ of the quantity of interest u at level l along

the dimension (i) is:

$$u(\boldsymbol{x},\boldsymbol{\xi}) \approx \mathcal{P}_l^{(i)}(u) \equiv \sum_{j=0}^{q_l^{(i)}} \mathbb{E}\left[u(\boldsymbol{x},\boldsymbol{\xi}) \cdot \phi_j^{(i)}(\boldsymbol{\xi})\right] \phi_j^{(i)}(\boldsymbol{\xi}) = \sum_{j=0}^{q_l^{(i)}} u_j(\boldsymbol{x})\phi_j^{(i)}, \tag{4.8}$$

where $q_l^{(i)}$ is the polynomial truncation[4]. In higher dimensions, its multivariate polynomial approximation variant would be for instance obtained via full tensor products:

$$\mathcal{P}_{\boldsymbol{l}}^{(\boldsymbol{N})}(u) \equiv \sum_{j_1=0}^{q_l^{(i)}} \dots \sum_{j_N=0}^{q_l^{(N)}} u_{\boldsymbol{j}}(\boldsymbol{x})\, \Phi_{\boldsymbol{j}}. \tag{4.9}$$

In the following, only univariate approximations are considered but the extension to N-variate approximations is straightforward. Assuming the expectations in Eq. (4.8) are perfectly evaluated *i.e.* the coefficients $\{u_j(\boldsymbol{x})_{j=0:q_l^{(i)}}\}$ are exact, the approximation error is only due to the *truncation* of the series. The convergence in the L_2 sense reads:

$$\left\| u - \mathcal{P}_l^{(i)}(u) \right\|_2^2 = \sum_{j=q_l^{(i)}+1}^{\infty} u_j^2 < \infty \tag{4.10}$$

In practice, the expansion coefficients are calculated thanks to some numerical approximations such as *e.g.* quadratures/cubatures $\mathcal{Q}_l^{(i)}$ at some level l. In this case, *pseudospectral* approximation $\mathcal{S}$ of Eq. (4.8) is defined as:

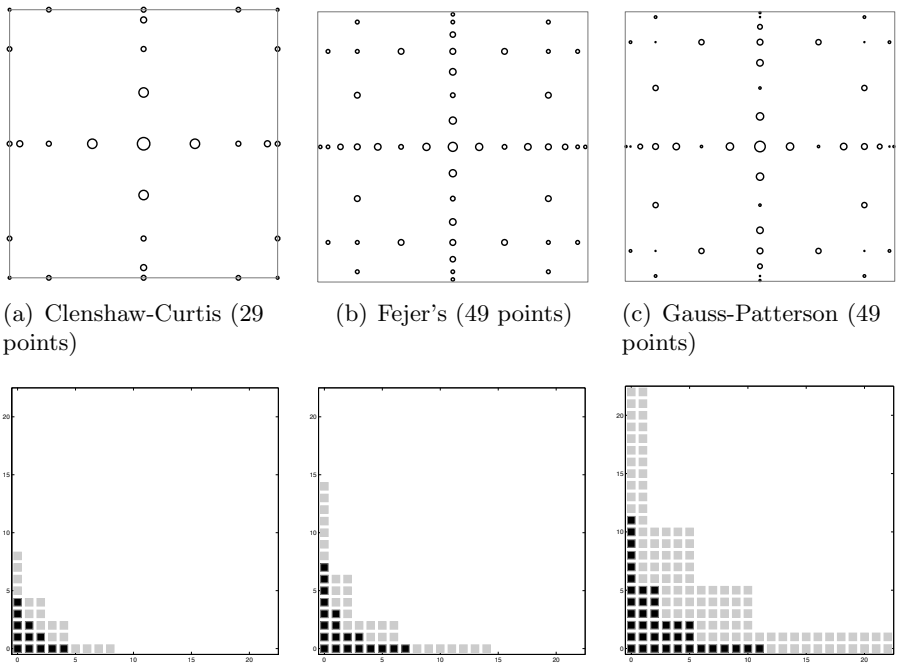
$$\mathcal{S}_l^{(i)}(u) \equiv \sum_{j=0}^{q_l^{(i)}} \mathcal{Q}_l^{(i)} \left(u \cdot \phi_j^{(i)}\right) \phi_j^{(i)} = \sum_{j=0}^{q_l^{(i)}} \tilde{u}_j \phi_j^{(i)} \tag{4.11}$$

Recent studies [135, 112], carried out in the context of adaptive stochastic approximations based on sparse grids, have shown that L_2 *aliasing* errors between $\mathcal{S}$ and $\mathcal{P}$ may be decomposed as follows[5]:

$$\begin{aligned} \left\| \mathcal{P}_l^{(i)}(u) - \mathcal{S}_l^{(i)}(u) \right\|_2^2 \;=\; & \sum_{j=0}^{q_l^{(i)}} \Bigg(\underbrace{u_j - \sum_{k=0}^{q_l^{(i)}} u_k\, \mathcal{Q}_l^{(i)} \left(\phi_j^{(i)} \cdot \phi_k^{(i)}\right)}_{\text{internal aliasing}} \\ & - \underbrace{\sum_{m=q_l^{(i)}+1}^{\infty} u_m\, \mathcal{Q}_l^{(i)} \left(\phi_j^{(i)} \cdot \phi_m^{(i)}\right)}_{\text{external aliasing}} \Bigg)^2 \end{aligned} \tag{4.12}$$

[4] As we will see in the following, here we have used on purpose for the upper bound the same notation as the polynomial degree of exactness of the quadrature introduced in the previous section.

[5] Here, the notations closely follow the detailed derivations from [112].



(a) Clenshaw-Curtis (29 points) (b) Fejer's (49 points) (c) Gauss-Patterson (49 points)

Figure 4.3: Examples of level $l = 3$ nested 2D sparse grids and corresponding polynomial exactness of $\mathcal{Q}_{\mathbf{3}}^{(\mathbf{2})}$; (top) quadrature points are located at the circle centers and quadrature weight magnitudes are proportional to the circle diameters. (bottom): quadrature polynomial exactness (grey symbols) and monomial coverage of the sparse pseudospectral representation with no internal aliasing (black symbols).

It becomes apparent that aliasing error comes from the inability to recover numerically discrete orthogonality. Internal aliasing is due to inaccuracies of the quadrature to correctly capture polynomials orthogonality *within* the expansion basis. In principle, aliasing errors should tend to zero for very large l (*i.e.* for powerful quadratures). In practice, for a given l, one is left with a choice of $q_l^{(i)}$ that should guarantee that the internal aliasing is null. Relating the cardinality of the polynomial approximation space to the quadrature integration capability is an easy choice for one(multi)-dimensional quadrature rules with sharp polynomial exactness deg (such as full tensor-based grids), where one can choose: $q_l^{(i)} = \lfloor deg(\mathcal{Q}_l^{(i)})/2 \rfloor$ based on the so-called half-accuracy set [112], required to cancel the internal aliasing term in Equation (4.12). It becomes more tedious for isotropic and anisotropic sparse grids with dimension-dependent polynomial exactness, for which the tensorization procedure is only partial.
A wise choice to avoid full tensor products is to take advantage of the fact that input dimensions may not be fully coupled: this is the strength of the Smolyak algorithm here applied to pseudospectral operators. In this case, the sparse pseudospectral approximations harnesses the full power of its underlying sparse grid by

projecting the solution onto a larger polynomial space that guarantees *no* internal aliasing, regardless of the admissible Smolyak multi-index set $\mathcal{K}$ retained [112]. The approximation may be summarized as:

$$u \approx \mathcal{A}^{(\boldsymbol{N})}_{\mathcal{S},\mathcal{K}}(u) \equiv \sum_{\boldsymbol{k}\in\mathcal{K}} c_{\boldsymbol{k}}\, \mathcal{S}^{(\boldsymbol{N})}_{\boldsymbol{k}}(u), \tag{4.13}$$

where $c_{\boldsymbol{k}}$ are Smolyak integers computed from the combinatorics of difference formulations and $\mathcal{S}^{(\boldsymbol{N})}_{\boldsymbol{k}}(u)$ is the general form of Equation (4.11). We refer to the excellent paper of Conrad *et al.* for all numerical details about the formulation [112]. The Smolyak algorithm is therefore a sum of different *full* tensor pseudospectral approximations, where each approximation is built around the polynomial accuracy of a single *full* tensor quadrature rule. It converges under similar conditions as the one-dimensional operators from which it is constructed. In fact, each polynomial $\Phi_{\boldsymbol{j}}$ in (4.13) is included in the approximation, only if $\exists\, \boldsymbol{k} \in \mathcal{K} \,:\, q^{(\boldsymbol{N})}_{\boldsymbol{j}} \leq \lfloor \deg(\mathcal{Q}^{(\boldsymbol{N})}_{\boldsymbol{k}})/2 \rfloor$, where $\mathcal{Q}^{(\boldsymbol{N})}_{\boldsymbol{k}}$ is the multivariate cubature based on multi-index $\boldsymbol{k} \in \mathcal{K}$. In the following, sparse pseudospectral approximation will be the method of choice when compared to stochastic collocation. Note that adaptive variants exist [112] but will not be pursued in this PhD thesis.

4.3.2 Stochastic collocation

Interpolating methods in (finite) multi-dimensions are often labelled as (stochastic) collocation (SC) methods [130, 119, 104]. They are non-intrusive in the sense that they are prone to parallelization of decoupled computations.

The key idea is to collocate the Eq. (4.1), $\mathcal{R}(\boldsymbol{x}, \boldsymbol{\xi})$, on a nodal set $\tilde{\boldsymbol{\Xi}}_N = \{\boldsymbol{\xi}_k\}_{k=1}^{\tilde{n}}$. Introducing a smooth function f, it is necessary to find a polynomial $\mathcal{I}(f) \in V_I$ such that $\mathcal{I}(f)(\boldsymbol{\xi}_k) = f(\boldsymbol{\xi}_k)$, $\forall k = 1, ..., \tilde{n}$ where V_I is the interpolation space corresponding to the random space $\boldsymbol{\Xi}$. This leads to the following relation

$$\mathcal{I}(f)(\boldsymbol{\xi}) = \sum_{k=1}^{\tilde{n}} f(\boldsymbol{\xi}_k) L_k(\boldsymbol{\xi}), \tag{4.14}$$

where $L_i(\boldsymbol{\xi}) \in V_I$, $L_i(\boldsymbol{\xi}_i) = \delta_{i,j}$, $1 \leq i, j \leq \tilde{n}$ are the Lagrangian polynomials. The solution (interpolant) approximation may be then rewritten as follows

$$u(\boldsymbol{x}, \boldsymbol{\xi}) \approx \hat{u}(\boldsymbol{x}, \boldsymbol{\xi}) \equiv \mathcal{I}u(\boldsymbol{x}, \boldsymbol{\xi}) = \sum_{k=1}^{\tilde{n}} u(\boldsymbol{x}, \xi_k) L_k(\boldsymbol{\xi}) \tag{4.15}$$

The solution approximation may be obtained also with some more local piecewise polynomials [136]. Using the property of Lagrange interpolation, the method leads to $\tilde{n}$ deterministic decoupled problems:

$$\mathcal{R}(\hat{u}(\boldsymbol{x}, \boldsymbol{\xi}))|_{\xi_k} = 0 \qquad \forall k. \tag{4.16}$$

The statistics may be computed as soon as the numerical solutions of (4.16) are obtained at all collocation points. The expectation reads:

$$\begin{aligned}\mathbb{E}(\hat{u}(\boldsymbol{x},\boldsymbol{\xi})) &= \sum_{k=1}^{\tilde{n}} u(\boldsymbol{x},\xi_k)\int_{\boldsymbol{\Xi}} L_k(\boldsymbol{\xi})\rho(\boldsymbol{\xi})d\boldsymbol{\xi} \\ &\simeq \sum_{k=1}^{\tilde{n}} u(\boldsymbol{x},\xi_k)\sum_{i=1}^{n_{\boldsymbol{l}}^{(\boldsymbol{N})}} L_k(\xi_i)w_i,\ \ \forall k=1,...,\tilde{n}, \end{aligned} \tag{4.17}$$

where the Gauss quadrature rule has been used. Please note that no constraints have been imposed on the nodal set $\tilde{\boldsymbol{\Xi}}_N$ yet. Let us further assume that $\tilde{\boldsymbol{\Xi}}_N = \{\boldsymbol{\xi}_i\}_{i=1}^{n_{\boldsymbol{l}}^{(\boldsymbol{N})}}$, *i.e.* the collocation points now coincide with the quadrature point set introduced in Section 4.2:

$$\mathbb{E}(\hat{u}(\boldsymbol{x},\boldsymbol{\xi})) \simeq \sum_{k=1}^{n_{\boldsymbol{l}}^{(\boldsymbol{N})}} u(\boldsymbol{x},\xi_k)L_k(\xi_k)w_k = \sum_{k=1}^{n_{\boldsymbol{l}}^{(\boldsymbol{N})}} u(\boldsymbol{x},\xi_k)w_k. \tag{4.18}$$

The method's efficiency is strictly linked with the chosen quadrature for the collocation points. In Section 4.2 isotropic sparse grids have introduced. The size of $n_{\boldsymbol{l}}^{(\boldsymbol{N})}$ may be further reduced by using adaptivity.

Adaptivity

In order to relax the isotropic construction inherent to the standard version of the Smolyak sparse grid algorithm, anisotropic ingredients may be introduced, *e.g.* [119]. Three main procedures are identified from the literature for greedy dimension-adaptive sparse grid applications: – variance-based decomposition, – spectral coefficient decay rates and – goal-oriented adaptation.
The first two lead to anisotropic tensor or sparse grids while the last one leads to generalized sparse grids (GSG). It seems that the goal-oriented adaptation technique is the most efficient in terms of convergence with respect to the number of numerical simulations, [120, 111]. In practice, efficient implementation of this algorithm in terms of efficient data structures for the storage and index sets bookkeeping is not straightforward. Indeed, the efficiency may be contested if the number of trial set is taken into consideration. With the procedure introduced by Gerstner *et al.* [120], only the trial sets that induces the largest change in the QoI statistics are selected. It seems that because not all of trial sets are being used in the approximation, some costly CFD computations that have been performed along the way may have been waisted. Another drawback resides in the fact that the algorithmic search might be put on hold if the next important set of indices is not admissible. In consequence, the variance-based decomposition methods are worth investigation.

ANOVA and GSA are at the base of the mentioned variance-based decomposition method. A dimension-adaptive procedure may be based on the SI [115]. These indices identify which (combinations of) variables $\xi^{(i)}$ influence the most the QoI $u(\boldsymbol{x}, \boldsymbol{\xi})$ in terms of total variance. Three indices may be identified that correspond to main (or first-order) $S^{(i)}$, cross- (or second-order) $S^{(i),(j)}$ directions, and total $S_T^{(i)}$ effects. Following the notations of Eldred [111], these indices are defined in the following fashion:

$$S^{(i)} = \frac{\mathbb{V}_{\xi^{(i)}}\left[\mathbb{E}\left[u|\xi^{(i)}\right]\right]}{\mathbb{V}\left[u\right]} \qquad \text{with} \quad 1 \leq i \leq N, \tag{4.19}$$

$$S^{(i),(j)} = \frac{\mathbb{V}_{\xi^{(i),(j)}}\left[\mathbb{E}\left[u|\xi^{(i),(j)}\right]\right]}{\mathbb{V}\left[u\right]} \qquad \text{with} \quad 1 \leq i, j \leq N, j \neq i, \tag{4.20}$$

$$S_T^{(i)} = 1 - \frac{\mathbb{V}\left[\mathbb{E}\left[u|\boldsymbol{\xi}^{(\sim i)}\right]\right]}{\mathbb{V}\left[u\right]} \qquad \text{with} \quad 1 \leq i \leq N, \tag{4.21}$$

where $\mathbb{V}[\cdot]$ is the variance operator and $\boldsymbol{\xi}^{(\sim i)} = \left(\xi^{(1)}, ..., \xi^{(i-1)}, \xi^{(i+1)}, \xi^{(N)}\right)$. Recent developments point out the advantages of the *anchored* ANOVA especially in term of computational savings [137].

The N-dimensional integrals are in general evaluated through Monte-Carlo techniques. Tang *et al.* [117] recently showed how to evaluate the above defined indices based on a SC approximation. As correctly pointed out by the authors, GSA should not be confused with UQ. In the present study a GSA has been performed as a subset of a UQ study thus, given uncertain inputs, the most influential ones have been identified thanks to relations (4.19-4.21).
A dimension-adaptive procedure based on SI is *not* a novel idea. Nevertheless, the literature about related approaches is very scarce and does not discuss algorithmic implementation [111]. Moreover, it relies only on total-order SI. Hereafter, an improved method together with its detailed algorithmic implementation is proposed. Its efficiency with respect to the few available data in literature is also investigated [111]. The refinement approach proposed in this study is based on the main *and* cross-direction effects monitored by the $S^{(i)}$ and $S^{(i),(j)}$ terms, respectively and not only based on the total direction effect $S_T^{(i)}$. With the present technique, the information arising from the GSA is richer and a more straightforward dimension-adaptive refinement procedure may be performed.

Algorithm 3 provides the main steps of the numerical implementation. The notation is in part inspired by the one of Gerstner *et al.* [120]. The algorithm inputs are: – the stochastic dimension N, – the previous (old) active index set

Data: $N, \boldsymbol{\mathcal{O}}, \boldsymbol{S}, nb_{IT}^{max}, nb_{iso}^{max}, l_i^{max}, cross$
Result: $\boldsymbol{\mathcal{A}}$
$\boldsymbol{\mathcal{A}} = \boldsymbol{\mathcal{O}}$
for $nb_{IT} \leq nb_{IT}^{max}$ ***or*** $\epsilon_{conv} > \epsilon_{conv_0}$ **do**
  if $|\sum_i [S^{(i)} + \sum_{j>i} S^{(i),(j)}] - 1| > \epsilon_1$ ***and*** $nb_{iso} \leq nb_{iso}^{max}$ ***and*** $aniso = 0$ **then**
    $\boldsymbol{\mathcal{A}} = \boldsymbol{\mathcal{A}} \cup \boldsymbol{j}_{1,...,N}^{+1}$
    $nb_{iso} + +$
    $nb_{IT} + +$
  else
    for $i \leq N$ **do**
      if $|S_{max}^{(i)} - S^{(i)}| / S_{max}^{(i)} \leq \epsilon_2$ ***and*** $l_i \leq l_i^{max}$ **then**
        $\boldsymbol{\mathcal{A}} = \boldsymbol{\mathcal{A}} \cup \boldsymbol{j}_i^{+1}$
        $aniso = 1$
        $l_i + +$
        $i + +$
      end
    end
    for *all* $\binom{N}{2}$ *combinations of* i, j **do**
      if $|S_{max}^{(i),(j)} - S^{(i),(j)}| / S_{max}^{(i),(j)} \leq \epsilon_3$ ***and*** $cross = 1$ **then**
        $\boldsymbol{\mathcal{A}} = \boldsymbol{\mathcal{A}} \cup \boldsymbol{j}_{i,j}^{+1}$
        $aniso = 1$
      end
    end
    $nb_{IT} + +$
  end
  Evaluation of ϵ_{conv}
end

Algorithm 3: Main steps of the Sobol' based dimension-adaptive Smolyak sparse grid (SC-Sobol) construction algorithm.

$\boldsymbol{\mathcal{O}}$ (typically from a grid of level l_2), – the SI vector $\boldsymbol{S} = \left(S^{(i)}, S^{(i),(j)}\right)$ estimated on the set $\mathcal{O}$ – the user-defined limiters nb_{IT}^{max}, nb_{iso}^{max} and l_i^{max} – and the user-defined boolean variable *cross* which allows the user to decide whether to use the second-order information from $S^{(i),(j)}$ or not. The algorithm returns: – the (new) active index set $\boldsymbol{\mathcal{A}}$. The index sets $\boldsymbol{\mathcal{O}}$ and $\boldsymbol{\mathcal{A}}$ are two sets gathering all indices $\boldsymbol{j} = (j_1, ..., j_N) \in \mathbb{N}^N$ (introduced in Section 4.2) which basically identify the quadrature levels. One main *if*-loop and two second *if*-loops may be identified. The first one states whether the refinement should be isotropic, *i.e.* following the standard Smolyak algorithm (noted in compact form as $\boldsymbol{j}_{1,...,N}^{+1}$), or anisotropic. If anisotropic, the refinement may be taken along the main direction (noted as $\boldsymbol{j}_i^{+1}$) and/or in a cross-direction (noted as $\boldsymbol{j}_{i,j}^{+1}$). These three loops are explained in more details hereafter:

1. Isotropic refinement (three main conditions need to be fulfilled)

   (a) $|\sum_i[S^{(i)} + \sum_{j>i} S^{(i),(j)}] - 1| > \epsilon_1$: check is necessary to avoid wrong refinement suggested by a poor initial estimation of $\boldsymbol{S}$. Namely, it is suggested that $\sum_i[S^{(i)}+\sum_{j>i} S^{(i),(j)}]$ should tend to 1 (condition coming from relations (4.19-4.21)). No condition has been imposed on single $S^{(i),(j)}$ since the cross-dimension effect may be exactly null.

   (b) $nb_{iso} \leq nb_{iso}^{max}$: isotropic refinement is user-bounded by nb_{iso}^{max}.

   (c) $aniso = 0$: checks whether the dimension-adaptive procedure has been already performed once. If this is the case, the isotropic refinement cannot be launched again. It is important then to carefully choose ϵ_1 and nb_{iso}^{max}.

   If these conditions are satisfied, the active set $\boldsymbol{\mathcal{A}}$ becomes the union of $\boldsymbol{\mathcal{O}}$ and the isotropic increment *in all direction* noted as $\boldsymbol{j}^{+1}_{1,\dots,N}$. The index diagram is then similar to Figure 4.2.

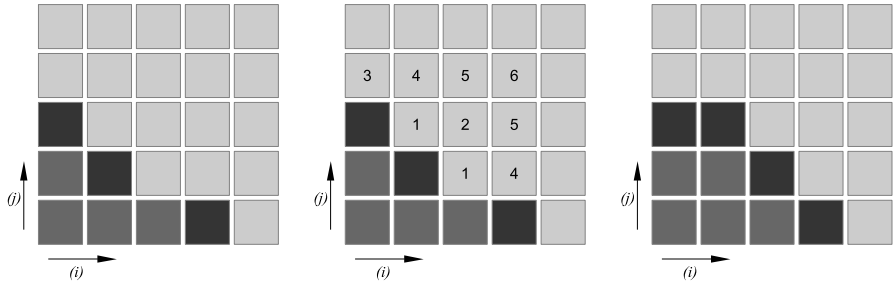
2. Anisotropic refinement.
   Introducing $S_{max}^{(i)} = \max_i(S^{(i)})$, $S_{max}^{(i),(j)} = \max_{i,j}(S^{(i),(j)})$, refinement may be taken along:

   (a) main direction if $|S_{max}^{(i)} - S^{(i)}|/S_{max}^{(i)} \leq \epsilon_2$. Supposing that there is at least one[6] dimension for which $S^{(i)}$ is large, the condition states whether the refinement in a main direction has to be done with respect to $S_{max}^{(i)}$. For $\epsilon_2 = 1$, the enrichment is in all main directions (i). This type of refinement is user-bounded by l_i^{max}. Figure 4.4 (a) shows the updated index diagram.

   (b) cross directions if $|S_{max}^{(i),(j)} - S^{(i),(j)}|/S_{max}^{(i),(j)} \leq \epsilon_3$. This is a similar condition as the previous one but expressed for cross-direction indices. Figure 4.4 (c) shows an example of index diagram. The refinement in a $((i),(j))$ plane may not be trivial since several intermediate refinement options may be identified. The chosen one is shown in Figure 4.4 (b). Please note the steps ordering. Even if some steps may be switched[7], *e.g.* 2 and 3, the method would not be consistent. Since the cross-direction refinement is addressed in this loop, all the possible cross-refinement options must be preferred. When cross-refinement is not allowed anymore and the refinement in the $((i),(j))$ plane has to be pursued, the refinement along the main direction (i) is allowed, *e.g.* step 3 in Figure 4.4 (b).

[6] If not, there is no reason for using adaptivity.

[7] since they are still admissible, please refer to Section 4.2



(a) Refinement along the main direction (i): $\boldsymbol{j}_i^{+1}$

(b) Progressive refinements along the cross-direction $\big((i),(j)\big)$

(c) Refinement along the cross-direction $\big((i),(j)\big)$: $\boldsymbol{j}_{i,j}^{+1}$

Figure 4.4: 2D index diagram for the Sobol' based dimension-adaptive Smolyak sparse grid algorithm.

Table 4.1: Suggested thresholds for limiters of Algorithm 3.

ϵ_1	ϵ_2	ϵ_3	nb_{iso}^{max}	l_i^{max}
$1 \cdot 10^{-3}$	0.8	0.8-0.9	1-3	6-10

The adaptation loop is stopped as soon as the convergence target ϵ_{conv_0} for the mean or the variance is achieved or the maximum number of loop iterations nb_{IT}^{max} is reached. Note that there is no stopping criterion based on the analysis of statistics between two (or more) consecutive refinement steps. This is one of the originalities of the proposed contribution. Without loss of generality, in this study, the dimension-adaptive approximation only uses up to second-order SI. A user guideline for the suggested thresholds for the ϵ_1, ϵ_2, ϵ_3, nb_{iso}^{max} and l_i^{max} limiters of Algorithm 3 are reported in Table (4.1).

4.4 Applications to test functions and 2D aerodynamic simulations

In this section, the methods introduced in Section 4.3 have first been applied to test functions, in Section 4.4.1, and then to airfoil aerodynamics simulations, in Section 4.4.2. The compared results are obtained from sample-based numerical approximations that all share *same* computational Smolyak-constructed *sparse* grids, unless otherwise acknowledged. The stochastic dimension N ranges from three (or four) for the test functions to $N = 8$ for the RANS aerodynamics simulations. In this section, the different results will be designated by compact acronyms as follows. Pseudospectral approximations results will be mentioned with notations containing the term 'gPC' to indicate that they rely on generalized Polynomial Chaos representations:

1. Standard projection approximation will be mentioned as gPC$^{\#D}$, with superscript referring to the type of construction retained for the polynomial approximation basis, *i.e.* gPC^{TD} for *total*, and gPC^{SD} for *sparse*[8], polynomial degree.

2. Sparse Smolyak projection approximations will be referred as SgPC.

Stochastic Collocation methods will be denoted as:

3. SC for the *uniform*/isotropic enrichment and

4. SC-Sobol for the Sobol'–based *anisotropic* one

4.4.1 Test functions

In this section, a *nonlinear* and *anisotropic* stochastic multi-variate test functions are considered. Low-order moments and sensitivity coefficients are exactly known for these functions and will be used to measure the performance of the numerical approximations. The gPC^{TD} results are obtained for a choice of TD polynomial order close, but not necessarily equal to the theoretical one provided by the polynomial exactness of the quadrature used. Several orders are tested and the one providing the lowest statistical error is selected.

Ishigami function - $N = 3$

The Ishigami or Homma-Saltelli function [138] reads as follows:

$$f(\boldsymbol{\xi}) = \sin(2\pi\xi^{(1)} - \pi) + 7\sin^2(2\pi\xi^{(2)} - \pi) + 0.1(2\pi\xi^{(3)} - \pi)^4 \sin(2\pi\xi^{(1)} - \pi), \tag{4.22}$$

for independent and equally distributed $\xi^{(i)} \in \mathcal{U}_{[0,1]}$. This function is characterized by two particular features which make the function particularly challenging for GSA studies: – it cancels at $\xi^{(1),(2),(3)} = 0.5$ and at bounds and – there is a strong *dependency* of the function to the single variables $\xi^{(1)}$ and $\xi^{(2)}$ and to the $(\xi^{(1)}, \xi^{(3)})$ combination.
SI seem a good candidate for studying the latter feature. The analysis of these indices shows, as expected, that S_1, S_2 and $S_{1,3}$ dominate. This example represents a typical case in which the $S^{(i),(j)}$ indices, monitoring the cross interaction between two random variables, are important.

The efficiency of the proposed dimension-adaptive method SC-Sobol based on SI has been tested. Convergence plots based on the relative error in the solution variance are shown in Figure 4.5. Results are compared from several methods including ones from the literature, *e.g.* Eldred *et al.* [111]. The comparison is done

[8] *Sparse* degree here refers to the monomial coverage of the underlying *sparse* quadrature. It projects onto the same basis as SgPC. Roughly speaking this is a polynomial degree of half the value of the gray symbols in Figure 4.3 but which does not guarantee the discrete orthogonality of the approximation [112].

both in terms of stochastic approximation methods and quadrature rules. The uniform refinement almost leads to the same convergence rate for both SC and SgPC with CC nodes while the slowest convergence is obtained for gPC^{TD}. Dimension-adaptive SC-Sobol methods perform better than isotropic approximations and the use of GP grids makes the convergence faster than CC nodes. For GP nodes, the method based on our algorithm reaches machine accuracy with around 3.5 times less evaluations than Eldred's implementation. As expected from the literature, the GSG goal-oriented method with the GP points performed the best [120].

The robustness of the proposed method has been investigated for different values of nb_{iso}^{max} and l_i^{max}, see for instance Figure 4.6. Another interesting feature of this method is the presence of step-wise convergence patterns. The existence of these *plateaus* is related to the refinement criterium illustrated in Figure 4.4 (b). For instance, in order to reach the index set #6, it may be necessary to go through several successive steps/simulations that may not pay off right away in term of variance convergence. This behavior gives grounds for the lack of stopping criterion based on the analysis of statistics between two (or more) consecutive refinement steps as mentioned before.
Convergence results for the main (a) and cross (b) SI are presented in Figure 4.7. For this function, the cross-dimension index is the hardest to capture, as its convergence pattern follows the variance convergence, cf. Figure 4.5 (b). Again the proposed Sobol' based adaptive approximation performs well, especially for the $S^{(1),(3)}$ index estimation.

In Figure 4.5-(b), gPC^{SD} results were not given. The reason lies in its unsatisfactory convergence pattern. Both methods are further explored in Figure 4.8. Figures 4.8 -(a) and -(b) show the monomial coverage emerging from the approximations while -(c) shows the polynomial approximations coefficients. It is clear from the colors distribution and the modal coefficients that the sparse pseudospectral representation correctly captures the sparsity and the cross interaction of the response reflecting a much better convergence and lower aliasing errors than the standard approximation.

Scalable Gerstner test function [120] - $N = 4$

The method can also be tested on functions with more additive effects, *e. g.* where the dependency on the N independent variables is determined by $\sum$ and $\prod$ operators of all terms. The scalable Gerstner [120] test problem, here extended to N (even) dimensions, reads as follows:

$$f(\boldsymbol{\xi}) = \sum_{i=1}^{N/2} \left[\exp^{-\left(\xi^{(2i-1)}\right)^2} + 10 \exp^{-\left(\xi^{(2i)}\right)^2} \right], \tag{4.23}$$

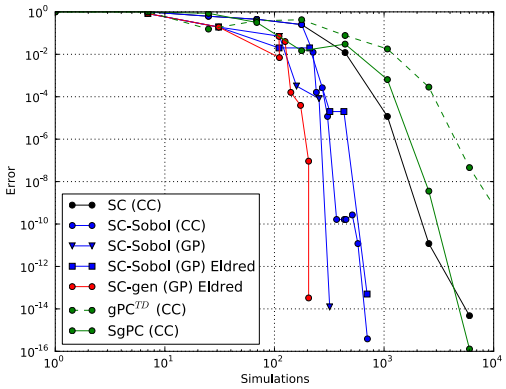
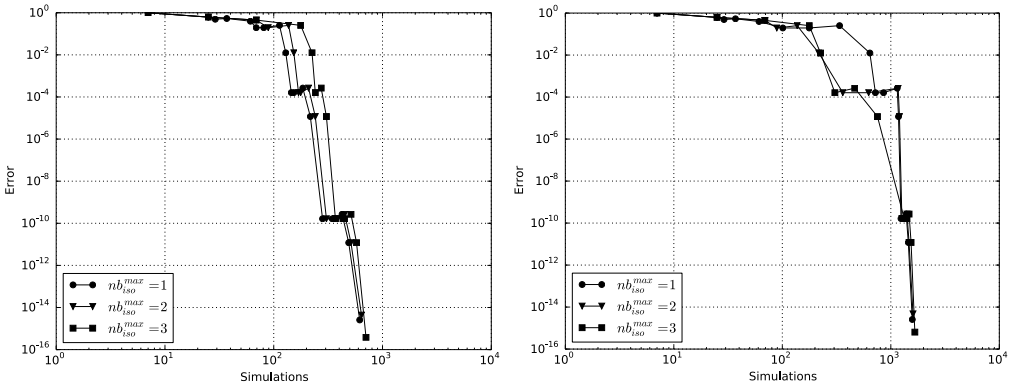
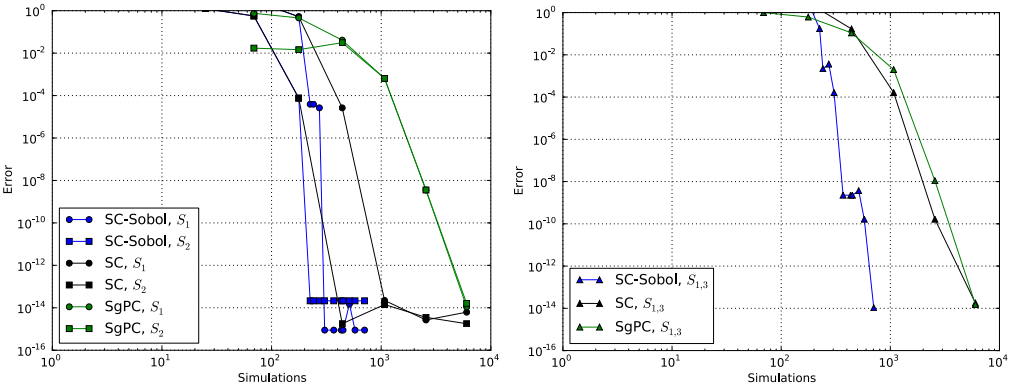


Figure 4.5: Variance convergence for the Ishigami function. Horizontal axis represents the total number of function evaluations. Comparison of sparse grids-based SC and gPC approximations. Clenshaw-Curtis and Gauss-Patterson nodes denoted respectively by (CC) and (GP); Eldred results from [111].



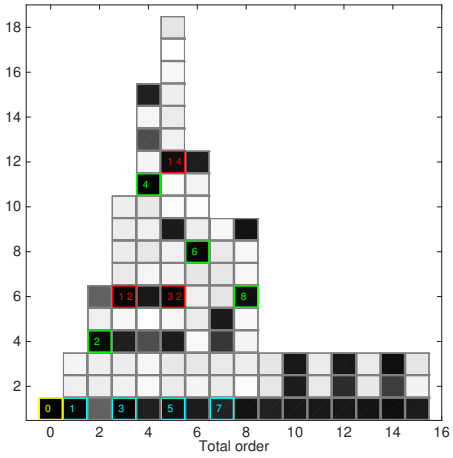
(a) $l_i^{max} = 6$ (b) $l_i^{max} = 10$

Figure 4.6: Variance convergence for the Ishigami function with SC-Sobol'. Influence of the choice of nb_{iso}^{max} for two different l_i^{max}.

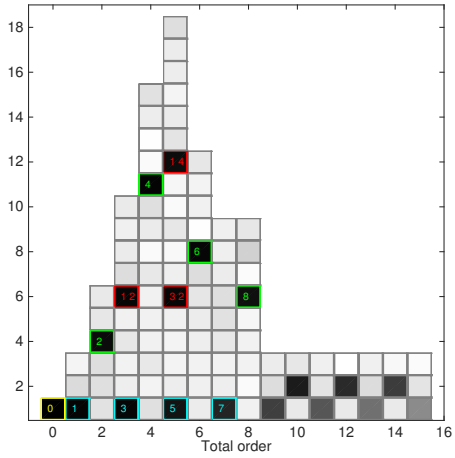


(a) First-order SI $S^{(i)}$ (b) Second-order SI $S^{(i),(j)}$

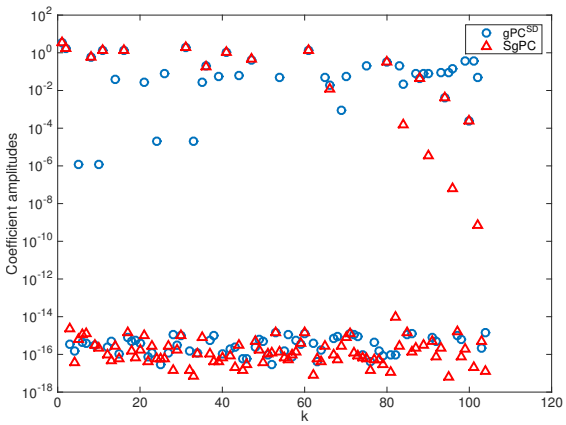
Figure 4.7: SI convergence for the Ishigami function; same caption as previous figure.



(a) Ishigami gPC^{SD} polynomial approximation

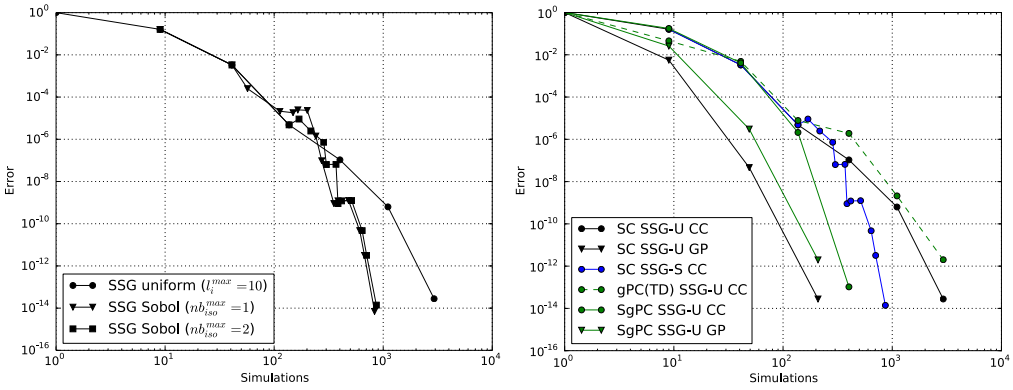


(b) Ishigami SgPC polynomial approximation



(c) Approximations coefficients

Figure 4.8: Comparison of standard gPC^{SD} vs. sparse SgPC pseudospectral approximations of the Ishigami function. Top two figures (a-b) display the monomial coverage emerging from the approximations. The horizontal axis refers to the order of each term from the expansion and the color represents $\log_{10}$ of the coefficients magnitude. Dark (respectively light) colors correspond to large (respectively low) magnitudes. Squares with cyan (respectively green and red) edge color refers to the correct dominant polynomial terms in the $\xi^{(1)}$ (respectively $\xi^{(2)}$ and $(\xi^{(1)}, \xi^{(3)})$) variables. Corresponding polynomial orders are indicated within those squares. (c) shows the magnitude of the polynomial approximation coefficients.



(a) $l_i^{max} = 6$ (b) Comparisons

Figure 4.9: Variance convergence for the Gerstner test problem. Tests at $nb_{iso}^{max} = 1, 2$ with two different l_i^{max}.

for independent and equally distributed $\xi^{(i)} \in \mathcal{U}_{[0,1]}$. In the following, results are presented and discussed for $N = 4$. Figure 4.9-(a) shows the variance convergence for two different nb_{iso}^{max} against the result obtained with uniform Smolyak sparse grid. The target convergence is achieved faster thanks to the adaptivity. Please note as well, that the computational speedup is lower with respect to the Ishigami function. The reason may lay again in the nature of $S^{(i),(j)}$ indices. In Figure 4.9-(b) the comparison with gPC^{TD} and SgPC results based on the same grid are also given for CC and GP nodes. For this C^{∞} class of functions, SgPC based on uniform grids performs better than any version of SC approximations based on the same type of quadrature rule for CC nodes. While, for GP nodes, the behavior of SC and SgPC is similar.

Discussion

- For uniform/isotropic refinement, the most straightforward approaches are SC and SgPC. They perform similarly unless the problem is very smooth with additive effects in which case the spectral accuracy of the SgPC approximation seems to outperform the SC approximations. A possible explanation lies in the *a priori* choice of the gPC polynomial space of approximation which is obviously influent but beyond the scope of this study. gPC$^{\#D}$ methods either converge slower or do not converge, the poor behavior of SD being due to strong aliasing errors. Overall, GP grids show better performance than CC grids for comparable number of nodes.

- For dimension-adaptive/anisotropic refinement, only the SC methods have been tested. Adaptive interpolation methods based on SI criteria are more accurate and less costly than their isotropic counterpart, in particular for this kind of function where accurate capture of high-order cross interactions

between the stochastic scales is required. Moreover, the algorithm together with its numerical implementation seems more performant than the one found in the literature. At this stage, it is hard to infer on the causes of such a difference due to the lack of algorithmic details provided in the referenced paper. It is conceivable that the control of the second-order $S^{(i),(j)}$ (and possibly third-order $S^{(i),(j),(k)}$) terms, may cause the efficiency improvement of the approximation.
Finally, generalized sparse grid methods based on adaptive tensor-product quadrature are shown to perform very well when the solution is smooth [120]. However the efficiency of GSG can be significantly improved when small local regions of the input space contribute to the model's variability [105]. Moreover, efficient implementation of the algorithm in terms of efficient data structures for the storage and index sets bookkeeping is not straightforward.

There exists a wide range of test functions for checking the efficiency of integration technique involved in statistics computation. The Genz's [139] is one of the most exploited benchmark packages. The Ishigami function behaves very differently from the Genz's test functions for which dependency of the N independent variables relate by definition to $\sum$ and $\prod$ operators of all the terms. On one hand, functions *with* explicit cross relation within certain variables, *e.g.* Ishigami; on the other hand, functions *without* explicit cross relation and more additive effects, *e.g.* Gerstner (and continuous Genz's) functions as in Section 4.4.1. In the latter case, $S^{(i)}$ indices alone are reasonably able to capture the main direction in which the refinement would be needed in term of analysis of the variance. Moreover, the $S^{(i),(j)}$ terms are less relevant as there is no explicit interactions of variables i with j and vice versa. It is then reasonable to think that a dimension-adaptive method as defined in Section 4.3.2 is not the most efficient approach to reproduce such functions.

Some remarks about the proposed SC-Sobol interpolation method: some results have shown the benefit of limiting the highest quadrature level along the main direction via the l_i^{max} parameter. This should not be seen as a weak point but as an asset. Indeed, for the level of accuracy foreseen for CFD applications (let say with $N > 4$), it will not be possible to reach beyond the l_6 level.
In the following Section, these stochastic approximation methods have been deployed in the framework of airfoil aerodynamics in subsonic regime subject to operational and geometrical uncertainties.

4.4.2 Airfoil aerodynamic simulations

The *first* step is the identification and quantification of the uncertainties [82] that are relevant to the realistic constraints encountered during an aircraft conception and testing, *i.e.* wind-tunnel and real flight conditions. Then, the physical

model approximation plays a crucial role in the QoI estimation. Several models exist which may give different values for the same QoI. Identified uncertainties may have either an higher or lower impact on the QoI with respect to the chosen physical approximations. In the following aerodynamic applications, models may have a higher impact on the QoI. Nevertheless the UQ studies are still valuable to get some insight on the sensitivity of *that* particular model with respect to the selected uncertainties. The present approach is based on the assumption that the discretization and truncation effects are similar in all computations and their impact on the observed quantities is minor. Due to the computational cost of reliable CFD and the high-dimensionality of the probability space, there is a need for efficient *adaptive* stochastic approximations. Approximations based on sparse grid techniques are potential candidates and are worth investigation in an aerodynamical framework.

The following studies address the aerodynamic performance prediction of two-dimensional subsonic steady flow at $Re = 1.95 \cdot 10^6$ with a freestream Mach number $M = 0.291$ about a NACA0015 airfoil in the presence of bounded operational and geometrical uncertainties. Two studies with different deterministic models have been carried out in this framework, in an effort to quantify the effect of different levels of accuracy and versatility of the models on the UQ analysis.
For the first study, in Section 4.4.3, the assessment of the effects of $N = 3$ simultaneous uncertainties on five different objectives has been made with different stochastic approximation methods. The deterministic code retained, XFOIL [8], is a fast interactive panel method based flow solver used for the design and analysis of airfoils.
For the second study, in Section 4.4.4, a fully turbulent RANS solver has been used. In this more challenging case, the stochastic problem has a dimension $N = 8$. Sections 4 and 4 motivate the choice of the sources of uncertainties that have been considered in these studies.Some experimental values are available thanks to the work of Piziali [29] performed at the NASA Ames Research Center.

Distribution laws

The probability distribution of the identified uncertain parameters has to be inferred. Here, only minimal and maximal uncertain value estimates are available, so the *uniform* distribution *should* be used, in order to fulfill the Jaynes' principle of maximum entropy [140]. At this stage, any other distribution will contain more information and less uncertainty thus less *entropy*. Normal distribution abundant in the literature is not always the best choice for three main reasons: 1. it is not bounded, 2. it is symmetric, 3. the mean value is more likely to occur. If there is no evidence of properties 2. and 3. in the experimental data, then the distribution would not fulfill the Jaynes' principle.

Table 4.2: Uncertain parameters for the XFOIL study (std values are expressed as percentage of mean values).

Uncertain param.	*iid* uniformly distributed $\mathcal{U}_{[\mu \pm \text{std}\%]}$
AoA	$8°$ $\pm$ 0.4%
M	0.291 $\pm$ 0.4%
N_{crit}	9 $\pm$ 33%

4.4.3 Simplified aerodynamics model

The present application concerns the estimation of the statistics of five different objectives in the presence of three simultaneous uncertainties. The five objectives are the lift C_l, drag C_d, pressure drag C_{dp}, momentum C_m coefficients and the transition point on the pressure side X_{tr}. As mentioned before, XFOIL developed by Drela [8] has been used. It employs an inviscid formulation to solve the outer domain and integral boundary layer (BL) formulation to solve the viscous inner domain, near the airfoil surface and in the wake. Its inviscid formulation is a linear-vorticity stream function panel method. Compressibility is included by applying a compressibility correction for Mach numbers up to sonic conditions. The interaction between the BL and the external flow is dealt via a surface transpiration model. Natural BL transition is predicted through an $e^{N_{crit}}$ method. Interesting comparison of XFOIL prediction of transitions with respect to RANS flow may be found in [141]. XFOIL is very convenient for UQ studies since it has a short evaluation time of the order of a few seconds. Use of a XFOIL solver in an UQ study of a WT experiment has been also adopted by Boon *et al.* [142].

The imposed uncertainties are listed in Table (4.2). The uncertain coefficient on the angle of attack (AoA) is justified by the experimental campaign of Piziali where it is stated that the instantaneous angle of attack may have a deviation of as mush as $0.3°$ [29]. The third chosen parameter relates to the second item in the list of Section 4. A standard value, for the aforementioned exponential factor N_{crit} needed for the prediction of the transition, is $N_{crit} = 9$. In practice, this value changes according to the WT condition: typically, $4 \leq N_{crit} \leq 8$ for dirty facilities and $10 \leq N_{crit} \leq 12$ for clean ones. In accordance, it has been chosen a variability of N_{crit} in the range between 6 and 12. The transition point at the suction side has been imposed very close to the leading edge at 0.5% x/c, in agreement with the experimental conditions. This leads to the cancellation of the laminar leading edge bubble. The simulations have been run with 245 panels.

Due to the very short computational time needed for each computation, it has been possible to perform refined stochastic simulations in order to obtain valuable reference statistical values for later comparison. The obtained estimations are shown in Table (4.3). Imposing a maximum number of simulation in the order of

4000, the results for the SC based on full tensor grid (SC-Full) of Gauss-Legendre nodes as well as Monte-Carlo simulations are compared. Based on the high polynomial exactness properties of the GL rule as well as other results not included in this manuscript, the SC-Full (GL) results have been retained as *reference* values.

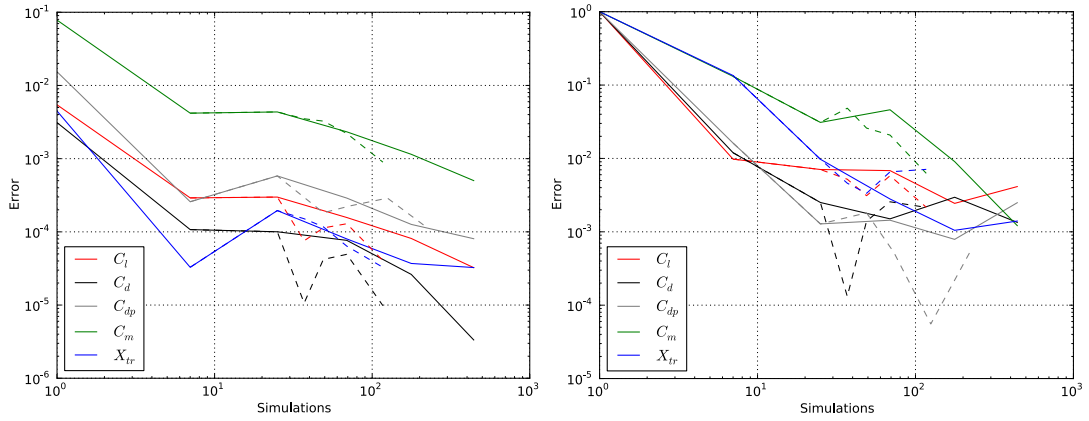
Table 4.3: XFOIL case. Reference objectives values with the available experimental data (Exp.), the nominal simulation with XFOIL and the statistics from SC-Full (GL nodes) and from MC.

Objective		Exp.	Nom. (1)	SC-Full ($n_{\boldsymbol{l}}^{(\boldsymbol{N})} = 4096$)	MC ($n_{\boldsymbol{l}}^{(\boldsymbol{N})} = 4000$)
C_l	mean $[\cdot 10^{-1}]$	8.528	8.978834	9.027929	9.032001
	std $[\cdot 10^{-2}]$	-	-	2.563730	2.569163
C_d	mean $[\cdot 10^{-2}]$	0.9	1.199885	1.203627	1.204123
	std $[\cdot 10^{-4}]$	-	-	2.466207	2.484428
C_{dp}	mean $[\cdot 10^{-3}]$	-	-2.322750	-2.359028	-2.355695
	std $[\cdot 10^{-4}]$	-	-	4.239083	4.163407
C_m	mean $[\cdot 10^{-2}]$	1.86	1.430439	1.326815	1.324301
	std $[\cdot 10^{-3}]$	-	-	1.827151	1.847993
X_{tr}	mean $[\cdot 10^{-1}]$	-	9.876674	9.832693	9.833519
	std $[\cdot 10^{-3}]$	-	-	9.764260	9.755916

Table (4.4) shows the relative errors of the statistics of the five objective obtained from the different methods with respect to the reference solution. It is important to emphasize that the obtained statistics accuracy is very high, *i.e.* $\mathcal{O}(10^{-5})$ for the mean (except for the C_m quantity, one order of magnitude higher) and $\mathcal{O}(10^{-3})$ for the standard deviation. Considering methods with uniform sparse enrichment, *i.e.* SC and all gPC methods reported here, the std errors are all of the same order of magnitude except for X_{tr} that is best estimated by SgPC and for C_m that is worst estimated by gPC^{SD}. It is also interesting to point out that for the same grid the SgPC approximation is always more accurate than the gPC$^{\#D}$ approximations.

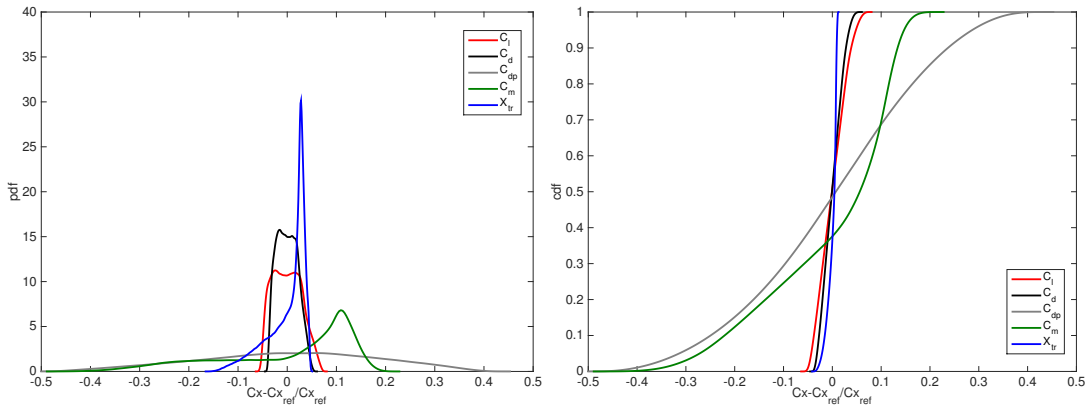
Moving to the adaptive SC-Sobol results, this part of the table should be reviewed together with Figure 4.10 that shows the statistics convergence against the number of samples. The SC-Sobol are better than the most refined isotropic approximation SC l_5 for the C_l and C_{dp} std, of comparable magnitude for C_d and worse for C_m and X_{tr}, which are the most difficult quantities to approximate. Interestingly, the adaptive greedy approach is always more accurate than the level SC l_4 with similar number of simulations, except for the X_{tr} std.

In order to deepen the analysis, Figure 4.11 shows the centered and normalized probability density functions (PDF) and the centered cumulative density functions



(a) Mean (b) Variance

Figure 4.10: XFOIL. Convergence of the relative error of SC (continuous line) and SC-Sobol (dashed line) with respect to SC-Full (GL with 4096 simulations).



(a) PDF (b) CDF

Figure 4.11: XFOIL-based PDF and CDF of aerodynamics coefficients Cx from SC (CC l_5) approximation with 1M samples. The functions are centered (cf. Table (4.3)) and normalized for the sake of comparaison. An additional ($\times 4$) factor is used for the normalization of the results of the X_{tr} PDF.

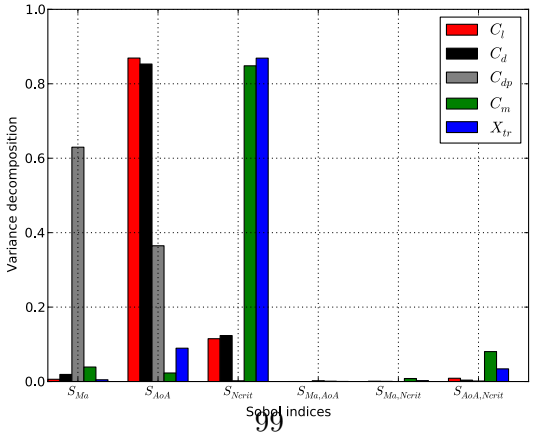


Figure 4.12: XFOIL-based first- and second-order SI obtained from the SC (CC l_5) approximation.

Table 4.4: XFOIL. Relative errors of the approximated mean and standard deviation with respect to the reference values, cf. SC-Full (GL) computations from Table (4.3). For each line, the lowest error in the statistics is identified by a black star*. Lowest errors between SC l_4 and SC-Sobol are pointed by a dagger† symbol.

Objective stat.		SC		SC-Sobol		gPC^{TD}	gPC^{SD}	SgPC
		$l_4(177)$	$l_5(441)$	(-)		(441)	(441)	(441)
C_l	mean [$\cdot 10^{-5}$]	8.1235	3.2413*	4.0502†	(117)	3.2413*	3.2413*	3.2413*
	std [$\cdot 10^{-3}$]	2.4408	4.1348	2.1616*†	(117)	4.5754	4.1806	2.1862
C_d	mean [$\cdot 10^{-5}$]	2.6226	0.3336*	0.9801†	(117)	0.3336*	0.3336*	0.3336*
	std [$\cdot 10^{-3}$]	2.9579	1.3439	2.1516†	(117)	1.5168	1.2281	0.7188*
C_{dp}	mean [$\cdot 10^{-5}$]	12.6417	8.0567*	11.5950†	(221)	8.0567*	8.0567*	8.0567*
	std [$\cdot 10^{-3}$]	0.7888	2.4916	0.5278*†	(221)	2.5718	1.4419	1.2549
C_m	mean [$\cdot 10^{-5}$]	114.6256	50.1033*	89.5465†	(117)	50.1033*	50.1033*	50.1033*
	std [$\cdot 10^{-3}$]	9.0573	1.2141	6.2652†	(117)	2.7514	18.3797	0.4919*
X_{tr}	mean [$\cdot 10^{-5}$]	3.6997	3.2530*	3.2762†	(117)	3.2530*	3.2530*	3.2530*
	std [$\cdot 10^{-3}$]	1.0472†	1.4022	7.0975	(117)	0.7522	5.0904	0.2214*

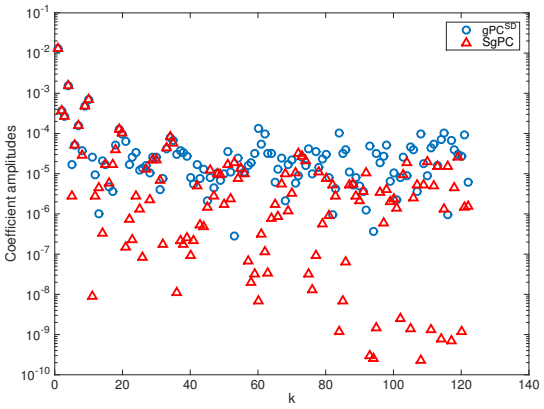
(CDF) for the five objectives from SC CC l_5 obtained trough Lagrange interpolation with 1 million (1M) MC samples. PDFs and CDFs approximations are obtained on 200 equally spaced data points, from a kernel density estimate package with normal kernel and optimized bandwidth following Silverman's rule of thumb. From the analysis of the profiles together with coefficient of variations, the following interesting observations may be derived:

1. C_{dp} and C_m are the most sensitive quantities with respect to the uncertain parameters while X_{tr} is the least sensitive;

2. None of the distributions are fully symmetric, but X_{tr} and C_m have the most skewed and nonlinear response with respect to the nominal deterministic value;

3. C_l and C_d have very similar profiles close to *uniform* distributions, while C_{dp} exhibits a hat-looking profile reminiscent of a sum of *uniform* distributions.

The variance decomposition in term of SI is depicted in Figure 4.12. Clear interaction between stochastic scales (here $S^{(AoA),(N_{crit})}$) are only present for the C_m response (and to some extent for the X_{tr} one), which is coherent with their multimodal PDF shape. All other physical quantities are strongly dependent either on one (here $S^{(AoA)}$ for C_l and C_d) or two (here $S^{(M)}$ and $S^{(AoA)}$ for C_{dp}) random inputs. While it makes sense that both C_l and C_d depend on the angle of attack and to a lesser extent to the coefficient affecting the transition location, it is interesting



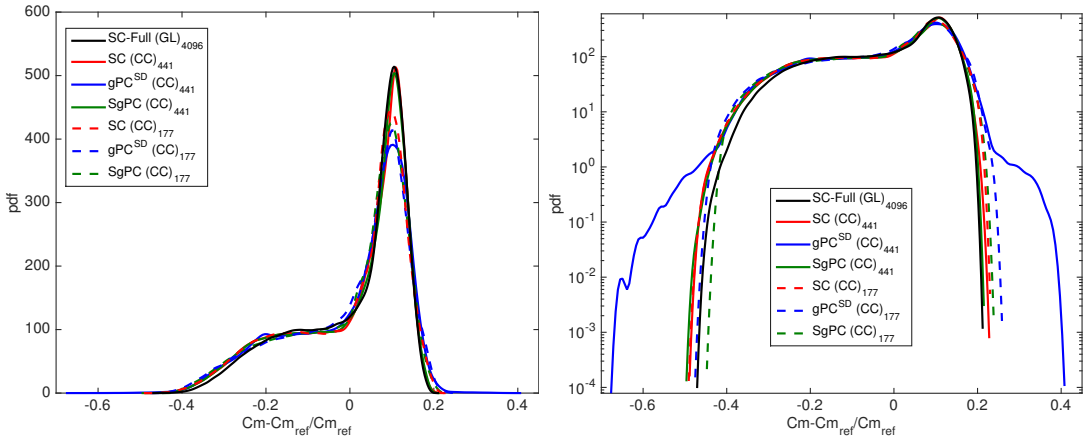
(a) Approximations coefficients

Figure 4.13: Comparison of the standard gPC^{SD} and the sparse SgPC pseudospectral polynomial approximation coefficients of the C_m aerodynamic coefficient considered in the study of Table (4.4).

to notice that due to removal of the friction effects, C_{dp} depends more evenly on both the Mach number and the airfoil incidence. As expected, the position of the transition point X_{tr} is strongly influenced by N_{crit}.

It seems interesting to investigate why sparse pseudospectral approximations are so successful in representing the C_m and X_{tr} statistics. The monomial coverage emerging from the pseudospectral approximations and the polynomial approximations coefficients are provided in Figure 4.13. Again, colors distribution and modal coefficients show that the sparse representation exhibits a more regular decay across the higher modes indicating a better convergence and lower aliasing errors than the standard approximation. It is also beneficial to compare the PDFs obtained from the other approximations, cf. Figure 4.14. It shows the distributions obtained from the coarse and fine standard gPC^{SD} and SgPC pseudospectral approximations compared to fine and coarse stochastic collocations (all based on uniform CC grids) of the C_m aerodynamic coefficient. The sparse approximations produce smoother and narrower distributions than the standard polynomial approximations due to the reasons exposed herebefore. Fine sparse polynomial approximation and stochastic collocation based distributions are very close. The comparison with the reference distribution obtained from the SC-Full (GL) grid shows that the peak and highest values are well resolved by these approximations. However the left tail is not completely well captured (the difference is more visible in the semi-logarithmic view).

The use of XFOIL is limited to moderate angle of incidence. The code indeed does not always converge for AoA$>$ 12° (severe flow detachment regime). For more realistic configurations with presence of both operational and geometrical



(a) standard view (b) semi-logarithmic view

Figure 4.14: Distributions obtained from the coarse and fine standard gPC^{SD} and SgPC pseudospectral approximations compared to fine and coarse SC (all based on uniform CC grids) of the C_m aerodynamic coefficient. The distributions have been centered and scaled according to the reference value of Table (4.4).

uncertainties, the use of a more complex (but more costly) aerodynamic simulator is needed. RANS modeling is a good candidate for this purpose.

4.4.4 Reynolds-Averaged Navier-Stokes model

The present study addresses the prediction of 2D subsonic turbulent flow about a NACA0015 airfoil. The computations are carried out by means of RANS simulations. The turbulence closure is assured by a Spalart-Allmaras turbulence model [1]. The ONERA finite-volume compressible CFD *elsA* [2] code has been used. The analysis of lift C_l, drag C_d and pressure C_p coefficients, in terms of mean and standard deviation has been carried out.

The flow about the airfoil has been studied at detached condition, AoA= 16° in the presence of eight simultaneous *uniform* uncertainties summarized in Table (4.5). Compared to the XFOIL study, the mean value of the Mach number has been kept the same. However, the uncertainty bounds has been raised to 6.25% (*i.e.* $\overline{M} \pm 0.0182$) as well as for the angle of attack (*i.e.* AoA±1°) for more pronounced effects.

Geometrical uncertainties have also been introduced. Since the geometric measurements data are scarce, an option that allows local control of the deformations consists in distributing *bumps* along the airfoil in order to reproduce at best available surface shape imperfections. One of the most exploited is the so-called Hicks-Henne [143] bump which use is common in literature for optimization pur-

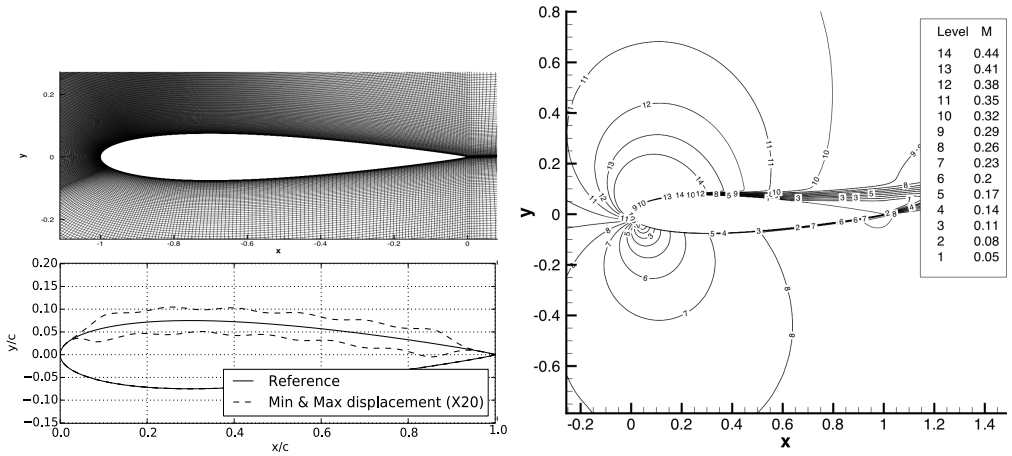
poses. Given the airfoil abscissa x, the height i-th bump is defined by the following formula

$$h_i(x) = A_i \left[\sin \left(\pi x^{\frac{\log 0.5}{\log t_1}} \right) \right]^{t_2}, \tag{4.24}$$

where A_i is the bump amplitude, t_1 controls the position maximum and t_2 the width. These small shape fluctuations may be applied to the airfoil camber, which guarantees a constant airfoil thickness, or directly to the upper and lower surface of the profile. Due to the very few available experimental assessments of the real airfoil surface in WT testing[9], the choice has been to keep *deterministic* bumps location along the profile but to allow for independent random bumps height. The amplitude A_i is then an uncertain parameter that should lay in the identified accuracy range. This approach implies that the typical wavelength of the geometrical fluctuations is kept constant while the amplitude of the bumps are random and in particular the relative elevation of a given bump compared to its neighbors. Please note that this shape modification may also induce in *side* alterations of airfoil characteristics, *i.e.* chord length and camber. In WT condition, similarly to the work of Piziali, these uncertainties lie in the range of $0.005\%\times$ chord (*i.e.* ± 0.05 mm). For this study, they have been multiplied by a factor of three in order to mimic RF condition. The six bumps have been centered on the suction side (which is the most sensitive to potential detachment condition) at $x_b \in [0.10, 0.25, 0.40, 0.55, 0.70, 0.85] \cdot x/c$ and $t_1 = 0.5$ and $t_2 = 5.0$ in Equation (4.24). The bump span is equal to $0.20 \cdot x/c$ projected on the airfoil surface. With this choice of bumps positioning, the chord length is not modified, namely $c = 1$. Figure 4.15 (a - bottom) shows the profile displacement envelop bounded by the local minimum and maximum surface position taken over all of the random events. This envelop has been magnified by a factor of 20 for the sake of visual clarity.

In the present study, it has been decided to keep the same flow solver setup and mesh refinement for each RANS simulation. The convergence is based on the residual of conservative variables. The chosen structured mesh available from Szydlowski *et al.* [144] has shown the best agreement with experiments [29] both in term of loads prediction and pressure distributions, in attached and detached cases. It is finer on the suction side, see Figure 4.15 (a - top). Moreover, the computational domain extends to 10 chords away from the airfoil and $y^+ = 1$ along the surface. The mesh is of C-type with 200 nodes on the pressure side, 750 nodes on the suction side, 48 in the wake and 105 in the direction normal to the wall, for a total of 109930 points. Please note that for each shape modification of the airfoil, the nominal mesh has been adapted accordingly by displacing nodes in the affected region of the shape modification. Sufficiently far from that region, the mesh is unchanged. This was an attempt to avoid unbalanced influence on results given by the generation of a new mesh. The numerical scheme is based on Roe

[9] In general these evaluations are expensive and are also covered by confidentiality.



(a) Mesh and bumps (b) M isolines, AoA= 16°

Figure 4.15: NACA0015 profile geometry/discretization and flow field. (a) mesh (top) and perturbed geometry (bottom) with displacement envelope (multiplication factor: 20). (b) Mach number isolines at AoA= 16°.

Table 4.5: Uncertain parametric ranges for the RANS study.

Uncertain param.	*iid* uniformly distributed $\mathcal{U}_{[\mu\pm\text{std}\%]}$
AoA	$16°$ $\pm$ 6.25%
M	0.291 $\pm$ 6.25%
$A_{1,..,6}$	0.0 $\pm$ 0.015% c

flux using a second-order MUSCL scheme (van Albada limiter) for the convective term, while first-order Roe flux for the turbulent flow. The viscous fluxes of the averaged and turbulent flows are discretized through a centred scheme with cell-centred gradients corrected at the cell faces. With this setup, the deficiency of *all* turbulence models to accurately predict the detachment phenomenon is well-known. Notwithstanding, more complex CFD simulations (*e.g.* RSM, DES, LES, DNS) nowadays remain too computationally demanding for such UQ study.

Figure 4.15 (b) shows the Mach number isolines for the nominal case, *i.e.* with no uncertainty on the input data. Due to the high computational cost of the RANS simulations, it was not possible to perform a reference estimation of the statistics as it was done in Section 4.4.3. A budget of around 1000 computations was the target. With this constraint, a level l_3 isotropic sparse grid with a maximum 849 CC nodes was constructed. Table (4.6) shows the estimated mean and standard deviation for C_l and C_d for all the stochastic approximation methods. Fast evaluation of coefficients of variation shows that C_d is about 20 times more

Table 4.6: RANS. C_l and C_d statistics. CC nodes. Lowest errors between SC l_2 and SC-Sobol with respect to SC l_3 are identified by a dagger† symbol.

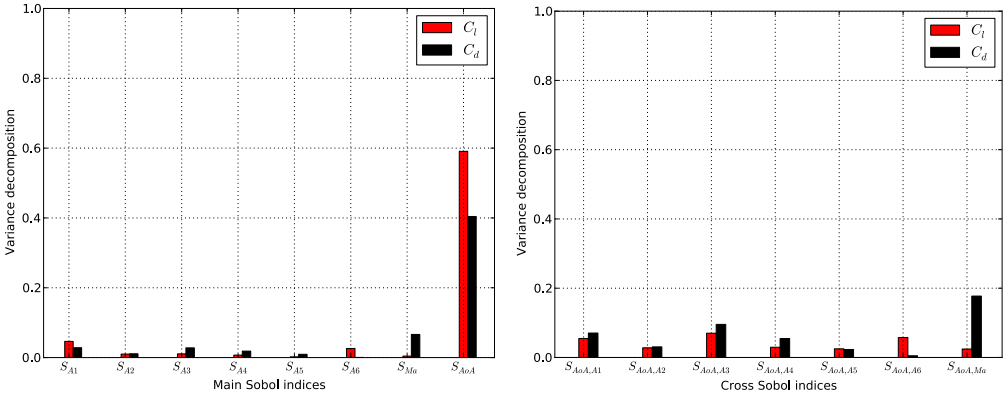
Method	Grid	$n_{\boldsymbol{l}}^{(\boldsymbol{N})}$	C_l		C_d	
			Mean	Std	Mean	Std
SC	$(CC)_{l_1}$	17	1.43012	0.0108767	0.0484011	0.0098380
SC	$(CC)_{l_2}$	145	1.42744	0.0156279	0.0421365	0.0061201
SC	$(CC)_{l_3}$	849	1.44612	0.0348340	0.0509102	0.0207006
SC-Sobol	(CC)	205-201	1.44284†	0.0314871†	0.0463688†	0.0171121†
gPC^{TD} (deg=2)	$(CC)_{l_3}$	849	1.44612	0.0417952	0.0509102	0.0237432
gPC^{TD} (deg=3)	$(CC)_{l_3}$	849	1.44612	0.0667229	0.0509102	0.0399846
gPC^{SD} (deg=4)	$(CC)_{l_3}$	849	1.44612	0.0903712	0.0509102	0.0430063
SgPC (deg=4)	$(CC)_{l_3}$	849	1.44612	0.0236920	0.0509102	0.0176705
Deterministic	-	-	1.44242	-	0.0452085	-
Experiment	-	-	1.0971	-	0.1262	-

sensitive than C_l in this case. For sufficiently fine grids, the obtained C_l and C_d mean values converge, while the standard deviation results do not agree. The best agreement between SC and gPC class of methods is obtained for SC l_3 and SgPC. Concerning the dimension-adaptive SC based on Sobol', it is interesting to observe that all results obtained from SC-Sobol get closer to SC l_3 results from below rather then SC l_2, despite the few more simulations needed. In fact the SC-Sobol std are 'in between' the SC l_3 and SgPC results. In particular, it gets very close to SgPC for the C_d std.

It is also beneficial to analyze the SI for both objectives. Figure 4.16 shows the histograms of the main (a) and of the non-negligible cross-dimensional (b) indices. The angle of attack strongly dominates, as in the XFOIL study, the variance estimation, and the Mach number has a smaller effect on C_d. The effects on the objectives of the geometrical uncertainties are small with the most influent bump being the first one (and also the last one for C_l). In Figure 4.17, the PDF (a) and CDF (b) of the aerodynamic coefficients are also reported. They exhibit non-*uniform* distributions and confirm the high sensitivity of C_d and its skewness towards high values.

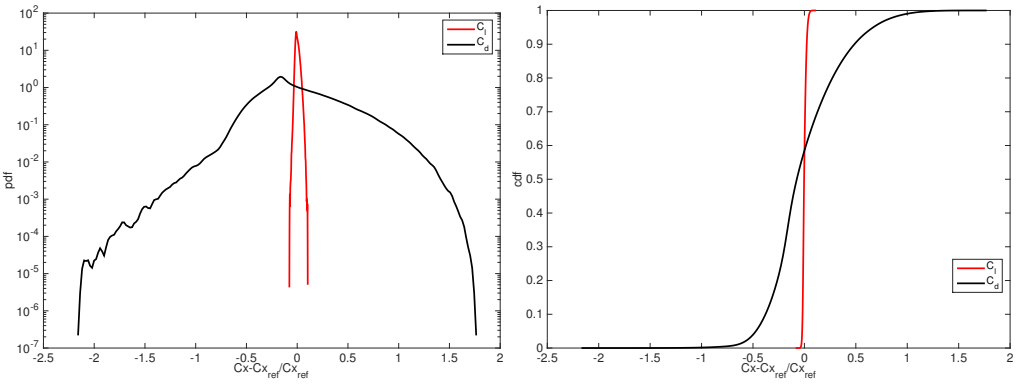
4.5 Some concluding remarks

This study was motivated first by recent literature introducing a way of constructing polynomial pseudospectral approximation based on the direct application of Smolyak algorithm. In this case, sparse sampling/approximations of linear oper-



(a) Main SI (b) Non-negligible cross SI

Figure 4.16: RANS-based first- and leading second-order SI obtained from the SC CC l_3 approximation. Bumps amplitude are denoted as $A_{1,..,6}$, where A_1 is placed downstream the leading edge and A_6 upstream the trailing edge.



(a) pdf (b) cdf

Figure 4.17: RANS-based PDF and CDF of aerodynamics coefficients Cx from SgPC approximation with 1M MC samples. The functions are centered (cf. Table (4.6)) and normalized for the sake of comparaison.

ators on tensor product spaces somewhat alleviate the curse of dimensionality, in particular if the number of random dimensions remain moderate (*i.e.* $\mathcal{O}(10-20)$) and if the quantity under investigation presents some levels of regularity. With this flexible formulation, the polynomial surrogate takes full advantage of sparse sampling and guarantees no internal aliasing errors. It was then natural to compare this approach with the well-known stochastic collocation approximation for *identical* (nested) computational grids. Both classes of approximations were then compared on regular isotropic sparse grids with different quadrature rules. During this study and inspired by the literature, a newly designed adaptive stochastic collocation with dimension-adaptivity driven by global sensitivity analysis of the quantity of interest was developed. The different stochastic approximations efficiency was assessed on nonlinear/anisotropic multi-variate test function as well as on airfoil aerodynamics simulations. The latter study addressed the probabilistic characterization of global aerodynamic coefficients derived from viscous subsonic steady flow about an airfoil in the presence of geometrical and different operational uncertainties. It was indeed envisioned that those global quantities resulting from aerodynamic forces balance about the geometry and depending on moderate number of random variables would be good candidate for the aforementioned numerical methods. Two studies with different deterministic models - aerodynamics simplified and RANS models - have been carried out in this framework. The idea was to quantify the effect of the balance between accuracy and versatility of the models and computational costs on the UQ analysis.

The results hints for some trends that are probably not extendable to all classes of stochastic problems. Nevertheless, it is fair to say that sparse pseudospectral and collocation approximations exhibit *close* level of performance, in terms of distributions and statistics, for isotropic sparse simulation ensembles. We have noticed that when the problem is very smooth with additive effects the pseudospectral approximation keeps an edge over the collocation approach. The right choice of the polynomial basis probably helps reducing external aliasing errors in this case. Concerning adaptive approaches, *i.e.* based on a substantially smaller number of samples and consequently at a smaller fraction of the computational cost, computational savings and accuracy gain of the proposed stochastic collocation driven by Sobol' indices are patent but remain problem-dependent. The approach is performant when there exists high-order cross interactions between the stochastic scales. Notwithstanding, it may be argued that the method can be used in a different framework. Indeed, an interesting point of view is the one where the adaptive collocation is utilized for the exploration of large (sparse) ensembles. This was the case for the aerodynamics studies (as often for engineering applications) where preexisting database collecting hundreds of realizations had been gathered, holding several levels of sparse isotropic refinements. Then the collocation with dimension-adaptivity driven by global sensitivity analysis, may be seen as a numerical tool that incrementally and anisotropically selects a subset of realizations

from the large isotropic ensemble structure, reaching the same (or even better) level of accuracy with much less information. This raises the question of the relevance of the systematic use of the largest available database at hand.
More efforts are needed but the present results provide support to the deployment of adaptive stochastic approximations for aerodynamics uncertainty propagation in a fully coupled and dynamic fashion. The development of greedy anisotropic refinements based on global sensitivity analysis for sparse pseudospectral approximation is the subject of ongoing work.

Conclusions & perspectives

Sensitivity analysis, verification and validation (V&V) as well as uncertainty quantification (UQ) are powerful tools in order to asses the validity and reliability of numerical simulations. The application of these methods in the Computational Fluid Dynamics (CFD) domain has risen significantly in the latter decades and the provided insight into CFD simulations differs from one approach to the other. The focus given in this PhD thesis research has been especially put on the numerical side of CFD code. Numerical solutions show strong dependency over the physical model, the discretization approach, the machine precision as well as the treatment of the model parametric input values. Special care has been given to the space discretization defined by the computational mesh and to the influence of the input values on the obtained solution. In order to improve the results, it was then essential to investigate the flow sensitivity with respect to the aforementioned factors.

Optimisation methods extensively use gradients and in turn gradient computation methods became an essential tool, nowadays available in many commercial CFD codes including *elsA*. The discrete direct differentiation and the discrete adjoint methods require the linearisation of the governing flow equations with respect to the conservative variables and to the metrics. In this PhD thesis research, important improvements in gradient computation has been implemented in the code. Namely, the full linearisation of the mean-flow viscous flux with respect to the conservative variables and the full linearisation of the Spalart-Allmaras turbulence model with respect to the metrics. The precision improvement has been investigated through relevant examples. The influence of the frozen turbulent viscosity hypothesis has been equally assessed and it turned out to be important for some applications. In aeronautics, the design process is highly demanding from a computational point of view, both for the numerous constraints given but also for the time needed for each simulation. Therefore, it is common to focus on some Quantity of Interest (QoI) of the flow (*e.g.* global aerodynamic functional) rather than on the entire flow domain. For this reason, goal-oriented methods have caught strong interest in the CFD community even for mesh adaptation. Goal-oriented mesh adaptation techniques where the adjoint vector of the QoI is used started to appear a couple of decades ago. An enhanced version of a mesh adaptation method

based on total derivative of aerodynamic functions (goal function J) with respect to mesh coordinates for one mesh level only has been given. The refinement indicator θ based on the $dJ/d\mathbf{X}$ has proven its efficiency in highlighting the crucial flow region in which a refinement is needed in a 2D RANS flow about an airfoil. In this thesis, only structured meshes have been considered which facilitate the handling of nodal addressing but in turn impose strong constraints on the construction of the mesh itself. The adapted drag coefficient-based mono-block mesh shows strong anisotropy according to the flow regime but local refinement unfortunately spreads all over the computational domain. The use of non-matching multi-block meshes relax this problem. The method has proven its flexibility also in the case where it is used in order to adapt the coarse farfield mesh blocks in the presence of a uniform refined boundary *frozen* mesh block.

CFD computations have been traditionally performed in a deterministic framework, *i.e.* the input data are at a fixed values, often the experimental nominal ones. Notwithstanding, reality exhibits fluctuations around the nominal values. Since about two decades ago, UQ methods have rapidly grown their influence in the V&V process by transforming CFD simulations into stochastic problems. The input values are then subject to a distribution law which should mimic the experimental recurrences or in any case fulfill the Jaynes' principle of maximum entropy. Given the high complexity of the *elsA* code, only non-intrusive approaches, which avoid major code modifications, have been considered. Monte-Carlo methods need generally an high number of computations which make their use quite rare in a CFD framework. Powerful methods have been developed in which the regularity of the solution through polynomial approximation is put to use. In particularly, generalized Polynomial Chaos (gPC) in its sparse pseudospectral form and Stochastic Collocation (SC) methods have been used. Both approaches have been constructed on Smolyak sparse grids which reduce the computational burden. The latter has been further reduced by proposing a new implementation of a dimension-adaptive algorithm driven by the Sobol' indices. The use of nested quadrature helps even more the computational savings while keeping the same quadrature precision with respect to Gauss-type formulae. The latter is proven to be true especially for non-polynomial functions. Multi-variate test functions as well as viscous aerodynamics applications have been presented. The performance enhancement provided by the gPC and the SC are similar for sparse isotropic grids. Nevertheless, when the problem is very smooth with additive effects the pseudospectral approximation keeps an edge over the collocation approach. The right choice of the polynomial basis probably helps reducing external aliasing errors in this case. Concerning anisotropy, the Sobol-based dimension-adaptive method has shown its efficiency in reducing the dimension of the stochastic problem but it remains problem dependent. It is reasonable to assume that for very non-smooth problem this approach is not sufficient and the reason relies on the sparse grid enrichment which follows the underlined quadrature rule.

The achievements reached during this PhD thesis point towards more reliable and robust CFD simulations. Indeed, more precise gradient computation, goal-oriented adapted mesh and quantitative insights of the effect of parametric input value uncertainties on the QoI are all beneficial. Improvements of this PhD thesis research are worth considering. On the gradient computation precision, minor approximations persist in the *elsA* code as well as some more important implementation linked with the linearisation of turbulence models other than the Spalart-Allmaras one. The extension to unstructured grid is undergoing and this will allow to capitalize the information from the refinement indicator θ without the strong mesh construction constraints imposed by the structured framework. On the UQ side, many improvements are conceivable. Stochastic approximations methods based on quadrature grid need an efficient and flexible way of grid enrichment which sometimes is limited by the underlined quadrature rule. These enhancements may lay on both new quadrature nested grids and on dimension-adaptive methods. The right choice of the polynomial basis for the gPC has shown to have an important influence on the QoI statistics precision, thus further studies in this direction are worth considering. Improvements in regression methods for high-dimensional problems are also interesting since the randomly selected computation points avoid the quasi-exponential growth of quadrature grids. Nevertheless, it is necessary to check that these points ensure some recovery properties (discrete orthogonality, mutual coherence et cet.) which for instance are only checked once as long as a quadrature grid-based method is used. It is finally important to underline one of the main perspective of this PhD thesis: a well-balanced goal-oriented deterministic/stochastic error control for CFD simulations. The integration of these approaches is essential in order to have reliable and validated results. The adaptation of both the mesh and the way random variables are approximated seems a research topic worth focusing on. The minimization of the discretization error (for instance for the computation of drag coefficient) over the variations range of the uncertain parameters (for instance Mach number, angle of attack and geometrical uncertainties) would enhance the quality of the UQ results. This would be possible by adapting the mesh for each parametric uncertainty. Another subject worth considering is the application of adjoint techniques in the approximation methods in order to identify the sources of uncertainty which impact the most the statistics of QoI. This would reduce the stochastic space dimensions and would allow UQ studies also for high-demanding computational simulations in aeronautics. Notwithstanding, the Author would like to recall once more the essential influence of the physical model which was not part of this PhD thesis research topics. The presented results are valuable in a fixed physical model approach. A complete control of the error on the CFD simulations cannot be detached from physical model considerations.

List of Figures

List of Tables

Bibliography

[1] P.R. Spalart and S.R. Allmaras. A one-equation turbulence model for aerodynamic flows. *AIAA Paper*, 1992. 92-0439.

[2] S. Heib L. Cambier and S. Plot. The onera elsa cfd software: input from research and feedback from industry. *Mechanics & Industry*, 14:159–174, 2013.

[3] P.L. Roe. Approximate riemann solvers, parameter vecotrs and difference schemes. *JCP*, 43:357–72, 1981.

[4] B. van Leer. Towards the ultimate conservative difference scheme. v-a second order sequel to godunov's method. *JCP*, 32:101–36, 1979.

[5] B. van Leer G.D. van Albada and W.W. Roberts. A comparative study of computational methods in cosmic gas dynamics. *Astronomy and Astrophysics*, 108:76–84, 1982.

[6] D. Destarac. *Far-Field / near-Field drag balance and applications of drag extraction in CFD.* VKI Lecture Series, Rhode Saint Genese, 2003.

[7] H. Toubin. Prediction and phenomenological breakdown of drag for unsteady flows. Technical report, ONERA, 2015.

[8] M. Drela. *XFOIL: an analysis and design system for low Reynolds number airfoils*, volume 54 of *Lecture Notes in Engineering*. Springer, 1989.

[9] D. Destarac J. Peter and D. Hue *et al.* Arf precotion. Technical Report RT 5/14969 DSNA, ONERA Confidentiel, 2010.

[10] P.J. Roache. *Verification and Validation in computational science and engineering*. Hermosa Publishers, Albuquerque, 1998.

[11] J.R. Roy. Review of code solution verification procedures for computational simulation. *Journal of Computational Physics*, 205:131–156, 2005.

[12] L. Richardson. The approximate arithmetical solution by finite difference problems involving differential equations, with an application to the stresses in a masonry dam. *Trans. R. Soc.*, 110:307–357, 1910.

[13] L. Richardson and J. Gaunt. The deferred approach to the limit. *Trans. R. Soc.*, 1927.

[14] L. Eça and M. Hoekstra. Proceedings. In *3rd workshop on CFD uncertainty analysis*, Lisbon, 2008.

[15] P.J. Roache. *Fundamentals of Verification and validation*. Hermosa Publishers, Albuquerque, 2009.

[16] AIAA standards. *Guide for verification and validation of computational fluid dynamics simulations. Technical Report AIAA G-077-1998*. American Institute of Aeronautics and Astronautics, 1998.

[17] W. L. Oberkampf and T. G. Trucano. Verification and validation in computational fluid dynamics. *Progress in Aerospace Sciences*, 38(3):209–272, 2002.

[18] T.G. Trucano L. P. Swiler T. Igusa W. L. Oberkampf and M. Pilch. Calibration, validation, and sensitivity analysis: What's what. *Reliability Engineering and System Safety*, 91(10-11):1331–1357, 2006.

[19] F. Jézéquel and J.-M. Chesneaux. Cadna : a library for estimating round-off error propagation. *Computer Physics Communications*, 178(12):933–955, 2008.

[20] M.J. Hemsch. Statistical analysis of cfd solutions from the drag prediction workshop. In *AIAA paper 2002-0842*, 2002.

[21] F. Stein and E. Stanewski. Wind tunnel flow quality and data accuracy requirements. Technical Report AR-184, AGARD, 1982.

[22] E.C. Carter and K.C. Pallister. Development of testing techniques in a large transonic wind tunnel to achieve a required drag in aerodynamics data accuracy and quality: requirements and capabilities in wind-tunnel testing. Technical Report CP-429, AGARD, 1988.

[23] D. Mavriplis. *Aerodynamics drag prediction using unstructured mesh solvers*. VKI Lecture Series, Rhode Saint Genese, 2003.

[24] D. Destarac R.H. Doe S. Leicher M.P. Carr J.W. Slooff, J.M. Bousquet and G. Wichmann. Garteur ag05: Accuracy study of transonic flow computations for 3d wings. Technical Report TP-030, GARTEUR, 1988.

[25] O.J. Boelens D. Destarac and O. Brode. Garteur ag39: Rans validation for transonic wing-body, phase 2. Technical Report TP-162, GARTEUR, 2007.

[26] M.J. Hemsch and J.H. Morrison. Statistical analysis of cfd solutions from the 2nd drag prediction workshop. In *AIAA paper 2004-0556*, 2004.

[27] E.N. Tinoco. *CFD uncertainty and validation for commercial aircraft applications*. NATO Symposium AVT 147, Athens, Greece, 2007.

[28] J. Kok. Resolving the dependence on freestream values for the $\kappa - \omega$ turbulence model. *AIAA Journal*, 38(7), 2000.

[29] R.A. Piziali. 2d and 3d oscillation wing aerodynamics for a range of angles including stall. Technical Report TM 4632, NASA, September 1994.

[30] P.J. Roache. Quantification of uncertainty in computational fluid dynamics. *Annu. Rev. Fluid. Mech.*, 1997.

[31] U. B. Mehta. Some aspects of uncertainty in computational fluid dynamics results. *Journal of Fluids Engineering*, 113(4):538–543, 1991.

[32] K.M. Hanson. A framework for assessing uncertainties in simulation predictions. *Physica D: Nonlinear Phenomena*, 113(1-4):179–188, 1999.

[33] J. Glimm and D.H. Sharp. Prediction and the quantification of uncertainty. *Physica D*, 133:152–170, 1999.

[34] F. Stern R. Wilson H.W. Coleman and E.G Paterson. Comprehensive approach to verification and validation of cfd simulations - part 1 : Methodologies and proceduresy. *Journal of Fluids Engineering*, 123:803–810, 2001.

[35] C. J. Roy and W.L. Oberkampf. A comprehensive framework for verification, validation, and uncertainty quantification in scientific computing. *Computer Methods in Applied Mechanics and Engineering*, 200(25 - 28):2131 – 2144, 2011.

[36] AA. VV. *National Research Council. Assessing the Reliability of Complex Models: Mathematical and Statistical Foundations of Verification, Validation, and Uncertainty Quantification.* Washington, DC: The National Academies Press, 2012.

[37] KAUST. Conceptual diagram for verification and validation and uncertainty quantification. `https://sri-uq.kaust.edu.sa/Pages/UQVV.aspx`, 2015.

[38] J.T. Oden and S. Prudhomme. Control of modeling error in calibration and validation processes for predictive stochastic models. *Int. J. Numer. Meth. Engng*, 87:262 – 272, 2011.

[39] W.N. Edeling P. Cinnella R.P. Dwight and H. Bijl. Bayesian estimates of parameter variability in the $k-\epsilon$ turbulence model. *Journal of Computational Physics*, 258:73 – 94, 2014.

[40] R.C. Almeida and J.T. Oden. Solution verification, goal-oriented adaptive methods for stochastic advection-diffusion problems. *Computer Methods in Applied Mechanics and Engineering*, 199:2472–2486, 2010.

[41] J.J. Alonso F. Palacios, K. Duraisamy and E. Zuazua. Robust grid adaptation for efficient uncertainty quantification. *AIAA Journal*, 50(7):1538–1546, 2012.

[42] J.A. Ekaterinarus G.R. Srinivasan and W.J. mcCroskey. Dynamic stall of an oscillatin wing. *AIAA Paper*, 93-3403, 1993.

[43] C. Laurent. *Etude d'ecoulements transitionnels et hors equilibre par des approches DNS et RANS.* PhD thesis, ONERA - ENSAM, Paris, France, 2012.

[44] G. Van der Plaats and R. Hicks. Numerical airfoil optimization using a reduced number of design coordinates. Technical Report TMX 73151, NASA, 1976.

[45] M. Méheut D. Destarac G. Carrier G. Anderson S. Nadarajah D. Poole J. Vassberg and D. Zingg. Gradient-based single and multi-points aerodynamic optimizations with the elsa software. In *53rd AIAA Aerospace Sciences Meeting - AIAA SciTech*, Kissimmee, Florida, 2015.

[46] A. Arntz M. Méheut and G. Carrier. Aerodynamic shape optimizations of a blended wing body configuration for several wing planforms. In *30th AIAA Applied Aerodynamics Conference*, New Orleans, 2012.

[47] A. Minelli I. Salah El Din G. Carrier A. Zerbinati and J.A. Désidéri. Cooperation and competition strategies in multi-objective shape optimization - application to low-boom/low-drag supersonic business jet. In *43rd AIAA Fluid Dynamics Conference and Exhibit*, San Diego, 2013.

[48] A. Dumont and G. Carrier. Multi-point aerodynamic optimization of a flexible transport aircraft wing using an aeroelastic adjoint method. In *6th European Conference on Computational Fluid Dynamics (ECFD VI)*, Barcelona, 2014.

[49] G. Vanderplaats. *Numerical Optimization Techniques for Engineering Design: with Applications.* McGraw Hill, 1984.

[50] Nielsen E and Park M. Using an adjoint approach to eliminate mesh sensitivities in computational design. *Amer. inst. Aeronaut.Astronaut.*, 44:948–953, 2005.

[51] D. C. Wilcox. Formulation of the k-omega turbulence model revisited. *AIAA Journal*, 46(11):2823–2838, 2008.

[52] Boman M. Johnson C, Rannacher R. Numerics and hydrodynamics theory: towards error control in cfd. *SIAM J Numer Anal*, 32:1058–1079, 1995.

[53] Rannacher R. Becker R. Weighted *a posteriori* error control in fe methods. In *Proceedings of ENUMATH-97, Heidelberg, World Scientific Publishing*, 1998.

[54] Rannacher R. Becker R. An optimal control approach to *a posteriori* error estimation in finite element methods. *Acta Numerica*, pages 1–102, 2001.

[55] Levenstam JM Giles M, Larson MG and S'uli E. Adaptative error control for finite element approximation of lift and drag coefficient in viscous flow. Technical Report NA-97/06, Comlab, Oxford University, 1997.

[56] Oden J. Prudhomme S. On goal oriented error estimation for elliptic problems: application to the control of pointwise error. *Computers Method in Applied Mechanics and Engineering*, pages 313–331, 1999.

[57] Barth TJ. Larson MG. *A posteriori* error estimation for discontinuous galerkin approximation of hyperbolic systems. Technical Report NAS 99-010, NASA, 1999.

[58] Patera AT. Machiels L, Peraire J. *A posteriori* finite element output bounds for the incompressible navier-stokes equations. application to a natural convection problem. *J Comput Phys*, 172:401–425, 2001.

[59] Houston P. Hartmann R. Adaptative discontinuous galerkin methods for the compressible euler equations. *J Comput Phys.*, 182(2):508–532, 2002.

[60] Hartmann R. Leicht T. Error estimation and anisotropic mesh refinement for 3d laminar aerodynamic flow simulations. *J Comput Phys.*, 229:7344–7360, 2010.

[61] Leicht T. Hartmann R, Held J. Adjoint-based error estimation and adaptative mesh refinement for the rans and (k-ω) turbulence model equations. *J Comput Phys.*, 230:4268–4284, 2011.

[62] Alauzet F. Loseille A, Dervieux A. Fully anisotropic goal-oriented mesh adaptation for 3d steady euler equations. *J Comput Phys.*, 229(2):2860–2897, 2010.

[63] Venditti DA and Darmofal DL. Adjoint error estimation and grid adaptation for functional outputs: application to quasi-one-dimensional flow. *J. Comput. Phys.*, 164:204–227, 2000.

[64] Venditti DA and Darmofal DL. Grid adaptation for functional outputs: application to two-dimensional inviscid flow. *Journal of Computational Physics*, 176:40–69, 2002.

[65] Venditti DA and Darmofal DL. Anisotropic grid adaptation for functional outputs: application to two-dimensional viscous flow. *J. Comput. Phys.*, 187:22–46, 2003.

[66] Nguyen-Dinh M Peter J and Trontin P. Goal oriented mesh adaptation using total derivative of aerodynamic functions with respect to mesh coordinates - with applications to euler flows. *Computers & Fluids*, 66:194–214, 2012.

[67] Nguyen-Dinh M Peter J Sauvage R Meaux M and Désidéri JA. Mesh quality assessment based on aerodynamic functional output total derivatives. *European Journal of Mechanics B/Fluids*, 45:51–57, 2014.

[68] Dwight RP. Goal-oriented mesh adaptation using a dissipation based error indicator. *Int J for Numer Methods Fluids*, 56(8):1193–2000, 2007.

[69] Dwight RP. Heuristic *a posteriori* estimation of error due to dissipation in finite volume schemes and application to mesh adaptation. *J Comput Phys*, 227:2845–2863, 2008.

[70] Turkel E. Jameson A, Schmidt W. Numerical solution of the euler equations by finite volume methods using runge-kutta time stepping schemes. *AIAA paper series*, 1259, 1981.

[71] G. Deng J. Wackers and M. Visonneau. On the feasibility of goal-oriented error estimation for ship hydrodynamics. In *VI International Conference on Adaptive Modeling and Simulation ADMOS 2013*, Lisbon, 2013.

[72] Darmofal DL Fidkowski KJ. Output-based error estimation and mesh adaptation in computational fluid dynamics: Overview and recent results. *AIAA J*, 49(4):673–694, 2011.

[73] Giles MB Pierce NA. Adjoint recovery of superconvergent functionals for pde approximations. *SIAM Review*, 42:247–264, 2000.

[74] Giles MB Pierce NA. Adjoint and defect error bounding and correction for functional estimates. *AIAA paper series*, 2003-3846, 2003.

[75] Le Pape MC Fillola G and Montagnac M. Numerical simulations around wing control surfaces. In *Proceedings of 24th ICAS*, 2004.

[76] McDonald MA Cook PH and Firmin MCP. Aerofoil rae 2822 - pressure distributions, boundary layer and wake measurements. Technical report, AGARD-AR-138, 1979.

[77] Li D and Hartmann R. Adjoint-based airfoil optimization with discretization error control. *Int. J. Numer. Meth. Fluids*, 77:1–17, 2015.

[78] Valero E Fraysse F and Ponsin J. Comparison of mesh adaptation using the adjoint methodology and truncation error estimates. *AIAA Journal*, 50(9), 2012.

[79] Leicht T Hartman R, Held JLT and Prill F. Error estimation and adaptive mesh refinement for aerodynamic flows. *Notes on Numerical Fluid Mechanics and Multidisciplinary Design*, 113:339–353, 2010.

[80] Vassberg JC and Jameson A. In pursuit of grid convergence, part i: two-dimensional euler solution. *AIAA paper series*, 4114, 2009.

[81] Todarello G. Goal oriented adaptation of unstructured meshes. application to finite volume methods. Technical report, ONERA - TUDelft. Master's thesis, 2014.

[82] P. Tattersall T.P. Evans and J.J. Doherty. Identification and quantification of uncertainty sources in aircraft-related cfd computations – an industrial perspective. In *Proceedings of RTO-AVT-147 symposium*, Athens, 2007.

[83] W.L. Oberkampf and F.G. Blottner. Issues in Computational Fluid Dynamics code verification and validation. *AIAA Journal,*, 36(5):687–695., 1998.

[84] D. Lucor, C. Enaux, H. Jourdren, and P. Sagaut. Stochastic design optimization: Application to reacting flows and detonation. *Comp. Meth. Appl. Mech. Eng.*, 196(49-52):5047–5062, 2007.

[85] D. Lucor J. Ko and P. Sagaut. Sensitivity of two-dimensional spatially developing mixing layers with respect to uncertain inflow conditions. *Phys. Fluids*, 2008.

[86] A. Birolleau, G. Poette, and D. Lucor. Bayesian inference acceleration using iterative polynomial chaos. *Communications In Computational Physics*, 16(1):1–34, 2014.

[87] M.C. Rochoux, S. Ricci, D. Lucor, B. Cuenot, and A. Trouvé. Reduced-cost ensemble kalman filter based on a polynomial chaos surrogate model for regional-scale wildfire spread. *Nat. Hazards Earth Syst. Sci.*, 14:2951–2973, 2014.

[88] A. Taylor M. Putko, P. Newman and L. Green. Approach for uncertainty propagation and robust design in CFD using sensitivity derivatives. *AIAA*, pages 2001–2528, 2001.

[89] G.J.A. Loeven and H. Bijl. Airfoil analysis with uncertain geometry using the probabilistic collocation method. In *AIAA 2008-2070*, 2008. 49th Structures, Structural Dynamics and Materials Conference, 7-10 April, Schaumburg.

[90] V. Schulz and C. Schillings. On the nature and treatment of uncertainties in aerodynamic design. *AIAA Journal*, 47(3):646–654, 2009.

[91] A. Litvinenko and H. G. Matthies. *Numerical Methods for Uncertainty Quantification and Bayesian update in Aerodynamics*. Springer, 2013. B. Eisfeld, H. Barnewitz, W. Fritz, F. Thiele, in book: Management and Minimisation of Uncertainties and Errors in Numerical Aerodynamics.

[92] P.S. Beran C.L. Pettit, M.R. Hajj. A stochastic approach for modeling incident gust effects on flow quantities. *Probabilistic Engineering Mechanics*, 25(1):153–162, 2010.

[93] F. Simon, P. Guillen, P. Sagaut, and D. Lucor. A gPC-based approach to uncertain transonic aerodynamics. *Computer Methods in Applied Mechanics and Engineering*, 199(17-20):1091–1099, 2010.

[94] R. Walters S. Hosder and R. Perez. A non-intrusive polynomial chaos method for uncertainty propagation in cfd simulations. *AIAA Paper*, 2006.

[95] J.-C. Chassaing, D. Lucor, and J. Trégon. Stochastic nonlinear aeroelastic analysis of a supersonic lifting surface using an adaptive spectral method. *JSV*, 331:394–411, 2012.

[96] V. E. Garzon and D. L. Darmofal. Impact of geometric variability on axial compressor performance. *Journal of Turbomachinery*, 125(4):692–703, 2003.

[97] M. Loeve. *Probability Theory*. Springer-Verlag, 1978. 4th edition, Vol. 2.

[98] J. Peter A. Resmini and D. Lucor. Sparse grids-based stochastic approximations with applications to aerodynamics sensitivity analysis. *Int. J. Numer. Meth. Engng.*, 2015.

[99] O.P. Le Maître and O.M. Knio. *Spectral Methods for Uncertainty Quantification - With Application to Computational Fluid Dynamics*. Scientific Computation. Springer, 2010.

[100] N. Wiener. The homogeneous chaos. *Amer. J. Math.*, 60:897–936, 1938.

[101] R. Ghanem and P. Spanos. *Stochastic Finite Elements: a spectral approach*. Springer-Verlag, New York, 1991.

[102] D. Xiu and G. Karniadakis. The wiener-askey polynomial chaos for stochastic differential equations. *SIAM J. Sci. Comput.*, 24:619–644, 2002.

[103] M.A. Tatang. *Direct incorporation of uncertainty in chemical and environmental engineering systems*. PhD thesis, Massachusetts Institute of Technology, Boston, USA, 1995.

[104] J.S. Hesthaven D. Xiu. High-order collocation methods for differential equations with random inputs. *SIAM J. Sci. Comput.*, 27(3):1118–1139, 2005.

[105] Griebel M. Garcke J. &. *Sparse Grids and Applications*, volume 88 of *Lecture Notes in Computational Science and Engineering*. Springer, 2013.

[106] J.A.S. Witteveen and G. Iaccarino. Simplex stochastic collocation with random sampling and extrapolation for nonhypercube probability spaces. *SIAM J. Sci. Comput.*, 34:814–838, 2012.

[107] K. Duraisamy and P. Chandrashekar. Goal-oriented uncertainty propagation using stochastic adjointss. *Computers & Fluids*, 66:10–20, 2012.

[108] I. M. Navon A. K. Alekseev and M. E. Zelentsov. The estimation of functional uncertainty using polynomial chaos and adjoint equations. *Int. J. Numer. Meth. Fluids*, 67:328–341, 2011.

[109] J.K. Romberg E.J. Candes and T. Tao. Stable signal recovery from incomplete and inaccurate measurements. commun. *Pure Appl. Math.*, 59(8):1207–1223, 2006.

[110] Tamellini L. Back J., Nobile F. and R. Tempone. Stochastic spectral galerkin and collocation methods for pdes with random coefficients: A numerical comparison. *Lecture Notes in Computational Science and Engineering*, 2010. Spectral and High Order Methods for Partial Differential Equations.

[111] L.P. Swiler M.S. Eldred. Towards goal-oriented stochastic design employing adaptive collocation methods. In *13th AIAA-ISSMO Multidisciplinary Analysis Optimization Conf.*, 2010. AIAA 2010-9125.

[112] P.R. Conrad and Marzouk Y.M. Adaptive smolyak pseudospectral approximations. *SIAM J. Scientific Computing*, 2013.

[113] Winokur J. Conrad P. Sraj I. Knio O. Srinivasan A. Thacker W.C. Marzouk Y. and Iskandarani M. A priori testing of sparse adaptive polynomial chaos expansions using an ocean general circulation model database. *Computational Geosciences*, 17(6):889–991, 2013.

[114] S. Tarantola A. Saltelli and K. P. S. Chan. A quantitative model-independent method for global sensitivity analysis of model output. *Technometrics*, 41:39–54, 1999.

[115] I.M. Sobol. Global sensitivity indices for nonlinear mathematical models and their monte carlo estimates. *Mathematics and Computers in Simulation*, 55:271–280, 2001.

[116] G. T. Buzzard and D. Xiu. Variance-based global sensitivity analysis via sparse-grid interpolation and cubature. *Communications in Computational Physics*, 9(3):542–567, 2010.

[117] M.S. Eldred G. Tang and L.P. Swiler. Global sensitivity analysis for stochastic collocation expansion. In *CSRI Summer Proceedings*, 2009.

[118] F. Nobile, R. Tempone, and C. G. Webster. An anisotropic sparse grid stochastic collocation method for partial differential equations with random input data. *SIAM Journal on Numerical Analysis*, 46(5):2411–2442, 2008.

[119] N. Zabaras X. Ma. An adaptive hierarchical sparse grid collocation algorithm for the solution of stochastic differential equations. *Journal of Computational Physics*, 228(8):3084–3113, 2009.

[120] T. Gerstner and M. Griebel. Dimension-adaptive tensor-product quadrature. *Computing*, 71(1):65–87, 2003.

[121] N. Zabaras X. Ma. An adaptive high-dimensional stochastic model representation technique for the solution of stochastic differential equations. *Journal of Computational Physics*, 229(10):3884–3915, 2010.

[122] M. Griebel and Holtz M. Dimension-wise integration of high-dimensional functions with applications to finance. *Journal of Complexity*, 26:455–489, 2010.

[123] I.M. Sobol'. Sensitivity estimates for nonlinear mathematical models. *MMCE*, 1(4):407–414, 1993.

[124] G. Geraci R. Abgrall, P.M. Congedo and G. Iaccarino. An adaptive multiresolution semi-intrusive scheme for uq in compressible fluid problems. *Int. J. Numer. Meth. Fluids*, 78:595–637, 2015.

[125] G. Fishman. *Monte Carlo: Concepts, Algorithms and Applications*. Springer-Verlag, 1996. New York.

[126] W. J. Morokoff and R. E. Caflisch. Quasi-monte carlo integration. *Journal of Computational Physics*, 122(2):218 – 230, 1995.

[127] S.A. Smolyak. Interpolation and quadrature formulas for tensor products of certain classes of functions. *Dokl. Akad. Nauk SSSR*, 4:240–243, 1963.

[128] T. Gerstner and M. Griebel. Numerical integration using sparse grids. *Numerical Algorithms*, 118(3-4):209–232, 1998.

[129] E. Novak V. Barthelmann and K. Ritter. High dimensional polynomial interpolation in sparse grids. *Advances in Computational Mathematics*, 12:273–288, 2000.

[130] R. Tempone F. Nobile and C.G. Webster. A sparse grid stochastic collocation method for partial differential equations with random input data. *SIAM J. Numer. Anal.*, 46(5):2309–2345, 2008.

[131] A.R. Curtis C.W. Clenshaw. A method for numerical integration on an automatic computer. *Numer. Math.*, 2:197–205, 1960.

[132] T. Patterson. The optimum addition of points to quadrature formulae. *Math. Comput.*, 22:847–856, 1968.

[133] L.N. Trefethen. Is gauss quadrature better than clenshaw-curtis? *SIAM Review*, 50(1):67–87, 2008.

[134] J. Ko. *Application des Polynômes de Chaos à la Simulation d'Ecoulements Convectifs Incertains.* PhD thesis, Université Pierre et Marie Curie, Paris, France, 2009.

[135] M. S. Eldred P. G. Constantine and E. T. Phipps. Sparse pseudospectral approximation method. *Computer Methods in Applied Mechanics and Engineering*, 229–232:1–12, 2012.

[136] J. Foo and G.E. Karniadakis. Multi-element probabilistic collocation method in high dimensions. *Journal of Computational Physics*, 229(5):1536–1557, 2010.

[137] P.M. Congedo K. Tang and R. Abgrall. Sensitivity analysis using anchored anova expansion and high order moments computation. *Int. J. Numer. Meth. Engng*, 102:1554–1584, 2015.

[138] T. Homma and Saltelli A. Importance measures in global sensitivity analysis of nonlinear models. *Reliability Engineering and System Safety*, 52(1):1–17, 1996.

[139] A. Genz. *Testing multidimensional integration routines.* Ed. B. Ford, J.C. Rault & F.Thomasset, 1984. In Tools, methods and languages for scientific and engineering computation.

[140] E. Jaynes. Information theory and statistical mechanics. *Phys. Rev.*, 106:620–630, 1957.

[141] E. Ferrer and X. Munduate. Cfd predictions of transition and distributed roughness over wind turbine airfoil. In *47th AIAA Aerospace Sciences Meeting*, Orlando, Florida, 2006.

[142] J.J.H.M. Sterenborg D.J. Boon, R.P. Dwight and H. Bijl. Reducing uncertainties in a wind-tunnel experiment using bayesian updating. In *14th AIAA Non-Deterministic Approaches Conference*, 2012.

[143] R. Hicks and P.A. Henne. Wing design by numerical optimization. *Journal of Aircraft*, 15:407–412, 1977.

[144] J. Szydlowski and M. Costes. Simulation fo flow around a naca0015 airfoil for static and dynamic stall configurations using rans and des. In *4th Decennial Specialist's conference on Aeromechanics*, San Francisco, California USA, January 2004.