Development of hp-adaptive techniques based on discontinuous Galerkin methods with application to aeronautical configurations

Francesca Basile



SORBONNE UNIVERSITÉ
École Doctorale de Sciences Mathématiques de Paris Centre

T H È S E
pour obtenir le grade de
DOCTEUR DE SORBONNE UNIVERSITÉ

Discipline : Mathématiques

Présentée et soutenue par
Francesca BASILE
le 23 septembre 2022

Development of *hp*-adaptive techniques based on discontinuous Galerkin methods with application to aeronautical configurations

---

Composition du jury :

Mme Paola CINNELLA Professeur · Sorbonne Université-UPMC	Président
Mme Héloïse BEAUGENDRE Professeur · Université de Bordeaux-Bordeaux INP	Rapporteur
M. Luca BONAVENTURA Professeur · Politecnico di Milano	Rapporteur
M. Michel VISONNEAU Directeur de Recherche · École Centrale de Nantes	Examinateur
M. Pascal FREY Professeur · Sorbonne Université-UPMC	Directeur de thèse
M. Jean-Baptiste CHAPELIER Ingénieur de Recherche · ONERA	Co-encadrant de thèse
M. Romain LARAUFIE Ingénieur · Airbus Commercial Aircraft	Invité

Abstract

In this thesis, novel *h*- and *hp*-adaptive strategies suited for the discontinuous Galerkin formulation of the compressible Navier-Stokes equations are established in the framework of the European collaborative project CODA (CFD ONERA DLR Airbus). The novel adaptive strategies target unstructured grids, considering a metric-based simplicial remeshing approach, with application to steady laminar (Navier-Stokes or NS), steady turbulent (Reynolds-Averaged Navier-Stokes or RANS) and unsteady turbulent scale-resolving configurations (Zonal Detached Eddy Simulation or ZDES). The definition of the size field (h) and the polynomial degree distribution (p) is driven by an *a posteriori* error estimator, combining the measure of the energy associated with the highest-order modes and the inter-element jumps. The choice of refining either in h or p is driven by a smoothness indicator based on the decay of the modal coefficients in each element. The performance of the *h*- and *hp*-adaptive algorithms is firstly assessed for 2D and 3D laminar flows, demonstrating a significant gain in accuracy of the adaptive simulations with respect to uniformly refined simulations. The adaptive *hp*-strategy is then extended to mixed-element meshes to account for the anisotropy of the boundary layers for the RANS and ZDES turbulence modeling approaches. Namely prisms located in the boundary layers can be *p*-enriched, while the tetrahedra discretizing the outer region of the flow can be both subject to *p*-enrichment and *h*-refinement. The adaptive *hp*-strategy is applied to a 3D turbulent jet issued from a nozzle, in both RANS and ZDES configuration (for which the adaptive algorithm is extended to unsteady flows), and validated through comparisons with the experiments and numerical results from the literature. The unsteady ZDES simulations are used to perform aeroacoustic analyses using the Ffowcs Williams-Hawkings (FW-H) method. Both fluid flow and acoustic results are shown to provide good agreement with the references while adapting the mesh, with a reduced number of degrees of freedom. The last part of this work focuses on the extension of the *h*-adaptation strategy to transonic flows considering complex geometries. For this purpose, the adaptation strategy is improved by allowing the projection of the adapted surface mesh on the CAD. This allows the adaptive strategy to take into account the exact representation of the geometry, which is of fundamental importance for complex industrial applications. A modification of the error estimator for the *h*-adaptive algorithm is proposed to account for shock capturing and transonic simulations. The performance of the developed algorithm is finally assessed for inviscid simulations of an isolated wing and a wing-body aircraft configuration.

Résumé

Dans cette thèse, de nouvelles stratégies adaptatives *h* et *hp* pour la formulation Galerkine discontinue des équations de Navier-Stokes compressibles sont établies dans le cadre du projet collaboratif européen CODA (CFD ONERA DLR Airbus). Les nouvelles stratégies adaptatives visent la simulation de configurations laminaires stationnaires (Navier-Stokes ou NS), turbulentes moyennées (Reynolds-Averaged Navier-Stokes ou RANS) et turbulentes avec résolution des grandes échelles (Zonal Detached Eddy Simulation ou ZDES). Une méthode de remaillage non structurée basée sur un champ de métrique a été développée en ce sens. La définition du champ de tailles caractéristiques des éléments (*h*) et de la distribution des degrés polynomiaux (*p*) est pilotée par un estimateur d'erreur *a posteriori*. Ce dernier combine la mesure de l'énergie associée aux modes d'ordre supérieur et les sauts entre les éléments. Le choix du raffinement en *h* ou en *p* est guidé par un indicateur de régularité basé sur la décroissance des modes polynomiaux dans chaque élément. La performance des algorithmes adaptatifs *h* et *hp* est premièrement évaluée pour des écoulements laminaires en 2D et 3D. Un gain de précision significatif des simulations adaptatives par rapport aux simulations uniformément raffinées est ainsi obtenu. La stratégie adaptative *hp* est ensuite étendue pour la prise en compte de couches limites, nécessitant des éléments anisotropes, afin de satisfaire les besoins de résolution des modélisations RANS et ZDES. En particulier, les prismes situés dans les couches limites sont enrichis en *p*, alors que les tétraèdres de la région extérieure sont à la fois soumis à un enrichissement en *p* et à un raffinement en *h*. Puis, l'algorithme adaptatif *hp* est étendu aux écoulements instationnaires et appliqué au jet turbulent 3D d'une tuyère, via une modélisation RANS et ZDES. Les simulations sont validées par des comparaisons avec les expériences et les résultats numériques de la littérature. Les simulations ZDES instationnaires permettent l'analyse aéroacoustique de l'écoulement en utilisant la méthode de Ffowcs Williams-Hawkings (FW-H). Les résultats de la dynamique de l'écoulement fluide et de son champ acoustique montrent un bon accord avec les références pour un nombre réduit de degrés de liberté. La dernière partie de ce travail se concentre sur l'extension de la stratégie d'adaptation *h* aux écoulements transsoniques considérant des géométries complexes. À cette fin, la méthode d'adaptation est améliorée en ajoutant au processus la projection du maillage de surface adapté sur la CAD. La représentation exacte de la géométrie est alors assurée, ce qui est d'une importance fondamentale pour les applications industrielles complexes. L'estimateur d'erreur de l'adaptation *h* est de plus enrichi pour une meilleure capture des chocs, caractéristiques des écoulements transsoniques. L'algorithme développé est finalement évalué pour la simulation non visqueuse d'une aile isolée et d'une configuration d'avion aile-fuselage.

Acknowledgments

I want to thank all the people that have been by my side during these three years, and have contributed to this work.

First of all, I am grateful to the members of the jury for their time and interest in my work, and to my thesis director Prof. Frey for supporting me and my supervisors during these three years.

My deepest gratitude goes to my ONERA and Airbus supervisors Jean-Baptiste and Romain. They have always motivated me and been open to discussion, and have always been available for me despite the distance depending on where I was, Paris or Toulouse. I have learned a lot from them, from their often different but complementary points of view, and I have appreciated working with them not only for their scientific and technical precious advice, but also for their human qualities. I also would like to thank Marta for supporting me during the beginning of my PhD journey.

I would like to thank all my colleagues at ONERA and Airbus for making me spend a nice time at work, from daily life conversations during the breaks, to sharing technical skills. In particular I would like to thank Gilles, Fabien, Maxime and Grégoire for the kind help they have provided for the set-up/analysis of the turbulent jet test case, which would have been a very difficult task without their help. I am also grateful to my hierarchy, especially to Renaud, Vincent and Pascal for supporting my work and appreciating my efforts.

Many thanks go to all the colleagues in the CODA community, in particular Tobias for developing crucial features in CODA in the beginning of this journey and for always being available for explanations. I have had the pleasure of regularly presenting my developments to the CODA developers and users on occasion of the several Airbus Program Increment Plannings during these three years. I am proud of having been part of such an ambitious project, in which I have decided to pursue my work for the next years.

This journey would have not been the same without all my friends.

I am extremely thankful to my longtime friends Isabella and Federico for making me feel at home in Paris.

Isabella, from my hometown, who happened to move to Paris in the same period as me, and became my flatmate for two years. I thank her for looking after me and supporting me during the stressful period of lock-down. But it is not only lock-down. It is also our sushi nights and evenings on the couch that helped me disconnect a bit from my PhD problems, but also have taught her something funny about my CFD world.

Federico, my old friend from the university period in Turin. He has been my study/party buddy since 2014, and having him here in Paris during these three years,

while sharing complaints and disappointments about PhD lives, but also joy and satisfaction for achieving our goals together, was truly cool.

All my friends from the Italian crew in Paris, Rocco, Marie (she counts as Italian), Chiara, Alberto, and the (few!) international friends as Javi and Matthias. I thank them for all the weekends and nights out spent together, and I thank those who were also my colleagues, for the funny coffee breaks at work. We have had a lot of fun and they have been a family for me during these three years full of experiences.

My friends in Toulouse, Minh, Malak, Andrea, Pratik, Luis, Romain, Ludovic. I am grateful for the time spent together while I was living there in the very first months and the last period of my thesis. They have warmly welcomed me in this new city of which I will keep very nice memories. Despite having lived only a few months in Toulouse, each of them has contributed to this work, and I hope our paths will cross again.

I want to thank my family, and in particular my mum, for always supporting me from afar since I left home for university. I want to thank her for always being there for me, for keeping me company on the phone when I was feeling down, for helping me with practical stuff I am terrible at, for giving me suggestions and advising me when decisions had to be taken, and for physically being there when achieving my successes.

A huge thanks goes to Fabio. His contribution to the success of my PhD has been multiple and falls into all the previous categories: he has been a friend, colleague, family, supervisor, reviewer. He has helped me deal with difficult moments thanks to his infinite patience, and has always encouraged me from the beginning to the end of my PhD experience during our everyday life together.

Thank you to everyone!

Communications

Publications in peer-reviewed journals

- Basile F, Chapelier J-B, de la Llave Plata M, Laraufie R, Frey P, *Unstructured h- and hp-adaptive strategies for discontinuous Galerkin methods based on a posteriori error estimation for compressible flows*, Computers & Fluids, Volume 233 (2022) - 105245, https://doi.org/10.1016/j.compfluid.2021.105245, 233:105245

- Balarac G, Basile F, Bénard P, Bordeu F, Chapelier J-B, Cirrottola L, Caumon G, Dapogny C, Frey P, Froehly A, et al. *Tetrahedral remeshing in the context of large-scale numerical simulation and high performance computing*, MathematicS In Action, Tome 11 (2022) no. 1, pp. 129-164, https://doi.org/10.5802/msia.22

- Basile F, Chapelier J-B, Laraufie R, Frey P, *Hybrid RANS/LES simulations and aeroacoustic analysis of jet flows using the discontinuous Galerkin method with hp-adaptive strategies*, Flow, Turbulence and Combustion (2022), https://doi.org/10.1007/s10494-022-00376-0

Conferences and proceedings

- Basile F, Chapelier J-B, de la Llave Plata M, Laraufie R, Frey P, *A high-order h-adaptive discontinuous Galerkin method for unstructured grids based on a posteriori error estimation*, AIAA SciTech 2021, Virtual

- Basile F, Chapelier J-B, Laraufie R, Frey P, *hp-adaptive hybrid RANS/LES simulations and aeroacoustic analysis of jet flows using a high-order discontinuous Galerkin method*, First DJINN Conference, 1-3 December 2021, VKI, Virtual

- Basile F, Chapelier J-B, Laraufie R, Frey P, *hp-adaptive hybrid RANS/LES simulations for unstructured meshes with the discontinuous Galerkin method*, AIAA SciTech 2022, San Diego

Secondments

- DLR Göttingen - "virtual" secondment 23/03/2020 - 03/04/2020. Handling of *p*-adaptive implementation in CODA and hanging-nodes *h*-adaptation. techniques

Airbus, ONERA and partners internal communications

- *Airbus - Product Increment (PI) Demostration. Four-monthly meeting*. Participation to seven meetings and presentation of project progress to four meetings.
- *ONERA - Journée des Doctorants (JDD)*. *Yearly meeting*. Presentation of project progress to three events.

Contents

1 Introduction **1**
- 1.1 Context and motivations . . . 1
- 1.2 Scientific and industrial objectives . . . 3
- 1.3 Overview of the thesis . . . 3
- 1.4 Fundings . . . 5

2 Flow equations and discretization **7**
- 2.1 The compressible Navier-Stokes equations . . . 7
- 2.2 Turbulence modeling . . . 8
  - 2.2.1 The Reynolds-Averaged Navier-Stokes equations . . . 8
  - 2.2.2 Scale-resolving simulations . . . 10
- 2.3 Numerical discretization of the fluid dynamics equations . . . 16
  - 2.3.1 The discontinuous Galerkin method . . . 16
  - 2.3.2 Time integration . . . 19
  - 2.3.3 Shock capturing . . . 19

3 State of the art of spatial resolution adaptive techniques **21**
- 3.1 *h*-adaptation techniques . . . 21
  - 3.1.1 Hanging nodes approach . . . 22
  - 3.1.2 Metric-based remeshing . . . 24
- 3.2 *p*-adaptation techniques . . . 32
- 3.3 *hp*-adaptation techniques . . . 33
- 3.4 *a posteriori* error estimators . . . 35
  - 3.4.1 High order-based error estimators . . . 37
  - 3.4.2 Jump error estimators . . . 38
  - 3.4.3 Combination of error indicators . . . 39

4 Validation on canonical test cases **41**
- 4.1 Study on the DG order of convergence on unstructured meshes - Isentropic CoVo vortex case . . . 41
- 4.2 Near-wall resolution requirements for DG/RANS - Turbulent flow over a flat plate at $Re = 5 \cdot 10^6$ . . . 44
- 4.3 Influence of the DES constant for DG simulations using unstructured grids - Taylor-Green vortex at $Re = 5000$ . . . 49
  - 4.3.1 DG computations at fixed number of dofs $\simeq 72^3$ . . . 50

4.3.2 Mesh sensitivity study . . . 53
4.4 Conclusion . . . 54

5 *h*- and *hp*-adaptation for steady computations 55
5.1 The steady *h*- and *hp*-adaptive algorithms . . . 55
5.1.1 *a posteriori* error estimator . . . 56
5.1.2 Smoothness indicator . . . 57
5.1.3 Metric prescription for *h*-adaptation . . . 58
5.1.4 Degree and metric prescription for *hp*-adaptation . . . 61
5.1.5 Choice of the target error . . . 65
5.1.6 Particular treatment for hybrid meshes . . . 66
5.2 Results on steady laminar configurations . . . 66
5.2.1 Laminar flow past a NACA0012 airfoil . . . 68
5.2.2 Laminar flow past a sphere at low Reynolds number . . . 72
5.2.3 Laminar flow past a delta wing at low Reynolds number and high angle of attack . . . 75
5.3 Results on a turbulent configuration: the isothermal subsonic turbulent jet flow issued from the PPRIME nozzle at $Re_D = 10^6$ . . . 78
5.3.1 Modified degree and metric prescription for the PPRIME nozzle . . . 79
5.3.2 Analysis of results . . . 80
5.4 Conclusions . . . 84

6 *hp*-adaptation for turbulent unsteady computations 87
6.1 The *h*/*hp*-adaptive algorithms for turbulent unsteady computations . . . 87
6.1.1 Smoothness indicator . . . 88
6.1.2 Strategy for unsteady flows . . . 89
6.2 Static adaptation for the CoVo test case . . . 89
6.3 DES of the subcritical flow past a sphere at $Re = 3700$ . . . 92
6.3.1 Flow field computational set-up . . . 92
6.3.2 Computational meshes and qualitative instantaneous flow field analysis . . . 93
6.3.3 Quantitative data analysis . . . 95
6.3.4 Discussion on the sensitivity of L_r . . . 100
6.3.5 Additional simulations and further investigation . . . 102
6.4 ZDES of the jet flow issued from the PPRIME nozzle at $Re_D = 10^6$ . . . 106
6.4.1 Motivation and state of the art of jet flow scale-resolving simulations 106
6.4.2 Flow field computational set-up . . . 109
6.4.3 Flow field results . . . 113
6.4.4 Far-field acoustics computational set-up . . . 125
6.4.5 Far-field acoustic results . . . 126
6.5 Conclusions . . . 129

7 Towards mesh adaptation for transonic flows over complex geometries: application to industrial configurations 131
7.1 Modifications to the adaptive algorithm . . . 132

7.1.1 Reprojection algorithm 132
7.1.2 Error estimator for transonic simulations 134
7.2 *h*-adaptation of the inviscid transonic flow past the ONERA M6 wing . . . 135
7.3 *h*-adaptation of the inviscid transonic flow past the CRM wing/body configuration 145
7.4 Conclusion 153

8 Conclusions and perspectives **155**
8.1 Conclusions 155
8.2 Perspectives 158

A Validation of the acoustic analysis **163**
A.1 Influence of the averaging procedure 163
A.2 Sensitivity study on the FW-H surfaces 164

Bibliography **170**

List of Figures

3.1 Isotropic (left) and anisotropic (middle and right) element splitting. . . . 23
3.2 Examples of unit elements with respect to a continuous element in 2D. Anisotropic (left) and isotropic (right) metric. 26
3.3 Example of mesh adaptation for the lid driven cavity. On the left the mesh obtained with isotropic elements splitting-based mesh adaptation by Offermans *et al.* [226] at $Re = 7500$ with the flow solver Nek5000. On the right, the mesh obtained with an isotropic metric-based adaptive strategy at $Re = 1000$, performed with the flow solver CODA and the h-adaptive strategy developed in this work. 30
3.4 Example of spatial resolution adaptation for the circular cylinder at $Re = 40$. On the left the mesh obtained with p-adaptation by Naddei *et al.* [219] with the flow solver AGHORA with variable $p = 2 - 6$. On the right, the mesh obtained with h-adaptation, performed with the flow solver CODA and the h-adaptive strategy developed in this work with fixed $p = 2$. 33
3.5 Example of spatial resolution adaptation for the sphere at $Re = 200$. On the left the polynomial distribution obtained with anisotropic p-adaptation by Rueda-Ramírez *et al.* [256] with $p = 2 - 7$. On the right, the mesh obtained with hp-adaptation with a variable $p = 1 - 3$, performed with the flow solver CODA and the adaptive strategies developed in this work. 36

4.1 CoVo at $M_\infty = 0.5$. Initialization of the streamwise velocity (top row) and of the pressure (bottom row). Left column: coarsest mesh (112 elements) with $p = 1$ reconstruction (336 dofs). Middle column: coarsest mesh (112 elements) with $p = 3$ reconstruction (1120 dofs). Right column: finest mesh (96936 elements) with $p = 3$ reconstruction (969360 dofs). . . 43
4.2 CoVo at $M_\infty = 0.5$. Error on entropy (left) and pressure (right) 44
4.3 DG/RANS of the flat plate at $Re = 5 \cdot 10^6$. The computational mesh G1 with stretching factor $\alpha = 1.05$ (left) and Mach contour on the whole domain of the flat plate (right). 45
4.4 DG/RANS of the flat plate at $Re = 5 \cdot 10^6$. Error on the value of the integrated friction drag coefficient C_{Dfric} along the flat plate. 47
4.5 DG/RANS of the flat plate at $Re = 5 \cdot 10^6$. Error on the value of the friction coefficient C_{fric} at $x = 0.1$ (left) and at $x = 0.97$ (right) of the flat plate. . . 47

4.6 DG/DES of the Taylor-Green vortex at $Re = 5000$. Computational mesh (left) for a $p = 3$ simulation with resolution 72^3, and respective iso-surface of $Q = 5(U_0/L)^2$ at $t_c = 10$, coloured by the streamwise velocity component (right). 50
4.7 DG/DES of the Taylor-Green vortex at $Re = 5000$. The 3D energy spectrum within the inertial sub-range at time $t_c = 10$ compared to the Kolmogorov scaling $E(k) \propto k^{-5/3}$. $p = 1, 2, 3, 4, 5$ simulations with $C_{DES} = 0.65$. 51
4.8 DG/DES of the Taylor-Green vortex at $Re = 5000$. Normalized enstrophy (left) and kinetic energy (middle) over the course of the simulation for a DGp3 discretization for different values of the C_{DES}. The 3D energy spectrum within the inertial sub-range at time $t_c = 10$ compared to the Kolmogorov scaling $E(k) \propto k^{-5/3}$ (right). 52
4.9 DG/DES of the Taylor-Green vortex at $Re = 5000$. Volume-averaged enstrophy. Different values of C_{DES} for three resolutions #dofs = $48^3, 96^3, 144^3$. 53
4.10 DG/DES of the Taylor-Green vortex at $Re = 5000$. The 3D energy spectrum within the inertial sub-range at time $t_c = 10$ compared to the Kolmogorov scaling $E(k) \propto k^{-5/3}$ for three resolutions #dofs = $48^3, 96^3, 144^3$ and different C_{DES} values. 54

5.1 Simplified central node with surrounding simplicial elements (marked for p-adaptation, for h-adaptation and not marked for adaptation). The nodal weighted average of the error estimator, the polynomial degree and the smoothness indicator for the prescription of the sizes is performed only on highlighted elements. 62
5.2 Laminar flow past a NACA0012 airfoil at $Re = 5000$. Zoom on the Mach number iso-contours of the cell-averaged solution (a) on the initial mesh (789 elements/4734 dofs), (b) on the 3rd adapted mesh (3446 elements/20676 dofs) and (c) on the 6th adapted mesh (8974 elements/53844 dofs), for the $h(p = 2)$ simulations. 69
5.3 Laminar flow past a NACA0012 airfoil at $Re = 5000$. Comparisons between the convergence history of the drag coefficient for hp-adapted meshes, h-adapted meshes and uniformly refined meshes in $p = 1$, $p = 2$, $p = 3$. C_D vs. number of dofs (left) and $|C_D - C_{D_{ref}}|/C_{D_{ref}}$ vs. number of dofs (right). 69
5.4 Laminar flow past a NACA0012 airfoil at $Re = 5000$. Comparisons between the convergence history of the lift coefficient for hp-adapted meshes, h-adapted meshes and uniformly refined meshes in $p = 1$, $p = 2$, $p = 3$. C_L vs. number of dofs (left) and $|C_L - C_{L_{ref}}|/C_{L_{ref}}$ vs. number of dofs (right). 70
5.5 Laminar flow past a NACA0012 airfoil at $Re = 5000$. Contours of the polynomial degree distribution (a) on the initial mesh (4734 dofs), (b) on the 3rd adapted mesh (18219 dofs) and (c) on the 6th adapted mesh (51834 dofs), for the hp simulations. 70
5.6 Laminar flow past a NACA0012 airfoil at $Re = 5000$. Zoom of figure 5.5. . 71

5.7 Laminar flow past a NACA0012 airfoil at $Re = 5000$. Study on the independence from the refinement factor on *hp*-adaptive simulations. 72
5.8 Laminar flow past a sphere at $Re = 200$. Close-up view of the error estimator iso-contours computed (a) on the initial mesh (6651 elements/133020 dofs), (b) on the 3rd adapted mesh (37920 elements/758400 dofs) and (c) on the 6th adapted mesh (154171 elements/3083420 dofs), for the $p = 3$ simulations. Slices of $y = 0$. 73
5.9 Laminar flow past a sphere at $Re = 200$. Comparisons between the convergence history of the drag coefficient for *hp*-adapted meshes, *h*-adapted meshes and uniformly refined meshes in $p = 1$, $p = 2$, $p = 3$. C_D vs. number of dofs (left) and $|C_D - C_{D_{ref}}|/C_{D_{ref}}$ vs. number of dofs (right). . 74
5.10 Laminar flow past a delta wing at $Re = 4000$. Slices of $y = 0$ and $x/L = 1.77$ planes of the reconstructed high-order solution on a finer post-processing mesh. Initial mesh (15100 dofs) at left, and 8th *hp*-adapted mesh (1543678 dofs) at right. Mach contour for volume slices, friction coefficient for surfaces, isosurface $M = 0.2$ transparence. 76
5.11 Laminar flow past a delta wing at $Re = 4000$. Comparisons between the convergence history of the drag coefficient for *hp*-adapted meshes, *h*-adapted meshes and uniformly refined meshes in $p = 1$, $p = 2$, $p = 3$. C_D vs. number of dofs (left) and $|C_D - C_{D_{ref}}|/C_{D_{ref}}$ vs. number of dofs (right). 76
5.12 Laminar flow past a delta wing at $Re = 4000$. Comparisons between the convergence history of the lift coefficient for *hp*-adapted meshes, *h*-adapted meshes and uniformly refined meshes in $p = 1$, $p = 2$, $p = 3$. C_L vs. number of dofs (left) and $|C_L - C_{L_{ref}}|/C_{L_{ref}}$ vs. number of dofs (right). 77
5.13 Laminar flow past a delta wing at $Re = 4000$. Contours of the polynomial degree distribution (a) on the initial mesh (15100 dofs), (b) on the 3rd adapted mesh (110836 dofs) and (c) on the 6th adapted mesh (563580 dofs), for the *hp* simulations. 77
5.14 Laminar flow past a delta wing at $Re = 4000$. Close-up view of figure 5.13 78
5.15 RANS PPRIME nozzle configuration. Slices of the initial mesh, entire domain (left) and zoom on the nozzle (right) 79
5.16 RANS PPRIME nozzle configuration. Contour of the cell-averaged Mach number for the initial mesh (left) and the mesh after 5 adaptation steps with $f_r = 1.5$ (10M dofs). 81
5.17 RANS PPRIME nozzle configuration. Velocity profiles on the centerline (left) and on the lipline (right). 81
5.18 RANS PPRIME nozzle configuration. Radial velocity profiles at $x/D = 1, 5, 10, 15$ (from left to right). 82
5.19 RANS PPRIME nozzle configuration. Zoom on *hp*-adapted meshes with around 10M dofs obtained with $f_r = 1.5$ (left) and $f_r = 2$ (right) 83
5.20 RANS PPRIME nozzle configuration. Velocity profiles on the centerline (left) and on the lipline (right) using $f_r = 1.5$ and $f_r = 2$ at around 10M dofs. 83

5.21 RANS PPRIME nozzle configuration. Radial velocity profiles at $x/D = 1, 5, 10, 15$ (from left to right) using $f_r = 1.5$ and $f_r = 2$ at around 10M dofs. 84
5.22 RANS PPRIME nozzle configuration. Comparisons between the convergence history of the norm of the integrated error along the profiles $r/D = 0, 1/2$ and $x/D = 1, 5, 10, 15$ (from top left to bottom right) for two different refinement factors $f_r = 1.5, 2$. $error$ vs. number of dofs. . . . 85

6.1 CoVo at $M_\infty = 0.5$. Velocity (left) and pressure (right) on the 11^{th} h-adaptation (4375 elements /43750 dofs) with a spatial $p = 3$ discretization. 90
6.2 CoVo at $M_\infty = 0.5$. Polynomial degree distribution (left), velocity (middle) and pressure (right) on the 15^{th} hp-adaptation (45609 dofs). 91
6.3 CoVo at $M_\infty = 0.5$. Convergence history of the entropy (left) and pressure (right) errors. 92
6.4 hpDG/DES of the subcritical flow past a sphere at $Re = 3700$. Zoom of the error estimator distribution around the body and near-wake. G0 on the left, hpG2 in the middle, hpG4 on the right. 94
6.5 hpDG/DES of the subcritical flow past a sphere at $Re = 3700$. Zoom of the polynomial degree distribution around the body and near-wake. G0 to hpG5, from top left to bottom right. 95
6.6 hpDG/DES of the subcritical flow past a sphere at $Re = 3700$. Iso-surface of $Q = 0.2(U_\infty/D)^2$, colored by the streamwise velocity component (right). 96
6.7 hpDG/DES of the subcritical flow past a sphere at $Re = 3700$. PSD of the radial velocity signal recorded at locations P1$(x/D = 2.4, r/D = 0.6)$ in the left panel and P2$(x/D = 3, r/D = 0.6)$ in the right panel 96
6.8 hpDG/DES of the subcritical flow past a sphere at $Re = 3700$. Mean streamwise velocity profiles on different locations in the wake $x/D = 0.2, 1.6, 3, 5, 10$ (from left to right). Comparison between hp-adaptive and experimental/numerical results. 98
6.9 hpDG/DES of the subcritical flow past a sphere at $Re = 3700$. Root-mean square of the streamwise velocity fluctuations profiles in the wake $x/D = 1.6, 3, 5, 10$ (from left to right). Comparison between hp-adaptive and experimental/numerical results. 99
6.10 hpDG/DES of the subcritical flow past a sphere at $Re = 3700$. Mean (left) and rms (right) streamwise velocity profiles on the wake axis $r/D = 0$. Comparison between hp-adaptive and experimental/numerical results. . . 100
6.11 hpDG/DES of the subcritical flow past a sphere at $Re = 3700$. Mean pressure coefficient $\overline{C_p}$ and mean normalized skin-friction coefficient $\overline{\tau_w}/(\rho_\infty U_\infty)Re^0.5$ on the surface of the sphere. Comparison between hp-adaptive and experimental/numerical results. 101
6.12 hpDG/DES of the subcritical flow past a sphere at $Re = 3700$. Zoom on the polynomial map distribution around the sphere for cases D, E, F, G. . . 104
6.13 hpDG/DES of the subcritical flow past a sphere at $Re = 3700$. Mean streamwise velocity profiles on the wake axis $r/D = 0$. Comparison between different hp-DG simulations. 104

6.14 *hp*DG/ZDES simulations of the PPRIME nozzle. Slice of the computational domain for the *hp*G1 discretization with polynomial degrees (left). High-order solution interpolated on a fine post-processing mesh of the pressure fluctuations in grey scale, with a sketch of the interface of the sponge layer zone. 110
6.15 *hp*DG/ZDES simulations of the PPRIME nozzle. Forcing of the interior of the nozzle to RANS mode. 112
6.16 *hp*DG/ZDES simulations of the PPRIME nozzle. Zoom of the polynomial degree map in the nozzle exit region and the early jet plume for the three *hp*-adapted meshes, *hp*G1 on the left, *hp*G2 in the middle, *hp*G3 on the right. 113
6.17 *hp*DG/ZDES simulations of the PPRIME nozzle. Zoom of the distribution of the error estimator in the nozzle exit region and the early jet plume for the *hp*-adapted meshes, *hp*G1 on the left and *hp*G2 on the right. 114
6.18 *hp*DG/ZDES simulations of the PPRIME nozzle. Iso-contours of the temperature T/T_∞ in colored-scale and the pressure fluctuations $(\mathrm{p}-\mathrm{p}_\infty)/\mathrm{p}_\infty$ in greyscale. Zoom of the *hp*G1 mesh (top), the *hp*G2 mesh (center) and the *hp*G3 mesh (bottom). 115
6.19 *hp*DG/ZDES simulations of the PPRIME nozzle. Iso-contours of vorticity at $x/D = 1, 2, 3$ (from left to right) for *hp*G1 (top), *hp*G2 (middle), *hp*G3 (bottom). 116
6.20 *hp*DG/ZDES simulations of the PPRIME nozzle. Boundary layer velocity profile at $x/D = 0.04$. Comparison between *hp*-adaptive and reference results. 117
6.21 *hp*DG/ZDES simulations of the PPRIME nozzle. Mean streamwise velocity profiles on the jet axis $r/D = 0$ (left) and on the lipline $r/D = 0.5$ (right). Comparison between *hp*-adaptive and reference results. 119
6.22 *hp*DG/ZDES simulations of the PPRIME nozzle. Root-mean square of the streamwise velocity fluctuation profiles on the jet axis $r/D = 0$ (left) and on the lipline $r/D = 0.5$ (right). Comparison between *hp*-adaptive and reference results. 121
6.23 *hp*DG/ZDES simulations of the PPRIME nozzle. Mean streamwise velocity profiles at different locations in the jet plume $x/D = 1, 5, 10, 15$ (from left to right). Comparison between *hp*-adaptive and reference results. 122
6.24 *hp*DG/ZDES simulations of the PPRIME nozzle. Root-mean square of the streamwise velocity fluctuation profiles at different locations in the jet plume $x/D = 1, 5, 10, 15$ (from left to right). Comparison between *hp*-adaptive and reference results. 123
6.25 *hp*DG/ZDES simulations of the PPRIME nozzle. PSD of the pressure signal recorded at locations P1($x/D = 2.62, r/D = 1.07$), P2($x/D = 3.42, r/D = 1.18$), P3($x/D = 4.47, r/D = 1.33$), P4($x/D = 5.85, r/D = 1.52$) from top left to bottom right. 124
6.26 *hp*DG/ZDES simulations of the PPRIME nozzle. Schematics of the acoustical set-up. Location of the FW-H surfaces. 126

6.27 *hp*DG/ZDES simulations of the PPRIME nozzle. Representation of the cylindrical (left) and polar array (right) with the chosen FW-H surface S1-D3. 127
6.28 *hp*DG/ZDES simulations of the PPRIME nozzle. PSD of the pressure on the cylindrical microphone array of radius r = 14.3D (left) and on the polar microphone array at 50D from the nozzle exit (right) for different angles of observation. 128
6.29 *hp*DG/ZDES simulations of the PPRIME nozzle. OASPL on the cylindrical microphone array of radius r = 14.3D (left) and on the polar microphone array at 50D from the nozzle exit (right) for different angles of observation.129

7.1 Adaptation algorithm for h-adaptation modified to handle complex geometries. 132
7.2 Inviscid flow past the ONERA M6 wing at $M_\infty = 0.84$. Zoom on the tip of the wing. The surface obtained from MMG on the left and the reprojected surface on the right. 133
7.3 Inviscid flow past the CRM wing/body configuration at $M_\infty = 0.85$. Zoom on the wind-shield. Left: intial mesh, center: mesh after 5 $h(p = 1)$ adaptations, right: mesh after 10 $h(p = 1)$ adaptations. 134
7.4 Inviscid flow past the ONERA M6 wing at $M_\infty = 0.84$. Pressure coefficient contour on the initial mesh with a $p = 1$ discretization. 136
7.5 Inviscid flow past the ONERA M6 wing at $M_\infty = 0.84$. Pressure coefficient contour on the 10^{th} $h(p = 1)$ adapted meshes with ϵ_{mom} in the left panel and $\epsilon_{\text{mom+pres}}$ in the right panel. 136
7.6 Inviscid flow past the ONERA M6 wing at $M_\infty = 0.84$. Convergence history of the integral quantities C_D and C_L vs. number of dofs for $h(p = 1)$ simulations for the two error estimators, compared to $p = 1$ uniformly refined simulations and the reference simulation by Balan *et al.* [32]. . . 137
7.7 Inviscid flow past the ONERA M6 wing at $M_\infty = 0.84$. Pressure coefficient profiles at $y/b = 0.2, 0.44, 0.65, 0.8, 0.9, 0.99$ for the two error estimators on the 10^{th} $h(p = 1)$ adaptation, compared to the reference simulation by Choi *et al.* [89] and the experiment. 138
7.8 Inviscid flow past the ONERA M6 wing at $M_\infty = 0.84$. Pressure coefficient contour on the 10^{th} $h(p = 1)$ adapted meshes obtained with different h_{min} values. $h_{min} = 5 \cdot 10^{-3}$ in the left panel, $h_{min} = 3 \cdot 10^{-3}$ in the middle panel (same as the right panel in figure 7.7) and $h_{min} = 1 \cdot 10^{-3}$ in the right panel. 139
7.9 Inviscid flow past the ONERA M6 wing at $M_\infty = 0.84$. Convergence history of the integral quantities C_D and C_L vs. number of dofs for $h(p = 1)$ simulations using different h_{min} values. 140
7.10 Inviscid flow past the ONERA M6 wing at $M_\infty = 0.84$. Pressure coefficient profiles at $y/b = 0.2, 0.44, 0.65, 0.8, 0.9, 0.99$ using different h_{min} values on the 10^{th} $h(p = 1)$ adaptation. 141
7.11 Inviscid flow past the ONERA M6 wing at $M_\infty = 0.84$. Pressure coefficient contour on the 7^{th} $h(p = 2)$ adapted meshes with ϵ_{mom} in the left panel and $\epsilon_{\text{mom+pres}}$ in the right panel. 142

7.12 Inviscid flow past the ONERA M6 wing at $M_\infty = 0.84$. Convergence history of the integral quantities C_D and C_L vs. number of dofs for $h(p = 2)$ simulations using different error estimators, compared to $h(p = 1)$ simulations and the reference simulation by Balan *et al.* [32]. 143
7.13 Inviscid flow past the ONERA M6 wing at $M_\infty = 0.84$. Pressure coefficient profiles at $y/b = 0.2, 0.44, 0.65, 0.8, 0.9, 0.99$ for two error estimators on the 7^{th} $h(p = 2)$ adaptation, compared to the reference simulation by Choi *et al.* [89], the experiment, and the 10^{th} $h(p = 1)$ adapted mesh. 144
7.14 Inviscid flow past the CRM wing/body configuration at $M_\infty = 0.85$. Pressure coefficient contour on the initial mesh with a $p = 1$ discretization. Detail of the wing in the right panel. 145
7.15 Inviscid flow past the CRM wing/body configuration at $M_\infty = 0.85$. Pressure coefficient contour on the 7^{th} (left) and the 10^{th} (right) $h(p = 1)$ adapted mesh. 146
7.16 Inviscid flow past the CRM wing/body configuration at $M_\infty = 0.85$. Pressure coefficient contour on the 7^{th} (left) and the 10^{th} (right) $h(p = 1)$ adapted mesh. Detail of the wing from figure 7.15. 146
7.17 Inviscid flow past the CRM wing/body configuration at $M_\infty = 0.85$. Initial mesh, 2^{nd}, 5^{th} and 9^{th} $h(p = 1)$ adapted meshes from top to bottom panels. Volume slice in the $y-z$ plane at $x = 2$ ($x \simeq 7c$) in left panels with pressure coefficient C_p contour, vorticity $||\omega||$ contour on planes $1.88 < x < 1.97$ in right panels. 148
7.18 Inviscid flow past the CRM wing/body configuration at $M_\infty = 0.85$. Artificial viscosity μ_{AV} contour. Initial mesh and $h(p = 1)$ adaptations 1,2,3,4,5,6,7 from top left to bottom right. 149
7.19 Inviscid flow past the CRM wing/body configuration at $M_\infty = 0.85$. Convergence history of the integral quantities C_D and C_L vs. number of dofs for $h(p = 1)$ and $h(p = 2)$ adapted simulations compared to $p = 1$ uniformly refined simulations. 150
7.20 Inviscid flow past the CRM wing/body configuration at $M_\infty = 0.85$. Pressure coefficient profiles at $y/b = 0.125, 0.37, 0.5024, 0.7268, 0.8456, 0.97$ for $h(p = 1)$ and $h(p = 2)$ adaptations. 151
7.21 Inviscid flow past the CRM wing/body configuration at $M_\infty = 0.85$. Smoothness contour on the 8^{th} $h(p = 2)$ adapted mesh with $\sigma_{thr} = 0.5$. Volume slices at $y/b = 0.15$ and $y/b = 0.78$. 152
7.22 Inviscid flow past the CRM wing/body configuration at $M_\infty = 0.85$. Pressure coefficient contour on the 7^{th} $h(p = 2)$ adapted mesh using $\sigma_{thr} = 0.5$ and $\sigma_{thr} = 1$. Detail of the wing. 153

A.1 hpDG/ZDES simulations of the PPRIME nozzle. PSD of the pressure on the cylindrical microphone array of radius r = 14.3D (left) and on the polar microphone array at 50D from the nozzle exit (right) for different angles of observation. Different parameters for the Welch PSD estimation. 166

A.2 hpDG/ZDES simulations of the PPRIME nozzle. PSD of the pressure on the cylindrical microphone array of radius r = 14.3D (left) and on the polar microphone array at 50D from the nozzle exit (right) for different angles of observation. Different FW-H surfaces of revolution. 167
A.3 hpDG/ZDES simulations of the PPRIME nozzle. PSD of the pressure on the cylindrical microphone array of radius r = 14.3D (left) and on the polar microphone array at 50D from the nozzle exit (right) for different angles of observation. Different closing FW-H surfaces. 168
A.4 hpDG/ZDES simulations of the PPRIME nozzle. OASPL on the cylindrical microphone array of radius r = 14.3D (left) and on the polar microphone array at 50D from the nozzle exit (right) for different angles of observation using different number of overlapping blocks for the Welch PSD estimation and different parameters for the Welch PSD estimation. 169
A.5 hpDG/ZDES simulations of the PPRIME nozzle. OASPL on the cylindrical microphone array of radius r = 14.3D (left) and on the polar microphone array at 50D from the nozzle exit (right) for different angles of observation using different FW-H surfaces of revolution. 169
A.6 hpDG/ZDES simulations of the PPRIME nozzle. OASPL on the cylindrical microphone array of radius r = 14.3D (left) and on the polar microphone array at 50D from the nozzle exit (right) for different angles of observation using different closing FW-H surfaces. 169

Chapter 1

Introduction

1.1 Context and motivations

More efficient, silent and cleaner aircraft designs require new concepts and increasingly complex physical modeling, integrating the multi-physical interactions between aerodynamics, acoustics, propulsion, structures and flight control systems. These new models must be carefully validated, so that Computational Fluid Dynamics (CFD) can be integrated as a primary tool in the certification process, reducing our reliance on costly experimental testing and enabling high-fidelity simulations that allow for a fine study of physical phenomena around complex geometries. However, high-fidelity simulations often require a very high computational effort due to the use of a very large number of spatial degrees of freedom (dofs) to capture accurately features stemming from complex physics.

In this context, adaptive strategies are of great interest for optimizing the cost and accuracy of the simulations. In practice, adaptive methods lead to the concentration of the dofs in regions of interest of the flow, guided by an error estimate [109, 182, 301]. Pure mesh adaptation, also referred to as h-adaptation, is performed to improve the resolution of the flow features employing both classical second order numerical methods [298, 20] and high-order methods [180, 76], by modifying the existing mesh according to an adaptation criterion. In the regions lacking resolution, h-adaptation aims at decreasing the size h of the elements, while increasing it in overrefined regions. When dealing with high-order methods, the increased resolution necessary to accurately solve the flow features can be achieved by locally increasing the order of the numerical method. In particular, for spectral element methods, increasing the order of accuracy corresponds to increasing the degree p of the polynomials which approximate the solution. This kind of spatial resolution adaptation is called p-adaptation [219, 14, 300].

In particular, the research community has begun to focus on hp-adaptive methods using piecewise polynomial approximations, in which, one not only locally adapts the size h of the mesh, but also the degree of the polynomials p, within the element. These adaptive methods offer better flexibility and efficiency than adaptive methods implementing only h-refinement or p-enrichment [182, 75, 34].

Among novel high-order numerical methods, discontinuous Galerkin (DG) schemes are a special class of spectral element methods which provide a natural framework for

hp-adaptation. In recent years, DG methods [167, 111] have become increasingly popular for the solution of nonlinear convection dominated flow problems [91]. DG methods are high-order finite element discretizations based on the variational formulation of the governing equations, and combine features from Finite Volume (FV) and Finite Element (FE) methods. DG methods lack of continuity constraints at the element boundaries, allowing for straightforward element-wise changes of the numerical features of the method. A number of aspects make DG methods especially interesting, such as their high-order of accuracy achieved on arbitrary unstructured meshes, accurate description of curved boundaries and suitability to parallel computing thanks to a compact stencil. Moreover, efficient, local, simple and low-computational cost *a posteriori* jump and spectral error indicators [202, 236, 139, 245] can be exploited for mesh or polynomial adaptation.

In the context of high-fidelity CFD simulations, the use of unstructured or hybrid meshes is often preferred over structured meshes for their higher flexibility in the meshing process and in the number of degrees of freedom needed to simulate a given configuration. DG is particularly suited for unstructured computational meshes, since the increased order derives from a local approximation of the differential operator. The computational stencil of DG methods thus remains compact independently of the order, and the quality of the approximation presents a lower sensitivity to the mesh regularity as compared to the high-order FV schemes [46]. However, for DG methods, the computational complexity and memory requirements increase with the polynomial order of the numerical approximation. As a consequence, for DG methods to become an industrially affordable alternative to FV solvers, it is essential to develop *hp*-adaptive techniques capable to reduce their computational cost.

We note that the low dissipation and dispersion properties of DG schemes make them the perfect candidates for the simulation of turbulent unsteady phenomena and their acoustic qualification, which require very accurate numerical schemes. Scale-resolving simulations are powerful tools that can help understanding turbulence and the noise generation mechanisms in aircraft, especially in the design of novel engine nozzles concepts, which are known to be the main sources of noise of an aircraft at take-off. Traditionally, scale-resolving simulations of aerodynamic problems have been performed by means of high-order FD and FV schemes, presenting, however, two important limitations. Both lose some of the parallel efficiency when employing an extended stencil, and their nominal order of accuracy near physical boundaries.

Concerning the physical modeling of turbulent flows, despite their extensive and well-assessed use for industrial configurations, Reynolds-averaged Navier-Stokes (RANS) simulations fail in predicting accurately non-equilibrium turbulent flows in which the turbulent large scales strongly affect the mean flow quantities [79], as for example with strongly detached and transitional flows [260]. RANS involves the solution of fluid dynamics equations considering time averaged quantities, such that the effects of the scales of turbulence are fully modeled. Large Eddy Simulation (LES) is designed to characterize explicitly the spatial and temporal dynamics of turbulent scales and the unsteady features encountered for example in transitional flows, gas turbine combustors, nozzles. Nonetheless, LES yields prohibitive costs to capture the wall-bounded turbulent dynamics at high Reynolds numbers.

Hybrid RANS/LES models have been proposed to combine the most attractive properties of the two approaches: the attached boundary layer is predicted by RANS models, while the unsteady 3D large eddies are captured by LES, decreasing the computational cost of separated turbulent flows with respect to LES. This allows for a significant reduction of the number of degrees of freedom, which wall-resolved LES would need to capture the smaller structures developed in the boundary layer. Hybrid RANS/LES models in general, together with Wall Modeled LES, that can be often seen as a subcategory of hybrid RANS/LES models, represent at the present time the only means to simulate relevant industrial configurations with sufficient accuracy and affordable computational time.

1.2 Scientific and industrial objectives

The main goal of the present PhD thesis is to provide a contribution towards the industrialization of DG methods, by optimizing their computational cost while maintaining their high accuracy for Airbus applications.

In particular this work is aimed at developing adaptive techniques in the context of the new unstructured flow solver CODA [183], developed in partnership by Airbus, ONERA and DLR, and targeting research and industrial aerodynamic problems of interest. This new CFD platform is designed for both innovative numerical methods and physical models, and efficient parallel and heterogeneous architectures, applying modern software techniques to a wide range of multidisciplinary applications. The object-oriented CODA framework permits the integration of advanced inter-operable CFD components, including different flow solvers (Finite Volume, modal and nodal discontinuous Galerkin schemes) applied to the Euler, Navier–Stokes, RANS, LES and hybrid RANS/LES equations.

The main research axis consists in the development of efficient adaptive strategies (both h and hp) for DG, suitable for industrial configurations, employing unstructured simplicial meshes. In particular, the MMG library [6, 97], supporting both 2D and 3D remeshing, has been chosen in this work as the remeshing tool. The adaptive algorithm is developed in the FlowSimulator environment [203] used in Airbus. This modular framework couples different applications of the CFD and the Computational Structural Mechanics (CSM) workflow on High Performance Computing (HPC) platforms, and the different modules of the FlowSimulator platform share a common data structure and software architecture.

The main target applications of this research are scale-resolving aeronautical industrial configurations. The final part of this thesis concerns steady aircraft configurations, in order to provide the basis for future DG/RANS adaptive simulations on transonic test cases, representative of Airbus aerodynamic design activities.

1.3 Overview of the thesis

In this thesis we propose novel metric based h- and hp-adaptive strategies tailored for DG schemes and tetrahedral elements. The introduction of an hp-decision strategy aims

at improving the efficiency and behavior of pure h-adaptive algorithms. An efficient *a posteriori* error estimator, which employs the solution itself to derive estimates of the discretization error, is used to control the element size for h-adaptation, and both the element size and the polynomial degree for hp-adaptation. Concerning the hp-adaptation algorithm, we propose that the choice on whether to h-refine or p-enrich an element be driven by the decay rate of the modal coefficients of the DG approximation, which characterizes the smoothness of the solution in the element. An isotropic mesh-refinement strategy is employed in this work. In the case of RANS and hybrid RANS/LES simulations this is fit for adapting the resolution in free-shear regions, while the anisotropic flow regions (e.g. boundary layers) can be meshed with fixed anisotropic prismatic elements whose resolution is modulated through p-adaptation.

Several numerical steady and unsteady test cases are studied to assess the efficiency of the developed adaptive algorithms. Adaptive simulations are compared to simulations performed on uniformly refined meshes and/or reference simulations from the literature, in order to measure the gain provided by the adaptive algorithms. The developed algorithm is finally applied to the hybrid RANS/LES scale resolving simulations of a subsonic jet at $Re = 10^6$ aimed at aeroacoustic analyses. The adaptation tool and the error estimator employed to drive the adaptation process are then adjusted for inviscid transonic simulations of wing/aircraft configurations.

The manuscript is organized as follows.

Chapter 2 presents the details of the discretization of the Navier–Stokes (and Euler) equations using the discontinuous Galerkin method. Different turbulence modeling approaches are presented, with a particular focus on the RANS equations employing the Spalart-Allmaras turbulence model, and the Detached Eddy Simulation (DES)/Zonal Detached Eddy Simulation (ZDES).

In Chapter 3 a bibliographic review of the different spatial resolution adaptive techniques is conducted. In particular different h-adaptation, p-adaptation and hp-adaptation techniques are presented, highlighting their main advantages and drawbacks, and their field of application. Some relevant *a posteriori* error estimators are finally presented.

In Chapter 4, the order of convergence of the employed DG discretization is first assessed for unsteady inviscid computations, thanks to the test case of the transport of an isentropic vortex. The properties of the DES/ZDES in the context of DG are then studied, focusing on the influence of the y^+ value by performing RANS simulations of the flow over a flat plate at Reynolds number equal to $5 \cdot 10^6$, and of the DES constant for DG simulations on a turbulent Taylor-Green Vortex (TGV) at Reynolds number equal to 5000.

Chapter 5 provides a thorough description of the h- and hp-adaptive algorithms developed in this work. The error estimator, the smoothness indicator and the degree/metric prescription are here detailed. The assessment of the implemented h- and hp-adaptation algorithms is then performed on three subsonic laminar configurations, namely the 2D flow past a NACA0012 airfoil, the 3D flow past a sphere and the 3D flow past a delta wing. Comparisons are made with respect to uniformly refined meshes, to highlight the gain provided by the adaptive strategies. The hp-adaptive algorithm is then applied to RANS simulations of a turbulent configuration of an isothermal subsonic jet

issued from the PPRIME nozzle [63], and validated by comparisons with experiments and a reference RANS simulation.

Chapter 6 extends the *h/hp*-adaptation algorithm to unsteady turbulent flows. The modifications of the adaptive procedure for such applications are here described. First, the test case of the transport of an isentropic vortex allows us to validate the unsteady static *h/hp*-adaptive implemented strategies, by comparison with uniformly refined meshes. Then the *hp*-adaptive strategy is tested on a hybrid RANS/LES (DES) configuration, the subcritical flow past a sphere. The results are validated and discussed by comparisons with other numerical references in the literature. The *hp*-adaptive strategy is finally applied to the unsteady configuration of the PPRIME nozzle, with a hybrid RANS/LES approach, namely the ZDES mode 1. Here the initial *hp*-mesh for starting the hybrid RANS/LES adaptation process is obtained via a series of low computational cost RANS *hp*-adaptations from Chapter 5. Results on the flow field and on the far-field acoustics, performed using a Ffowcs Williams-Hawkings approach, are presented and compared to the experiment and the numerical references from the literature.

In Chapter 7, the *h*-adaptive strategy is improved by introducing the possibility to reproject the geometrical surfaces to its CAD. This important feature allows for treating more complex geometries typical of the Airbus industrial context. Thanks to this development, the complete surface mesh adaptation can be performed without degrading the geometry representation over the course of the different adaptation steps. Moreover, the availability of an artificial viscosity technique for DG methods in the CODA flow solver allows us to apply the *h*-adaptive algorithm to inviscid transonic configurations of the ONERA M6 wing and the body-wing configuration of the CRM. The error estimator is here adapted to the different physics of the flow presenting a strong physical discontinuity as a shock.

In Chapter 8 concluding remarks and directions for future research are discussed.

1.4 Fundings

This CIFRE PhD thesis has been funded by ANRT and Airbus under the CIFRE contract n. 2019/0568, and it has been carried out at ONERA in Châtillon (Hauts de Seine, FR) and Airbus Commercial Aircraft in Toulouse (Haute Garonne, FR). The computations in this thesis have been performed using HPC resources from GENCI-CINES (Grant 2020-A0082A11470 and Grant 2021-A0102A11470) and internal ONERA and Airbus HPC resources.

Chapter 2

Flow equations and discretization

This chapter describes the equations for solving compressible fluid flow problems in Section 2.1, as well as the discontinuous Galerkin discretization implemented in the flow solver CODA used in the present work in Section 2.3.

2.1 The compressible Navier-Stokes equations

The motion of a compressible fluid in a three-dimensional domain $\Omega \in \mathbb{R}^3$ is described by the compressible Navier-Stokes (NS) equations, which take the form

$$\begin{aligned} \partial_t \mathbf{u} + \nabla \cdot (\mathbf{F}_c(\mathbf{u}) - \mathbf{F}_v(\mathbf{u}, \nabla\mathbf{u})) = 0, \quad \forall \mathbf{x} \in \Omega,\ t > 0 \\ \mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0(\mathbf{x}), \quad \forall \mathbf{x} \in \Omega \end{aligned} , \tag{2.1}$$

with appropriate boundary conditions prescribed on $\partial\Omega$. The vector $\mathbf{u}$ represents the conservative variables, with ρ being the density, $\mathbf{v}$ being the velocity vector and E the specific total energy. The convective and diffusive fluxes $\mathbf{F}_c(\mathbf{u})$ and $\mathbf{F}_v(\mathbf{u}, \nabla\mathbf{u})$ are defined, respectively, as:

$$\mathbf{u} = \begin{bmatrix} \rho \\ \rho\mathbf{v}^{\mathrm{T}} \\ \rho E \end{bmatrix}, \ \mathbf{F}_c(\mathbf{u}) = \begin{bmatrix} \rho\mathbf{v}^{\mathrm{T}} \\ \rho\mathbf{v}\mathbf{v}^{\mathrm{T}} + \mathrm{p}\mathbf{I} \\ \rho E\mathbf{v}^{\mathrm{T}} + \mathrm{p}\mathbf{v}^{\mathrm{T}} \end{bmatrix} \text{ and } \mathbf{F}_v(\mathbf{u}, \nabla\mathbf{u}) = \begin{bmatrix} 0 \\ \boldsymbol{\tau} \\ \mathbf{v}^{\mathrm{T}}\boldsymbol{\tau} - \mathbf{q}^{\mathrm{T}} \end{bmatrix} \tag{2.2}$$

with:

$$\boldsymbol{\tau} = 2\mu\mathbf{S}^D = \mu(T)\left(\nabla\mathbf{v} + (\nabla\mathbf{v})^{\mathrm{T}} - \frac{2}{3}(\nabla \cdot \mathbf{v})\mathbf{I}\right), \quad \mathbf{q} = -\lambda\nabla T \ , \tag{2.3}$$

$$\mathrm{p} = \rho RT = (\gamma - 1)(\rho E - \frac{1}{2}\rho\mathbf{v} \cdot \mathbf{v}) \ . \tag{2.4}$$

where p is the static pressure, $\gamma = \dfrac{C_p}{C_v}$ is the ratio of specific heats, $\mu(T)$ is the dynamic viscosity, $\mathbf{S}^D$ is the deviatoric component of the strain-rate tensor $\mathbf{S} = \dfrac{1}{2}(\nabla\mathbf{v} + (\nabla\mathbf{v})^{\mathrm{T}})$, T is the temperature, R is the specific gas constant, and $\lambda = \mu\dfrac{C_p}{Pr}$ is the thermal conductivity, with Pr the Prandtl number. In this work, a value of $Pr = 0.72$ is used. The perfect gas

law is used to close the system of equations, and the dependence of the viscosity on the temperature is expressed using the Sutherland's law.

The behavior of inviscid flows is governed by the Euler equations. They correspond to the Navier–Stokes equations with zero viscosity $\mu = 0$ and zero thermal conductivity $\lambda = 0$. They then take the form of equations (2.1) with $\mathbf{F}_v = [0]$.

2.2 Turbulence modeling

Different physical modeling approaches can be used for the simulation of turbulent flows. These range from Direct Numerical Simulation (DNS) (the most precise and computationally expensive), which solves all turbulent scales, to the Reynolds-Averaged Navier-Stokes (RANS) approach, which describes the average motion by modeling all the turbulent scales (the least precise and cheapest from a computational point of view). Intermediate steps in a scale of precision and subsequent computational cost are the Large Eddy Simulation (LES) and the hybrid RANS/LES methods. In this section we present the different approaches mentioned here, and detail the specific formulations used in this work.

2.2.1 The Reynolds-Averaged Navier-Stokes equations

Reynolds developed a statistical description of turbulence by averaging in time the Navier-Stokes equations.

The conservative variables $\mathbf{u}$ can be decomposed as follows:

$$\mathbf{u}(\mathbf{x},t) = \overline{\mathbf{u}(\mathbf{x},t)} + \mathbf{u}'(\mathbf{x},t) \text{ with } \overline{\mathbf{u}(\mathbf{x},t)} = \frac{1}{\Delta t}\int_{-\frac{\Delta t}{2}}^{\frac{\Delta t}{2}} \mathbf{u}(\mathbf{x},t+\tau)d\tau \ , \tag{2.5}$$

where $\overline{\mathbf{u}(\mathbf{x},t)}$ is the time averaged value and $\mathbf{u}'(\mathbf{x},t)$ the instantaneous fluctuations. This description corresponds to the classical RANS approach for statistically steady flows, where the averaged quantity is constant in time $\overline{\mathbf{u}(\mathbf{x})}$. If the averaged quantity is time-dependent, we refer to an approach called URANS with $\overline{\mathbf{u}(\mathbf{x},t)}$. The time interval Δt must be chosen to be large enough with respect to the time scale of the turbulence, but at the same time sufficiently small with respect to the time scale of the global unsteady phenomena.

Averaging the convective term, the fluctuating velocity gives rise to an extra term containing the correlation between the components of the fluctuating velocities, the Reynolds stress tensor $-\overline{\rho\mathbf{v}' \otimes \mathbf{v}'}$.

In RANS simulations, the Reynolds stresses are unknown and need to be modeled in order to close the system of equations. The system can be closed with the Boussinesq hypothesis, expressing the Reynolds stress tensor using averaged quantities

$$\overline{\boldsymbol{\tau}_\mathbf{t}} = 2\mu_t\overline{\mathbf{S}^D} - \frac{2}{3}\overline{\rho}\overline{k}\mathbf{I}, \quad \overline{\mathbf{q}_\mathbf{t}} = -\frac{\mu_t}{Pr_t}C_p\nabla\overline{T} \ , \tag{2.6}$$

where $Pr_t = 0.9$ is the turbulent Prandtl number, μ_t is the turbulent dynamic viscosity and $\overline{k}$ is the turbulent kinetic energy.

In order to correctly capture velocity gradients near the wall, in RANS simulations the first element at the wall should be small enough to ensure $y^+ \sim 1$ in the wall-normal direction. Usually RANS meshes are highly anisotropic in the boundary layers, since very high x^+ and z^+ can be generally employed in the two directions tangential to the wall.

The modeling of the turbulent viscosity and of the turbulent heat fluxes can be realised with different RANS models. Among the most popular, we can find the $k-\epsilon$ model by Launder and Spalding [178], the one equation model by Spalart & Allmaras [276], the $k-\omega$ by Wilkox [305] and the $k-\omega$ by Menter [206]. In this work we will use the approach by Spalart & Allmaras, described more in detail in the next paragraph.

The Spalart-Allmaras turbulence model

The mean flow equations are coupled with the one-equation turbulence model of Spalart-Allmaras (SA) [276] to solve a modeled transport equation for the kinematic eddy turbulent viscosity. In particular we describe the more recent formulation, known as the "negative" version of the SA model, which keeps into account the modification proposed by the original authors in [23], including the modifications for compressibility effects. The Reynolds-Averaged Navier-Stokes equations with the SA turbulence model can be written in conservative form by adding a source term to equation (2.1) as:

$$\begin{aligned} \partial_t \mathbf{u} + \nabla \cdot (\mathbf{F}_c(\mathbf{u}) - \mathbf{F}_v(\mathbf{u}, \nabla\mathbf{u})) = \mathbf{S}(\mathbf{u}, \nabla\mathbf{u}), \quad \forall \mathbf{x} \in \Omega, t > 0 \\ \mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0(\mathbf{x}), \quad \forall \mathbf{x} \in \Omega \,, \end{aligned} \tag{2.7}$$

where we assume that $\mathbf{u}$ is the vector of time-averaged conservative variables over a given time interval and t is a pseudo time. Here and in the remainder of this section we have dropped the symbol $\overline{(\cdot)}$ for clarity. The turbulent conservative variable $\rho\tilde{\nu}$ is added to the vector of conservative variables and the convective and diffusive fluxes $\mathbf{F}_c(\rho\tilde{\nu})$, $\mathbf{F}_v(\rho\tilde{\nu}, \nabla(\rho\tilde{\nu}))$ are defined as:

$$\mathbf{F}_c(\rho\tilde{\nu}) = \rho\tilde{\nu}\mathbf{v}^{\mathrm{T}}, \quad \mathbf{F}_v(\rho\tilde{\nu}, \nabla(\rho\tilde{\nu})) = \frac{1}{\sigma}(\mu + f_{n1}\rho\tilde{\nu})\boldsymbol{\nabla}\tilde{\nu}^{\mathrm{T}} \,. \tag{2.8}$$

In the diffusive fluxes, the turbulent stress tensor $\boldsymbol{\tau_t}$ and the turbulent heat fluxes $\mathbf{q_t}$ in equation (2.6) are added respectively to $\boldsymbol{\tau}$ and $\mathbf{q}$ defined in equation (2.3). In particular the term depending from k is ignored for this type of one-equation model, because k is not readily available. The turbulent dynamic viscosity μ_t is defined as:

$$\mu_t = \begin{cases} \rho\tilde{\nu} f_{v1}(\chi) & \text{for } \tilde{\nu} \geq 0 \\ 0 & \text{for } \tilde{\nu} < 0 \end{cases}, \quad f_{v1}(\chi) = \frac{\chi^3}{\chi^3 + c_{v1}^3}, \quad \chi = \frac{\rho\tilde{\nu}}{\mu} \tag{2.9}$$

The source terms act on the conservation equation only for the turbulent variable $\rho\tilde{\nu}$ and read:

$$\mathbf{S}(\rho\tilde{\nu}, \nabla(\rho\tilde{\nu})) = -\rho(P - D) - \frac{c_{b2}}{\sigma}\rho\nabla\tilde{\nu}\cdot\nabla\tilde{\nu} + \frac{1}{\sigma}(\nu + f_{n1}\tilde{\nu})\,\nabla\rho\cdot\nabla\tilde{\nu} \tag{2.10}$$

where the production and destruction terms, P and D, are defined by:

$$P = \begin{cases} c_{b1}(1-f_{t2})\tilde{\omega}\tilde{\nu} & \text{for } \tilde{\nu} \geq 0 \\ c_{b1}(1-c_{t3})\omega\tilde{\nu} & \text{for } \tilde{\nu} < 0 \end{cases} \qquad D = \begin{cases} \left(c_{w1}f_w - \dfrac{c_{b1}}{\kappa^2}f_{t2}\right)\left(\dfrac{\tilde{\nu}}{\mathrm{d}_w}\right)^2 & \text{for } \tilde{\nu} \geq 0 \\ -c_{w1}\left(\dfrac{\tilde{\nu}}{\mathrm{d}_w}\right)^2 & \text{for } \tilde{\nu} < 0 \end{cases} \tag{2.11}$$

and

$$f_{n1} = \begin{cases} 1 & \text{for } \tilde{\nu} \geq 0 \\ \dfrac{c_{n1}+\chi^3}{c_{n1}-\chi^3} & \text{for } \tilde{\nu} < 0 \end{cases}, \qquad f_{t2} = c_{t3}\exp(-c_{t4}\chi^2), \qquad f_w = g\left(\frac{1+c_{w3}^6}{g^6+c_{w3}^6}\right)^{\frac{1}{6}} \tag{2.12}$$

with

$$g = r + c_{w2}(r^6 - r), \qquad r = \min\left(r_{lim}, \frac{\tilde{\nu}}{\omega\kappa^2\mathrm{d}_w^2}\right), \tag{2.13}$$

where d_w is the distance to the nearest wall and ω the vorticity magnitude.

The modified vorticity magnitude $\tilde{\omega}$ is given by

$$\tilde{\omega} = \begin{cases} \omega + \overline{\omega} & \text{for } \overline{\omega} > -c_{v2}\omega \\ \omega + \dfrac{\omega(c_{v2}^2\omega + c_{v3}\overline{\omega})}{(c_{v3}-2c_{v2})\omega - \overline{\omega}} & \text{for } \overline{\omega} < -c_{v2}\omega \end{cases}, \tag{2.14}$$

where $\overline{\omega}$ and f_{v2} are given by

$$\overline{\omega} = \frac{\tilde{\nu}}{\kappa^2\mathrm{d}_w^2}f_{v2}, \qquad f_{v2} = 1 - \frac{\chi}{1+\chi f_{v1}}. \tag{2.15}$$

For the sake of completeness, we give the values of the constants in the above expressions: $c_{v1} = 7.1$, $\sigma = 2/3$, $c_{b1} = 0.1355$, $c_{b2} = 0.622$, $\kappa = 0.41$, $c_{w2} = 0.3$, $c_{w3} = 2$, $r_{lim} = 10$, $c_{t3} = 1.2$, $c_{t4} = 0.5$, $c_{v2} = 0.7$, $c_{v3} = 0.9$, $c_{n1} = 16$.

2.2.2 Scale-resolving simulations

While RANS (and URANS) well predicts the average and the instantaneous forces on the body, the temporal fluctuations are lost, and they prove not reliable for highly unsteady flows. For computations as acoustic analysis, scale-resolving simulations (DNS, LES or hybrid RANS/LES) are mandatory in order to obtain good results.

Direct Numerical Simulation

The DNS method is the most straightforward approach to solve unsteady turbulent flows, since the entire spectrum of the turbulence is represented without the need of any modeling. DNS does not involve approximations, other than those due to the numerical discretization. The unsteady, three-dimensional Navier-Stokes equations are numerically solved, and all the motion scales are resolved, requiring a mesh fine enough to capture the smallest scale of turbulence.

However, the high computational cost of this method makes it prohibitive for most engineering problems, where typical Reynolds are in the range of $Re = 10^5 - 10^8$.

In fact, the computational cost of DNS depends on the Reynolds number: the higher the Reynolds number, the higher the spatial requirements in order to solve all the scales. This can be demonstrated from the Kolmogorov analysis [171] for homogeneous isotropic turbulence, according to which the ratio between the largest scales L of the flow and the smallest scales η is proportional to a power of the Reynolds number related to the large scales Re_L

$$\frac{L}{\eta} \propto Re_L^{\frac{3}{4}} . \tag{2.16}$$

The computational cost is largely determined by the resolution requirements, since the grid spacing must be small enough to resolve the dissipative scales η, and the computational box size must be large enough to represent the energy-containing motions with scale L.

Given the three-dimensionality of turbulence, the required number of degrees of freedom to compute the entire complete flow, scales with $Re_L^{\frac{9}{4}}$.

Moreover, it has been shown [88] that the simulation of wall-bounded flows increases the dependence of the number of degrees of freedom on the Reynolds number to $Re_L^{3.5}$, due to the higher resolution needed by turbulent boundary layers. While being a powerful research tool to carry out experiments which are difficult or impossible to realize in the laboratory, DNS cannot be considered for engineering design. The resolution of the Navier-Stokes equations is for the moment restricted to low Reynolds flows.

Large Eddy Simulation

For this reason, LES was introduced in the 60s [274]. Here turbulence scales smaller than the mesh size are filtered and modeled, while the large scales containing most of the energy of the flow are fully simulated. However, this method requires a very fine resolution in the three directions near the body because the very small scales of turbulence present in the thin boundary layer along the walls must have a reasonably high resolution in order to correctly evolve. Usually Wall Resolved LES requirements impose $\Delta y^+ = 1$, $\Delta x^+ = 100$ and $\Delta z+ = 20$ [83], where y is the direction normal to the wall, x is the streamwise and z is the spanwise direction.

Wall Resolved LES methods are still too cost demanding for the industrial context. Wall-Modeled LES (WMLES) [237, 177] are LES methodologies where the near-wall region is modeled, relaxing the $y^+ = 1$ constraint, and allowing to simulate a time-dependent wall-bounded flow with affordable computational resources.

Even though LES will never explicitly appear in the computations in this work, a small overview of the LES classical approach is given here for the sake of completeness and to help the understanding of hybrid RANS/LES methods.

The flow quantities may be decomposed into a large scale contribution and a subgrid contribution by applying a spatial filter to the Navier-Stokes equations. The conservative variables $\mathbf{u}$ are therefore decomposed as follows

$$\mathbf{u}(\mathbf{x}, t) = \widehat{\mathbf{u}}(x, t) + \mathbf{u}'(\mathbf{x}, t) . \tag{2.17}$$

The large scales $\widehat{\mathbf{u}}$ are adequately resolved on a relatively coarse grid, whereas the effects of the small scales $\mathbf{u}'$ contribution on the large scale flow are described by a model. Two important differences with Reynolds decomposition in equation (2.5) are that $\widehat{\mathbf{u}}$ is obtained by a spatial filter instead of a time average, and that it contains a part of the most energetic turbulent fluctuations, that control the evolution of the flow. This is in contrast with the Reynolds decomposition, where all the turbulent scales are filtered out by the time average operator. In particular $\widehat{\mathbf{u}}$ corresponds to the filtered field with a filter function G_Δ associated with a length Δ.

In the case of compressible flow problems, such as the ones considered in the present work, we introduce a filter which is weighted by the density, the Favre filtering operator $\widetilde{(\cdot)}$. For any quantity f we define a Favre filtered quantity as $\tilde{f} = \dfrac{\widehat{\rho f}}{\widehat{\rho}}$.

Similarly to what has been obtained in RANS modeling with the introduction of $\boldsymbol{\tau}_t$, the filtered momentum equation involves an additional term expressing the effects of the modeled scales on the resolved scales, the subgrid stress tensor

$$\boldsymbol{\tau}_{\mathbf{SGS}} = -\widehat{\rho \mathbf{v} \otimes \mathbf{v}} + \hat{\rho} \tilde{\mathbf{v}} \otimes \tilde{\mathbf{v}} \ . \tag{2.18}$$

The subgrid stress tensor $\boldsymbol{\tau}_{\mathbf{SGS}}$ can be split into a deviatoric and a spherical part:

$$\boldsymbol{\tau}_{\mathbf{SGS}} = \boldsymbol{\tau}^S_{\mathbf{SGS}} + \boldsymbol{\tau}^D_{\mathbf{SGS}} \ . \tag{2.19}$$

The spherical contribution has been included in the modified filtered pressure

$$\hat{p}\mathbf{I} \equiv \hat{p}\mathbf{I} - \boldsymbol{\tau}^S_{\mathbf{SGS}} \ , \tag{2.20}$$

while the subgrid stress tensor $\boldsymbol{\tau}^D_{\mathbf{SGS}}$ needs to be modeled in order to close the system of the filtered Navier-Stokes equations.

The most commonly used subgrid closure is of the eddy-viscosity type, which consists in approximating $\boldsymbol{\tau}^D_{SGS}$ in terms of filtered flow variables

$$\boldsymbol{\tau}^D_{\mathbf{SGS}} \approx \hat{\rho} \nu_{sgs} \widetilde{\mathbf{S}^D} \ , \tag{2.21}$$

by introducing the concept of an eddy viscosity ν_{sgs}. The subgrid heat flux can be modeled with a similar eddy-viscosity approach.

For high Reynolds numbers, the eddy viscosity can be modeled as $\nu_{SGS} = l_S^2 |\tilde{\mathbf{S}}|$, with l_S a characteristic length scale of the model depending on the filter width Δ. The Smagorinsky model is the most commonly used LES model, and reads $\nu_{SGS} = (C_S \Delta)^2 |\tilde{\mathbf{S}}|$, with C_S the Smagorinsky constant.

Despite the difficulties associated with determining an appropriate C_S, the value $C_S = 0.18$ was obtained with the procedure proposed by Lily in [187]. However, the Smagorinsky subgrid model has been successfully used to carry out LES calculations in a number of different applications with a $C_S = 0.1 - 0.2$.

Among the most popular approaches for the modeling of the eddy viscosity, we can find also the dynamic model by Germano [144], the WALE [224], the Vreman [296] and the VMS [157] models.

Hybrid RANS/LES methods

Hybrid RANS/LES methods allow to combine a LES type resolution far from the walls and a RANS modeling in the boundary layer. This method allows for using highly anisotropic meshes (RANS like) at the walls, while maintaining a good resolution of the turbulence far from them.

Among the several hybrid RANS/LES models that have been proposed over the years, and carefully reviewed in [131, 79, 154], we can find the Detached Eddy Simulation (DES), the Partially-averaged Navier-Stokes (PANS), the Partially Integrated Transport Modeling (PITM) and the Scale Adaptive Simulation (SAS). The key differentiating factor between these methods is the approach employed to control the turbulent viscosity in different regions of the flow. By progressively reducing the turbulent viscosity in regions away from the wall, the different models allow the development and resolution of the largest turbulent scales.

In DES models this is obtained by applying an upper limit to the RANS length scale in a destruction term. In LES zones, the limiting is active and the eddy viscosity is decreased by increasing the destruction term.

PANS, PITM, SAS models modify the RANS model coefficients in RANS equations such that RANS equations can develop fluctuating solutions.

In the PANS and PITM, the turbulent viscosity is reduced by decreasing the destruction term in the dissipation (ϵ) equation. This decreases the turbulent viscosity both because the dissipation term has increased, and because the increased dissipation term decreases the turbulent kinetic energy. These models do not use a filter width, while resolving a large part of the turbulence spectrum, and for this reason they have been classified by Fröhlich and von Terzi [131] as "second-generation URANS models".

In the SAS model based on the $k-\omega$ model, an additional source term in the ω equation is active in resolved turbulence, while it is inactive for steady flows. When the source term is active, it increases ω, which in turn decreases the turbulent viscosity.

In the next paragraph we will revise more in detail the approaches of Detached Eddy Simulation and the Zonal Detached Eddy Simulation.

Detached Eddy Simulation One of the most popular hybrid RANS/LES approaches is the Detached Eddy Simulation (DES), originally developed by Spalart *et al.* [277]. It is based on the consideration that RANS modeling efficiently simulates attached boundary layers, while presenting many difficulties in providing realistic solutions for separated flows, which require an LES approach. Thus the main idea is to solve a RANS model close to the wall and an LES model farther away.

In DES formulation, the wall distance d_w of the SA turbulent variable transport equation, is replaced with the hybrid length scale $\tilde{\mathrm{d}}_{DES}$.

$$\tilde{\mathrm{d}}_{DES} = \min(\mathrm{d}_w, C_{DES}\tilde{\Delta}_{DES}) \ , \tag{2.22}$$

where $\tilde{\Delta}_{DES}$ is the maximum grid extension $\Delta_{max} = \max(\Delta_x, \Delta_y, \Delta_z)$ in the original formulation. C_{DES} is a coefficient usually set to 0.65 which has been calibrated from decaying homogeneous turbulence simulations [270].

In practice this modification is active in the destruction term of the RANS-SA equation. Using this model, the eddy viscosity of the RANS model progressively turns into a subgrid viscosity $\nu_t \propto \tilde{\omega}\Delta^2$ of the same type of Smagorinsky turbulent viscosity in the LES areas in the free stream and flow separation away from the walls, when production $P \sim \tilde{\omega}\tilde{\nu}$ and destruction $D \sim \left(\frac{\tilde{\nu}}{\tilde{\mathrm{d}}_{DES}}\right)^2$ terms are balanced.

A main drawback of the modeling was found in the so-called "Modeled-Stress Depletion" (MSD) phenomenon [278], usually accompanied by a decrease of the skin friction coefficient. This phenomenon occurs when the grey area (the region where the model switches from the fully modeled turbulence (RANS) to mostly resolved turbulence (LES) and $\mathrm{d}_w \simeq \Delta$) is located in the boundary layer, but the available resolution is not sufficient to resolve the relevant turbulent scales, typically when the streamwise and spanwise grid spacings are similar to the boundary layer thickness Δ_x or $\Delta_z \simeq \delta$. In this case, if the mesh is not sufficiently fine to correctly capture the turbulent scales via the LES mode, and, not being the model able to convert the modeled turbulent kinetic energy into a resolved one, the Reynolds stress levels are underestimated compared to the RANS model. In some cases, as a consequence, the separation line moves forward leading to a "Grid-Induced Separation" (GIS) [208].

Several modifications to the original DES formulation have been proposed over the years, in order to solve the aforementioned shortcomings. The most popular are the Delayed Detached Eddy Simulation (DDES) [278] and the Improved Delayed Detached Eddy Simulation (IDDES) [273]. In DDES, the DES length is modified to incorporate a shielding function f_d depending on the velocity gradients, built to ensure that boundary layers are fully treated in RANS, based on measures depending on flow characteristics and not only from the grid design. The IDDES model combines DDES with Wall Modeled LES functionalities. In general, the IDDES formulation provides different response depending on the presence or not of inflow turbulent content. In the first case, it reduces to WMLES, otherwise a DDES response is obtained. However IDDES is only of limited use in industrial flows as it does not provide sufficient shielding capabilities in RANS mode [207].

The Zonal Detached Eddy Simulation The hybrid RANS/LES method that we use in this work is the Zonal Detached Eddy Simulation (ZDES). It was proposed by Deck [101] and was initially based on the Spalart Allmaras (SA) [276, 23] RANS model, as an efficient solution to achieve a faster decay of the eddy viscosity in the LES mode with respect to classical DES/DDES. ZDES prevents then the "Grid-Induced Separation" phenomenon and the "Modeled-Stress Depletion".

In the ZDES framework, the wall distance d_w of the SA turbulent variable transport equation is replaced with the hybrid length-scale $\tilde{\mathrm{d}}_{ZDES}$. The employed hybrid length scale, corresponding to a particular "mode" of ZDES, depends on the problem of interest. Mode 1 is used for flows where the separation is fixed by the geometry, mode 2 for flows where the separation is induced by a pressure gradient on a smooth surface, while mode 3 concerns flows in which the separation is strongly influenced by the dynamics of the boundary layer.

In ZDES, the user has to specify the mode which needs to be used in each of the

regions, depending on the flow characteristics. The length scale $\tilde{d}_{ZDES}$ for each of the modes is indicated as:

$$\tilde{d}_{ZDES} = \begin{cases} d_w & mode\ 0 \\ \tilde{d}^{I}_{DES} & mode\ 1 \\ \tilde{d}^{II}_{DES} & mode\ 2 \\ \tilde{d}^{III}_{DES} & mode\ 3 \end{cases} \tag{2.23}$$

where the exact definitions of $\tilde{d}^{(\cdot)}_{DES}$ are given later, and depend on a $\tilde{\Delta}$, the subgrid length scale.

The original definition proposed for $\tilde{\Delta}$ in ZDES is the cubic root of the volume of the cell $|K|$, $\Delta_{vol} = \sqrt[3]{|K|}$, and not the maximum grid extension $\Delta_{max} = \max(\Delta_x, \Delta_y, \Delta_z)$ as Spalart *et al.* proposed in their DES formulation. More recently, Chauvet *et al.* [85] proposed an alternative definition for $\tilde{\Delta}$ taking into account the local vorticity of the flow.

In this work we use the original ZDES formulation $\tilde{\Delta} = \Delta_{vol}$, but adapted for high-order schemes. This length scale is divided by the number of 1D degrees of freedom for a DG method, $p+1$, in order to obtain the high-order sub-grid length scale of the DES model $\tilde{\Delta}_{DG} = \frac{\Delta_{vol}}{p+1}$, as already considered in the LES literature for both hexahedral and tetrahedral meshes [99, 190]. The hybrid length scale for each mode takes the following definition:

$$\tilde{d}^{I}_{DES} = \min(d_w, C_{DES}\tilde{\Delta}^{I}_{DES}) \tag{2.24}$$

$$\tilde{d}^{II}_{DES} = d_w - f_d \max(d_w - C_{DES}\tilde{\Delta}^{II}_{DES}) \tag{2.25}$$

$$\tilde{d}^{III}_{DES} = \begin{cases} d_w & \text{if } d_w < d_w^{\text{interface}} \\ \tilde{d}^{I}_{DES} & \text{otherwise .} \end{cases} \tag{2.26}$$

In the above, f_d is the original DDES shielding function, $\tilde{\Delta}^{I}_{DES}$ corresponds to the aforementioned $\tilde{\Delta}$ (that in this work corresponds to $\tilde{\Delta}_{DG}$ to take into account the DG scaling), and $\tilde{\Delta}^{II}_{DES}$ depends on the value of f_d (see Deck [102]). Recently Deck [103] proposed a new formulation of the ZDES mode 2 - EP (Enhanced Protection) to avoid the failure of the previous method when considering fine meshes or adverse pressure gradient. The hybrid length scale $\tilde{d}^{III}_{DES}$ relies on the parameter $d^{\text{interface}}$ which is the distance from the wall of the RANS/LES interface where the switch between the two models occurs.

While in the ZDES mode 1 and 2 the attached boundary layers are treated in RANS, ZDES mode 3 can be seen as a Wall Modeled LES approach in which the outer part of attached boundary layers is resolved in LES. While the first two methods can handle typical RANS streamwise and spanwise resolutions in the wall region, the spacing requirements for the third mode are more strict and tend to those for LES [250, 135].

Another aspect differentiating ZDES from standard DES/DDES approaches is that the near-wall corrections f_{v1}, f_{v2}, f_w are the same as the RANS Spalart-Allmaras model in RANS mode, while in LES mode they are modified:

$$f_{v1} = 1, \qquad f_{v2} = 0, \qquad f_w = 1, \tag{2.27}$$

which are their asymptotic values far from the wall. This formulation avoids a drop of the subgrid viscosity that could be caused by the damping functions of the RANS model in the resolved LES zones presenting low eddy viscosity levels. In this work, the modifications in equation 2.27 and the length scale $D\tilde{e}lta_{DG}$, are used also for DES computations in Chapters 4 and 6.

2.3 Numerical discretization of the fluid dynamics equations

2.3.1 The discontinuous Galerkin method

Here we present a brief description of the discontinuous Galerkin (DG) method implemented in the CODA solver. A more detailed presentation of this class of methods is provided in [91, 111].

We start by defining a shape-regular partition of the domain Ω, into a tessellation $\mathcal{T}_K$ of N non-overlapping and non-empty elements K of characteristic size h. We also define the sets $\mathcal{E}_i$ and $\mathcal{E}_b$ of interior and boundary faces in $\mathcal{T}_K$, such that $\mathcal{E}_h = \mathcal{E}_i \cup \mathcal{E}_b$.

Let $\mathcal{V}_h^p = \{\phi_h \in L^2(\Omega) : \phi \mid_K \in \mathcal{P}^p(K), \forall K \in \mathcal{T}_K\}$ be the functional space formed by piece-wise polynomials of total degree at most p, and $\Phi_K^p = (\phi_K^1, ..., \phi_K^{N_p}) \in \mathcal{P}^p(K)$ a hierarchical and orthonormal basis of $\mathcal{P}^p$, of dimension N_p, confined to K.

The solution in each element is thus expressed as

$$\mathbf{u}_h(\mathbf{x}, t) = \sum_{l=1}^{N_p} \phi_K^l(\mathbf{x}) \mathbf{u}_K^l(t), \forall \mathbf{x} \in K, \quad K \in \mathcal{T}_K, \forall t \geq 0 \, . \tag{2.28}$$

The polynomial coefficients $(\mathbf{u}_K^l)_{1 \leq l \leq N_p}$ represent the degrees of freedom of the discrete problem in element K. To generate the orthonormal basis within each element, we employ the methodology developed by Bassi *et al.* [39], which consists in defining a starting set of monomial basis functions in each (arbitrarily shaped) element and applying a modified Gram-Schmidt orthonormalization procedure. The resulting basis yields a diagonal mass matrix in each element of the discretization, simplifying the resolution of the sets of equations with the variational formulation. Additionally, Φ_K^p is a hierarchical basis if it is contained in higher-order basis, that is $\Phi_K^p \subset \Phi_K^{p+1}$. In this case, equation (2.28) can be rewritten for an element K of degree p_K as:

$$\mathbf{u}_h(\mathbf{x}, t) = \sum_{q=0}^{p_K} \sum_{l \in d_q} \phi_K^l(\mathbf{x}) \mathbf{u}_K^l(t), \forall \mathbf{x} \in K, \quad K \in \mathcal{T}_K, \forall t \geq 0 \, , \tag{2.29}$$

where $d_0 = \{1\}$ and $d_q = \left\{ l \in 2...N_p(K) \mid \phi_K^l \in \mathcal{P}^q(K) \setminus \mathcal{P}^{q-1}(K) \right\}$ is the set of indices of the basis functions of total degree equal to q.

The number of degrees of freedom for a polynomial degree of the DG approximated solution p in two and three dimensions are respectively:

$$N_{p,2D} = \frac{(p+1)(p+2)}{2} \qquad \text{and} \qquad N_{p,3D} = \frac{(p+1)(p+2)(p+3)}{6} \, . \tag{2.30}$$

The conservation law is discretized in physical space by using a discontinuous Galerkin method and the semi-discrete variational form of the system of equations (2.1) thus reads: find $\mathbf{u}_h$ in $\mathcal{V}_h^p$ such that $\forall \phi_h \in \mathcal{V}_h^p$ we have

$$\int_{\mathcal{T}_K} \phi_h \partial_t \mathbf{u}_h dV + \mathcal{L}_c(\mathbf{u}_h, \phi_h) + \mathcal{L}_v(\mathbf{u}_h, \phi_h) = 0 \, . \tag{2.31}$$

In equation (2.31) $\mathcal{L}_c$ and $\mathcal{L}_v$ represent the weak form of the convective and viscous terms respectively. The following notations are introduced: for a given interface e in $\mathcal{E}_i$ we define the average operator as $\{\{\mathbf{u}\}\} = (\mathbf{u}^+ + \mathbf{u}^-)/2$, the jump operator is defined as $[[\mathbf{u}]] = \mathbf{u}^+ \otimes \mathbf{n} - \mathbf{u}^- \otimes \mathbf{n}$ where $\mathbf{u}^+$ and $\mathbf{u}^-$ are the traces of the variable $\mathbf{u}$ at the interface between elements K^+ and K^-. The DG discretization of the convective terms then reads

$$\mathcal{L}_c(\mathbf{u}_h, \phi_h) = -\int_{\mathcal{T}_K} \mathbf{F}_c(\mathbf{u}_h) \cdot \nabla_h \phi_h dV + \int_{\mathcal{E}_i} [[\phi_h]] \mathbf{h}_c(\mathbf{u}_h^+, \mathbf{u}_h^-, \mathbf{n}) dS + \int_{\mathcal{E}_b} \phi_h^+ \mathbf{F}_c(\mathbf{u}_b) \cdot \mathbf{n} dS \, , \tag{2.32}$$

where the boundary values $\mathbf{u}_b = \mathbf{u}_b(\mathbf{u}_h^+, \mathbf{u}_{ext}, \mathbf{n})$, with $\mathbf{u}_{ext}$ a reference external state computed such that the boundary conditions are satisfied on $\mathcal{E}_b$. The numerical flux $\mathbf{h}_c$ is chosen such that it is consistent and conservative. In this work we use the Roe flux [254] with an entropy fix similar to that of Harten [151] for all simulations. In general, Roe-type fluxes, presenting or not an entropy fix or other modifications of the flux function dissipation [43, 99, 47, 174, 299], are a common choice for DG simulations for different types of flows, together with the LLF [218, 43, 99, 249] numerical flux.

The discretization of the viscous terms is performed using the BR1 approach of Bassi & Rebay [41]. This approach relies on the definition of the conservative variable gradients as auxiliary variables $\boldsymbol{\sigma} = \nabla \mathbf{u}$ which verify the following equations:

$$\boldsymbol{\sigma} - \nabla \mathbf{u} = 0 \, , \tag{2.33}$$

$$\partial_t \mathbf{u} + \nabla \cdot (\mathbf{F}_c(\mathbf{u}) - \mathbf{F}_v(\mathbf{u}, \boldsymbol{\sigma})) = 0 \, . \tag{2.34}$$

This leads to the introduction of the so called global lifting operator $\mathbf{L}_h$ such that:

$$\boldsymbol{\sigma}_h = \nabla_h \mathbf{u}_h + \mathbf{L}_h \, , \tag{2.35}$$

and $\mathbf{L}_h$ satisfies the following condition

$$\int_{\mathcal{T}_K} \phi_h \mathbf{L}_h dV = -\int_{\mathcal{E}_i} \{\{\phi\}\} [[\mathbf{u}_u]] dS - \int_{\mathcal{E}_b} \frac{\phi^+}{2} (\mathbf{u}_h^+ - \mathbf{u}_b) \otimes \mathbf{n} dS \, . \tag{2.36}$$

The discrete variational form of the viscous term for the BR1 method therefore takes the form

$$\begin{aligned} \mathcal{L}_v(\mathbf{u}_h, \phi_h) = &\int_{\mathcal{T}_K} \mathbf{F}_v(\mathbf{u}_h, \nabla_h \mathbf{u}_h + \mathbf{L}_h) \cdot \nabla_h \phi_h dV - \int_{\mathcal{E}_i} [[\phi_h]] \{\{\mathbf{F}_v(\mathbf{u}_h, \nabla_h \mathbf{u}_h + \mathbf{L}_h)\}\} \cdot \mathbf{n} dS \\ &- \int_{\mathcal{E}_b} \phi_h^+ \mathbf{F}_v(\mathbf{u}_b, \nabla \mathbf{u}_b + \mathbf{L}_h) \cdot \mathbf{n} dS \, . \end{aligned} \tag{2.37}$$

The integrals in the formulas above are computed by numerical integration, with the specific quadrature formulas depending on the type of element/face being considered. We here therefore briefly list the quadrature rules used in this work, and the corresponding number of quadrature points.

Quadrilateral and hexahedral elements use tensor-product formula obtained from the 1D Gauss-Legendre quadrature rule. Numerical integration on triangles and tetrahedra is efficiently performed by means of the optimized quadrature rules proposed by Witherden et al. [307]. Prismatic elements employ a combination of a Gauss-Legendre quadrature in the extrusion direction and the aforementioned optimized quadrature for triangles in the other two directions. When the required integration degree id, which indicates the maximum degree of a polynomial which is exactly integrated by the formula, is superior to 7, these optimized quadrature rules are replaced by the tensor-product rules for every type of element.

By default, the integration degree used for a given polynomial degree p of the DG discretization is defined as $id = 2p + 1$. The number of integration points N_{qp} required by each quadrature rule for $p \leq 5$ are reported in table 2.1. For higher polynomial or integration degrees, the number of integration points is simply computed as $N_{qp} = \left[\text{INT}\left(\frac{id+2}{2}\right)\right]^d$, where INT is the integer division and d is the number of dimensions of the element/face.

polynomial degree p	integration degree id	#dofs 2D $N_{p,2D}$	#dofs 3D $N_{p,3D}$	#quadrature points N_{qp}				
				tri	quad	tetra	prisms	hexa
1	3	3	4	6	4	8	12	8
2	5	6	10	7	9	14	21	27
3	7	10	20	15	16	35	60	64
4	9	15	35	25	25	125	125	125
5	11	21	56	36	36	216	216	216

Table 2.1: Number of dofs per element in 2D and 3D in the first two columns. Default number of quadrature points per element for 2D elements (triangles and quadrilaterals) and 3D elements (tetrahedra, prisms, hexahedra) used in this work, without overintegration.

Overintegration may be performed by using an increased integration degree with respect to the default for a given polynomial degree. This means that, for example, for $p = 2$ in a tetrahedron with integration degree $id = 7$ there will be $N_{qp} = 35$ quadrature points, and in any $p = 4$ 3D element with integration degree $id = 11$ there will be $N_{qp} = 216$ quadrature points. In this work standard integration is always used with the exception of the convected vortex test case analyzed in Chapters 4 and 6, where overintegration is performed.

2.3.2 Time integration

The spatially discrete problem can be written as an initial value problem of the following form: being $\mathbf{U}$ the vector of degrees of freedom $u_K^i(t)$, given $\mathbf{U}(0)$, find $\mathbf{U}(t)$ such that

$$\mathcal{M}\frac{d\mathbf{U}}{dt} + \mathcal{R}(\mathbf{U}) = 0 \,, \tag{2.38}$$

where $\mathcal{R}$ is the spatial residual vector, and $\mathcal{M}$ is the block-diagonal mass matrix. The mass matrix of the element K is the K-th block of $\mathcal{M}$ and has components

$$\mathcal{M}_{ij}^K = \int_K \phi_i \phi_j dV \,. \tag{2.39}$$

The system (2.38) can be solved with any ordinary differential equations integration scheme in order to obtain the evolution of the dofs $u_K^i(t)$.

Two different types of temporal discretization schemes are employed in this work.

The time integration for unsteady problems can usually be performed by using explicit Runge-Kutta schemes [67, 269] or implicit methods, such as Rosenbrock-Wanner, ESDIRK and BDF methods [302]. In the current work only the explicit third or fourth order classical Runge-Kutta schemes [67] are employed for unsteady simulations.

For steady problems we use an implicit integration scheme. One of the most used methods is the backward Euler method to advance the solution in time from an initial condition until the steady state solution is achieved. In this work the initial condition is either a uniform flow, or is projected from a simulation obtained on a discretization with a lower resolution (in terms of h or p or both). In order to solve the resulting linear system, the two main classes of iterative methods are the stationary iterative methods, among which we find the Jacobi method and the Gauss-Seidel method, and the more general Krylov subspace methods, such as the preconditioned GMRES [258]. In this work we use a matrix-free GMRES method with a block Jacobi preconditioner, using an LU block decomposition.

2.3.3 Shock capturing

The modal DG discretization of the flow equations (Navier Stokes, Euler, RANS) employed in this work requires the addition of an artificial viscosity to obtain stable solutions in the presence of shocks. The results presented in the last chapter of this manuscript use the modification by Glaubitz [145] of the original artificial viscosity method by Persson and Peraire [236].

Using the Laplacian formulation of the artificial viscosity, the equation (2.1) is modified adding a term $\nabla \cdot (\epsilon_{AV} \nabla \mathbf{u})$:

$$\begin{aligned} \partial_t \mathbf{u} + \nabla \cdot (\mathbf{F}_c(\mathbf{u}) - \mathbf{F}_v(\mathbf{u}, \nabla \mathbf{u})) &= \nabla \cdot (\epsilon_{AV} \nabla \mathbf{u}), \quad \forall \mathbf{x} \in \Omega, t > 0 \\ \mathbf{u}(\mathbf{x}) &= \mathbf{u}_0, \quad \forall \mathbf{x} \in \Omega \end{aligned} \,. \tag{2.40}$$

In particular ϵ_{AV} is the dynamic artificial viscosity μ_{AV} in all the equations, except for the total energy equation, where μ_{AV} is replaced by an artificial thermal conductivity due to the artificial viscosity $\lambda_{AV} = \mu_{AV} \dfrac{C_p}{Pr}$.

A modal resolution-based shock sensor S in equation (2.41) is introduced in order to define the location and strength of the dissipation added by the artificial viscosity method. It is piecewise constant and it is defined using the density as a characteristic quantity for the shock detection.

$$S = \frac{||(\rho)_{h,p} - (\rho)_{h,p-1}||^2_{L^2(K)}}{||(\rho)_{h,p}||^2_{L^2(K)}} \ . \tag{2.41}$$

The sensor measures the norm of the difference between the numerical solution $\rho_{h,p}$ and the projection of the numerical solution on the reduced-order space $\mathcal{V}_h^{p-1}$, $\rho_{h,p-1}$, divided by the norm of the density.

Once the shock has been detected by the sensor S, and artificial viscosity is activated, the amount of viscosity to add is determined by the smooth viscosity ν_{AV} defined as

$$\nu_{AV} = \begin{cases} 0 & , s < s_{ref} - k_r \\ \frac{\nu_{AV}^{max}}{2}\left(1 + \sin\frac{\pi(s - s_{ref})}{2k}\right) & , s_{ref} - k_r \leq s \leq s_{ref} + k_r \\ \nu_{AV}^{max} & , s > s_{ref} + k_r \end{cases} \tag{2.42}$$

The maximum value of the artificial viscosity $\nu_{AV}^{max} \propto \frac{h}{p}$ is defined as:

$$\nu_{AV}^{max} = \frac{1}{2}\max\left(\left|\frac{\partial f}{\partial u}\right|\right) \cdot \frac{h}{p} \ , \tag{2.43}$$

with $\max\left(\left|\frac{\partial f}{\partial u}\right|\right)$ the maximum eigenvalue of the convective term over the integration points of all the elements, h the element characteristic size, p the polynomial degree, and k_r a case-dependent ramp parameter, usually chosen equal to 1.

The sensor s in equation (2.42) is modified from the original version in [236] and chosen as

$$s = \log_{10} F \text{ with } F = \min\left(cp^4 \cdot S, 1\right) \ , \tag{2.44}$$

with $s_{ref} = -2$ and c a parameter defined to control the sensor sensitiveness. The value of the parameter c increases with the nonlinearity of the underlying equations. In this work $c = 1$ is employed.

Chapter 3

State of the art of spatial resolution adaptive techniques

Spatial resolution adaptive techniques aim at locally increasing (or decreasing) the degrees of freedom in a computational domain, in order to obtain an accurate numerical solution with a reduced computational cost (memory requirements and CPU time) as compared to standard approaches based on uniform refinement or manual meshing of the computational domain.

Generally, there are three categories of techniques for adapting the spatial resolution: h-adaptation, where the element sizes vary while the order of the numerical approximation is constant; p-adaptation, where the order of the numerical approximation varies while the element sizes are constant; hp-adaptation, which is a combination of both h- and p-adaptation.

The vast majority of the studies reported in the literature focuses on h-adaptivity, which is the only option for low order numerical discretizations. For high-order methods, h-adaptivity is mostly well-suited for non-smooth solutions with singularities such as shocks, or geometrical singularities (corners or sharp edges). However, in regions of the computational domain where the solution is smooth but still presenting large numerical errors, p-adaptivity is preferred since it provides the fastest convergence rate for smooth problems. This has led to the development of hp-adaptive techniques combining both h- and p-refinement options.

We present in this chapter an overview of the existing h-, p- and hp-adaptive strategies, and introduce the fundamental background for the adaptive techniques developed in this work.

3.1 h-adaptation techniques

Concerning pure h-adaptation, two main approaches, reviewed more in detail in the following sections, are identified:

- Element subdivision employing non-conforming elements and hanging nodes;
- Metric-based remeshing.

Other two less common approaches that can be found in the literature are:

- Conformal element subdivision. Some examples using triangular/tetrahedral meshes can be found in [251, 28, 320, 308]. In the case of quadrilateral/hexahedral elements, either quadrilaterals/hexahedra can be obtained after the adaptation [138, 150] or a combination of elements, such as quadrilaterals and triangles in 2D or pyramids and tetrahedra in 3D [223]. This type of adaptation, especially in the case of quadrilaterals/hexahedra, generally requires complex algorithms in order to avoid poor quality elements.
- *r*-adaptation [22, 90, 316], consists in adapting the mesh by node relocation with constant connectivity, offering the possibility to introduce mesh adaptation capabilities into existing flow solvers with minimal intrusivity, as there is no need for updating data structures and connectivities.

These last two approaches are less common, given their lower degree of flexibility and their strong dependence on the initial mesh.

3.1.1 Hanging nodes approach

In the element splitting with hanging nodes technique, the produced meshes are characterized by nodes on edges and faces that are not vertices of all the cells sharing those edges or faces. When refining a mesh with this approach, every initial "parent cell", selected for refinement by a given marking criterion, is divided into "children cells". On the other hand, coarsening is performed by reverting a refinement step, which consists in agglomerating children cells into the initial parent cells. Although the hanging node scheme provides significant mesh flexibility, it requires additional memory to maintain the mesh hierarchy which is used by the refinement/coarsening mesh adaptation operations.

Element splitting-based mesh adaptation is mainly performed on (initially) structured hexahedral meshes, however some examples of non-conformal element splitting based on triangular/tetrahaedral meshes can be found in the literature [323]. The element splitting technique allows the refinement to remain local but introduces non-conformal nodes which lie on the edges of neighboring elements. Despite the ability of DG methods to easily handle non conforming meshes from a numerical point of view [175, 75, 181, 78], those meshes can be cumbersome to handle at the implementation level.

The element splitting can be isotropic or anisotropic. In isotropic refinement, cells marked for refinement are split along all the directions. In anisotropic refinement, the division is applied in one (or even two in 3D) dominant direction. Anisotropic refinement is usually employed when the flow features are predominantly to be resolved in one direction, as for example boundary layers and shocks. However, one main disadvantage of such techniques is that the possible stretching directions of the elements are strongly dependent on the initial mesh topology.

From a practical point of view, isotropic refinement for a quadrangular element would produce four new children elements, while anisotropic refinement would only generate two.

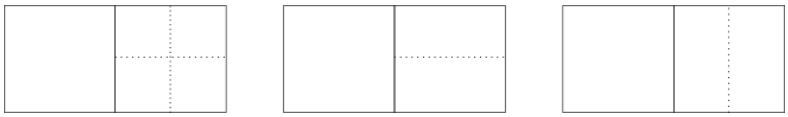


Figure 3.1: Isotropic (left) and anisotropic (middle and right) element splitting.

Hanging nodes-based isotropic mesh adaptation

Some examples of isotropic mesh adaptation by element splitting can be often found the literature for laminar [226] or low-to-moderate Reynolds numbers scale resolving flows [164, 26]. Their use for higher Reynolds number flows in RANS and hybrid RANS/LES simulations is mainly established in the context of overset grids [77, 306].

Here elements presenting an error above a certain threshold are refined by dividing a parent cell into four (in two dimensions) or eight (in three dimensions) children cells. Usually this threshold is defined by a fixed-fraction marking strategy, where a fixed fraction of all elements are selected for refinement and/or coarsening based on the absolute values of the local indicators, or by a maximum marking strategy, in which the element is marked when its error is greater than a given fraction of the maximum local error.

Usually, researchers enforce that the mesh presents at most a degree of nonconformity of two. This constraint means that an element can have in 2D a maximum of two neighboring elements along a single edge, whereas in 3D it cannot have more than 4 neighboring elements along a single face. This is done for ease of implementation, and to avoid strong cell-to-cell size variations.

Hanging nodes-based anisotropic mesh adaptation

Anisotropic mesh adaptation requires the automatic detection of the dominant directions along which additional resolution is required. In classical second order FV/FE approaches, this information is often provided by the gradients [162, 321] or the Hessian of the solution [298]. However, different criteria can be found as regards LES simulations, where anisotropic error indicators, estimating the directional small-scale energy, can be employed to adapt a grid in an anisotropic manner [282].

In the context of DG simulations for advection-diffusion-reaction problems, Georgoulis *et al.* [142] implemented an anisotropic adaptive algorithm to control of the error in a prescribed target functional. This approach involves exploiting both local isotropic and anisotropic mesh refinement, based on choosing the most competitive subdivision of elements marked for refinement from a series of trial local refinements.

Using a similar approach, Ceze & Fidkowski [74] proposed to prescribe the anisotropy of the elements naturally arising from the minimization of a cost function including both an output error estimate and a count of the additional degrees of freedom for each refinement option, with application to laminar and RANS configurations. With this approach, the anisotropy is not directly prescribed from direct assumptions on the solution anisotropy.

In a DG *hp*-adaptive framework with unsteady laminar applications, Chalmers *et al.* [78] used a directional estimator built from the coefficients of the Legendre polyno-

mials, and performed directional *h*-refinement based on the value of this estimator in each direction.

An alternative to the Hessian approach for high-order methods has been used for 2D and 3D non-conformal refinement of quadrilateral/hexahedral elements by Leicht & Hartmann [180, 181]. Here inter-element jumps in the solution can provide a simpler approximation for the $(p+1)^{th}$ order derivatives, providing comparable results to an approach using derivative information. While elements requiring refinement are determined with an adjoint-based error indicator, the anisotropy jump indicator decides whether to split the element along a particular direction, or in an isotropic manner. This approach has been successfully applied to 2D and 3D inviscid and laminar viscous simulations, and extended to RANS simulations in a later work on *hp*-adaptation [182].

3.1.2 Metric-based remeshing

Metric-based remeshing adaptive approaches are used essentially for simplicial meshes. These consist in generating a new mesh for the entire computational domain, adapted according to some metric fields that depend on physical or geometrical data. The original mesh is used to store the characteristics of the new mesh during regeneration. The new mesh is described using a Riemannian metric, based on the assumption that in an optimal mesh all edge lengths will have unit measure in the metric space.

The use of a metric-based approach provides high flexibility in terms of prescribing precise sizes to an element. This avoids the constraint imposed by the element splitting history, and the implementation of complex data structures needed by an element splitting approach.

However, despite the great flexibility offered by the remeshing strategy, the prescription of the metric is not straightforward, requiring a characteristic size for each node (or element), in addition to anisotropic characteristics (orientation and aspect ratio) in anisotropic approaches. The criteria for adapting the mesh are often more sophisticated than for element splitting-based mesh adaptation. In fact, for splitting-based *h*-adaptation, the prediction of the number of dofs in the new mesh is exact and this can be used to tackle more easily the definition of the error target above which an element should be marked for refinement (or below which it should be marked for coarsening when possible). In metric-based adaptation there exist some criteria to predict the number of elements in the new mesh, namely guided by the ratio of new and old sizes for isotropic meshes [47] and adding the contribution of aspect ratios for anisotropic meshes [76], or by the notion of "complexity", which is a quantity proportional to the number of vertices or elements in the mesh [191]. However these criteria strongly depend on the ability of the remesher to strictly respect the imposed metric, depending in turn also on the chosen smoothing parameters (e.g. h_{grad} in the case of the remesher MMG, as detailed later), which may affect this prediction.

The metric-based approach is extensively used in the literature, and a number of remeshing software is available, often under open-source licensing. Among the many mesh adaptation software using the metric-based strategy, based on the Delaunay method, we can find *BAMG* from University Pierre et Marie Curie-LJLL and INRIA [153], *Feflo.a* from INRIA [195], *refine* from NASA [210], *Omega_h* from Rensselaer Polytech-

nic Institute and subsequently by Sandia National Laboratories [159], *pragmatic* from Imperial College London [146], *EPIC* from Boeing [209], *Metrix* from INRIA [19], *MMG* from Bordeaux INP, CNRS, Sorbonne Université and INRIA [97].

Both the isotropic [98, 47, 232] and the anisotropic [191, 97, 314, 185] variants have shown promising results in the literature. However, while for isotropic remeshing approaches, more easily adaptable to 3D, the metric field can be built by defining only one size per element, anisotropic approaches would need an efficient approximation of a $(p+1)^{th}$ derivative tensor. The application of high-order anisotropic methods to configurations relevant for aeronautic purposes is part of very recent research.

After briefly recalling some theoretical notions underlying the metric-based mesh generation, we revise here some anisotropic and isotropic remeshing approaches. At the end of the section some features of the remesher used in this study, MMG, are presented.

Basic notions of metric-based mesh generation

The mesh is modeled as a continuous medium: at the continuous level, we consider mesh elements being represented by ellipsoids (ellipses in 2D). In this geometric representation, the size of the element is its volume (area in 2D), its shape is associated with the ratio of the lengths of its semi-axes and its orientation is provided by its principal axis vectors [107]. Therefore, the control of the element size can be achieved by specifying a metric tensor $\mathcal{M}(\mathbf{x})$ to prescribe the size, shape and orientation of mesh elements over the whole domain. $\mathcal{M}(\mathbf{x})$ is a $d \times d$ symmetric positive definite matrix, with d the number of dimensions of the problem. This metric tensor guides the generation of a quasi-uniform mesh of Ω in the metric $\mathcal{M}$.

In the continuous framework a metric tensor $\mathcal{M}$ is a continuous element [191], that can be geometrically represented by its unit ball. A discrete element K is said to be unit with respect to a continuous element $\mathcal{M}$ if the length of all its edges is unit in the metric $\mathcal{M}$. The standard Euclidean scalar product is then modified using a proper metric tensor field, in order to prescribe a unit edge length

$$l_{\mathcal{M}}(\mathbf{e}) = ||\mathbf{e}||_{\mathcal{M}} = \sqrt{{}^{\mathrm{T}}\mathbf{e}\mathcal{M}\mathbf{e}} = 1 \ , \tag{3.1}$$

where $l_{\mathcal{M}}(\mathbf{e})$ is the distance between the extrema of the vector $\mathbf{e}$. The set of points satisfying the relation above describes an ellipsoid in three dimensions (ellipse in two dimensions), for which the length of each semi-axis is given by the square root of the inverse of the corresponding eigenvalues of $\mathcal{M}$. As the metric is not constant in an element but varies across the domain, after parametrizing the curve $\mathbf{e}(s) : [0,1] \rightarrow \mathbb{R}^d$ the average length of a mesh edge $\mathbf{e}$ is introduced:

$$l_{\mathcal{M}}(\mathbf{e}) = \int_0^1 \sqrt{{}^{\mathrm{T}}\mathbf{e}\mathcal{M}(s)\mathbf{e}}\, ds \ . \tag{3.2}$$

The desired adapted mesh is then a unit mesh, i.e. a mesh such that for each edge $l_{\mathcal{M}}(\mathbf{e}) \simeq 1$. In practice, the remesher builds a mesh such that the edges lengths are close to 1. In particular every edge of the mesh l verifies: $\frac{1}{\sqrt{2}} < l < \sqrt{2}$.

In order to deal with a single metric at the vertices, metric intersection and interpolation procedures are then employed. For a more complete treatment of these specific subjects the reader should refer to [129, 107].

The metric tensor can be diagonalized and decomposed as:

$$\mathcal{M}(\mathbf{x}) = \mathcal{R}(\mathbf{x})\,\Lambda(\mathbf{x})\,\mathcal{R}^{\mathrm{T}}(\mathbf{x}) \tag{3.3}$$

where $\mathcal{R}$ is the square matrix whose i-th column is the eigenvector $(\mathbf{v}_i)_{i=1,..,d}$ of $\mathcal{M}$ and Λ is the diagonal matrix whose diagonal elements are the corresponding eigenvalues $\Lambda_{ii} = \lambda_i$. The matrix Λ thus prescribes the size and shape of the element, while the matrix $\mathcal{R}$ prescribes its orientation.

The element sizes are then linked to the eigenvalues by $h_i = 1/\sqrt{\lambda_i}$ with $i = 1, ..., d$. The metric tensor can be rewritten as:

$$\mathcal{M}(\mathbf{x}) = \mathcal{D}^{2/d}(\mathbf{x})\mathcal{R}(\mathbf{x})\,\zeta(\mathbf{x})\,\mathcal{R}^{\mathrm{T}}(\mathbf{x}) \tag{3.4}$$

where the density function is defined as $\mathcal{D}(\mathbf{x}) = \sqrt{\det\mathcal{M}(\mathbf{x})} = \left(\Pi_j^d \lambda_j\right)^{1/2} = \left(\Pi_j^d h_j\right)^{-1}$. The anisotropic quotients tensor $\zeta(\mathbf{x})$ is defined as a diagonal matrix with the d anisotropic quotients equal to $r_k(\mathbf{x}) = h_k^d \left(\Pi_j^d h_j\right)^{-1}$ as diagonal entries.

For an isotropic metric the semi-axes of the ellipsoid (ellipse in 2D) have the same length, and the d eigenvalues associated with the ellipsoid coincide, i.e. $h_1 = h_2 = h_3 = h_n$, the *j-th* eigenvectors are simply the *j-th* coordinate vectors, and the anisotropic quotients are all unitary. We obtain then the sphere (and the circle in 2D) of radius 1.

For a thorough review about metric-based generation theory, the reader can refer to [141, 129, 130, 191].

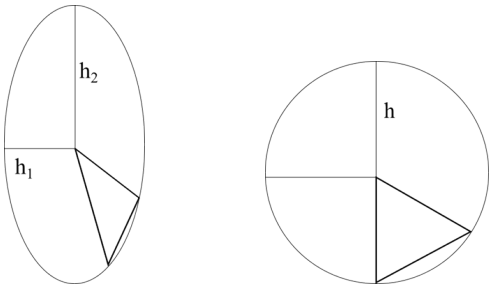


Figure 3.2: Examples of unit elements with respect to a continuous element in 2D. Anisotropic (left) and isotropic (right) metric.

Metric-based anisotropic mesh adaptation

Concerning anisotropic mesh adaptation, the most common strategy, typically used in FE/FV contexts, consists in computing a metric based on the Hessian matrix of the solution [73, 129, 192].

One of the most popular approaches for second order methods is the multi-scale metric method developed by Loseille & Alauzet [191, 192]. Using their approach, the

error on the mesh is minimized analytically, using calculus of variations. The resulting multi-scale metric field is found to be the optimal, under a specified target complexity. The multi-scale method has been successfully applied to a wide range of applications, including RANS simulations of the ONERA M6 wing, the NASA Common Research Model (CRM) high lift configuration and the JAXA Standard Model (JSM) [30, 209, 211, 257]. The method has been successively extended in order to control the error in an output functional in [194]. In previous works, Formaggia *et al.* [125] and Venditti & Darmofal [292] had already made the first attempts to combined output-error indicators with the Hessian of the solution. The multi-scale anisotropic mesh adaptation approach by Loseille & Alauzet was as well extended to unsteady flows and moving geometries in [35, 21] using a fixed-point algorithm, and was employed by Park *et al.* [234] for hybrid RANS/LES adaptive simulations.

In the context of DG methods, the Hessian-based metric was used by Remacle *et al.* [248] and by Alauzet *et al.* [18] using second-order discretizations ($p = 1$) to prescribe the anisotropy to the element. For higher order DG discretizations, Remacle [247] exploited the Hessian as a correction indicator to compute the direction of shocks and other discontinuities, after a smoothness indicator was employed as a shock detector. Later, Ceze & Fidkowski [76], based on previous findings [122] that the directions obtained from the Hessian often correlate reasonably well with directions obtained from approaches using higher-order derivatives, employed the information from the Hessian, for high-order DG schemes, combining it with output-based error estimates, for the simulation of laminar and RANS configurations.

However, except for some approximations, this strategy is not directly applicable to higher orders, as the standard Hessian matrix approach found in the literature for low-order methods is based on a linear approximation of a scalar quantity. The error on a piecewise linear interpolation of a smooth function is bounded by a quadratic term involving the second derivatives (the Hessian) of the function. The definition of a two-dimensional metric tensor is more difficult for high-order methods, where the interpolation error depends on a term involving the $(p+1)^{th}$ derivative of the solution. This complication is the reason why the 3D metric-based anisotropic mesh adaptation for high-order methods is today still an open research topic, and only few works with realistic 3D applications have been published.

An attempt to extend anisotropic mesh adaptation to high-order methods was made by Pagnutti *et al.* [227], where the authors developed a method to calculate a metric for arbitrary p using a Fourier series representation of $p+1$ order terms. While the method should be valid for any order of accuracy in both two and three dimensions, they focused only on third order schemes in two dimensions in [227] and the method was not extended to 3D/higher-order methods.

Coulaud & Loseille [94] extended the multi-scale approach of Loseille & Alauzet [192] to high-order numerical methods, by controlling the high order interpolation error. The method is based on an iterative algorithm to derive a local optimal metric to approximate the interpolation error to a quadratic form. This optimal local metric is then globally optimized via a calculus of variation to obtain the optimal distribution of the degrees of freedom. This method however was initially applied only to 3D analytical functions, and successively it was employed for high-order mesh generation [119, 252],

rather than anisotropic mesh adaptation.

Carabias *et al.* [69] proposed to use a least-square method to fit a third-order error into a quadratic term. An optimization problem was then analytically solved to obtain the optimal mesh metric distribution. The method was applied to an acoustic propagation benchmark. However, the method was not extended to 3D and/or viscous configurations due to the high computational cost of their CENO third-order approximation scheme, and the authors intended to switch to a different numerical scheme, as for example DG.

Shakoor *et al.* [266] developed a method directly approximating the third-order error as a quadratic form through a geometric averaging operation. The method was applied to 2D and 3D inviscid configurations but was limited to quadratic FEM approximations of the solution, without extending it to higher-order approximations.

The most fruitful approaches for high-order anisotropic metric-based mesh adaptation up to date appear to be the Mesh Optimization via Error Sampling and Synthesis (MOESS) initially developed by Yano & Darmofal [312], and the methods based on the error model initially developed by Dolejší [110] for *hp*-adaptation.

At first, Yano *et al.* [314] applied the fixed-fraction Mach-anisotropy method (FFMA), a modification of the algorithm developed by Fidkowski & Darmofal [122] to 2D subsonic, transonic and supersonic RANS configurations. Their adaptation strategy was based on an output-based error estimate, to be minimized for a given cost. In 2D, they set the anisotropy stretching ratio equal to the $p+1$ root of the ratio between the maximum derivative of the Mach and the derivative normal to this direction. However, the search over all directions to determine the maximum $p+1$ derivative does not appear to be extensible in a straightforward manner to three dimensions.

In a later work, Yano & Darmofal [312] developed a unified framework for anisotropic *h*-adaptation for problems with localizable error estimates. They solved a continuous constrained optimization problem of the tensor valued field corresponding to a Riemaniann metric tensor using the mesh-metric duality by Loseille & Alauzet [192]. In particular their adaptive method iterates towards a mesh minimizing an error functional while constrained by a maximum cost functional, by performing local error sampling steps. The anisotropic adaptation is driven by an output-based error estimate and the cost functional is represented by the maximum allowed number of degrees of freedom.

They tested this optimization algorithm for advection–diffusion problems [312], for inviscid and laminar simple configurations [311] and for 2D/3D boundary layers problem [313]. They presented as well results on a 3D laminar configuration and on 2D RANS simulations of the flow past an RAE 2822 airfoil and multi-element airfoils in Yano's PhD thesis [313], highlighting the overall better performance of the newly developed MOESS approach as compared to the previous FFMA approach.

However, despite the perfect scalability of the adaptation algorithm and its suitability for multi-process computing, they measured that the time spent on error estimation and adaptation was a significant fraction of the flow solve time (60% for the NACA 0006 Euler shock propagation problem and %16 for the transonic RANS RAE 2822). Although they observed a decrease of the relative cost with the increase of the complexity of the test case, this aspect could realistically lead to a very high computational cost for industrial problems requiring a significantly large number of dofs.

The MOESS algorithm was extended to two dimensional inviscid and laminar unsteady flows by Krakos *et al.* [173] and by Doetsch *et al.*[108], and was employed by Carson *et al.* [71] to simulate 3D laminar configurations.

Focusing now on the second mentioned approach for high-order anisotropic mesh adaptation, the error model developed by Dolejší [110] is a generalization for high-order of the multi-scale metric method developed by Loseille & Alauzet for second-order methods in [191]. While the author presented this approach in the context of *hp*-adaptation, we focus here only on the *h*-adaptation part of the methodology. The approach consists in a number of steps, which aim at minimizing the interpolation error. The first step consists in computing an estimate of the $(p+1)^{th}$ order derivatives of the solution. This is computed by evaluating a higher-order patch reconstruction for each element by exploiting the information over all its neighboring elements (see [113] for a detailed discussion). The computed derivatives are then used to identify the optimal anisotropy (orientation and aspect ratio). Finally, the mesh sizes are defined by an analytical formula which corresponds to the equidistribution of the interpolation error for a given bound.

In recent work, Rangarajan *et al.* [239, 243] improved the approach modifying only the definition of the mesh sizes by solving a global optimization problem, which presents an analytical solution, for a given number of dofs. This approach was further extended to goal-oriented adaptation in [244, 242], by including a suitable adjoint-based error estimate in the error model.

In a similar manner, Balan *et al.* [34] based the definition of the error size on adjoint estimates, while employing the error model proposed by Dolejší [110] for the computation of the local anisotropy. With this approach, the area of the mesh elements is reduced or increased according to a fixed-fraction marking criterion. A user-specified mesh fraction is used to define the reference error level. The elements with errors higher than the reference are refined and those with errors lower than the reference are coarsened using an heuristic formulation of the new mesh sizes depending on their local error.

These approaches have been tested on a variety of 2D and 3D configurations, ranging from analytical problems and simple inviscid and laminar configurations [110, 34, 239, 243], with more complex RANS configurations recently reported in [241].

Despite the very promising results, these methods have been developed for steady configurations, and they do not appear to be extensible in a straightforward manner to unsteady turbulent configurations.

Dolejší's error estimates have been employed to develop as well *hp*-adaptive strategies, as will be presented in Section 3.3 dedicated to *hp*-adaptation techniques.

Metric-based isotropic mesh adaptation

Isotropic approaches are present in the literature mostly in the context of unsteady complex applications.

Bernard *et al.* [47] used a dynamic mesh adaptation strategy for ocean modelling, solving two saddle point optimization problems that can be solved in a closed form for constant polynomial degree. The first minimizes the global error in the domain while keeping constant the number of elements. The second is based on minimizing the

number of elements while keeping the global error constant in the domain. They do not perform node repositioning but only edge swap, edge collapse and edge split.

In the context of LES, Daviller *et al.* [98] used a sensor on the time-averaged dissipation of kinetic energy, provided as field function to the remesher, and applied the adaptive procedure to a swirler with two counter-rotating passages. Agostinelli *et al.* [16] generalized the original contribution proposed by Daviller to complex turbulent reacting flows. Odier *et al.* [225] applied the approach by Daviller for complex Wall Modeled turbomachinery flows. For Wall Modeled simulations, they modified the definition of the metric to account for both the losses occurring out of the boundary layer, and a good prediction of the boundary layer, by imposing a different metric at wall cells, while using Daviller's original metric in other cells.

The strategy developed by Benard *et al.* [45] defined the new mesh sizes in order to respect two criteria: one minimizes a quantity depending on the second derivatives of the time-averaged flow velocity and one ensures that a sufficient part of turbulent scales is explicitly resolved. Their strategy was initially applied to the simulation of the isothermal flow in a complex meso-combustor, and more recently to wind turbine wakes [319] as well as to a lean-premixed semi-industrial burner [215].

In figure 3.3 we show an example of the two main h-adaptive strategies on the lid-driven cavity test case (the Reynolds numbers are different, but the flow characteristics are very similar). The h-adapted meshes with a quadrilateral-hanging node approach by Offermans *et al.* [226] is in the left panel and the mesh obtained with a simplicial metric-based approach is depicted in the right panel. The metric-based h-adaptation is performed with the h-adaptive strategy developed during this thesis.

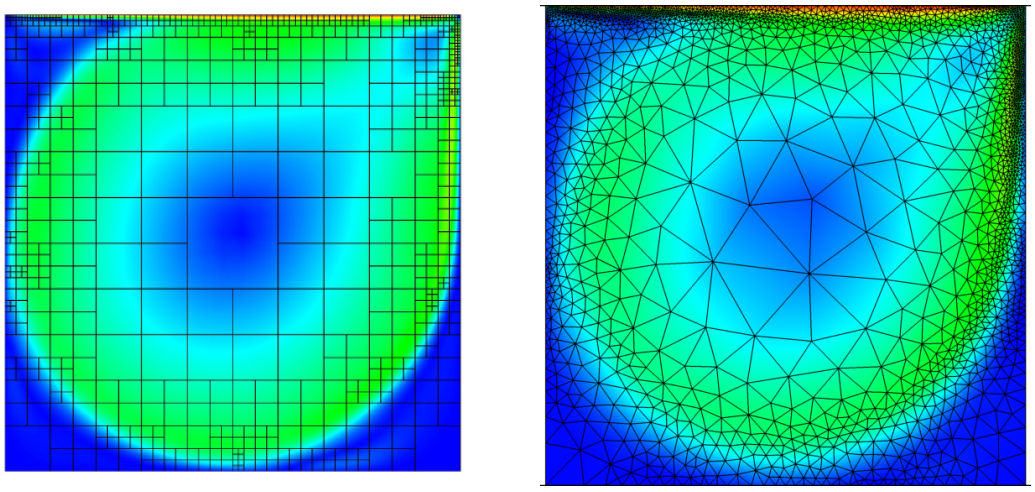


Figure 3.3: Example of mesh adaptation for the lid driven cavity. On the left the mesh obtained with isotropic elements splitting-based mesh adaptation by Offermans *et al.* [226] at $Re = 7500$ with the flow solver Nek5000. On the right, the mesh obtained with an isotropic metric-based adaptive strategy at $Re = 1000$, performed with the flow solver CODA and the h-adaptive strategy developed in this work.

MMG library

The remeshing library used in this work is MMG. MMG is an open source software for simplicial remeshing [6, 97], supporting both 2D and 3D remeshing (MMG2D and MMG3D). The MMG library has been already used extensively in the context of FV and FE methods [98, 45, 61, 259]. We provide a short description of the MMG mesh adaptation and optimization algorithm.

MMG's goal is to remesh $\mathcal{T}_{K,i}$, where i is the index corresponding to the mesh adaptation step, into a new mesh $\mathcal{T}_{K,i+1}$, which is a close approximation of the computational domain Ω, adapted to a specified local size feature. From the user's point of view, MMG takes as inputs a mesh $\mathcal{T}_{K,i}$ and a discrete metric map $\mathcal{M}^*_{i+1}$ defined at each vertex. We employ the superscript * to remind that the desired metric field $\mathcal{M}^*_{i+1}$ is prescribed by the user on the mesh $\mathcal{T}_{K,i}$, even though the real metric associated with the mesh $\mathcal{T}_{K,i+1}$ is $\mathcal{M}_{i+1}$, which is different due to quality constraints. These constraints need to be satisfied by MMG, in addition to reproducing the prescribed metric field. MMG defines a continuous metric field using interpolation schemes, local modifications and quality checks. It then outputs the mesh $\mathcal{T}_{K,i+1}$ with a corresponding metric field $\mathcal{M}_{i+1}$, which is an approximation of the desired metric $\mathcal{M}^*_{i+1}$.

The approach MMG uses for three-dimensional domain remeshing is a local, iterative remeshing procedure. It consists in conducting local modification operations both on the surface and volume parts, which affect very limited areas of the meshes at hand, and a sequence of meshes is produced, until it converges toward the final optimal mesh $\mathcal{T}_{K,i+1}$. Among the local mesh modifications that the software performs inside the volume there are edge split, edge collapse, edge swap and node relocation.

Regarding the handling of boundaries, MMG locally reconstructs the "ideal" surface from the discrete geometry of the input mesh, using cubic Bézier triangles. The Hausdorff distance, defined as the distance between the ideal and the discrete mesh surface is controlled in order to ensure a good boundary approximation.

The parameters needed by MMG to perform the adaptation are listed below :

- h_{max}: the maximum allowed size of an edge in $\mathcal{T}_{K,i+1}$;
- h_{min}: the minimum allowed size of an edge in $\mathcal{T}_{K,i+1}$;
- h_{grad}: the gradation value which controls the ratio between two adjacent edges (for a given gradation of h_{grad}, two adjacent edges h_1 and h_2 in $\mathcal{T}_{K,i+1}$ must respect that $\frac{1}{h_{grad}} \leq \frac{h_1}{h_2} \leq h_{grad}$);
- h_{Hausd}: controls the geometric approximation of the boundaries $\partial\mathcal{T}_{K,i+1}$. It imposes the maximal distance between the piecewise linear representation of the boundary and the reconstructed ideal boundary.

The parameters h_{max}, h_{min} and h_{Hausd} depend on the length scale of the problem, while h_{grad} controls the quality of the mesh. The lower the gradation, the better the mesh quality. However, higher gradation values impose less constraints, and allow for a better match between the prescribed metric $\mathcal{M}^*_{i+1}$ and the actual metric $\mathcal{M}_{i+1}$ computed by MMG, without introducing extra elements.

A more detailed explanation of MMG features can be found in [97].

3.2 *p*-adaptation techniques

p-adaptation consists in modifying the local polynomial degree for the approximation of the solution within the element, while not requiring a modification of the topology of the mesh. It provides a faster convergence than *h*-adaptation for smooth flows (the error is proportional to h^{p+1}, $\epsilon = O(h^{p+1})$), while it is not desirable for non-smooth flows where the convergence rate is fixed by the singularities ($\epsilon = O(h^1)$). In the case of discontinuities, DG methods employing high polynomial degrees may suffer from stability issues, for which usually shock capturing techniques are applied while potentially deteriorating the convergence order of the solution.

The error estimators can often be used indistinctly for *p*- or *h*-adaptation, and the selected criteria for adapting the mesh are usually very similar to those employed for element splitting-based mesh adaptation, which are more straightforward than for metric-based *h*-adaptation. In fact, the fixed-fraction marking strategy can be easily used, given the exact prediction of the number of the dofs in the new mesh when using *p*-adaptation. Moreover, the choice of the new degree of the polynomial in the element constitutes in general a more constrained and less flexible choice with respect to selecting a precise size for each element. This can be seen both as a limitation with respect to *h*-adaptation, and a strength, allowing to refine the mesh in a smoother and more robust way. This might be preferable in some cases as for example for scale-resolving simulations.

However, the lower degree of flexibility of *p*-adaptation with respect to *h*-adaptation may lead to the necessity of sufficiently fine initial meshes.

Another complication introduced by *p*-adaptive techniques is the requirement of *ad hoc* load balancing treatments in the flow solver. These need to take into account, in the mesh partitioning process, the variable computational load due to the non-uniform polynomial degree distribution, and not only the equidistribution of the elements, as done by classical mesh partitioners.

Applications to steady inviscid and laminar computations can be found in the literature [256, 219, 116, 172]. Ekelschot *et al.* [116] in particular highlighted the necessity of implementing *h*-adaptive techniques in the vicinity of shocks and strong discontinuities.

Despite these applications, its faster convergence rate for smooth flows, together with the consistent reduction of dissipation and dispersion errors that high-orders of accuracy enable, make *p*-adaptation particularly interesting especially for scale-resolving simulations of turbulent flows, for which they are mainly employed.

In the context of LES/ILES simulations, pure *p*-adaptation has been successfully applied to DG methods [81, 100, 14, 218, 300] and is more popular than *h*-adaptation, which is usually preferred for FV and FE methods [216, 164, 303, 282, 98, 45].

Static *p*-adaptation of unsteady flows was performed by Chapelier *et al.* [81] on a channel flow at friction Reynolds number $Re = 392$, by Tugnoli *et al.* [284] on the flow around a square section cylinder at Reynolds number $Re = 22000$ and in the flow over periodic hills at $Re = 2800$, by Naddei [218] on the transitional flow past a NACA0012

airfoil at $Re = 50000$, and by de la Llave Plata *et al.* [100] and Bassi *et al.* [38] on the flow past a circular cylinder at Reynolds numbers respectively $Re = 140000$ and $Re = 3900$. Abbà *et al.* and Wang *et al.* [14, 300] followed a dynamic approach for the simulation of turbulent flows, such as the temporally evolving mixing layer, the interaction of a vortex and a square cylinder [14], and the transitional flows over a cylinder with $Re = 3900$ and the SD7003 wing with $Re = 60000$ [300].

Two meshes obtained respectively with p- and metric-based h-adaptation are compared in figure 3.4. The test case is the flow past a circular cylinder at $Re = 40$. The polynomial degree distribution obtained with p-adaptation by Naddei *et al.* [219] is depicted in the left panel. Their adaptive strategy was controlled by the SSED error estimator, described in equation (3.6) with a variable polynomial degree from $p = 2$ to $p = 6$. The h-adapted mesh in the right panel has been obtained with the metric-based h-adaptive strategy developed in this work with a fixed polynomial degree $p = 2$.

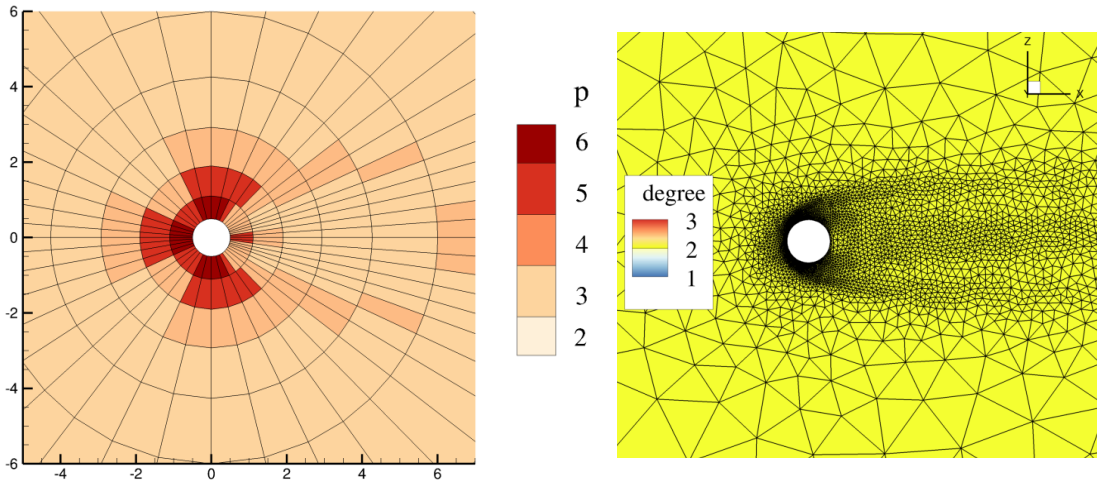


Figure 3.4: Example of spatial resolution adaptation for the circular cylinder at $Re = 40$. On the left the mesh obtained with p-adaptation by Naddei *et al.* [219] with the flow solver AGHORA with variable $p = 2 - 6$. On the right, the mesh obtained with h-adaptation, performed with the flow solver CODA and the h-adaptive strategy developed in this work with fixed $p = 2$.

3.3 *hp*-adaptation techniques

Regarding *hp*-adaptivity, the local error estimator is not sufficient to guide the adaptive algorithm. While the error estimator is still used to identify the elements requiring improved resolution, an additional strategy needs to be defined to identify whether h-refinement or p-enrichment should be employed. This is usually referred to as the *hp*-decision strategy. Generally, this choice is made according to an estimate of the smoothness of the solution within an element. If the solution is sufficiently smooth, the adaptive

algorithm opts for p-enrichment, while non-smooth regions are h-refined. Thanks to hp-adaptation, the best of the two approaches are combined. While theoretically achieving exponential convergence (with respect to h-adaptive methods which converge with an algebraic rate), the constraint on the initial mesh typical of p-adaptive methods are relaxed [105]. Moreover, pure p-adaptation needs the use of very high polynomial degrees to ensure a sufficient level of flexibility, which can be cumbersome to optimize in terms of load balancing and cost per dof, and can lead to strong numerical instabilities. Hence the interest of employing an hp-technique for DG schemes (applied to LES), where the use of very high order polynomials can be replaced by the possibility to adapt also the size of the elements. As already mentioned for pure p-adaptation, the implementation of *ad hoc* load balancing treatments is necessary also for hp-adaptation. In their absence, high levels of load imbalance might offset any computational time reduction produced by the locally adaptive approach, and significantly increase the computational cost of simulations.

Several hp-decision indicators have been proposed in the literature.

In the context of octree-based mesh adaptation, Gui & Babuška [148], followed by Adjerid *et al.* [15], assessed the solution smoothness thanks to the ratio between error estimates based on p and $p-1$ approximations. Houston *et al.* [155] developed an hp-algorithm based on the estimation of the local Sobolev regularity index of a given function, by monitoring the decay rate of its Legendre expansion coefficients. These two approaches, however, cannot be applied to $p = 1$ computations because they are based on lower-order estimates.

Mavriplis [202] determined the local smoothness of the solution by computing the decay rate of the Legendre expansion coefficients of the solution under the assumption that, for non-smooth solutions, the discontinuities in the solutions deteriorate this decay rate. Leicht *et al.* [182] and Chalmers *et al.* [78] extended this approach to multiple dimensions, the first for 2D/3D inviscid, laminar and RANS flows, the second for unsteady laminar simulations.

A sensor based on a measure of the inter-element jumps of quantities such as the density, pressure and velocities in the solution, was used by Burgess and Mavriplis [66] and by Wang *et al.* [301] for respectively compressible laminar and inviscid 2D cases. The latter proposed to couple the jump-based sensor with a criterion based on the ratio of the flow quantity obtained by using the truncated expansion of modal coefficients, to the one obtained with the full expansion [236], in order to assess the smoothness of the solution.

The aforementioned approaches (and the related ones) have been largely exploited in the literature in later works on DG hp-adaptation [143, 117, 33, 308, 231] in several fields of application. However they usually require the manual setting of different parameters that can often be case-dependent, especially in the definition of the threshold between smooth and non-smooth behavior of the solution.

In [137], Gao & Wang proposed to identify the non-smooth regions with an accuracy-preserving Total Variation Diminishing marker, presenting no case-dependent parameters.

In the work by Ceze & Fidkowski [75], the choice between h- and p-adaptation, as well as the anisotropy prescription, is made by optimizing a merit function that incor-

porates both an output sensitivity and a measure of the solution cost on the new mesh, without performing smoothness measurements. The cost was estimated considering two measures of CPU work, the number of degrees of freedom and the number of floating point operations. The adaptive strategy was validated on both 2D and 3D flows, for inviscid, laminar and RANS configurations.

In [212], Mitchell and McClain summarized several strategies proposed over the years for the determination between *h*- and *p*-adaptation.

In the context of remeshing adaptation strategies, previously described in Section 3.1.2, the methodologies presented by Dolejší, Balan, Rangarajan *et al.* [110, 34, 112, 240, 241] can be modified to produce anisotropically adapted *hp*-meshes. For this purpose the anisotropic features as computed by the error model of Dolejší, and the optimal mesh size are computed by considering for each element a modified local polynomial degree $(p-1,\ p,\ p+1)$. The polynomial degree to employ for the target resolution is then defined either as the one which provides the minimum number of dofs for a given error [112, 240] or the minimum bound of the interpolation error [34]. We remark however that when including a variable polynomial degree, the optimal density distribution cannot be computed analytically, but requires an iterative solution process.

These adaptation methodologies were validated by performing numerical experiments, including boundary layer problems, as well as compressible flow test cases such as inviscid and laminar configurations, and RANS simulations of the turbulent flow past a multi-element airfoil.

To the best of the authors' knowledge, the presented approaches (and the related works) relying on Dolejší error estimates, constitute the only study in the literature of *hp*-adaptive techniques for metric-based mesh adaptation.

In figure 3.5, we present a comparison between the spatial discretizations obtained with *p*- (left panel) and *hp*-adaptation (right panel) of the flow past a sphere at $Re = 200$. The *p*-adaptation has been performed by Rueda-Ramírez *et al.* [256] on a fixed mesh with a variable polynomial degree from $p = 2$ to $p = 7$ with an anisotropic approach (in the legend N_{av} is the average polynomial degree in the three directions). The *hp*-adapted mesh on the right has been obtained with the approach developed in this work, with the polynomial degree ranging from $p = 1$ to $p = 3$ (presented in Chapter 5).

3.4 *a posteriori* error estimators

The error estimators driving adaptive strategies that can be found in the literature mainly belong to three main groups:

- Discretization error-based indicators formally measure the difference between the numerical and the exact solution. The discretization error is mainly caused by the interaction between the spatial discretization scheme, the mesh resolution, the mesh quality, and the behavior of the solution itself and its derivatives. They include estimates of higher-order solutions and estimates based on the residual, such as the truncation error [255].

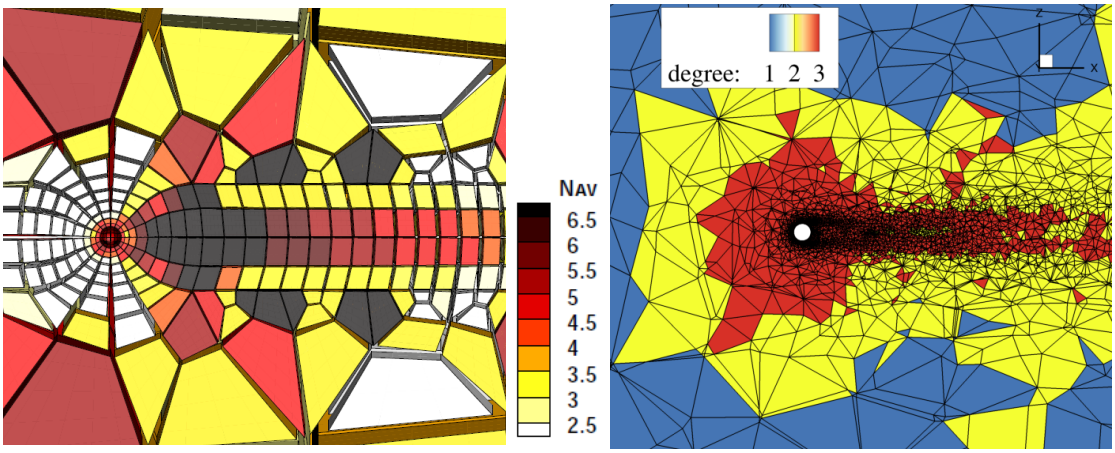


Figure 3.5: Example of spatial resolution adaptation for the sphere at $Re = 200$. On the left the polynomial distribution obtained with anisotropic p-adaptation by Rueda-Ramírez *et al.* [256] with $p = 2 - 7$. On the right, the mesh obtained with hp-adaptation with a variable $p = 1 - 3$, performed with the flow solver CODA and the adaptive strategies developed in this work.

- Feature-based indicators identify particular physical phenomena which need to be accurately captured, as for example shock, boundary layer or vortex detection [236, 87, 166].
- Goal-oriented error indicators identify regions in which a lack of resolution influences the most a scalar output function, which in aeronautic applications is often the lift or the drag coefficient. This is performed by solving an adjoint problem [123, 152].

Despite being inexpensive and simple to implement, feature-based estimators usually lack robustness and require some experience from the user in the studied phenomenon. On the other hand, adjoint-based mesh adaptation, even if in the last twenty years has been shown to be successful in a large range of steady applications [291, 232, 193, 240] and unsteady Euler and low Reynolds number flows [199, 44, 121, 108], only very recently has become an option for turbulent scale-resolving simulations [40]. In fact, in typical unsteady adjoint approaches, a backward-in-time integration is required, which is considerably expensive for three-dimensional simulations from a memory requirements and computational cost point of view. However, those mentioned recent studies rely on entropy-adjoint error estimates for moderate Reynolds number flows, while goal-oriented mesh adaptation to minimize the error in specific target quantities such as lift and drag still seems out of reach in an industrial context for scale-resolving simulations.

In the next subsections we list some of the discretization error-based and feature-based error estimators found in the literature and useful for the purpose of this manuscript. This is not intended as a comprehensive overview of these classes of *a posteriori* error estimators, for which the reader is redirected to [219] and the cited works.

3.4.1 High order-based error estimators

This class of error estimators relies on the assumption that the error on the high-order polynomial approximation of the solution is strictly related to the content of the highest-order mode. The estimators presented in the next paragraphs are completely local and require only the knowledge of the polynomial expansion within the element.

Small Scale Energy Density (SSED)

This error estimator is based on an approximation of the error $\mathbf{E}_K$ of the numerical solution $\mathbf{u}$ with respect to the exact solution $\mathbf{u}_{ex}$ in the element by computing the norm of the difference between the numerical solution $\mathbf{u}_{h,p}$ and the projection of the numerical solution on the reduced-order space $\mathcal{V}_h^{p-1}$, namely $\mathbf{u}_{h,p-1}$:

$$\mathbf{E}_K = ||\mathbf{u}_h - \mathbf{u}_{ex}|| \simeq ||\mathbf{u}_{h,p} - \mathbf{u}_{h,p-1}|| \tag{3.5}$$

The error based on the norm of the high-order modes of the momentum vector is normalized by the volume of the element (as done by Naddei *et al.* [219] for laminar and LES p-adaptive simulations). This has been found to provide better results as compared to the classical SSED indicator not normalized by the volume proposed in [175] (LES-VMS simulations) when a mesh with large variation in element size in the domain is employed. The final formulation reads:

$$\epsilon^2_{\text{SSED},K} = \frac{\int_K ||(\rho\mathbf{v})_{h,p} - (\rho\mathbf{v})_{h,p-1}||^2 dV}{|K|} = \frac{||(\rho\mathbf{v})_{h,p} - (\rho\mathbf{v})_{h,p-1}||^2_{L^2(K)}}{|K|}\,, \tag{3.6}$$

where $(\rho\mathbf{v})_h$ is the momentum vector and $|K|$ is the volume of the element.

Spectral Decay (SD)

Persson and Peraire first introduced in [236] the Spectral Decay estimator as a shock capturing sensor in the formulation presented in Chapter 2, in equation 2.41. It was then successively used by Tumolo *et al.* [285] for p-adaptive simulations for the shallow water equations. The Spectral Decay indicator, following the approach of Tumolo *et al.* [285] and Naddei *et al.* [219] is defined as:

$$\epsilon^2_{\text{SD},K} = \frac{\int_K ||\mathbf{u}_{h,p} - \mathbf{u}_{h,p-1}||^2 dV}{\int_K ||\mathbf{u}_{h,p}||^2 dV} = \frac{||\mathbf{u}_{h,p} - \mathbf{u}_{h,p-1}||^2_{L^2(K)}}{||\mathbf{u}_{h,p}||^2_{L^2(K)}}\,. \tag{3.7}$$

When applied to the momentum variable $\rho\mathbf{v}$ as done in [219], the SD estimator in equation (3.7) can be seen as a normalization of the SSED in equation (3.6), by the norm of all the modes of the momentum, corresponding to the total "energy" in the cell. Other variables can be used to estimate the error, other than the momentum, as for example the pressure, employed by Bassi *et al.* [38] for p-adaptation of the implicit LES of incompressible transitional flows.

Tugnoli *et al.* [284] and Flad *et al.* [124] employed a similar formulation with the norm of the momentum, but they removed the contribution of the cell-average in the computation of the total energy at the denominator.

3.4.2 Jump error estimators

This class of error estimators is based on the assumption that the exact solution is continuous across elements, except when a physical discontinuity is present in the flow configuration. Therefore a jump in the numerical solution can be considered as a measure of the error. These indicators only require the knowledge of the projection at the interface of the solution from the direct neighbors. They usually constitute a convenient indicator from a practical implementation point of view because they require the same jump information needed for the computation of the numerical fluxes.

Average jump error estimator

This estimator was used by Bernard *et al.* [47] for ocean modeling. The average value of the conservative variables at a given interface e in $\mathcal{E}_i$ is used to approximate the exact solution $\mathbf{u}^+_{ex}$:

$$\mathbf{u}^+_{ex} \simeq \frac{1}{2}(\mathbf{u}^+_h + \mathbf{u}^-_h) \,, \tag{3.8}$$

where $\mathbf{u}^+_h$ and $\mathbf{u}^-_h$ are respectively the traces of the variable $\mathbf{u}$ at the interface between elements K^+ and K^-. They then define the error $\mathbf{E}_e$ on the interface e as half the jump of the variable traces across the interfaces:

$$\mathbf{E}_e = ||\mathbf{u}^+_h - \mathbf{u}^+_{ex}|| \simeq \frac{1}{2}||\mathbf{u}^+_h - \mathbf{u}^-_h|| \tag{3.9}$$

and they consider an averaged error over each interface e, for the norm of the conservative variables:

$$\epsilon^2_{\text{JUMP},e} = \frac{\int_{\partial K_e} ||\mathbf{u}^+_h - \mathbf{u}^-_h||^2 dS}{4|\partial K_e|} = \frac{||\mathbf{u}^+_h - \mathbf{u}^-_h||^2_{L^2(\partial K_e)}}{4|\partial K_e|} \,. \tag{3.10}$$

The error indicator is then normalized by the area of each interface $|\partial K_e|$ (that in 2D is the length of the edge e, while in 3D is the area of the face e). For each element K they compute the error estimator by the following simple rule:

$$\epsilon^2_{\text{avgJUMP},K} = \frac{1}{N_e}|K| \sum_{e=1}^{N_e} \epsilon^2_{\text{JUMP},e} \tag{3.11}$$

with N_e the number of edges in 2D as in the original formulation [47], or the number of faces of the element in 3D.

The authors also use the relative error defined as

$$\epsilon^2_{\text{avgJUMP}_{rel,K}} = \frac{\epsilon^2_{\text{avgJUMP},e}}{2||\mathbf{u}_h||^2_{L^2(K)}} \,. \tag{3.12}$$

Maximum jump error estimator

The following discontinuity sensor was used at first by Krivodonova *et al.* [174] for shock detection and limiting for hyperbolic conservation law purposes, then by Gassner *et*

al. [139, 140] for spatial adaptation purposes. The estimator is defined as the maximum jump of a variable, such as the density or the pressure, at the element e interfaces

$$\epsilon_{\text{maxJUMP},K} = \max_e \max_j \left| \frac{(\mathbf{u}_h)^+(x_j) - (\mathbf{u}_h)^-(x_j)}{(\mathbf{u}_h)^+(x_j) + (\mathbf{u}_h)^-(x_j)} \right| \tag{3.13}$$

$$= \max_e \max_j \left| \frac{\left[\left[(\mathbf{u}_h)^+(x_j)\right]\right]}{2\left\{(\mathbf{u}_h)^+(x_j)\right\}} \right| , \tag{3.14}$$

where the x_j denotes the location of the j-th interpolation/quadrature point on the element interface e.

3.4.3 Combination of error indicators

Small-Scale Lifted indicator (SSL)

In his thesis work [218], Naddei improved the performances of the SSED indicator by proposing as refinement indicator the "Small-Scale Lifted" indicator. It consists in a combination of the SSED indicator in equation (3.6) and an additional term which approximates the error in the gradient of the momentum $\frac{h_K}{|K|^{1/2}} \sum_{i=1}^{d} \left\| \frac{\partial(\rho\mathbf{v})_{h,p}}{\partial x_i} - \frac{\partial(\rho\mathbf{v})_{ex}}{\partial x_i} \right\|$.

In order to compute the additional term, he employed an approach inspired by the BR1 and BR2 schemes of Bassi *et al.* [41] described in section 2.3.1, by approximating the exact gradient by the lifted gradient of the momentum. The resulting error estimator reads

$$\epsilon_{\text{SSL},K} = \epsilon_{\text{SSED},K} + \frac{h_K}{|K|^{1/2}} \sum_{i=1}^{d} ||\mathbf{i} \cdot \mathbf{L}_h^{\rho\mathbf{v}}||_{L^2(K)} , \tag{3.15}$$

with $\mathbf{i}$ the i-th unit vector defining the physical coordinate system. The second term includes information on the jumps of the solution at the interfaces of the elements.

In equation (3.15) the gradient term is scaled by the characteristic element size, defined as $h_K = |K|^{1/d}$. An alternative definition of h_K, better suited for general anisotropic meshes, was also proposed.

Maximum jump+Spectral Decay error estimator

Another example of combinations of two error estimators is the indicator used by Colombo *et al.* [92]. They observed by numerical experiments that an error estimator based on the jumps at the interfaces of the elements is reliable for any polynomial degree, whereas an error indicator based on the high-order modes, the SD in their work, is reliable only for $p \geq 2$. The SD estimator is accurate but can yield strong cell-to-cell variations, hence the interest of coupling such indicator with a jump error estimator. The latter identifies a larger region for adaptation and smooths out the overall indicator map. According to these observations, they implemented a combination of the two indicators.

The two estimators appear in the formulations respectively in equations (3.13) and (3.7), and are computed using the pressure variable.

Both indicators are normalized by their respective maximum and minimum values over the whole domain (min-max normalization) before the coupling:

$$\epsilon_{\text{SD+maxJUMP},K} = \text{INT}\left(\frac{p_K}{\max(2, p_K)}\right)\epsilon_{\text{SD},K} + \frac{1}{\max(1, p_K)}\epsilon_{\text{maxJUMP},K} \tag{3.16}$$

where p is the polynomial degree of the numerical solution and INT represents an integer division (meaning that $\epsilon_{\text{SD},K}$ is set to 0 for $p \in [0, 1]$).

Chapter 4

Validation on canonical test cases

This chapter is dedicated to the validation of the numerical methods and models described in Chapter 2, and to the study of the impact of several numerical parameters relevant in the context of RANS and scale-resolving simulations.

In Section 4.1 we verify the numerical error order of convergence of the DG method for the convection of an isentropic vortex, featuring various polynomial degrees and unstructured triangular grids.

In Section 4.2 we study the admissible near-wall resolution in terms of wall normal spacing for DG RANS from high Reynolds number flat plate simulations.

Finally, in Section 4.3, the influence of the DES constant C_{DES} is studied from canonical Taylor-Green vortex computations, featuring various polynomial degrees and unstructured tetrahedral grids.

4.1 Study on the DG order of convergence on unstructured meshes - Isentropic CoVo vortex case

The test case of a vortex convected by a uniform flow (referred to as CoVo in the following) is usually aimed at testing the capability of numerical methods to preserve vorticity in an unsteady inviscid flow. The accurate transport of vortices is indeed very important for LES and DES, as well as for rotorcraft unsteady applications. An analytical solution of the CoVo test case is available, allowing us to conduct an error analysis aimed at verifying the order of accuracy of the DG scheme implemented in CODA.

The "fast" configuration with $M_\infty = 0.5$ is chosen from the High-Order Workshop [12]. The vortex has a characteristic radius $R_c = 0.005$ and strength $\beta = 0.2$. The domain is a square of side $L_x, L_y = 0.1$, with periodic boundary conditions in the x and y directions, and the vortex is initialized with its center at $(x_c, y_c) = (0.05, 0.05)$.

The flow is initialized as:

$$\begin{aligned} T(\mathbf{x},t) &= T_\infty - \frac{U_\infty^2 \beta^2}{2C_p} e^{-r^2} \\ u(\mathbf{x},t) &= U_\infty \left(1 - \beta \frac{y-y_c}{R_c} e^{-r^2/2}\right) \\ v(\mathbf{x},t) &= U_\infty \beta \frac{x-x_c}{R_c} e^{-r^2/2} \end{aligned} \tag{4.1}$$

with the heat capacity defined as $C_p = R\frac{\gamma}{\gamma-1}$, the free-stream velocity $U_\infty = M_\infty\sqrt{\gamma R T_\infty}$ and the non-dimensional distance to the (initial) vortex core position $r = \frac{\sqrt{(x-x_c)^2+(y-y_c)^2}}{R_c}$. The fluid pressure p, temperature T and density ρ are prescribed such that the vortex is a steady solution of the reference free-stream flow (without the vortex):

$$\begin{aligned} \rho(\mathbf{x},0) &= \rho_\infty \left(\frac{T(\mathbf{x},0)}{T_\infty}\right)^{\frac{1}{\gamma-1}} \\ \mathrm{p}(\mathbf{x},0) &= \rho(\mathbf{x},0) R T(\mathbf{x},0) \end{aligned} \tag{4.2}$$

Relatively long simulations are usually performed, where the solution is advanced in time for many time periods, and the solution obtained at the last instant is compared to its analytical counterpart.

The time necessary for the vortex to advect through the computational domain is $T_{CoVo} = \frac{L}{U_\infty}$. In the following analysis, simulations are performed for $10T_{CoVo}$, lower than the $50T_{CoVo}$ suggested in the guidelines of the High-Order Workshop, but still representative to assess the dissipation and dispersion properties of the numerical scheme.

We present the errors measured on two quantities, the pressure and the entropy. The first is the L^2 norm of the relative error on the pressure, computed as:

$$error_{\text{pressure}} = \sqrt{\frac{\int_{\mathcal{T}_K} \left(\frac{\mathrm{p}-\mathrm{p}_{exact}}{\mathrm{p}_{exact}}\right)^2 dV}{|\mathcal{T}_K|}} \tag{4.3}$$

where p_{exact} is the analytical pressure, used also as initialization in equation (4.2), and $|\mathcal{T}_K|$ is the volume of the entire computational domain.

Moreover, in an Euler simulation without shocks, no sources of dissipation (other than numerical) are present, and the entropy is supposed to be equal to the entropy of the free-stream reference state. Defining the approximated entropy as $s = \frac{\mathrm{p}}{\rho^\gamma}$, we can compute the L^2 norm of the relative error on the entropy as:

$$error_{\text{entropy}} = \sqrt{\frac{\int_{\mathcal{T}_K} \left(\frac{s-s_\infty}{s_\infty}\right)^2 dV}{|\mathcal{T}_K|}} \tag{4.4}$$

where s_∞ is the approximated entropy of the infinite reference state.

For all the tested meshes, overintegration has been employed to benefit from a more precise computation of the integral error in the solution. Time steps are chosen small enough to measure dissipation and dispersion due to the spatial scheme and not due to the time integration.

The CoVo simulations are performed on six uniformly refined unstructured meshes counting respectively 112, 404, 1596, 6226, 24529, 96936 elements, for the three discretizations $p = 1$, $p = 2$ and $p = 3$. We remind that in 2D a $p = 1$, $p = 2$ and $p = 3$ element respectively contains 3, 6, 10 dofs, the number of dofs of our simulations ranges approximately from 300 to one million.

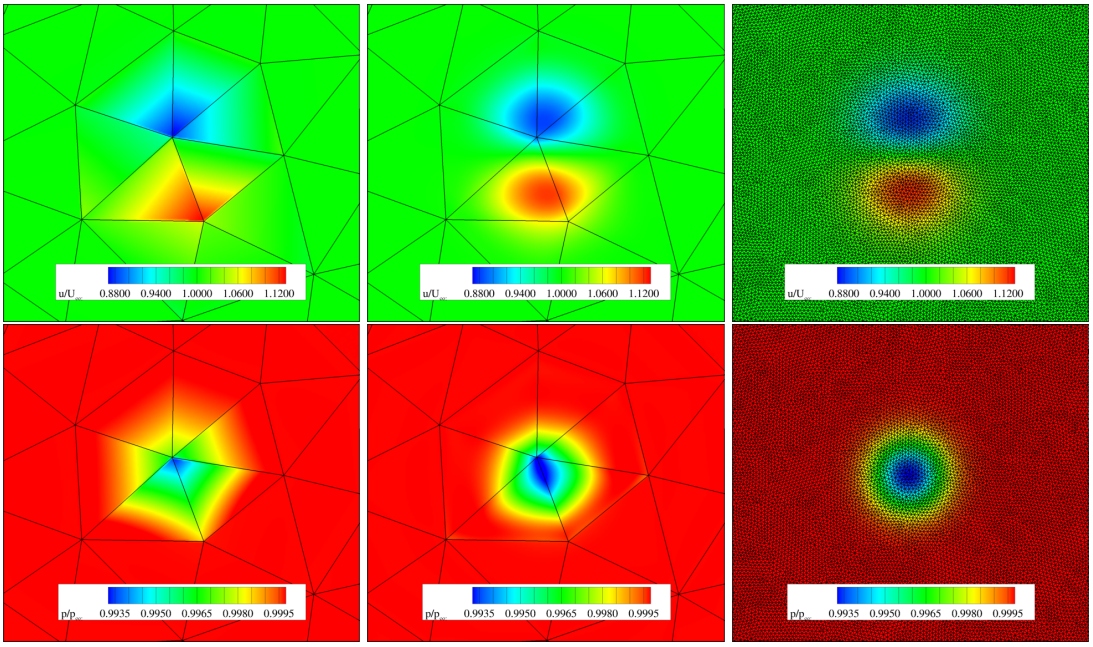


Figure 4.1: CoVo at $M_\infty = 0.5$. Initialization of the streamwise velocity (top row) and of the pressure (bottom row). Left column: coarsest mesh (112 elements) with $p = 1$ reconstruction (336 dofs). Middle column: coarsest mesh (112 elements) with $p = 3$ reconstruction (1120 dofs). Right column: finest mesh (96936 elements) with $p = 3$ reconstruction (969360 dofs).

In figure 4.1 a zoom of the initial condition of the CoVo in terms of the streamwise velocity and pressure are respectively presented in the top and the bottom panels. The left panels refer to the coarsest discretization (the first mesh with $p = 1$), the middle panels refer to the coasest mesh but with a $p = 3$ discretization, while the right panels refer to the finest discretization (the last mesh with $p = 3$). The visible mesh edges in figure 4.1 are those used for the computations, while the initial solutions are projected on a very fine post-processing mesh. This simple exercise shows how poor is the reconstruction of solution for the coarse $p = 1$ mesh. Here the interfaces between the triangular elements are clearly visible, and the low order $p = 1$ discretization on these very coarse elements provides a reconstruction which barely resembles, from a visual point of view, to the

prescribed initialization of the CoVo. On the other hand, we can see in the middle plots, how the fourth order $p = 3$ resolution provides a much more faithful representation of the CoVo, despite the very coarse mesh. The right panels present the solution on the finest available discretization, showing how the prescribed CoVo should look like with a very low interpolation error.

In figure 4.2 we present the error on the entropy, computed as in equation (4.4), and on the pressure, computed as in equation (4.3), for the six meshes, for $p = 1$ in blue, $p = 2$ in red and $p = 3$ in black solid lines. In dotted lines the theoretical second, third and fourth order are plotted respectively in blue, red and black.

First of all, we observe from figure 4.2 that on the coarsest mesh the simulations are still far from the asymptotic range of convergence, and similar error levels are obtained with different polynomial degrees.

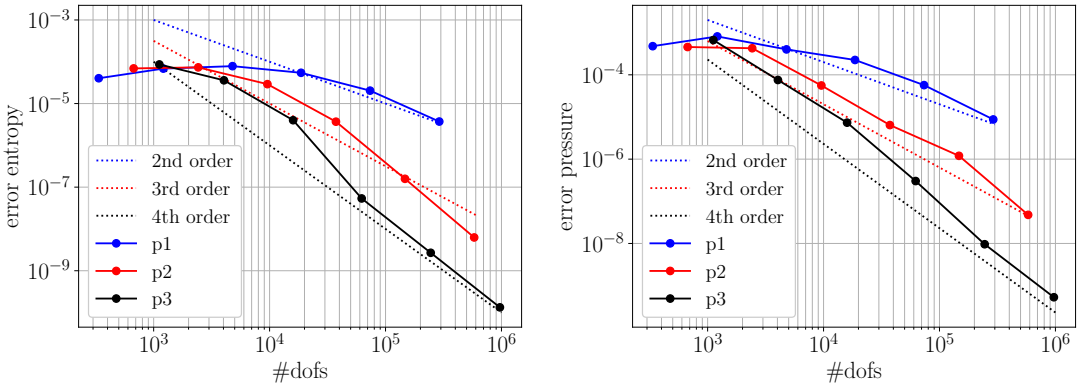


Figure 4.2: CoVo at $M_\infty = 0.5$. Error on entropy (left) and pressure (right)

We can notice that the asymptotic range has been achieved from the first mesh for $p = 3$ computations, while approximately from the second for $p = 2$ computations and from the third mesh for $p = 1$ computations. Once the asymptotic range has been established, we can compare the curves obtained with numerical simulations with the theoretical curves representing the convergence order. Keeping in mind that for a smooth problem, the order of convergence of a DG method should be $p + 1$, we can confirm that the theoretical orders are achieved with the current DG implementation on triangular meshes, given that numerical curves match very closely theoretical orders.

4.2 Near-wall resolution requirements for DG/RANS - Turbulent flow over a flat plate at $Re = 5 \cdot 10^6$

The objective of the study presented in this section is to determine the acceptable near-wall resolution in terms of y^+ for high-order DG methods and provide a comparison with the best practices for classical second-order schemes, for which a value of $y^+ = 1$ is commonly employed.

The non-dimensional wall coordinate y^+ is defined as:

$$y^+ = \frac{y_w u_\tau}{\nu}, \qquad \text{with } u_\tau = \sqrt{\frac{\tau_w}{\rho}} \tag{4.5}$$

where y_w is the distance of the first cell from the wall, u_τ is the friction velocity and τ_w is the wall shear stress.

The influence of the y^+ value of the mesh is studied for three discretization orders $p = 1, 2, 3$ by performing simulations of the turbulent flow over a flat plate. The analysis is here carried out by performing RANS simulations. However, we expect these results to be applicable to ZDES simulations mode 1, for which the boundary layers are fully solved in RANS mode.

The configuration used in this study is the "2D Zero Pressure Gradient Flat Plate Verification Case" for turbulence modeling by NASA [13]. The case is run at $M_\infty = 0.2$ and Reynolds number of $Re_L = 5 \cdot 10^6$ based on length $L = 1$. The plot in figure 4.3 shows the layout of the flat plate grids used for this study, with the boundary conditions employed. The stagnation pressure $p_t/p_\infty = 1.02828$ and the stagnation temperature $T_t/T_\infty = 1.008$ are imposed at the inflow, far-field Riemann conditions are imposed at the top boundary at $y = L$, and the static pressure $p/p_\infty = 1$ is imposed at the outflow at $x = 2L$. The flat plate at the bottom boundary extends from $x = 0$ to $x = 2L$ at $y = 0$, and is modeled as a solid adiabatic wall, while symmetric boundary conditions extend from $x = -0.24L$ to $x = 0$, in the bottom boundary preceding the flat plate.

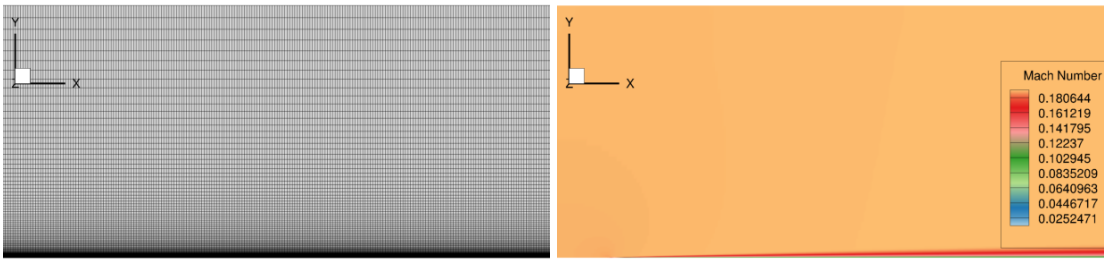


Figure 4.3: DG/RANS of the flat plate at $Re = 5 \cdot 10^6$. The computational mesh G1 with stretching factor $\alpha = 1.05$ (left) and Mach contour on the whole domain of the flat plate (right).

Since DG methods feature a polynomial variation of the solution inside the mesh cells, the relevant minimum height of the first cell for a DG method is considered as $y_{w,DG} = y_w/(p+1)$, where y_w is the height of the first element. For a cell-based FV method, the minimum wall normal spacing is computed as the distance of the cell center from the wall $y_{w,FV} = y_w/2$. For $p = 1$ simulations $y_{w,DG} = y_w/2$, therefore the computation of the relevant height of the first cell is the same as classical FV.

Simulations with different wall normal spacings y_w of the first element on the wall are performed. In the wall-normal direction the mesh is stretched with a geometric progression, and the stretching factor α is the parameter governing the wall normal spacing of the first element $y_w = L/(\sum_{i=0}^{N_y-1} \alpha^i)$.

	#dofs (per elem)	#elems grid G1	total #dofs grid G1	#elems grid G2	total #dofs grid G2
DG $p1$	3	224x97	65184	224x97	65184
DG $p2$	6	224x97	130368	158x69	65412
DG $p3$	10	224x97	217280	123x53	65190

Table 4.1: DG/RANS of the flat plate at $Re = 5 \cdot 10^6$. Resolution of the employed grids G1 and G2.

The study has been conducted considering two mesh configurations. The first, called hereafter G1, consists in structured $N_x \times N_y = 224 \times 97$ grids, represented in figure 4.3. The streamwise spacing is kept at a constant value $\Delta x = 0.01$ (equivalent to an average $\overline{x^+} \simeq 1800$ over the flat plate - notice that no DG scaling has been applied to this value, as will be explained for y^+ in the following). G1 simulations results are plotted in straight lines and circles in figures 4.4 and 4.5. For a DG method, several dofs are present in each element in each direction depending on the employed discretization order, thus G1 simulations provide a different number of dofs for each discretization (table 4.1).

The different global resolution between DG discretizations is likely to affect the study, as the results depend not only on the DG-relevant height of the first cell y^+_{DG}, but also on the DG-relevant streamwise spacing $x^+_{DG} = x^+/(p+1)$. Therefore, we decided to conduct the study also defining the same number of dofs for the three discretizations, denominated here as G2. The $p = 1$ mesh counting 224x97 elements is kept the same, yielding around 65K dofs, while a 158x69 elements mesh is employed for $p = 2$ simulations and a 123x53 elements mesh is employed for $p = 3$ simulations in the G2 configuration. Also in this case the Δx spacing is constant for each set of $p = 1$, $p = 2$, $p = 3$ meshes and the different wall normal spacing of the first cell are achieved by changing the stretching factor of the geometric progression. Therefore all the G2 simulations performed feature around 65K dofs and are represented in dotted lines and crosses respectively in figures 4.4 and 4.5.

Different simulations are performed for $p = 1$, $p = 2$, $p = 3$ discretizations varying the height of the first cell. For each simulation we compute $\overline{y^+}_{DG}$, which we define as the height of the first cell $y_{w,DG}$, multiplied by the average of u_τ/ν over all the cells in the first layer of elements of the boundary layer, computed from the reference simulation. This value is chosen as the one to control the spatial resolution, since the value of y^+_{DG} varies along the streamwise direction. We then monitor the error in the friction drag coefficient as well as the skin friction coefficient at two different locations of the flat plate, $x = 0.1$ and $x = 0.97$. These values are reported for each simulation as a function of $\overline{y^+}_{DG}$ in figures 4.4 and 4.5. Since for this test case the pressure drag coefficient is zero, the friction is the only contribution to the drag coefficient. The reference for the computation of the error is performed using a $p = 4$ discretization on a mesh featuring the smallest employed wall-normal spacing, and half the streamwise spacing of that employed in G1.

Typically, for wall-resolved RANS simulations, in order to ensure a proper representation of the velocity profile with accurate gradients, the chosen value for FV computations

is $y^+_{FV} = 1$ for the first layer. In this section we want to assess the possibility to employ higher values of y^+_{DG} for higher-order methods.

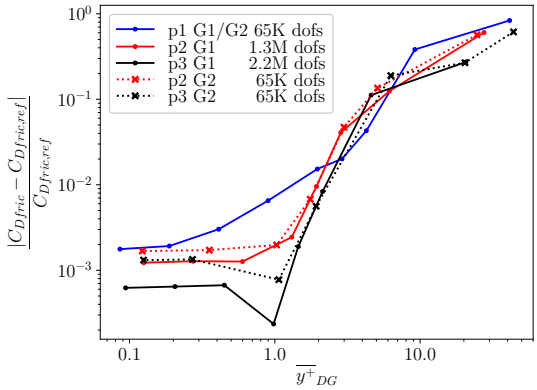


Figure 4.4: DG/RANS of the flat plate at $Re = 5 \cdot 10^6$. Error on the value of the integrated friction drag coefficient C_{Dfric} along the flat plate.

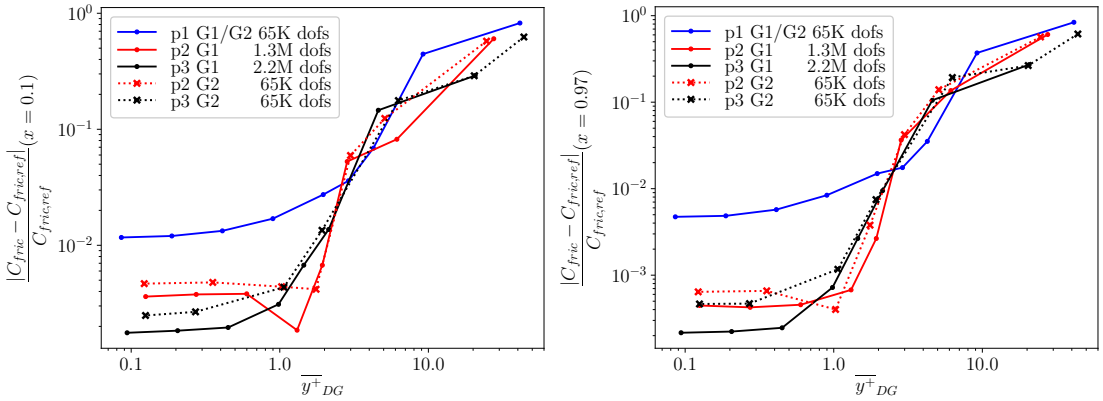


Figure 4.5: DG/RANS of the flat plate at $Re = 5 \cdot 10^6$. Error on the value of the friction coefficient C_{fric} at $x = 0.1$ (left) and at $x = 0.97$ (right) of the flat plate.

The first observation regarding figures 4.4 and 4.5 concerns the dependence of the C_{fric}, and consequently the C_{Dfric}, on the number of dofs in the streamwise direction. While the mesh G1 coincides with G2 for $p = 1$ computations, we can compare G1 and G2 curves for $p = 2$ and $p = 3$ computations and analyze the differences.

For all the three plots, we notice that straight (G1) and dotted (G2) curves are almost superimposed for both $p = 2$ and $p = 3$ discretizations, for values $\overline{y^+}_{DG}$ higher than 1. Below this threshold, the discrepancies between G1 and G2 results are due to the higher streamwise resolution of G1 simulations. This means that the convergence of the C_{fric} (and C_{Dfric}) value is guided only by the height of the first cell, except when dealing with very low values of the error, where a higher streamwise resolution leads to better results. We are typically not interested in meshes with $\overline{y^+}_{DG}$ significantly lower than 1. In fact

employing very refined meshes at the wall would impose severe constraints on the time step. The dependence of C_{fric} only on the height of the first cell is thus demonstrated and the following observations will be made independently from the use of G1 or G2 meshes.

Looking at the blue curves in figures 4.4 and 4.5, in order to accurately capture the gradients in the near-wall region and achieve 1% errors on the C_{fric} (and C_{Dfric}), the requirement $y^+_{DG} \simeq 1$ seems an adequate choice for the second order method. It follows that the second order DG discretization $p = 1$ imposes basically the same y^+ constraint as a FV discretization.

While for $\overline{y^+}_{DG} > 2.5$ no substantial difference is observed between the three DG discretizations, the C_{fric} at two locations $x = 0.1$ and $x = 0.97$, and the integrated C_{Dfric} values, obtained with the higher order discretizations $p = 2$ and $p = 3$, appear to be less sensitive to the value $\overline{y^+}_{DG}$ when lower values of $\overline{y^+}_{DG}$ are employed.

The $p = 2$ and $p = 3$ discretizations achieve the same accuracy with respect to the integrated C_{Dfric} obtained by the second-order discretization at $\overline{y^+}_{DG} = 1$, with larger wall spacing of respectively $\overline{y^+}_{DG} \simeq 1.8$ and $\overline{y^+}_{DG} \simeq 2.1$ (figure 4.4). Considering the error on the skin friction coefficient C_{fric} at $x = 0.1$ in the left panel of figure 4.5, a value of $\overline{y^+}_{DG} \simeq 2.3$ is sufficient for $p = 2$ and $p = 3$ discretizations to achieve the same accuracy as compared to $p = 1$ simulations with $\overline{y^+}_{DG} = 1$, while $\overline{y^+}_{DG} \simeq 2.1$ is sufficient for the C_{fric} at $x = 0.97$.

In turn, for the same value of $\overline{y^+}_{DG} = 1$, a reduction of the error of almost one order of magnitude is obtained by using $p = 2$ and $p = 3$ as compared to the $p = 1$ simulations.

The second and third columns of table 4.2 show the $\overline{y^+}_{DG}$ values necessary to obtain respectively a 1% error on the C_{Dfric} and the error on the C_{Dfric} at $\overline{y^+}_{DG} = 1$ for the three discretizations.

	$\overline{y^+}_{DG}$ @1% error on C_{Dfric}	error on C_{Dfric} @ $\overline{y^+}_{DG} \simeq 1$[1]
DG $p1$	1.32(G1,G2)	0.7%(G1,G2)
DG $p2$	1.96(G1)	0.12%(G1)
	1.96(G2)	0.17%(G2)
DG $p3$	2.24(G1)	0.06%(G1)
	2.32(G2)	0.13%(G2)

Table 4.2: DG/RANS of the flat plate at $Re = 5 \cdot 10^6$. The $\overline{y^+}_{DG}$ value needed to achieve 1% error on C_{Dfric}, and the achieved error on C_{Dfric} at $\overline{y^+}_{DG} \simeq 1$ [1] for the three discretizations.

The curves representing errors in figures 4.4 and 4.5 stagnate below $\overline{y^+}_{DG} \simeq 1$ due to

[1]Note that $p = 3$ simulations in figure 4.4, both G1 and G2, present a numerical undershoot of the error on the C_{Dfric}, for $\overline{y^+}_{DG}$ exactly equal to 1. This behavior is spurious and not representative of the real behavior of the error on the C_{Dfric}. Since results obtained by $p = 2$ and $p = 3$ simulations appear to converge to a precise error level for each discretization for $\overline{y^+}_{DG} < 1$, we decided to employ in the last column of table 4.2, for both $p = 2$ and $p = 3$, the values of the error on the C_{Dfric} at which each simulation converges, namely the error values obtained at $\overline{y^+}_{DG} < 0.3$.

other sources of errors (mostly the singular point of the flat plate) which do not depend only on the height of the first cell, but also on the streamwise resolution. The spurious undershoots that can be seen in the three figures (for G1$p3$ in figure 4.4, G1$p2$ in the left panel of figure 4.5 and G2$p2$ in the right panel of figure 4.5) are due to the coefficient oscillating around the reference value. This can lead to an artificial yet localized sudden decrease of the error for very low error levels.

In conclusion, when handling high order discretizations (i.e. greater than 2), it appears from our analysis that the classical constraint of $y^+_{DG} = 1$ can be relaxed, and a higher value of $y^+_{DG} \simeq 2.5$ seems adequate in order to retrieve results similar to second order discretizations with $y^+_{DG} \simeq 1$. Equivalent DG wall normal spacings for high-order methods can be then chosen more than twice than those employed for second order methods, allowing for higher time steps in time explicit simulations, while still providing good accuracy.

4.3 Influence of the DES constant for DG simulations using unstructured grids - Taylor-Green vortex at $Re = 5000$

In this section we assess the value of the DES model constant C_{DES} in the context of DG simulations for unstructured tetrahedral grids. This constant determines the amplitude of the subgrid dissipation in regions of the flow featuring developed turbulence. Shur *et al.* [270] found the optimal value of the C_{DES}, which verifies a Kolmogorov inertial range up to the grid cut-off, for simulations of the homogeneous decaying isotropic turbulence, using a centered fourth-order accurate differencing scheme.

In the same spirit, we choose here to compute the 3D Taylor-Green vortex case to assess the capacity of DES to predict the physics of turbulence for different discretization orders of a modal DG method. This case describes freely decaying isotropic turbulence in a periodic box with length of each spatial direction $L_x = 2\pi L$ with $L = 1$. The initial conditions with Mach $M_0 = 0.1$, and $U_0 = M_0\sqrt{\gamma R T_0}$, are given by:

$$\begin{aligned} p(\mathbf{x},0) &= \rho_0 U_0^2 \left[\frac{1}{\gamma M_0^2} + \frac{1}{16} \cdot (\cos(2x/L) + \cos(2y/L)) \cdot (\cos(2z/L) + 2) \right] \\ u(\mathbf{x},0) &= U_0 \cdot \sin(x/L) \cdot \cos(y/L) \cdot \cos(z/L) \\ v(\mathbf{x},0) &= -U_0 \cdot \cos(x/L) \cdot \sin(y/L) \cdot \cos(z/L) \\ w(\mathbf{x},0) &= 0 \end{aligned} \tag{4.6}$$

This initial condition corresponds to large vortical structures in the computational domain, which progressively break up into smaller structures and generate an energy cascade. The flow is initialized as isothermal ($p(\mathbf{x},0)/\rho(\mathbf{x},0) = p_0/\rho_0 = RT_0$). The Reynolds number is $Re = \rho_0 U_0 L/\mu_0 = 5000$, and the unsteady simulation is performed for a time $T = 15t_c$, where $t_c = L/U_0$ is the characteristic convective time.

The pseudo-eddy viscosity field $\tilde{\nu}$ is initialized with the turbulent viscosity obtained from the Smagorinsky model

$$\tilde{\nu}(\mathbf{x},0) = (C_S \cdot \Delta)^2 \cdot \sqrt{2 S_{ij}(\mathbf{x},0) S_{ij}(\mathbf{x},0)}\ , \tag{4.7}$$

where $C_S = 0.17$ is the Smagorinsky constant, $\Delta = \frac{\sqrt[3]{|K|}}{p+1}$ is the characteristic size of each mesh element K (computed as the cubic root of its volume) divided by the number of 1D degrees of freedom for a DG method, and S_{ij} is the strain rate tensor obtained analytically from equations (4.6).

4.3.1 DG computations at fixed number of dofs $\simeq 72^3$

In this section we perform simulations at relatively low fixed number of dofs #dofs $\simeq 72^3$ (the number is not exact due to the tetrahedral unstructured meshes for which an exact control of the number of elements is not straightforward) for different discretization orders, from $p = 1$ to $p = 5$. Given that $p = 1, 2, 3, 4, 5$ discretizations count respectively 4, 10, 20, 35, 56 degrees of freedom per element, meshes including respectively around $72^3/4$, $72^3/10$, $72^3/20$, $72^3/35$, $72^3/56$ elements are employed for each discretization (figure 4.6). Note that the number of dofs per element is different from Sections 4.1 and 4.2 because here the problem is three-dimensional.

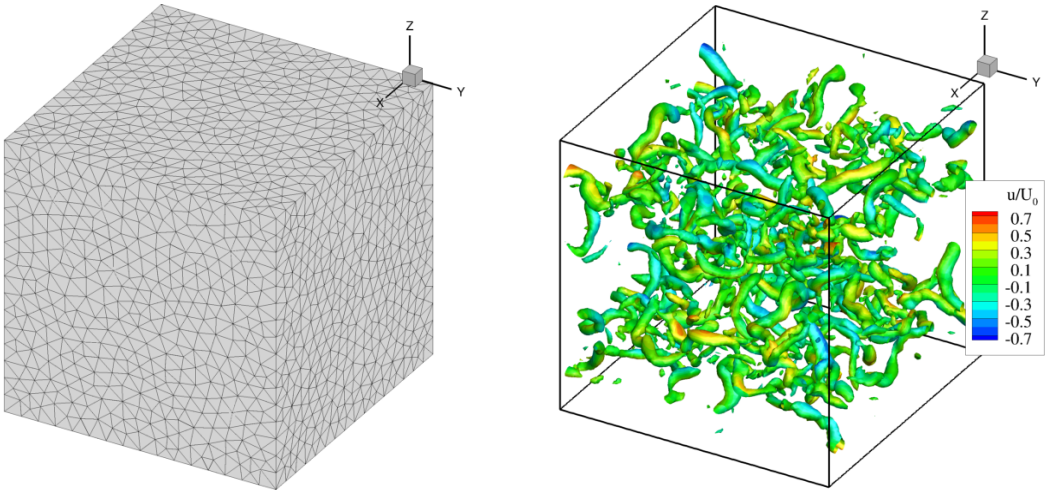


Figure 4.6: DG/DES of the Taylor-Green vortex at $Re = 5000$. Computational mesh (left) for a $p = 3$ simulation with resolution 72^3, and respective iso-surface of $Q = 5(U_0/L)^2$ at $t_c = 10$, coloured by the streamwise velocity component (right).

The quantities we monitor in order to assess the quality of the representation of the turbulence are the normalized enstrophy ξ and the normalized kinetic energy E_k

$$\xi = \frac{1}{\rho_0 U_0^2 |\mathcal{T}_K|} \int_{\mathcal{T}_K} \frac{\rho \boldsymbol{\omega} \cdot \boldsymbol{\omega}}{2} dV \tag{4.8}$$

$$E_k = \frac{1}{\rho_0 U_0^2 |\mathcal{T}_K|} \int_{\mathcal{T}_K} \frac{\rho \mathbf{v} \cdot \mathbf{v}}{2} dV \tag{4.9}$$

The reference DNS simulations have been performed on a mesh employing 480^3 dofs [82] with a eighth order Spectral Difference method.

We compute the energy spectra of the solution at time $t = 10$, when all scales of the flow are developed.

When analyzing LES results obtained with DG methods it is assumed that the resolution limit is defined by the cut-off wavenumber $k_{1/3} = \sqrt[3]{\#\text{dofs}}/3$. This is justified by the fact that the DG discretization represents accurately the spectrum for wavenumbers up to $k_{1/3} = \sqrt[3]{\#\text{dofs}}/3$, followed by a steep decrease of the energy for higher wavenumbers [220]. This is only due to the properties of the DG approximation. The presence of extra numerical dissipation might lead to additional discrepancies between DNS and LES also at lower wavenumbers.

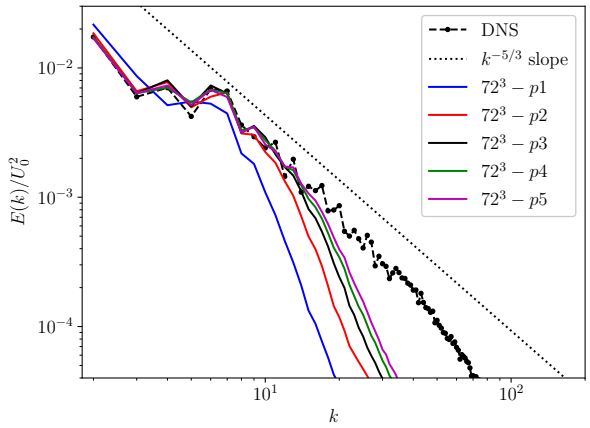


Figure 4.7: DG/DES of the Taylor-Green vortex at $Re = 5000$. The 3D energy spectrum within the inertial sub-range at time $t_c = 10$ compared to the Kolmogorov scaling $E(k) \propto k^{-5/3}$. $p = 1, 2, 3, 4, 5$ simulations with $C_{DES} = 0.65$.

In figure 4.7, the theoretical slope of the energy spectrum $k^{-5/3}$ is clearly observed in the inertial range for $p = 3, 4, 5$ simulations, with a good fit with the DNS reference computation. The damping of energy occurs at wavenumbers between 14 and 16 for the three discretizations $p = 3, 4, 5$, before the limit value $k_{1/3} = 72/3 = 24$. As the discretization order is increased, the cut-off occurs at higher wavenumbers and the decay of the spectrum is slower. However the three curves are very close to each other, meaning that the maximum wave length resolution limit is almost the same for $p \geq 3$, and no further significant improvements can be expected further increasing the discretization order.

The $p = 2$ simulation presents a damping at a lower wavenumber compared to $p = 3$ but still presents the theoretical slope at low wavenumbers, while the $p = 1$ simulation provides results far from the theoretical behavior, damping a significant part of the energy associated with the large scales. This is due to the intrinsic numerical dissipation that low order DG schemes provide. The higher the order of the DG discretization, the lower the numerical dissipation. This means that for a low-order scheme the numerical dissipation is too strong to correctly represent the turbulent behavior, unless the resolution is dramatically increased.

It is thus preferable to employ high-order $p \geq 2$ discretizations when an accurate

representation of turbulence is sought with a fixed number of degrees of freedom.

For *hp*-adaptation to be attractive from the computational time point of view, an *ad hoc* load balancing technique is mandatory to fully exploit the potential of variable order degree simulations. Since this goes beyond the scope of this work, and no sophisticated load balancing techniques are employed here, we choose to employ a maximum polynomial degree discretization $p = 3$ in the adaptive algorithm for the sphere and the PPRIME nozzle in Chapter 6. As shown in figure 4.7, the high-order $p = 3$ discretization provides a good representation of turbulent phenomena, on par with higher-order discretizations.

All the following analysis are then conducted on a $p = 3$ discretization, representative of our final test cases.

The sensitivity of the model to the constant C_{DES} is studied and the corresponding results are displayed in figure 4.8. The results of DES simulations are presented in terms of the time evolution of the enstrophy and the kinetic energy, with respect to the filtered DNS data, and the 3D energy spectrum with respect to the DNS spectrum. For a fair comparison between the DES and the reference DNS computations, especially for the levels of enstrophy which are strongly affected by the resolution, the DNS is filtered using the transfer function associated with the DG discretization introduced by Chapelier *et al.* [80]. The DNS velocity and vorticity fields are filtered and the corresponding spatially-integrated quantities are computed with a sampling $\Delta t_c = 1$ between $t_c = 1$ and $t_c = 20$.

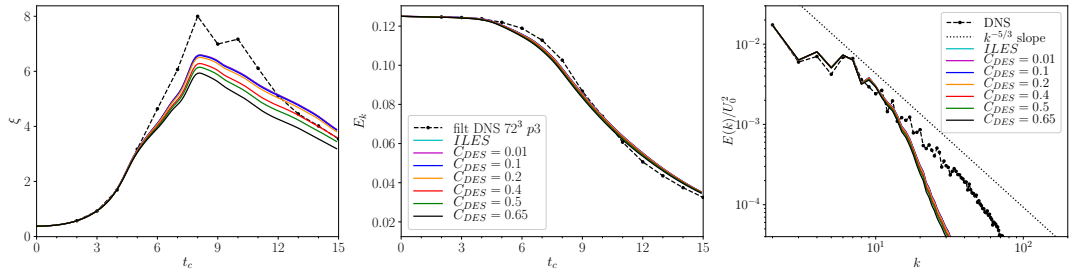


Figure 4.8: DG/DES of the Taylor-Green vortex at $Re = 5000$. Normalized enstrophy (left) and kinetic energy (middle) over the course of the simulation for a DGp3 discretization for different values of the C_{DES}. The 3D energy spectrum within the inertial sub-range at time $t_c = 10$ compared to the Kolmogorov scaling $E(k) \propto k^{-5/3}$ (right).

Analyzing the plots of the enstrophy over the course of the simulation in the left panel of figure 4.8, the decrease of the C_{DES} constant from the original value $C_{DES} = 0.65$ to $C_{DES} = 0.01, 0.1, 0.2, 0.4, 0.5$ leads to a slightly better representation of the peak of enstrophy, due to a weaker dissipation introduced by the DES model. When employing low values of the $C_{DES} = 0.01, 0.1, 0.2$, the model seems not to provide a sufficient dissipation after the peak of enstrophy, with an overestimation of the enstrophy for $t \geq 12$. At the same time, the kinetic energy evolution in the middle panel of figure 4.8 shows a low sensitivity to variations of the DES constant, meaning that the model predicts accurately the decay of the kinetic energy employing several values of the C_{DES} constant.

The value of the constant C_{DES} seems to affect marginally the energy spectra in the

right panel of figure 4.8. The simulations performed with different C_{DES} values provide indeed a similar decay of the turbulent energy spectrum.

The method appears therefore to be robust to variations of the C_{DES}, and the standard value $C_{DES} = 0.65$ does not appear to introduce an excessive amount of dissipation.

4.3.2 Mesh sensitivity study

In this subsection we study the sensitivity of DG-DES simulations on the mesh resolution for a fixed polynomial degree. We perform simulations on different resolutions, and compare them with the filtered DNS data at the respective cut-off wave number $k_{1/2}$. The $p = 3$ discretization is considered, and simulations counting #dofs $= 48^3, 96^3, 144^3$ are performed on tetrahedral meshes, in addition to #dofs $= 72^3$, presented in the previous section. Three different values of the constant $C_{DES} = 0.1, 0.4, 0.65$ are compared.

We made comparisons for the enstrophy development for the three resolutions. In the following we only report and analyze the evolution of the enstrophy. Kinetic energy is not shown here because only very small differences can be found between the different computations. Note the different scale of the enstrophy ξ levels for the different resolutions considered.

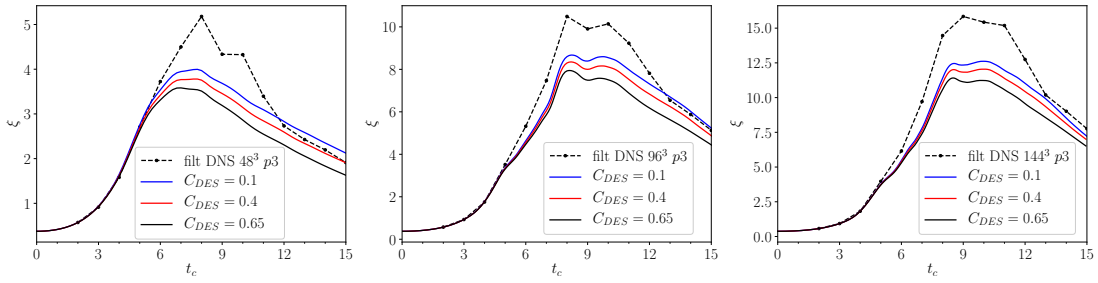


Figure 4.9: DG/DES of the Taylor-Green vortex at $Re = 5000$. Volume-averaged enstrophy. Different values of C_{DES} for three resolutions #dofs $= 48^3, 96^3, 144^3$.

As already observed for the simulations using 72^3 dofs, increasing C_{DES} leads to a stronger dissipation and lower levels of the enstrophy. A low value of the constant $C_{DES} = 0.1$ does not provide a sufficient dissipation after the peak of enstrophy has been reached. In particular, higher levels of the enstrophy than the respective filtered DNS are observed for $t_c \geq 12$ for the two coarse resolutions with 48^3 (left panel in figure 4.9) and 72^3 dofs (left panel in figure 4.8). On the other hand, when increasing the resolution to 96^3 and 144^3 dofs, the model always provides sufficient dissipation to the unresolved scales, without requiring high values of the C_{DES}.

The energy spectrum is also analyzed to provide a further insight in the mechanism of the turbulent decay. As already pointed out for the resolution corresponding to 72^3 dofs, the energy spectra at $t_c = 10$ obtained with 3 different values of the C_{DES} are almost indistinguishable, for each resolution. This confirms the overall low criticity of this constant for the generation and development of turbulent scales.

According to Sagaut *et al.* [260] the DES approach can be considered analogous to the well-known Smagorinsky model for free-shear flows, and a value of $C_{DES} = 0.65$ is

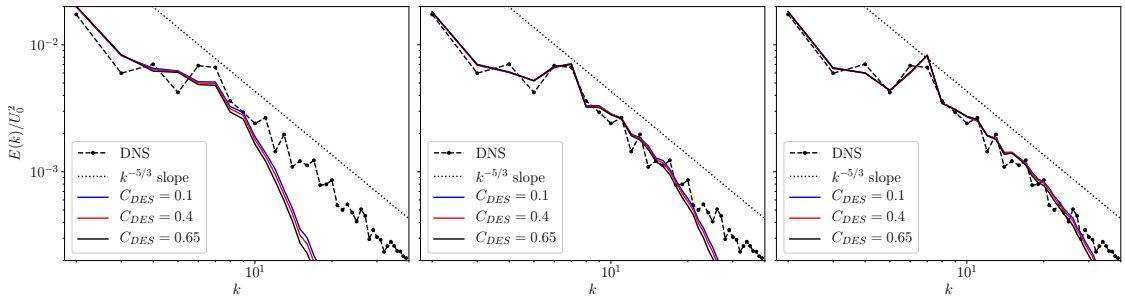


Figure 4.10: DG/DES of the Taylor-Green vortex at $Re = 5000$. The 3D energy spectrum within the inertial sub-range at time $t_c = 10$ compared to the Kolmogorov scaling $E(k) \propto k^{-5/3}$ for three resolutions #dofs = $48^3, 96^3, 144^3$ and different C_{DES} values.

equivalent to the Smagorinsky constant $C_S \simeq 0.2$. Due to the intrinsic dissipation of DG methods, a lower $C_S = 0.1$ is often employed [100] for LES/DG simulations. We could think that decreasing the constant to $C_{DES} = 0.3$ or 0.4 would be an appropriate choice to allow the subgrid model to provide a sufficient amount of dissipation, without over-dissipating the turbulent structures. These values would correspond to a $C_S \simeq 0.1 - 0.13$ according to Sagaut's relations. However, for the purposes of this work we prefer to use the classical constant value $C_{DES} = 0.65$, since very few differences are observed in the time evolution of the enstrophy by changing the C_{DES} value, and almost no differences are found out in the decay of the turbulent spectrum. A much wider analysis would be necessary to assess the effective benefits that changing this constant would bring, and eventually propose a different value.

4.4 Conclusion

In this chapter, some features of the DG methods coupled with RANS/DES are analyzed for canonical test cases. The first validation test case, the transport of an isentropic vortex, has been used to verify the convergence order of the employed DG method on unsteady simulations performed on unstructured triangular meshes. A sensitivity study on the minimum wall spacing y^+ for RANS has been carried out for the flow over a flat plate at Reynolds number equal to $5 \cdot 10^6$ using structured quadrilateral meshes, showing that increasing the order of accuracy of the DG approximation allows for relaxing the constraints on the near-wall resolution. Finally, the value of the DES constant has been tested for DG on grids composed of tetrahedra, from the Taylor-Green Vortex test case, representing the breakdown into turbulence and the following decay of homogeneous isotropic turbulence. It has been shown that the modulation of the DES constant does not impact significantly the turbulent spectrum. However, the choice of the polynomial degree has a significant impact, and at least a fourth order approximation is required to obtain a good representation of turbulent structures in the fully-developed turbulent regions, at least for relatively coarse simulations.

Chapter 5

h- and *hp*-adaptation for steady computations

In the present chapter, we describe in detail the adaptation strategies used in this research, based on the coupling of accurate DG-based indicators extracted from the flow solver CODA, with the external remeshing library MMG, and we apply them to steady flow test cases.

Section 5.1 provides a thorough description of the *h*- and *hp*-adaptive procedures.

In Section 5.2, the performance of the developed adaptation algorithms is assessed for the 2D laminar viscous flow past a NACA0012 airfoil, and for the 3D laminar viscous flows past a sphere and past a delta wing, by comparisons with uniform refinement. Simulations are performed using polynomial degrees $p = 1, 2, 3$.

In Section 5.3, the *hp*-adaptive algorithm is applied and validated on a turbulent configuration of an isothermal subsonic jet issued from a round nozzle, by comparing RANS adaptive simulations with the experimental data and a numerical reference.

Finally, the main conclusions of this study are presented in Section 5.4.

5.1 The steady *h*- and *hp*-adaptive algorithms

The adaptation procedure is driven by an *a posteriori* error estimator which controls the solution accuracy within the domain, identifying the regions lacking the requested resolution. The resolution in these regions is improved by either decreasing the size of the element or increasing the polynomial degree which approximates the solution. A smoothness indicator guides the *hp*-decision, leading to *p*-enrichment for smooth regions and *h*-refinement for non-smooth regions.

The final applications targeted in this first part of the thesis work involve hybrid RANS-LES simulations of turbulent jets, and in general 3D configurations with a high number of degrees of freedom. The possibility to extend the algorithm to 3D in a straightforward manner is, in this context, an essential requirement. This is why an isotropic mesh refinement-based strategy is selected, which is fit for adapting the resolution in free-shear regions, while the flow regions presenting highly anisotropic features (e.g. boundary layers) can be meshed with fixed anisotropic elements whose resolution

is modulated through p-adaptation.

A metric-based hp-mesh combines continuous and discrete settings, to be prescribed respectively to the nodes and to the elements of the mesh. The geometry of elements changes smoothly, and is prescribed to the remesher by means of a nodal size map. In contrast, the approximation polynomial degree is a discrete quantity unequivocally defined for each element. The following sections describe the strategies to create an hp-mesh in the context of metric-based mesh adaptation, without using complex optimization algorithms. We briefly summarize the two procedures in algorithms 1 and 2.

We remind that the remesher MMG handles only linear meshes, presenting a linear approximation of the mesh elements. Curved meshes, which represent the geometry by using higher degree polynomials, have not been employed to perform simulations in this work. We remark that in this work we make use of the compact wording, e.g. "$p = 2$ elements", to refer to those elements which are characterized by a local spatial DG discretization using polynomials of degree $p = 2$. They must not be confused with curved elements presenting a quadratic approximation of the boundaries.

5.1.1 *a posteriori* error estimator

The first step in the construction of the present hp-adaptation strategy is to devise an accurate and simple error estimator suited to various polynomial degrees.

The interest of coupling two different indicators has been emphasized in Section 3.4.3. We follow therefore the same idea as Colombo *et al.* [92] and Bassi *et al.* [38] by combining two error estimators, one based on the energy of the highest-order modes, the so-called Small Scale Energy Density (SSED) estimator ϵ_{SSED} (presented in equation 3.6), and the second one based on the jumps across element interfaces, $\epsilon_{\text{JUMPmom}}$ (with a similar formulation of the estimator from equations (3.9), (3.10) and (3.11)).

In our work the solution component on which we estimate the error is the norm of the momentum, while Colombo and Bassi employed the pressure. Moreover the error estimators we use in this work have the dimensions of a momentum $[\rho\mathbf{u}]$, while Bassi and Colombo employed non dimensional estimators (before the coupling and the further normalization over the minimum and the maximum values).

Concerning the first error estimator based on the highest-order modes of the solution, several variants are found in the literature. Naddei *et al.* [219] compared different error estimators, among which we find the SD with the formulation used by Tumolo *et al.* [285], identical to that of Bassi and Colombo, except that the error is estimated from the momentum rather than the pressure. In the context of turbulent flows, Tugnoli *et al.* [284] and Flad *et al.* [124] employed a similar formulation with the norm of the momentum, but they decided to remove the contribution of the cell-average in the computation of the total energy. Tugnoli *et al.* discussed that this choice was made to avoid an underestimation of the error in zones of the domain presenting a strong mean flow. Alternatively, in recirculation zones, a low value of the momentum can yield an overestimation of the momentum-based normalized error estimator.

Based on these considerations and on the work of Naddei *et al.,* who observed better performances of the SSED estimator with respect to the "normalized version" of Tumolo *et al*. We therefore decided to employ the SSED estimator in the formulation of equation

(3.6), measuring the error only from the highest-order modes, and to formulate the JUMP error estimator accordingly.

The second error estimator is a modified version of the one proposed by Bernard *et al.* [47] presented in equations (3.9), (3.10) and (3.11). We then define the error $\mathbf{E}_e$ on the interface e as in equation (3.9) and we consider an averaged error over each interface e, for the norm of the momentum vector, as done in equation (3.10) for a generic variable:

$$\epsilon^2_{\text{JUMPmom},e} = \frac{\int_{\partial K_e} ||(\rho\mathbf{v})^+_h - (\rho\mathbf{v})^-_h||^2 dS}{4|\partial K_e|} = \frac{||(\rho\mathbf{v})^+_h - (\rho\mathbf{v})^-_h||^2_{L^2(\partial K_e)}}{4|\partial K_e|} . \tag{5.1}$$

For consistency with the dimensions of the SSED indicator in equation (3.6), the error indicator here is normalized by the area of each interface $|\partial K_e|$ (or for 2D problems the length of the edge e). For each element K we compute the error estimator by the same rule as the original formulation, except that here we do not multiply the numerator by the volume of the cell:

$$\epsilon^2_{\text{JUMPmom},K} = \frac{1}{N_e} \sum_{e=1}^{N_e} \epsilon^2_{\text{JUMPmom},e} , \tag{5.2}$$

with N_e the number of faces of the element in 3D (or edges in 2D).

Both indicators are normalized by their respective maximum and minimum values over the whole domain $\mathcal{T}_K$ (min-max normalization) before the coupling:

$$\epsilon_K = \epsilon_{\text{SSED},K,norm} + \epsilon_{\text{JUMPmom},K,norm} \tag{5.3}$$

$$= \frac{\epsilon_{\text{SSED},K} - \min(\epsilon_{\text{SSED}})_{\mathcal{T}_K}}{\max(\epsilon_{\text{SSED}})_{\mathcal{T}_K} - \min(\epsilon_{\text{SSED}})_{\mathcal{T}_K}} + \frac{\epsilon_{\text{JUMPmom},K} - \min(\epsilon_{\text{JUMPmom}})_{\mathcal{T}_K}}{\max(\epsilon_{\text{JUMPmom}})_{\mathcal{T}_K} - \min(\epsilon_{\text{JUMPmom}})_{\mathcal{T}_K}} . \tag{5.4}$$

In references [92, 38] which focus on p-adaptation and our preliminary work [36] focused on h-adaptation, the contribution of the error estimator based on the solution jump decreased as the polynomial degree increased and the contribution of the modal estimator was switched off for $p = 1$. In this work we prefer a uniform approach to combine the two error estimators for all polynomial degrees. We observed that even though the estimator based on the highest order modes is not always optimal for $p = 1$ computations, its coupling with a jump error estimator still performs well for low-degree polynomials.

This error estimator will be the only estimator used in all the chapters of the thesis, except in the last Chapter 7, where a different formulation of the error estimator taking into account also the pressure will be introduced for the adaptation of transonic flows.

5.1.2 Smoothness indicator

The difficulty in hp-adaptive methods lies in the choice whether to adapt an element with h-refinement or p-enrichment. In this work we follow the approach proposed in 1D by Mavriplis [202]. It is based on the assumption that the decay rate of the spectrum of the DG modal coefficients is related to the convergence rate of the solution. This information is exploited to evaluate the smoothness of the solution in the mesh elements. It

is assumed that for a 1D Legendre expansion with coefficients $a^{(q)}$ with $q = 0, ..., p_K$, the modal coefficients decay exponentially fast after the asymptotic range has been reached:

$$|a^{(q)}| \simeq C \exp(-\sigma q) \ , \tag{5.5}$$

where C and σ are constants determined by a least-squares best fit of $\log(a^{(q)})$ vs. q (see equation 2.29). The decay coefficient σ is then used as smoothness indicator. The reliability of the smoothness estimate increases with the polynomial degree. High decay rates imply that the solution is smooth, while the solution deviates from analytical behavior in the presence of low decay rates. In this work, following Mavriplis' choice, we use a threshold value of 1: elements with $\sigma_K > 1$ are assumed to be smooth, and subject to p-enrichment; if this condition is not fulfilled, then h-refinement is used.

For 2D and 3D computations, several modal coefficients can contribute to an index q. Therefore we need to gather in some way the coefficients of the modes to retrieve one single value per index q. Based on Mavriplis' work [202], the extension to 2D has been performed by Chalmers *et al.* [78], who computed a one-dimensional smoothness indicator for each mode, and used the minimum value for σ. Leicht *et al.* [182] proposed to extend the approach in 3D through the accumulation of all coefficients of the Legendre polynomials of the corresponding multi-dimensional degree. They tested the classical threshold between smooth and non-smooth element $\sigma = 1$ as well as different threshold values.

The approach we follow in our work to retrieve one single coefficient for indices q for 2D and 3D configurations consists in computing the coefficient $a^{(q)}$ as the L^2-norm of $u^{(q)}$ of the coefficients of the polynomial basis as:

$$a_K^{(q)} = \sqrt{\sum_{l \in d_q} {u_K^{(l)}}^2} \quad \forall q \in (1, p_K) \ , \tag{5.6}$$

with d_q the set of indices of the basis functions of total degree equal to q (introduced in equation (2.29)). We notice that the coefficient associated with $q = 0$ corresponds to the average of the solution over the element, which can severely bias the decay rate. Therefore we chose not to employ the coefficient $a^{(0)}$ in the log-linear regression $\log(a^{(q)})$ vs. q. As a result, $p = 1$ computations lack information about the smoothness of the solution, and the smoothness indicator can be computed only for high-order elements, i.e. $p \geq 2$. In this work $p = 1$ elements are always fictitiously marked as smooth elements, always requiring p-adaptation. A different strategy, employing smoothness indicators better suited for shock capturing, should be used in future work for test cases with strong physical discontinuities, where a low polynomial degree is desired in these regions. The decay rate is computed from the modal coefficients associated with the norm of the momentum. The smoothness indicator can also serve as an indicator of the convergence rate of the solution, which will be exploited in the mesh refinement strategy described below.

5.1.3 Metric prescription for h-adaptation

At the end of the simulation i, the error estimator in each mesh element is computed from the solution and indicated as $\epsilon_{K,i}$. The updated element size $h^*_{K,i+1}$ is then prescribed

for each mesh element (K, i), similarly to the approach proposed by Bernard *et al.* [47] and by Remacle *et al.* [246] for finite elements. We follow the assumption that the local error converges asymptotically to zero at a given convergence rate. In particular, for a DG method the L^2-norm of the error in the solution decays asymptotically as $O(h^{p+1})$ [111].

In a pure h-adaptive context, under the assumption of asymptotic regular behavior, the error for one element K can be thus seen as:

$$\epsilon_K \simeq C h_K^{\min(k,p+1)}, \qquad \forall K \in \mathcal{T}_{K,i} \ , \tag{5.7}$$

with C depending only on the solution, for a fixed polynomial degree, and the Sobolev index k depending on the regularity of the solution. In the presence of a physical or geometrical discontinuity, the smooth convergence order p+1 is lost. Houston *et al.* [155] approximated the Sobolev index k solution to estimate the regularity of the solution. Remacle *et al.* [247] defined *a priori* a given resolution to resolve the discontinuities, by prescribing a fixed small desired size to elements presenting non-smooth behavior. In our work we exploit the fact that the convergence order of the solution is limited in the vicinity of discontinuities to $\epsilon_K \simeq O(h^1)$. Therefore we impose the convergence rate $m = p + 1$ if the solution is smooth, and $m = 1$ if the solution is non-smooth. For $p = 1$ computations for which the smoothness indicator cannot be computed, the convergence rate is always set to $m = p + 1$.

Since the input size-field that MMG requires is node-based, while the error estimator $\epsilon_{K,i}$ is element-based, a volume-weighted average of the error estimator is computed based on its values in the elements surrounding each node. This node-based error estimator will be denoted as $\epsilon_{n,i}$.

The h-adaptation algorithm therefore acts as follows.

The first step consists in obtaining from the initial mesh the metric $\mathcal{M}_0$, which is calculated by MMG from a preliminary remeshing step, and the initial characteristic lengths h_0. For each subsequent step, the metric $\mathcal{M}$ is automatically produced by MMG.

In an isotropic context h_n measures the length of the edge of the equilateral tetrahedra surrounding the node n in 3D (triangle for 2D). In practical terms MMG provides as output of the remeshing process a non-scalar matrix, which means that it introduces some anisotropy. In this case the eigenvalues associated with the metric are not identical. In order to make the output compatible with our isotropic strategy, we extract the characteristic length h_n from MMG by recalling the notions of density with respect to the metric tensor $\mathcal{M}$ introduced in Section 3.1.2. The characteristic length is then considered equal to the cubic root (square root in 2D) of the inverse of the mesh density in the metric $\mathcal{M}$ at the node n

$$h_n = \mathcal{D}_n^{-1/d} = (\Pi_{j=1}^d h_j)^{1/d} = (\Pi_{j=1}^d \lambda_j)^{-1/(2d)} \ , \tag{5.8}$$

where d is the number of dimensions and λ_j are the eigenvalues associated with the metric tensor output of MMG.

According to the value of the averaged error estimator $\epsilon_{n,i}$ at each node n of the mesh, the new characteristic length imposed to the node n is decreased if $\epsilon_{n,i} > \epsilon_{i+1}^*$ or is kept unchanged if $\epsilon_{n,i} \leq \epsilon_{i+1}^*$, with ϵ_{i+1}^* being the global desired error level for the

$i+1$ simulation, and defined later in Section 5.1.5. We will call hereafter $\mathcal{T}_{h,i} \subseteq \mathcal{T}_K$ the subset of the refined nodes, while $\mathcal{T}_{=,i} \subseteq \mathcal{T}_K$ is the subset of the nodes not marked for refinement.

Adapting equation (5.7) to nodal values, the new characteristic sizes $h^*_{n,i+1}$ imposed at the adaptation step $i+1$ at mesh nodes $\mathcal{T}_{h,i}$ are expressed as

$$h^*_{n,i+1} = h_{n,i} \left(\frac{\epsilon^*_{i+1}}{\epsilon_{n,i}} \right)^{\frac{1}{m_n}} , \qquad \forall n \in \mathcal{T}_{h,i} , \tag{5.9}$$

where subscript n refers to quantities defined at nodes. $h^*_{n,i}$ is the characteristic size of the element n at the adaptation step i and m_n is the assumed convergence rate on the node n.

In order to define the value of m_n, the element-wise smoothness indicator σ_K is computed first, and the corresponding nodal value σ_n is obtained by the means of a volume-weighted average. When the solution is considered smooth, i.e. $\sigma_n > 1$, we choose $m_n = p+1$. For the rest of the nodes corresponding to non-smooth regions, the value $m_n = 1$ is chosen.

A user-defined parameter, the maximum refinement factor r_h, is needed to avoid uncontrolled or excessive size changes between two subsequent adaptation steps. The value of $1/r_h$ represents the minimum allowed ratio $h^*_{n,i+1}/h_{n,i}$ such that we have $h^*_{n,i+1} \geq h_{n,i}/r_h$, i.e. in one adaptation step the edge can decrease its size by at most r_h times.

Once the new characteristic size has been identified, the input size-field is provided to MMG for each node of the mesh i as a diagonal matrix $\mathcal{M}^*_{i+1} = \Lambda^*_{i+1} = (h^*_{n,i+1})^{-2}\mathcal{I}$, where $\mathcal{I}$ is the identity matrix. This input field allows MMG to perform the refinement process, which is seamless for the user.

Once the new mesh has been output by MMG, the solution from the previous mesh is projected onto the adapted mesh by means of an L^2 projection.

Algorithm 1: h-adaptation algorithm

```
set i=0;
create the initial mesh T_0;
obtain from MMG the initial characteristic lengths field h_0;
while #dofs < #dofs_max do
    perform the simulation on T_i;
    compute the a posteriori error estimator ε_{K,i} on T_i;
    compute the smoothness indicator σ_{K,i} on T_{K,i};
    choose the target error ε*_{i+1} in order to fulfill the given refinement criterion;
    average the error estimator at the nodes ε_{n,i} (volume-weighted average);
    average the smoothness indicator at the nodes σ_{n,i} (volume-weighted
     average);
    if ε_{n,i} > ε*_{i+1} then
        adapt the current size h_{n,i} to the new h*_{n,i+1} provided by the formulation
         (5.9);
    else
        leave the current size h*_{n,i+1} = h_{n,i} unmodified;
    end
    give the new metric to MMG, which generates the refined mesh T_{K,i+1};
    project the solution of T_{K,i} on T_{K,i+1};
    i+=1;
end
```

5.1.4 Degree and metric prescription for *hp*-adaptation

Similarly to h-adaptation, at the end of the simulation i, we compute the error estimator as well as the smoothness indicator from the solution in each mesh element as $\epsilon_{K,i}$. Then both the updated polynomial degree $p_{K,i+1}$ and the new element size $h_{K,i+1}$ are imposed for each mesh element K, as described below. We start by a constant polynomial degree map $p = p_{\min}, \quad \forall K \in \mathcal{T}_K$. We can define three non overlapping sets of elements of the mesh

$$\mathcal{T}_{K,i} = \mathcal{T}_{h,i} \cup \mathcal{T}_{p,i} \cup \mathcal{T}_{=,i} \ , \tag{5.10}$$

where $\mathcal{T}_{h,i}$ is the subset of the elements marked for h-adaptation, $\mathcal{T}_{p,i}$ is the subset of the elements marked for p-adaptation and $\mathcal{T}_{=,i}$ is the subset of the elements not marked for any adaptation. An element cannot be selected for both h- and p-adaptation.

Firstly the smooth elements, i.e. the elements with a smoothness indicator $\sigma_K > 1$, and the $p = 1$ elements, which lack in information about their smoothness, are marked for p-adaptation if they present an error estimator $\epsilon_{K,i}$ greater than a target error ϵ^*_{i+1}, and their polynomial degree is increased by one. A threshold on the maximum polynomial degree $p_{\max}$ that the algorithm can employ is defined by the user. After the threshold is attained, h-adaptation is performed also on elements presenting a smooth solution. Thus, for smooth and $p = 1$ elements lacking resolution, the new polynomial

degree is defined as:

$$p^*_{K,i+1} = \max(p_{K,i} + 1, p_{\max}), \qquad \forall K \in \mathcal{T}_{p,i} \ . \tag{5.11}$$

We use the superscript * also for the polynomial degree to remind that this corresponds to the desired polynomial degree on the mesh $i+1$, but is imposed on the mesh i. The degree map is not going to be exactly preserved in each location of the the mesh $i+1$, due to the interpolation of the polynomial degree from one mesh to another.

Once the elements subject to p-refinement have been chosen for the subset of the mesh $\mathcal{T}_{p,i}$

$$\mathcal{T}_{p,i} = \{K \in \mathcal{T}_i \mid \epsilon_{K,i} > \epsilon^*_{K,i+1} \text{ and } [(\sigma_{K,i} > 1 \text{ and } p_{K,i} + 1 \leq p_{\max}) \text{ or } p_{K,i} = 1]\} \ , \tag{5.12}$$

we still need to address the remaining elements $K \in \mathcal{T}_{h,i} \cup \mathcal{T}_{=,i}$. Similarly to equation (5.7), the error in an element K at the adaptation step i verifies:

$$\epsilon_K \simeq C h_K^{m_K}, \qquad \forall K \in \mathcal{T}_{h,i} \cup \mathcal{T}_{=,i} \ . \tag{5.13}$$

Assuming that C remains constant for non p-refined elements $K \in \mathcal{T}_{h,i} \cup \mathcal{T}_{=,i}$ between iterations i and $i+1$, we can use the formula (5.9) to compute the new sizes for nodes $n \in \mathcal{T}_{h,i}$.

This allows for h-refining the elements belonging to the subset $\mathcal{T}_{h,i}$:

$$\mathcal{T}_{h,i} = \{K \in \mathcal{T}_i \mid \epsilon_{K,i} > \epsilon^*_{K,i+1} \text{ and } [\sigma_{K,i} \leq 1 \text{ or } (\sigma_{K,i} > 1 \text{ and } p_{K,i}{+}1 > p_{\max})] \text{ and } p_{K,i} \neq 1\} \ . \tag{5.14}$$

To be consistent with the node-based input size-field required by MMG, the quantities $\epsilon_{n,i}$, $p_{n,i}$ and $\sigma_{n,i}$ are computed at nodes using a volume-weighted average with contributions only from surrounding elements that are not marked for p-enrichment at the current step. This choice is illustrated in figure 5.1. This prevents an excessive refinement in p-refined zones, and keeps valid the hypothesis of a constant C between two adaptive steps in equation (5.13). This selective weighted average allows for an hp-choice despite the fact that p is evaluated for each element and h is defined at the nodes.

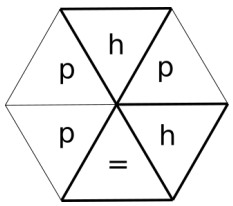


Figure 5.1: Simplified central node with surrounding simplicial elements (marked for p-adaptation, for h-adaptation and not marked for adaptation). The nodal weighted average of the error estimator, the polynomial degree and the smoothness indicator for the prescription of the sizes is performed only on highlighted elements.

The size $h^*_{n,i+1}$ is then imposed at the nodes n as done in equation (5.9). The theoretical convergence order m_n is $p_n + 1$ if $\sigma_n > 1$ and 1 if $\sigma_n \leq 1$, as considered in the previous section.

It should be noted that the nodal polynomial degree p_n is no longer an integer after the nodal average. Again, $p = 1$ elements are considered as smooth and the prescribed convergence order is $m_n = p_n + 1$.

The regions presenting a higher error estimator value than the target, with smooth solution or $p = 1$ approximation, are p-enriched, while the regions with non-smooth solution or already at their maximum polynomial degree are subject to h-refinement, again when the error is higher than the target.

The input size-field $h^*_{n,i+1}$ is provided to MMG for each node of the mesh i analogously to what is done for h-adaptation. MMG returns the new adapted mesh $i + 1$, and the new polynomial degree map $p^*_{K,i+1}$ is interpolated from mesh $\mathcal{T}_{K,i}$ to mesh $\mathcal{T}_{K,i+1}$ with a nearest-neighbor interpolation: the new degree prescribed to an element of the old mesh is assigned to all the elements of the new mesh presenting their centers inside the old cell.

Algorithm 2: hp-adaptation algorithm

set i=0;
create the initial mesh $\mathcal{T}_0$;
obtain from MMG the initial characteristic lengths field h_0;
set an initial polynomial degree map $p_0 = p_{\min}$;
while #dofs < $\#\text{dofs}_{\max}$ **do**
  perform the simulation on $\mathcal{T}_{K,i}$;
  compute the *a posteriori* error estimator $\epsilon_{K,i}$ on $\mathcal{T}_{K,i}$;
  compute the smoothness indicator $\sigma_{K,i}$ on $\mathcal{T}_{K,i}$;
  choose the target error ϵ^*_{i+1} in order to fulfill the given refinement criterion;
  if $\epsilon_{K,i} > \epsilon^*_{i+1}$ **and** $[(\sigma_{K,i} > 1$ **and** $p_{K,i} + 1 \leq p_{\max})$ **or** $p = 1)]$ **then**
    adapt the current polynomial degree of the elements $K \in \mathcal{T}_{p,i}$ as $p^*_{K,i+1} = p_{K,i} + 1$;
  else
    leave the current polynomial degree of the element $p^*_{K,i+1} = p_{K,i}$ unmodified;
  end
  average the polynomial degree $p_{K,i}$ of elements $K \in \mathcal{T}_{h,i} \cup \mathcal{T}_{=,i}$ at the nodes $p_{n,i}$;
  average the error estimator $\epsilon_{K,i}$ of elements $K \in \mathcal{T}_{h,i} \cup \mathcal{T}_{=,i}$ at the nodes $\epsilon_{n,i}$;
  average the smoothness indicator $\sigma_{K,i}$ of elements $K \in \mathcal{T}_{h,i} \cup \mathcal{T}_{=,i}$ at the nodes $\sigma_{n,i}$;
  if $\epsilon_{n,i} > \epsilon^*_{i+1}$ **then**
    adapt the current size $h_{n,i}$ to the new $h^*_{n,i+1}$ provided by the formulation (5.9);
  else
    leave the current size $h^*_{n,i+1} = h_{n,i}$ unmodified;
  end
  give the new metric to MMG, which generates the refined mesh $\mathcal{T}_{K,i+1}$;
  project the new polynomial degree map $p^*_{K,i+1}$ from $\mathcal{T}_{K,i}$ to $\mathcal{T}_{K,i+1}$;
  project the solution of $\mathcal{T}_{K,i}$ on $\mathcal{T}_{K,i+1}$;
  i+=1;
end

5.1.5 Choice of the target error

To complete the adaptation algorithm, we need to specify how the target error ϵ^*_{i+1} is defined. This is given by a revisited fixed-fraction marking strategy, which allows for a rapid convergence of the mesh adaptation algorithm towards the optimal mesh. The classical version of the fixed-fraction marking strategy [181, 314, 264] controlling the percentage of the mesh to refine (and often the percentage to coarsen) shows poor results in this context. This is due to the fact that the prescribed decrease of sizes here directly depends on the magnitude of the error estimator. A strong gap in the error estimator magnitude between the highest value and the mean value over the domain would lead to a localized refinement only in zones with the highest intensity of the error estimator, and an insufficient refinement in zones with medium intensity of the estimator. This leads to an early and unwanted stop of the refinement algorithm.

The strategy we propose in this work allows for controlling the number of dofs in the mesh, in order to prescribe the appropriate target error. We assume that the desired number of elements $\#\text{elt}^*_{i+1}$ in the new mesh can be computed thanks to the prescribed reduction of element sizes [47, 246]:

$$\mathrm{N}^*_{i+1} = \#\text{elt}^*_{i+1} = \sum_K \left(\frac{h_{K,i}}{h^*_{K,i+1}} \right)^d , \tag{5.15}$$

where d is the number of dimensions of the mesh, and the characteristic sizes of the elements $h_{K,i}$ and $h^*_{K,i+1}$ considered here are an average of the lengths associated with the nodes n of the element K.

In the context of hp-adaptation the formula above can be rewritten keeping into account a variable polynomial degree for the element K.

$$\mathrm{N}^*_{i+1} = \#\text{dofs}^*_{i+1} = \sum_K N_p(p^*_{K,i+1}) \left(\frac{h_{K,i}}{h^*_{K,i+1}} \right)^d , \tag{5.16}$$

where $N_p(p^*_{K,i+1})$ is the number of dofs associated with the desired polynomial degree in the element K.

We introduce a parameter $f_r > 1$ used to define the target error, which corresponds to the increase of dofs wanted at each adaptation step.

The target error is then defined as the value ϵ^*_{i+1} such that $\mathrm{N}^*_{i+1}(\epsilon^*_{i+1}) = f_r \cdot \mathrm{N}_i$. This value is in practice obtained by an iterative procedure, using the bisection method from the Python package SciPy to solve the equation:

$$g(\epsilon^*_{i+1}) = \mathrm{N}^*_{i+1}(\epsilon^*_{i+1}) - f_r \cdot \mathrm{N}_i = 0 , \tag{5.17}$$

where N_{i+1} is the number of dofs wanted in the following adaptation step. Using a bisection method for numerically solving the equation (5.17) requires that the function $g(\epsilon^*_{i+1})$ is a continuous function defined on an interval $[a, b]$, where $g(a)$ and $g(b)$ have opposite signs. The interval is bounded by the minimum and maximum values of the error estimator at the adaptation step i on the domain $\mathcal{T}_K$ $[\min(\epsilon)_{\mathcal{T}_{K,i}}, \max(\epsilon)_{\mathcal{T}_{K,i}}]$. With $f_r > 1$, the inequality $g(b) = g(\max(\epsilon)_{\mathcal{T}_{K,i}}) < 0$ always applies, because $\mathrm{N}^*_{i+1}(\max(\epsilon)_{\mathcal{T}_{K,i}}) = \mathrm{N}_i$ and no

elements are marked for refinement. On the other hand, although $g(a) = g(\min(\epsilon)_{\mathcal{T}_{K,i}}) > 0$ is verified most of the time, in some cases the approximated formulas in equations (5.15) and (5.16) can provide negative values of the function g. This is caused by the constraints imposed by the smoothness, maximum polynomial degree, or maximum refinement factor (or a combination of the three). Marking all elements for refinement could still provide $\mathrm{N}^*_{i+1}(\min(\epsilon)_{\mathcal{T}_{K,i}}) < f_r \cdot N_i$. In this case, a target error $\epsilon^*_{i+1} = 0$ is directly chosen without launching the root-finding algorithm, and all the elements are marked for refinement, while still respecting all the aforementioned constraints.

For h-adaptation, the new lengths are computed from the equation (5.9), the new number of elements is computed from the equation (5.15) and the iterative procedure results in the target error ϵ^*_{i+1} needed to obtain $\#\mathrm{elt}^*_{i+1} = f_r \cdot \#\mathrm{elt}_i$ elements. For hp-adaptation, the new lengths and the new polynomial degrees are computed respectively with the equations (5.9) and (5.11), the new number of dofs is computed with the equation (5.16) and the iterative procedure results in the target error ϵ^* needed to obtain $\#\mathrm{dofs}^*_{i+1} = f_r \cdot \#\mathrm{dofs}_i$.

5.1.6 Particular treatment for hybrid meshes

The use of prisms or hexahedra in boundary layers is beneficial for computing the gradients with high-accuracy or satisfying precisely near-wall resolution in terms of y^+ for turbulent flows. A common practice involves the definition of a structured (extruded regular quadrilaterals) or pseudo-structured (extruded unstructured quadrilaterals or triangles) zone capturing the boundary layer around the body, which blends with an outer region composed of tetrahedra [200, 27, 281]. The adaptation methodology proposed in Sections 5.1.3 and 5.1.4 is straightforwardly extended to hybrid meshes. During the adaptation process, boundary layer structured or pseudo-structured elements are maintained while the remeshing is applied only on tetrahedra. Regarding the hp-adaptation approach, the p-refinement applies to any type of element, overcoming the constraint that elements with a fixed geometry impose in a pure h-adaptive context. This allows for improving the resolution in boundary layers as well, if the error indicator is active in these regions. No specific treatment is required concerning MMG, which preserves quadrilaterals and prisms by default. The preservation of hexaedra, pyramids and general polygonal elements can be implemented in a straightforward manner.

5.2 Results on steady laminar configurations

In this section we assess the presented adaptive algorithms on three 2D/3D steady laminar test cases: the 2D flow past a NACA0012 airfoil, the 3D flow past a sphere and the 3D flow past a delta wing.

The parameters of the adaptation process for the laminar test cases are chosen as follows:

- $h_{grad} = 1.5$, which is higher than the default value of MMG $h_{grad} = 1.3$. This value shows a good compromise between excessive refinement induced by a low value of h_{grad}, and the loss of isotropy and mesh quality induced by a high value.

- h_{Hausd} = 0.01 for the sphere, which has constant curvature and diameter equal to 1, and h_{Hausd} = 0.002 for the NACA0012, which has a chord size equal to 1, but maximum thickness equal to 0.12 and a strong curvature at the leading edge. In general h_{Hausd} = 0.01 is a suitable value for an object of size 1 in each direction [6]. For smaller (respectively larger) objects, the value of the Hausdorff parameter needs to be decreased (respectively increased). This parameter is irrelevant for the delta wing, which does not present curved boundaries.

- $h_{\min}$ is set to a low value, which is never reached in the presented adaptation processes.

- $h_{\max}$ is set as the size of the domain where far-field boundary conditions are imposed.

- r_h = 2 such that between two steps of the adaptation process, the edge of the equilateral triangle can at most halve its size.

- f_r = 1.5 is used for all the laminar test cases. A sensitivity study of the adaptation process to this parameter is carried out for the NACA0012 case.

- $p_{\min}$ = 1 is the minimum polynomial degree of the spatial discretization allowed for *hp*-adaptation. The initial simulations are performed with a uniform polynomial degree p = 1 for *hp*-adaptation.

- $p_{\max}$ = 3 is the maximum polynomial degree that the spatial discretization is allowed to reach for *hp*-adaptation. We have chosen not to employ higher polynomial degree discretizations, since *ad hoc* load balancing techniques, mandatory to fully exploit the potential of *hp*-adaptation, are not here employed. Moreover, increasing the order in case of implicit time integration, increases the stiffness of the linear problem, which requires stronger and more robust preconditioners, and leads to a significant increase of the computational time. The fourth order p = 3 discretization is considered here as a good compromise between the benefits provided by high order methods and the improved robustness of lower order methods.

Initial meshes are easily created with MMG by providing a uniform coarse metric and forcing the location of the mesh nodes on the wall. By using the mesh smoothing capability of MMG with h_{grad} = 1.5, the resulting mesh is uniform and very coarse far away from the body and presents a smooth refinement towards the solid boundaries. By using this approach, the initial metric at the nodes needed by the adaptation module is provided directly by the remesher.

The error on drag and lift coefficients is computed as the absolute value of the difference between the coefficient of the adaptive simulation at the i^{th} step, Coef_i, and a reference coefficient Coef_{ref} extracted from a simulation with a finer resolution, normalized by the latter.

$$|\Delta\text{Coef}| = \left| \frac{\text{Coef}_i - \text{Coef}_{\text{ref}}}{\text{Coef}_{\text{ref}}} \right|$$

The error on h- and hp-adapted meshes is compared to uniformly refined meshes, obtained with MMG by prescribing at each nodes half the corresponding size on the previously refined mesh.

5.2.1 Laminar flow past a NACA0012 airfoil

The h- and hp-adaptive algorithms are assessed in the present section for a flow representative of aerodynamic applications. We consider the steady, subsonic and viscous flow around the NACA0012 airfoil, which is a well-known test case used in CFD for validation purposes (turbulence models, shape optimization, numerical schemes), and among others for steady mesh adaptation [180, 227, 311, 33].

The configuration studied here has a free-stream Mach number of $M_\infty = 0.5$, angle of attack $\alpha = 1°$ and Reynolds number of $Re = 5000$. The main feature of this flow is the thin, laminar boundary layer developing over the airfoil.

Reference results are obtained on a finer mesh with a $p = 3$ discretization highly refined around the body. Those results compare favorably with other results present in the literature, as seen in Table 5.1.

	C_D	C_{Dfric}	C_{Dpres}	C_L
Swanson *et al.* (NASA report) [280]	0.055980	0.032758	0.023222	0.018464
Yano & Darmofal [311]	0.055317	-	-	0.018274
Haga *et al.* [5]	0.05589	-	-	0.018923
Balan *et al.* [33]	0.055317	-	-	-
current	0.055662	0.032801	0.022861	0.019983

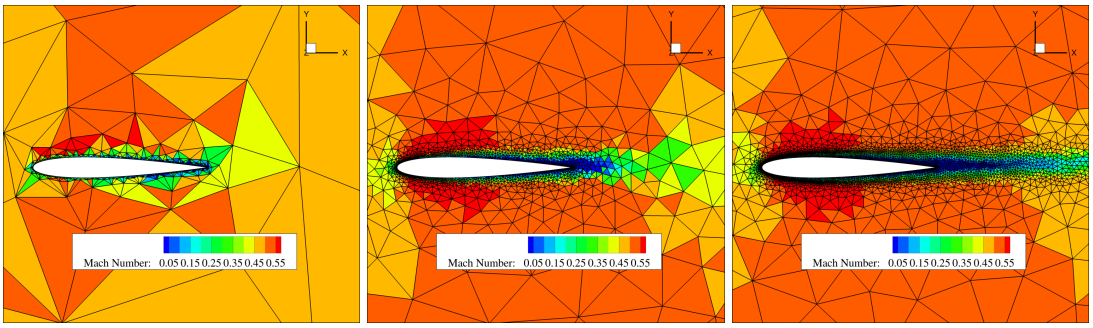
Table 5.1: Laminar flow past a NACA0012 airfoil at $Re = 5000$. Integral flow quantities found in the literature and for the present reference.

The small discrepancies in the C_L with respect to other references are due to the different analytical definition of the NACA0012 used in this paper. We employed the revised definition [1], while the reference values provided in table 5.1 have been obtained using the exact formula of the NACA0012 airfoil.

The mesh shown in the left panel in figure 5.2 contains 789 triangular elements and is used as initial mesh. It is a C-type mesh, centered at the airfoil mid chord with a radius of 80 chords and extends for 100 chords in the rear region. Adiabatic viscous conditions have been imposed on the walls of the airfoil, and free-stream far-field boundary conditions have been imposed on the external boundaries.

Seven series of computations are performed: three uniformly refined series with $p = 1, 2, 3$, three pure h-adaptation series with $p = 1, 2, 3$ referred later as $h(p = 1)$, $h(p = 2)$, $h(p = 3)$ and one hp series. Figure 5.2 presents the meshes generated during the adaptation steps for the $h(p = 2)$ series. We can see that the adaptation yields a concentration of the refinement in the boundary layer region around the airfoil, the stagnation region near the leading edge and the wake in the region downstream of the airfoil.

The error on the drag coefficient and lift coefficient integrated over the walls of



(a) (b) (c)

Figure 5.2: Laminar flow past a NACA0012 airfoil at $Re = 5000$. Zoom on the Mach number iso-contours of the cell-averaged solution (a) on the initial mesh (789 elements/4734 dofs), (b) on the 3rd adapted mesh (3446 elements/20676 dofs) and (c) on the 6th adapted mesh (8974 elements/53844 dofs), for the $h(p = 2)$ simulations.

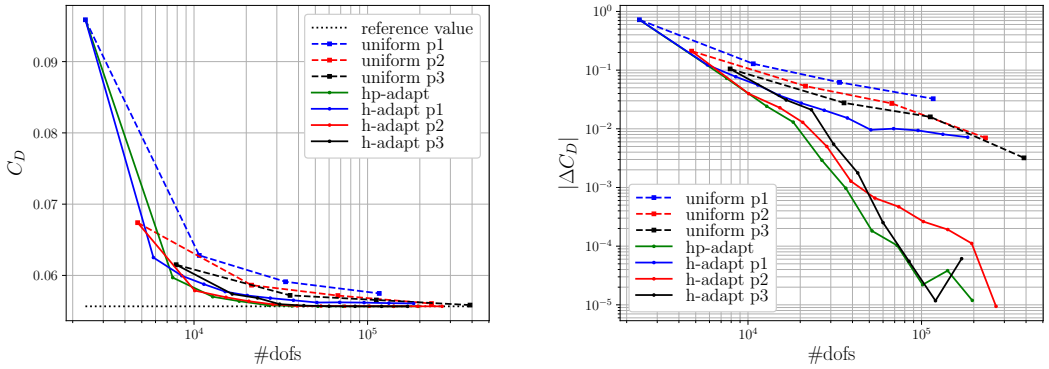


Figure 5.3: Laminar flow past a NACA0012 airfoil at $Re = 5000$. Comparisons between the convergence history of the drag coefficient for *hp*-adapted meshes, *h*-adapted meshes and uniformly refined meshes in $p = 1$, $p = 2$, $p = 3$. C_D vs. number of dofs (left) and $|C_D - C_{D_{ref}}|/C_{D_{ref}}$ vs. number of dofs (right).

the NACA0012 airfoil, C_D and C_L, is evaluated for the different meshes by computing the difference between the values obtained on the adapted mesh and those from the reference $p = 3$ simulation.

The convergence history of the C_D and C_L (left) and the error (right) versus the number of dofs of the performed simulations are shown respectively in figures 5.3 and 5.4. For all pure *h*-adaptive simulations, four adaptation steps are sufficient for $p = 1$, $p = 2$ and $p = 3$ to reach a level of error on the drag coefficient lower than the error obtained on the corresponding finest uniformly refined meshes. The *hp*-adaptive process is also capable in only four adaptations to reach an error level lower than the uniform $p = 3$ most refined mesh, and with less degrees of freedom than $p = 1$, $p = 2$ and $p = 3$ *h*-adaptive simulations. A similar behavior is observed for the lift coefficient C_L in figure 5.4.

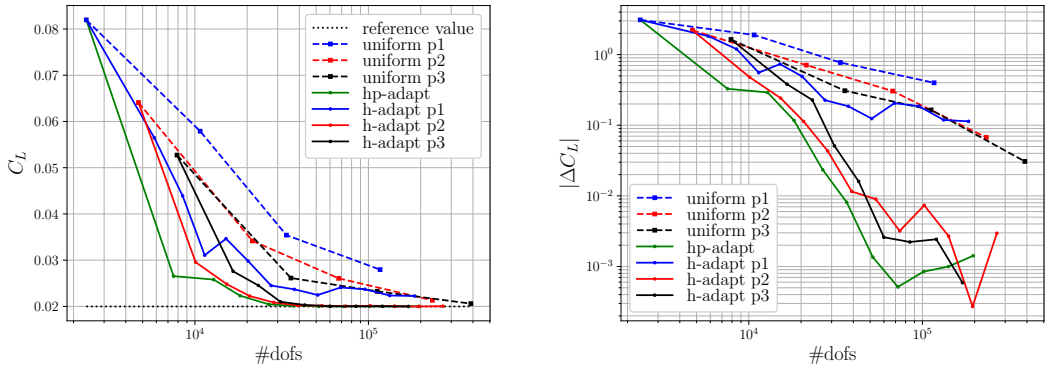
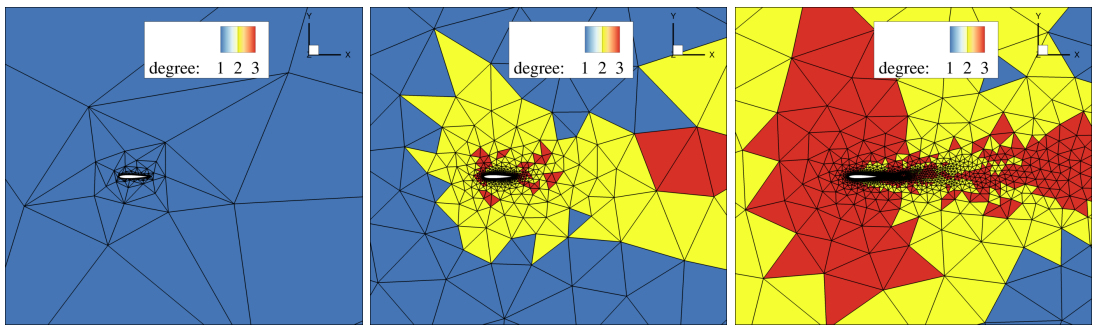


Figure 5.4: Laminar flow past a NACA0012 airfoil at $Re = 5000$. Comparisons between the convergence history of the lift coefficient for hp-adapted meshes, h-adapted meshes and uniformly refined meshes in $p = 1$, $p = 2$, $p = 3$. C_L vs. number of dofs (left) and $|C_L - C_{L_{ref}}|/C_{L_{ref}}$ vs. number of dofs (right).

For the higher order adapted simulations, $h(p = 2)$, $h(p = 3)$ and the hp-simulations, we observe that at the end of the adaptive process, the error in the drag and lift coefficients is reduced by around two orders of magnitude with respect to the uniformly refined simulations. Note that the stagnating behavior of the error below a certain threshold is explained by the relative uncertainty about the reference value, which is not exact but obtained from a computation on a very fine mesh.

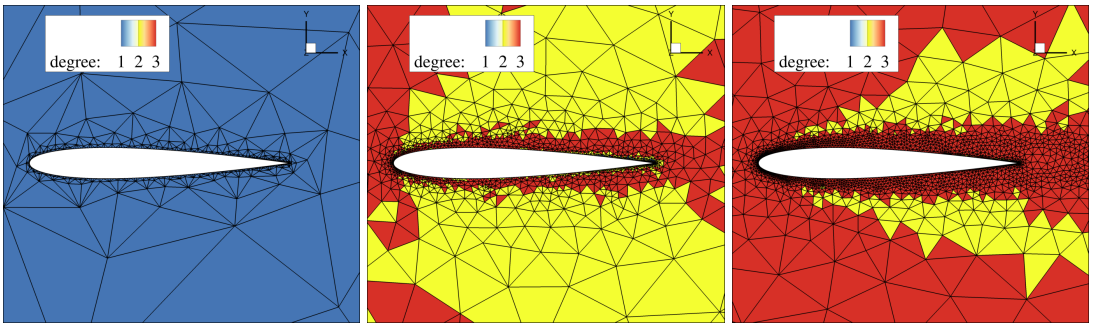
The error on second order $p = 1$ simulations decreases at a lower rate than $h(p = 2)$, $h(p = 3)$ and hp simulations, which is due to the combination of several factors. Namely, the initial lower count of dofs, the increased numerical error compared to higher orders and, as discussed in Section 5.1.1, the lower accuracy of the SSED part of the error estimator for low-order discretizations because of the related limited spectral content inside the cells.



(a) (b) (c)

Figure 5.5: Laminar flow past a NACA0012 airfoil at $Re = 5000$. Contours of the polynomial degree distribution (a) on the initial mesh (4734 dofs), (b) on the 3rd adapted mesh (18219 dofs) and (c) on the 6th adapted mesh (51834 dofs), for the hp simulations.

Figure 5.5 displays the polynomial degree distribution for the first, third and sixth adaptation steps obtained from the *hp*-adaptation process. A part of the wake region is first detected as non-smooth where the elements are still too large to accurately capture the physical features, and present values of the smoothness indicator lower than 1. As the mesh is refined, these regions are detected as smooth and the wake is adapted first by increasing the polynomial degree and then by decreasing the size of the element (see figures 5.5(b) and 5.5(c)).



(a) (b) (c)

Figure 5.6: Laminar flow past a NACA0012 airfoil at $Re = 5000$. Zoom of figure 5.5.

A close-up view of the p distribution around the airfoil is displayed in figure 5.6. Since this flow presents globally smooth features, the general trend of the *hp*-adaptation is to first increase the polynomial degree, and then decrease the size when the polynomial degree in the element saturates at the maximum allowed value $p_{\max}$. We can still notice from figure 5.6(c) that the exterior part of the shear layers of the airfoil remains at $p = 2$ over the course of the adaptation process, while the element size decrease in that area. This behavior is explained by the gradation of the remesher (see Section 3.1.2), which ensures a smooth increase in the element size between the highly refined near wall area and the outer part. This results in a slight over-refinement in zones where the error estimator is not necessarily high. However, this behavior is compensated by the *hp*-adaptation algorithm, which does not increase p in these regions.

The influence of the refinement factor f_r, which controls the increase of number of dofs at each adaptation step has also been assessed for *hp*-adaptive simulations. The corresponding error plots are presented in figure 5.7. The algorithm is tested by selecting seven different values of f_r from 1.2 to 2, and by analyzing the effect of this parameter on the convergence history of the C_L and C_D values. As can be seen from figure 5.7, all values of f_r drive the integral errors to convergence. This parameter, however, affects the rate of convergence in terms of number of dofs. In particular, by increasing f_r, more elements are selected for refinement, including some elements presenting a low error. On the other hand, a small value of f_r mainly refines the zones presenting a strong error, leading to a more rapid convergence in terms of dofs. However, despite the fact that a very small value of f_r makes the algorithm converge to the same value with much less dofs, more adaptation steps are necessary to achieve convergence. This process can be time consuming since more remeshing steps and more flow simulations must be realized. In addition, too small refinement factors lead to oscillations in the error because too few

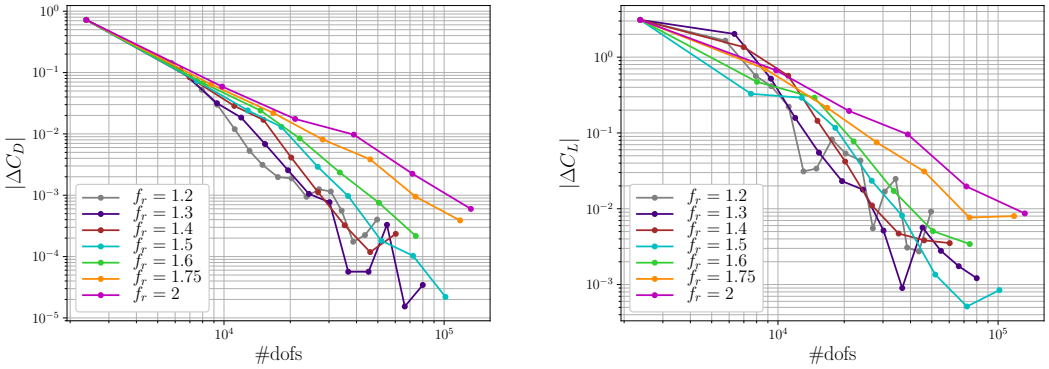


Figure 5.7: Laminar flow past a NACA0012 airfoil at $Re = 5000$. Study on the independence from the refinement factor on hp-adaptive simulations.

elements are adapted at each step, as can bee seen for $f_r = 1.2$. The choice of the increment in terms of number of dofs from one step to another can thus be seen as a trade-off between the number of adaptation steps and the computational time. For this reason, in the present section, the three laminar configurations are adapted using a value $f_r = 1.5$, which proved to be an efficient choice to ensure a good convergence for a low number of iterations.

5.2.2 Laminar flow past a sphere at low Reynolds number

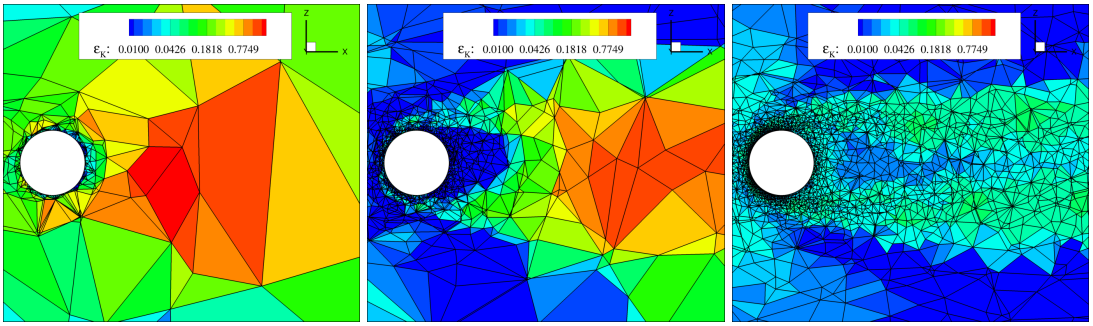
In this section, the h- and hp-adaptive algorithms are validated performing simulations of the context of the three-dimensional flow past a sphere at $M_\infty = 0.2$ and low Reynolds number $Re = 200$. This particular configuration, already studied by Rueda *et al.* [256] in the context of p-adaptive DG methods, is characterized by a separation determined by the viscous processes at the wall, which creates a recirculation region behind the body. This flow regime presents a steady and axi-symmetrical behavior, which is lost for higher Reynolds numbers. Reference results are obtained from a $p = 3$ computation on a highly-refined mesh around the body and compared to results from the literature [256, 126, 118] in table 5.2.

	C_D
Rueda *et al.* [256]	0.7771
Fadlun *et al.* [118]	0.7567
Fornberg [126]	0.7683
current	0.7782

Table 5.2: Laminar flow past a sphere at $Re = 200$. Integral flow quantities found in the literature and for the present reference.

The initial mesh counts 6651 tetrahedral elements (plot in figure 5.8(a)): the density of the mesh on the surface of the body is high enough to obtain an accurate representa-

tion of the geometry. This allows the remesher to keep the right curvature of the surface when the mesh is successively refined during the adaptation steps. The mesh is spherical with a radius equal to 100 diameters of the sphere of unit diameter, and its origin is located at the center of the solid body. The surface of the sphere is treated as a viscous adiabatic wall, and the external boundaries are modeled with free-stream far-field boundary conditions.



(a) (b) (c)

Figure 5.8: Laminar flow past a sphere at $Re = 200$. Close-up view of the error estimator iso-contours computed (a) on the initial mesh (6651 elements/133020 dofs), (b) on the 3rd adapted mesh (37920 elements/758400 dofs) and (c) on the 6th adapted mesh (154171 elements/3083420 dofs), for the $p = 3$ simulations. Slices of $y = 0$.

Three h-adaptive simulations named $h(p = 1)$, $h(p = 2)$, $h(p = 3)$ are performed for 3 different polynomial degrees. An hp-adaptive simulation is also performed using $p_{\max} = 3$ and a constant $p = 1$ degree distribution on the initial mesh.

Close-up views of the $h(p = 3)$ meshes are shown in figure 5.8, namely the initial mesh (figure 5.8(a)) and the meshes obtained after 3 and 6 adaptation steps (respectively figures 5.8(b) and 5.8(c)). The adaptation procedure is able to refine the mesh in flow regions of interest, around the body and in the sphere wake. The iso-values of the error estimator computed at the three steps of the adaptive process are presented, in which we have used the same color-scale for the three plots. The initial mesh presents very high values of the error estimator in the wake, resolved here by a few very coarse elements, and low values on some small surface elements, which appear to be already sufficiently refined. We remind here that the values of the error estimator used for the adaptation are normalized by their min/max values in the domain. As the adaptation process progresses, the error estimator varies in a more limited range of values. This is a manifestation that by improving the resolution, the distribution of the error estimator becomes more regular. At the 6th adaptation step, the zone of interest of the flow does not present any element with a significant difference in the error estimator value from the average. The adaptation process can be considered as converged, as from this point on, the h-refinement is going to take place mainly in the far wake, where the error estimator remains relatively high until the wake is fully refined up to the downstream boundary. When this happens, it is not necessary to pursue the adaptation process.

Figure 5.9 shows the convergence history of the drag coefficient (right) and its error with respect to the reference value (left) for h-/hp-adapted and uniformly refined

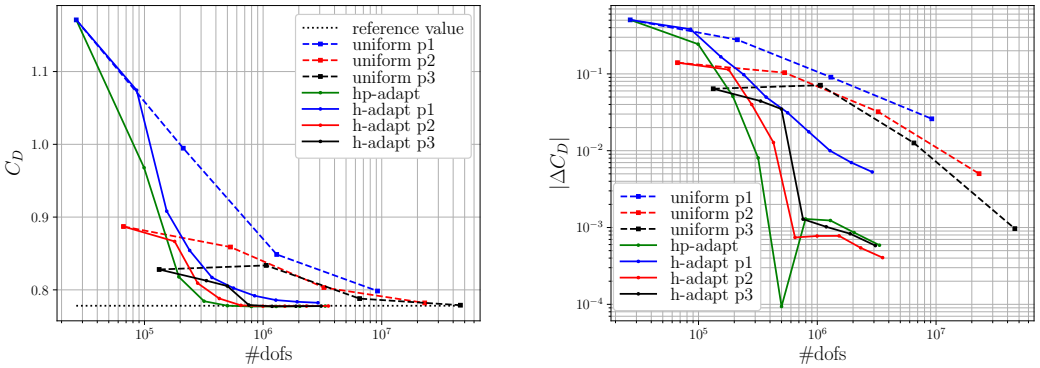


Figure 5.9: Laminar flow past a sphere at $Re = 200$. Comparisons between the convergence history of the drag coefficient for hp-adapted meshes, h-adapted meshes and uniformly refined meshes in $p = 1$, $p = 2$, $p = 3$. C_D vs. number of dofs (left) and $|C_D - C_{D_{ref}}|/C_{D_{ref}}$ vs. number of dofs (right).

simulations. We can observe how the adaptive algorithms are capable of reaching the correct values of C_D with less dofs with respect to the uniform meshes. Pure $h(p = 1)$ adaptation converges more slowly than high-order h-adaptive simulations, as expected for the considerations already made for the 2D flow past a NACA0012 airfoil. Nonetheless a consistent gain of around 10 times the number of dofs is observed with respect to $p = 1$ uniformly adapted meshes. The hp-adaptive procedure converges slightly faster than $h(p = 2)$ and $h(p = 3)$ adaptations in the first steps.

When considering stiff problems (e.g. featuring geometrical, physical singularities or turbulence), it can be challenging to obtain a high-order solution on the initial coarse mesh without increasing gradually the polynomial degree first, due to robustness issues. This difficulty is alleviated when considering our hp-adaptation method as the progressive p-refinement based on smoothness detection is integrated in the adaptation process and reinforces the high-order solver robustness. We choose to assess the performance of the two h- and hp-adaptive strategies in terms of number of dofs only (represented by the plots *error vs. #dofs* for the three laminar test cases), rather than the performance in terms of CPU time which can vary depending on the efficiency of the implicit time stepping approach used. In order to be relevant, a comparison based on CPU time would require an accurate load balancing strategy and optimized implicit schemes. We also note that since we are dealing with static adaptive strategies, the CPU time necessary to perform one adaptive step (error estimation, metric map computation, degree map computation, remeshing, interpolation) is negligible with respect to the CPU time necessary to perform the implicit steady flow simulation.

5.2.3 Laminar flow past a delta wing at low Reynolds number and high angle of attack

In this section, the DG *h*- and *hp*-adaptation strategies are assessed for a delta wing configuration at laminar conditions with inflow Mach number equal to $M_\infty = 0.3$, the angle of attack $\alpha = 12.5°$ and Reynolds number, based on the root chord length $L = 1$, $Re = 4000$. The delta wing features a sharp leading edge and a blunt trailing edge.

The main feature of this case is the rolling-up of the flow at the leading edge yielding the development of a vortex together with a secondary vortex, which persists in the wake.

This test case has been studied in the first three High-Order Workshops [5], as well as by Leicht & Hartmann [180], Ceze & Fidkowski [75] and Tsolakis *et al.* [283] in the framework of *h/hp*-adaptation. The reference values of drag and lift coefficients C_D and C_L for the current computations have been obtained by extrapolation of the last points of $p = 3$ uniform simulations. The results are shown in table 5.3 and compared to values used in the literature.

	C_D	C_L
Hartmann *et al.* [5]	0.1658	0.347
Ceze *et al.* [5]	0.16578	0.34771
current	0.165617	0.34716

Table 5.3: Steady laminar flow past a delta wing at $Re = 4000$. Integral flow quantities found in the literature and for the present reference.

The initial mesh of 3775 tetrahedral elements, as for the two previous test cases, is slightly refined near the surface of the wing, with a fast coarsening when moving towards the far-field boundaries. The geometry and the boundaries are built according to the High-Order Workshop meshes: the domain is a box of size $10.62L \times 5.31L \times 12.74L$, respectively in the directions x, y, z in figure 5.10. Half of the model is simulated with symmetric boundary conditions applied on the wall-normal/streamwise plane of symmetry. The wing surface is modeled as a no slip isothermal wall with $T_w = T_\infty$, and the external boundaries as free-stream far-field boundary conditions.

In figure 5.10 the high-order solution on the initial coarse mesh (left) and on the eighth iteration of the *hp*-adaptive algorithm, has been reconstructed on a very fine post-processing mesh, and sliced on the $y = 0$ and $x = 1.77L$ planes. The Mach number iso-contour map is shown in the sliced planes, the skin friction coefficient colors the surface of the wing, and the transparent iso-contour of the Mach number represents $M = 0.2$. Looking at the surface of the wing, we can notice the clustering of the elements along the sharp leading edge, and especially in the rear part, where the geometry changes abruptly into the blunt trailing edge. This refinement allows for an accurate generation of the trailing vortices. A sharp representation of these vortices is also observed further away in the wake, as seen from the x-cut plane and the Mach iso-surfaces enveloping the vortical structures behind the body.

Figures 5.11 and 5.12 collect the results obtained on the sequences of uniformly refined meshes with $p = 1, 2, 3$, h-adapted meshes ($h(p = 1)$, $h(p = 2)$, $h(p = 3)$) and

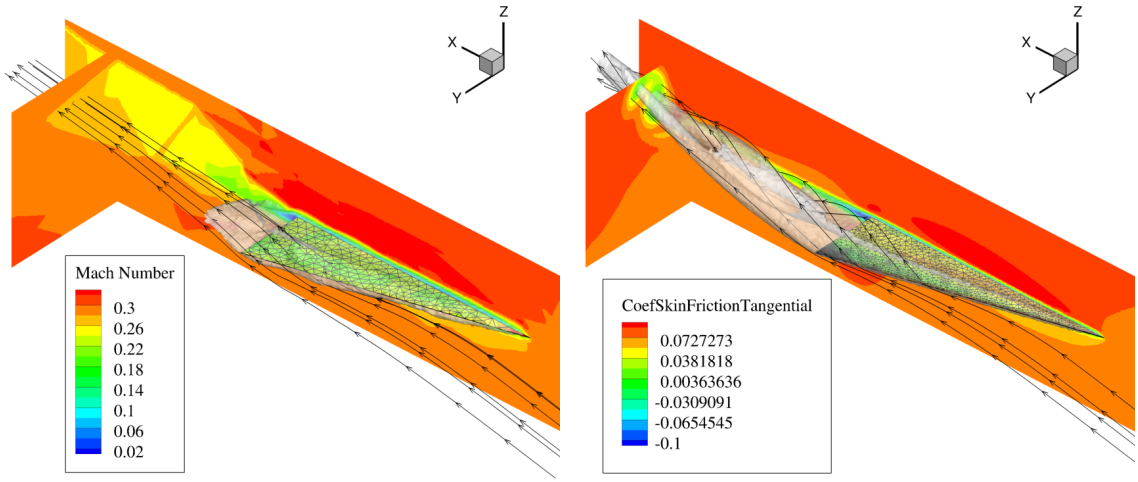


Figure 5.10: Laminar flow past a delta wing at $Re = 4000$. Slices of $y = 0$ and $x/L = 1.77$ planes of the reconstructed high-order solution on a finer post-processing mesh. Initial mesh (15100 dofs) at left, and 8^{th} hp-adapted mesh (1543678 dofs) at right. Mach contour for volume slices, friction coefficient for surfaces, isosurface $M = 0.2$ transparence.

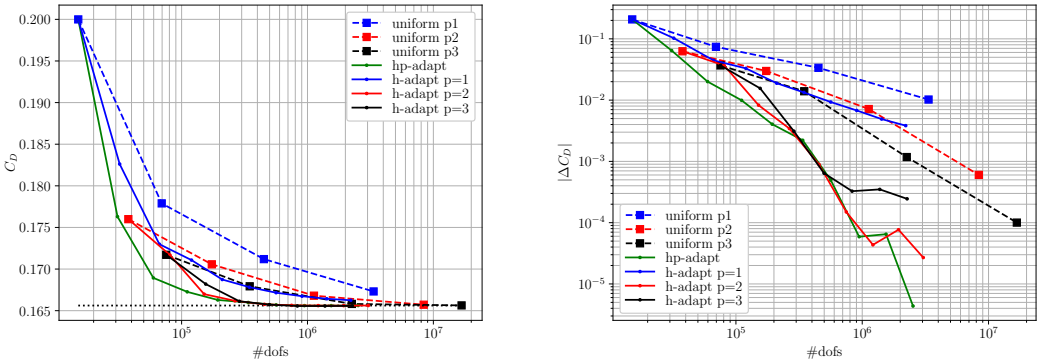


Figure 5.11: Laminar flow past a delta wing at $Re = 4000$. Comparisons between the convergence history of the drag coefficient for hp-adapted meshes, h-adapted meshes and uniformly refined meshes in $p = 1$, $p = 2$, $p = 3$. C_D vs. number of dofs (left) and $|C_D - C_{D_{ref}}|/C_{D_{ref}}$ vs. number of dofs (right).

hp-adapted meshes. C_D and C_L values are shown in left plots, while their errors with respect to the reference solution can be found in right plots. $h(p = 1)$ simulations, despite being clearly the least competitive strategy to reduce the number of dofs, still reaches the same level of accuracy on the errors on target quantities compared to the most uniformly refined $p = 1$ mesh with around respectively 90% (for C_D) and 50% (for C_L) less dofs. As observed for the previous test cases, the use of an hp-adaptive strategy ensures a faster convergence of the error for the first steps of the adaptive process, while

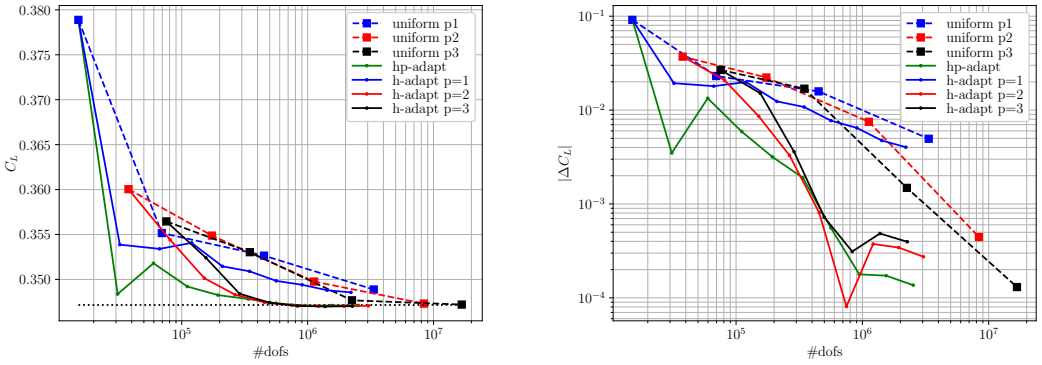
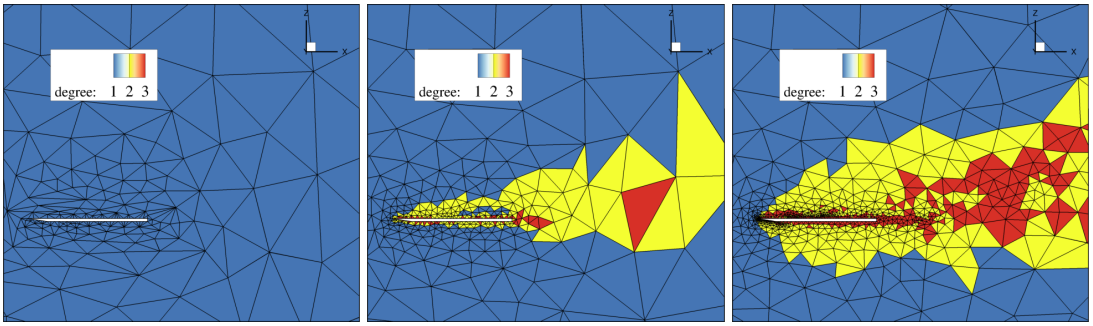


Figure 5.12: Laminar flow past a delta wing at $Re = 4000$. Comparisons between the convergence history of the lift coefficient for *hp*-adapted meshes, *h*-adapted meshes and uniformly refined meshes in $p = 1$, $p = 2$, $p = 3$. C_L vs. number of dofs (left) and $|C_L - C_{L_{ref}}|/C_{L_{ref}}$ vs. number of dofs (right).

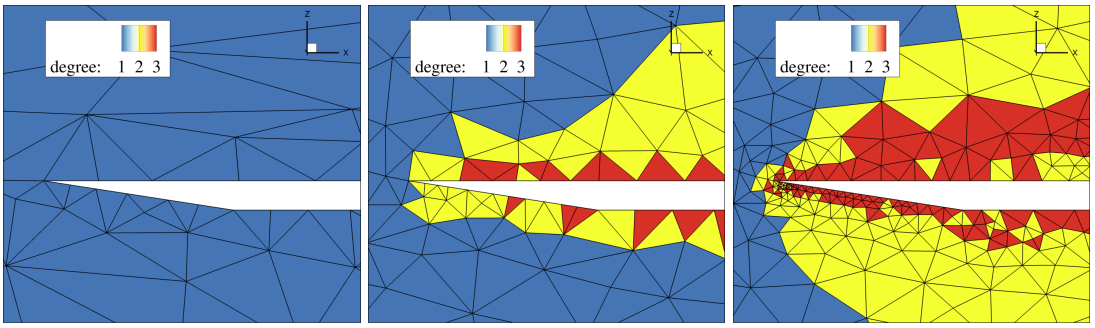
the error curve is superposed to that from the $h(p = 2)$ simulations for a higher number of dofs (and lower error). For this test case slightly lower performances of the adaptive process are observed for $h(p = 3)$ simulations, where the dofs in regions far from the body have a strong impact on the dofs count, while not improving the solution in a satisfactory way. However the slower convergence achieved for $h(p = 3)$ simulations with respect to $h(p = 2)$ and *hp* simulations starts showing for values of the error on the C_D and C_L lower than 10^{-3}, in a region where the extrapolation procedure of the reference value could affect the reliability of the evaluation of the error.

The same remarks made for the previous test cases regarding the peaks of errors in the low error region of the plot are still valid. In this region, oscillations around the reference value can lead to strong low and high peaks lacking of physical meaning in the adaptive procedure.



(a) (b) (c)

Figure 5.13: Laminar flow past a delta wing at $Re = 4000$. Contours of the polynomial degree distribution (a) on the initial mesh (15100 dofs), (b) on the 3rd adapted mesh (110836 dofs) and (c) on the 6th adapted mesh (563580 dofs), for the *hp* simulations.



(a) (b) (c)

Figure 5.14: Laminar flow past a delta wing at $Re = 4000$. Close-up view of figure 5.13

For the hp-adaptive simulations, figures 5.13 and 5.14 display the polynomial degree distribution on the xz plane at $y = 0$ for the initial mesh, as well as the meshes obtained after 3 and 6 adaptation steps. The geometric singularities are refined first, then the shear layers around the body and the wake are progressively refined over the course of the adaptation process. In particular, the elements around the leading edge singularity keep a $p = 2$ polynomial degree for several adaptive steps. This is due to the smoothness indicator marking this zone as non-smooth. As the singular region gets better resolved, those $p = 2$ elements located around the corner progressively switch to $p = 3$.

5.3 Results on a turbulent configuration: the isothermal subsonic turbulent jet flow issued from the PPRIME nozzle at $Re_D = 10^6$

In this section, the DG-hp methodology is assessed for RANS simulations of a turbulent jet configuration described in [63], for which experiments have been performed at the Bruit & Vent jet-noise facility of the Institut PPRIME, Poitiers, France, and reference data are available [63]. Computational references are available as well in the literature. This configuration has been simulated using LES by Brès *et al.* [63] to analyze features such as near-wall adaptive mesh refinement, synthetic turbulence and wall modeling inside the nozzle. Hybrid RANS/LES simulations have been performed by Gand *et al.* [135] to assess the generation of a turbulent inflow, and RANS simulations by Neifeld *et al.* [222] for jet noise prediction purposes with an eddy relaxation source model.

The operating conditions are defined in terms of the total pressure ratio $p_t/p_\infty = 1.7$ and total temperature ratio $T_t/T_\infty = 1.15$. The jet is isothermal ($T_j/T_\infty = 1$), the Mach number is $M_j = U_j/c_j = 0.9$, and the Reynolds number $Re_D = \rho_j U_j D/\mu_j \simeq 10^6$, where the subscript j refers to jet properties. U_j is the mean jet exit longitudinal (x-direction) velocity, c is the speed of sound, $D = 0.05m$ is the exit diameter of the nozzle, ρ is the density and μ is the dynamic viscosity.

The axi-symmetric computational domain extends from approximately -10D (the length of the nozzle in the x-direction) to 50D in the streamwise direction x, and from

$-30D$ to $30D$ in the radial direction, and includes the nozzle geometry with its exit centered at $(x, y, z) = (0, 0, 0)$ (slices shown in figure 5.15).

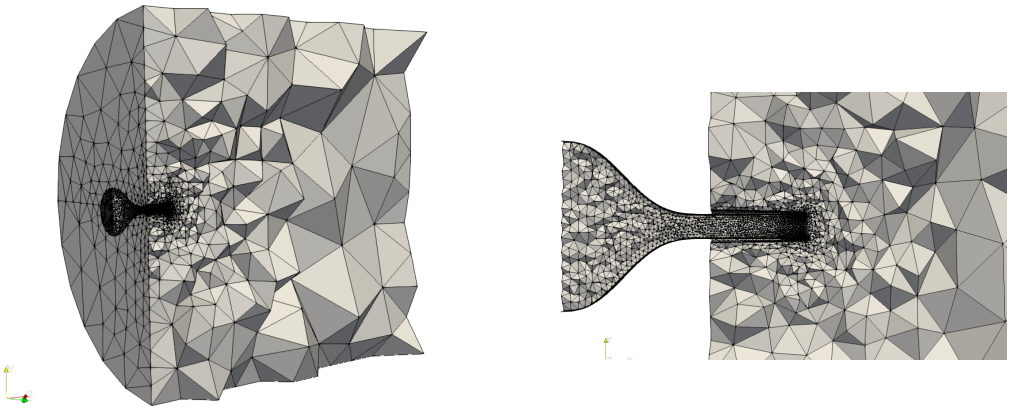


Figure 5.15: RANS PPRIME nozzle configuration. Slices of the initial mesh, entire domain (left) and zoom on the nozzle (right)

A slow coflow at Mach number $M_\infty = 0.01$ is imposed outside the nozzle (the flow is at rest in the experiment). This prevents spurious recirculation and facilitates flow entrainement. All solid surfaces of the nozzle are treated as no-slip adiabatic walls.

The initial mesh is generated with the mesh generation software ANSA [3]. The geometry of the body and the far field boundaries are created and meshed using triangles, and the surface of the body (the internal and external walls of the nozzle) is projected normal to the wall to generate the prismatic boundary layer, whose first cell measures $0.0003D$ in the wall-normal direction. The wall-normal spacing of these near-wall prisms is set such that we obtain an accurate representation of the turbulent boundary layer using the SA model with $p_{\max} = 3$, following the guidelines in Section 4.2.

The rest of the volume is then filled with tetrahedra. The initial mesh created by ANSA is processed by MMG employing the size field output of a first "analysis step". The remesher is actually able to output an isotropic size field from any given mesh. The parameters considered for the *hp*-adaptive simulation are the same as those used in the previous section. The exception is the higher $h_{min} = 8 \cdot 10^{-4} = 0.016D$, which is the length of the smallest edge of the boundary layer prisms constituting the interface with tetrahedra.

5.3.1 Modified degree and metric prescription for the PPRIME nozzle

In the field of nozzle/jet configurations an unstructured mesh approach has been extensively used in the literature, both for RANS simulations [106, 2] and resolved LES of jets [324, 190]. Nevertheless, the use of anisotropic elements on boundary layer regions is of crucial importance for RANS equations, where the constraint of $y^+ \simeq 1$ is fundamental to capture the correct boundary layer profile.

During the RANS adaptation process, boundary layer prisms (which include the surface) are kept unchanged, and the remeshing adaptive algorithm acts only for tetrahedra. Due to the twofold nature of the RANS test case chosen here, which couples internal and external aerodynamics, it has been decided not only to preserve the boundary layer, but also all the tetrahedral elements internal to the nozzle. The reason is that a remeshing of the tetrahedra inside the nozzle, which are constrained by the fixed surface sizes can severely affect the quality of the mesh. This step is handled by MMG, which is able to preserve a set of tetrahedra specified by the user.

The *hp*-adaptive algorithm is modified according to the consideration of a frozen region of mesh elements. We can now define three main zones of the mesh $\mathcal{T}_K$: $\mathcal{T}_{BL,i}$, $\mathcal{T}_{TI,i}$, $\mathcal{T}_{TE,i}$, which are respectively the prismatic layer of the near-body pseudo-structured region, the portion of tetrahedral mesh inside the nozzle, and the outer tetrahedral region.

In the regions $\mathcal{T}_{BL,i}$ and $\mathcal{T}_{TI,i}$, which are not subject to h-adaptation, the choice between h-refinement and p-enrichment is relaxed, and an under-resolved element does not require anymore smooth features to be p-adapted. A value of the error estimator greater than the target value is sufficient to mark for p-refinement. The tetrahedral region external to the nozzle $\mathcal{T}_{TE,i}$ is free to be h- and p-adapted according to the usual criteria adopted for laminar computations (see Section 5.1.4).

Therefore, to sum up, the only condition that elements with a fixed geometry (prisms and internal tetrahedra, regions $\mathcal{T}_{BL,i}$ and $\mathcal{T}_{TI,i}$,) need to fulfill to be p-enriched is to present an error estimator greater than the target error. Elements allowed to be h- or p- adapted (tetrahedra external to the nozzle $\mathcal{T}_{TE,i}$) are h- or p-enriched following the procedure described in Section 5.1.4.

While this approach proves very powerful for flows which weakly depend on the discretization of the geometry, as in the case of the presented nozzle, some enhancements are still required in order to fully extend this approach to every type of configuration. The representation of curve boundaries through high-order elements and the use of much higher polynomial degrees are essential to well capture the features of the flow while keeping the surface mesh fixed. This would allow the accurate simulation of industrial flow configurations as wings or aircraft, without strongly depending on the initial discretization of the geometry.

5.3.2 Analysis of results

We present here the results obtained with *hp*-adaptation on the PPRIME configuration. The initial mesh, shown in figure 5.15, contains around 1.5 millions of dofs, with a constant polynomial degree $p = 1$ over the whole computational domain. In figure 5.16, a first qualitative comparison can be done on the cell-averaged Mach number solution obtained on the initial mesh and on the *hp*-adapted mesh after 5 adaptation steps, with $f_r = 1.5$, counting 10 million dofs. The initial coarse mesh clearly leads to an early dissipation of the jet and produces strongly asymmetric features. In contrast, the *hp*-adapted mesh in the right panel of figure 5.16 clearly symmetrizes the flow, and the improved resolution permits a better development of the jet.

The quality of *hp*-adaptation results is quantitatively assessed by extracting the

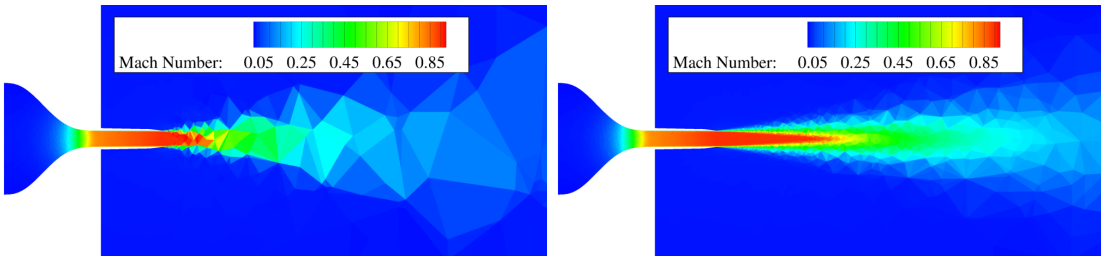


Figure 5.16: RANS PPRIME nozzle configuration. Contour of the cell-averaged Mach number for the initial mesh (left) and the mesh after 5 adaptation steps with $f_r = 1.5$ (10M dofs).

streamwise velocity profiles on the jet centerline ($r = 0$) and lipline ($r = D/2$) in figure 5.17 and in the jet at four different positions from the nozzle exit, $x/D = 1, 5, 10, 15$ (see figure 5.18). The curves obtained for different steps of the adaptation process using a refinement factor of $f_r = 1.5$ are compared to the reference experimental results [63]. Moreover, in order to assess our DG *hp*-adaptive RANS results, we performed a highly resolved second order FV RANS computation on a hexahedral structured mesh counting 48 million elements, and used it as a numerical reference.

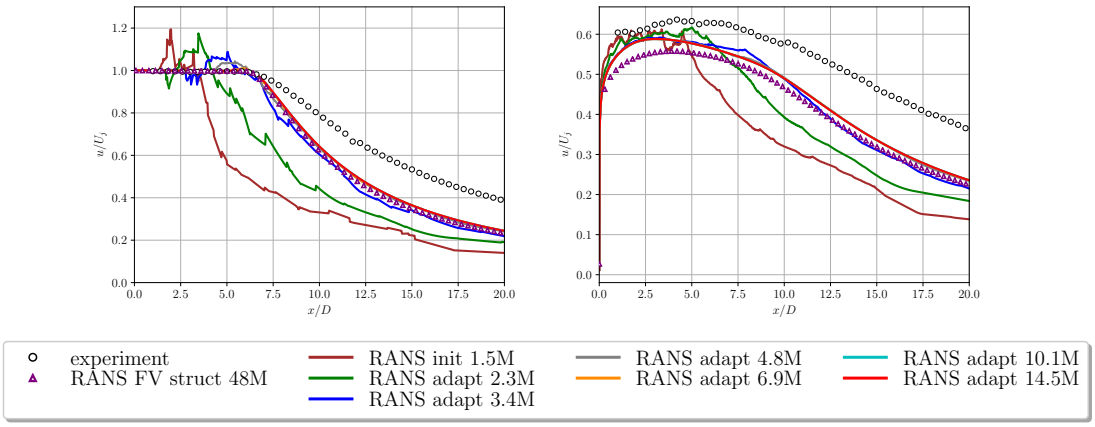


Figure 5.17: RANS PPRIME nozzle configuration. Velocity profiles on the centerline (left) and on the lipline (right).

The results yielded by the three last iterations of the *hp*-algorithm with respectively 6.9, 10.1 and 14.5 million dofs practically collapse, showing that mesh convergence has been achieved for the adaptive RANS simulations in four adaptation steps.

The RANS computations, both DG adaptive and the structured reference FV RANS, show a significant underprediction of the centerline velocity for $x/D \geq 5$. This effect is associated with a lower length of the potential core. This behavior is a well-known flaw of RANS models, see [2]. Despite these differences with the experimental results, which are attributed to the RANS model, the adaptive simulations provide results in overall good agreement with the fine FV reference computation. At $x/D = 1$ the shape of the

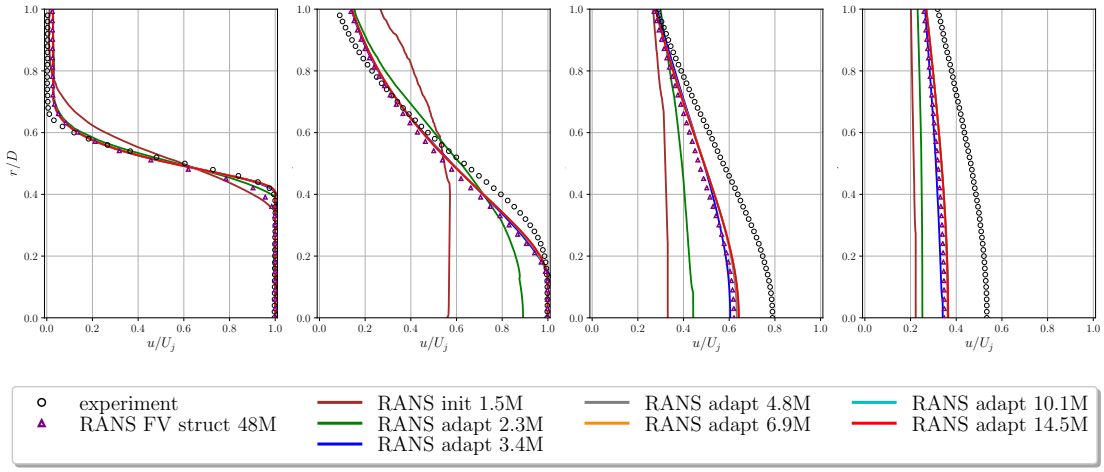


Figure 5.18: RANS PPRIME nozzle configuration. Radial velocity profiles at $x/D = 1, 5, 10, 15$ (from left to right).

velocity profile is improved at each adaptation step, and the finest adaptive simulations match almost perfectly the experiment. At $x/D = 5$, converged adaptive simulations are still very close to the experiment, yet showing a slight overprediction of the spreading rate of the jet. At $x/D = 10$ and $x/D = 15$, the radial velocity profiles obtained from RANS simulations show significant discrepancies with respect to the experimental results, yet achieving very close results to the FV fine simulation.

Overall, the results obtained from hp-adaptation all converge towards the same solution with a low number of adaptation steps, with 15 to 30% the number of degrees of freedom of the structured FV simulation, i.e. 7, 10 and 15 versus 48 million dofs. As a matter of fact, as already mentioned before, we observe that the hp-adaptive procedure starting from a low order polynomial distribution gradually improves the robustness of the solution of the implicit system of equations. Moreover, the use of an automated adaptive process circumvents the difficulties that a classical manual structured meshing process may involve, especially when handling complex geometries.

The last aspect we have studied is the influence of the refinement factor f_r on the adaptive process. We show in figure 5.19 two meshes containing both approximately 10 million dofs, one obtained after five adaptation steps based on a refinement factor $f_r = 1.5$ (left panel in figure 5.19) and the other after three adaptation steps based on a refinement factor $f_r = 2$ (right panel in figure 5.19). We clearly see how the mesh adaptation algorithm is capable, in both cases, of detecting the zones of interest. Most of the dofs are concentrated around the potential core and in the outmost part of the jet shear layer. In particular, for the mesh refined with a lower refinement factor (and thus more adaptation steps), we can remark a concentration of elements in the zone presenting the highest gradients in the solution. This corresponds to the cone enclosing the potential core. This feature is less marked if a higher refinement factor is used. In the second case (see right panel in figure 5.19), the adaptation focuses on a wider zone of interest and provides a smoother mesh size distribution. This results in a slower

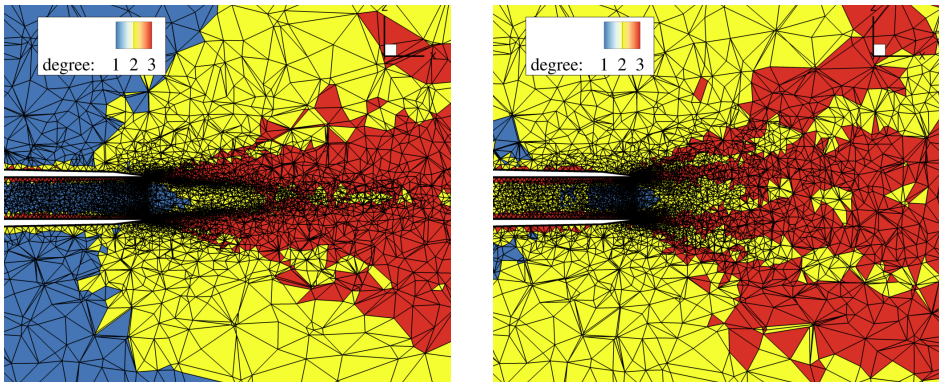


Figure 5.19: RANS PPRIME nozzle configuration. Zoom on *hp*-adapted meshes with around 10M dofs obtained with $f_r = 1.5$ (left) and $f_r = 2$ (right)

convergence towards the optimal mesh, as observed for the NACA0012 test case based on different refinement factors (see figure 5.7 in Section 5.2.2).

In figures 5.20 and 5.21 we present a comparison at around 10M dofs between the adapted meshes using $f_r = 1.5$ and $f_r = 2$ (respectively the fifth and the third adaptations).

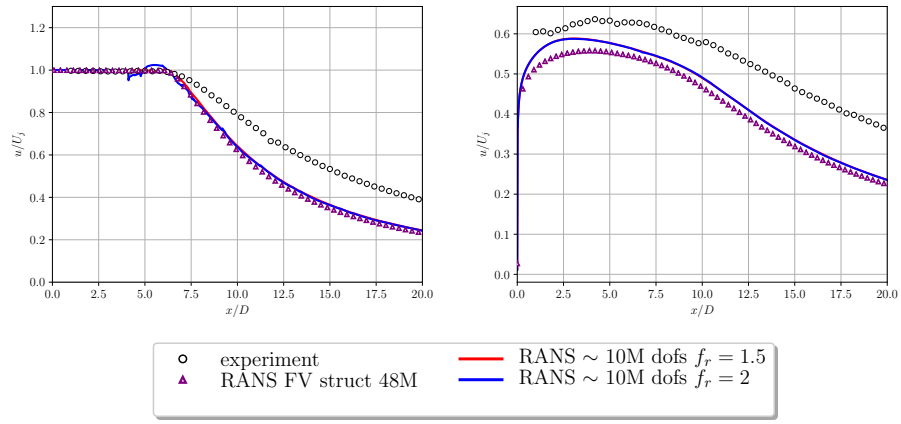


Figure 5.20: RANS PPRIME nozzle configuration. Velocity profiles on the centerline (left) and on the lipline (right) using $f_r = 1.5$ and $f_r = 2$ at around 10M dofs.

While the velocity profiles at $x/D = 1, 5, 10, 15$ and the lipline velocity profile are basically identical for the mesh obtained with $f_r = 1.5$ and $f_r = 2$, some small discrepancies are present in the velocity profile in the centerline. The irregularities at the end of the potential core are due to still coarse elements. The impossibility to spatially average the solution in the centerline may accentuate the discontinuous behavior of the solution in proximity of coarse elements interfaces, due to the discontinuous nature of DG.

The differences in the results shown in the two meshes in figure 5.19 can be quantified by computing the L^2-norm of the error on the velocity profiles at $x/D = 1, 5, 10, 15$ along the radial line, and at $r = 0, r = 0.5$ along the respective streamwise line for each

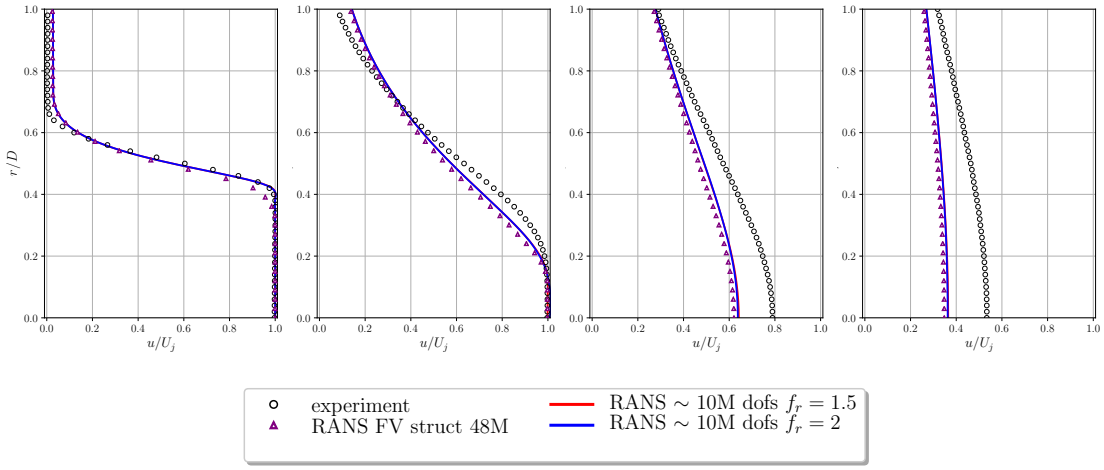


Figure 5.21: RANS PPRIME nozzle configuration. Radial velocity profiles at $x/D = 1, 5, 10, 15$ (from left to right) using $f_r = 1.5$ and $f_r = 2$ at around 10M dofs.

adaptation step as in equation (5.18)

$$error = \frac{||u - u_{ref}||_{L^2}}{||u_{ref}||_{L^2}} = \sqrt{\frac{\int_l (u - u_{ref})^2 dx}{\int_l u_{ref}^2 dx}} \tag{5.18}$$

where l is the line on which the profile is extracted, and u_{ref} are the profiles obtained from the fifth adaptation step of the adaptive process based on $f_r = 2$, counting 41 million dofs. The error plot as a function of number of dofs is shown in figure 5.22. As expected, the norm of the error on the velocity profiles decreases faster, yet requires more adaptation steps, for a lower value of f_r. In contrast, a higher value value of f_r reduces the convergence rate of the error, though globally needs less adaptation steps. As already pointed out, the optimal value yields a trade-off between maximizing the convergence rate and minimizing the number of adaptations.

This is a delicate topic if we keep in mind the possibility to generate adapted RANS meshes to use as the starting point of a static adaptive procedure for unsteady turbulent flows, as will be presented in Chapter 6. For scale-resolving simulations, the necessity of reducing the overall computational cost is crucial. In this case, due to the costly transient and the time-averaging of the error estimator needed at each adaptation step, the user might prefer to use a higher refinement factor with the consequent lower number of adaptation steps.

5.4 Conclusions

In this chapter, h- and hp-adaptive strategies suitable for discontinuous Galerkin methods have been proposed and assessed, for solving steady fluid flow problems on unstructured meshes. An *a posteriori* error estimator based on both the measure of the energy

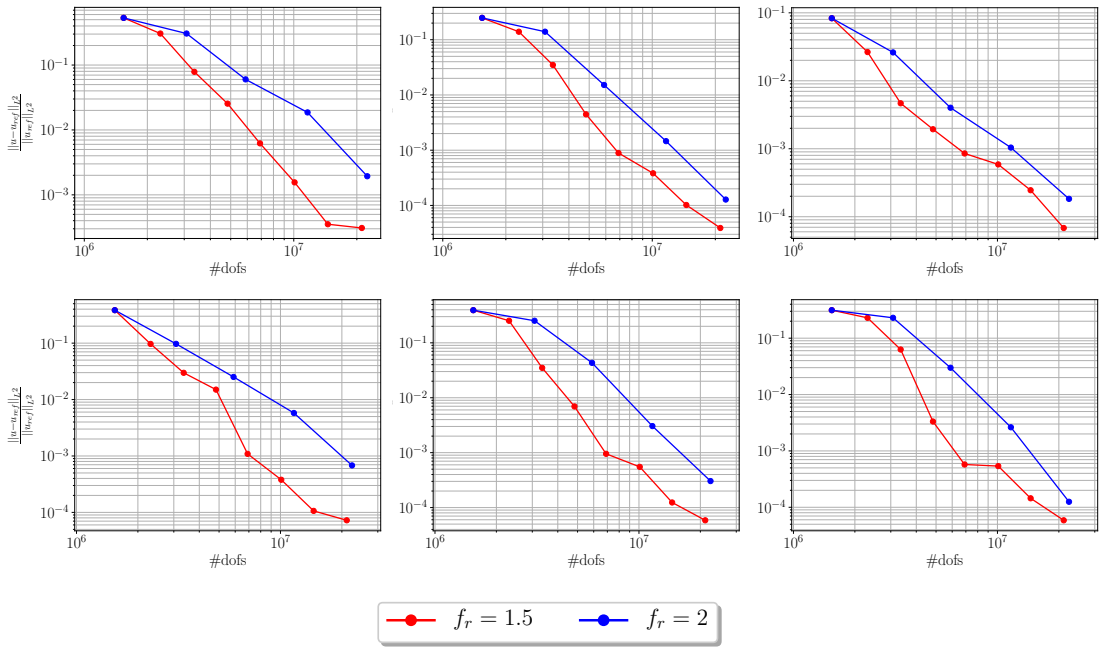


Figure 5.22: RANS PPRIME nozzle configuration. Comparisons between the convergence history of the norm of the integrated error along the profiles $r/D = 0, 1/2$ and $x/D = 1, 5, 10, 15$ (from top left to bottom right) for two different refinement factors $f_r = 1.5, 2$. *error* vs. number of dofs.

contained in the highest order polynomial modes and the jumps at the element interfaces has been assessed in the framework of isotropic metric-based *h*- and *hp*-adaptation. These methodologies have been combined with a smoothness indicator guiding both the choice between *h*- and *p*-adaptation and the reduction of the size of elements marked for *h*-adaptation.

Adaptive DG computations of three configurations of 2D/3D laminar steady flows on triangular/tetrahedral meshes have been performed based on polynomial degrees $p = 1$, $p = 2$ and $p = 3$. The overall adaptation strategies are found to capture accurately the zones in which the solution needs higher refinement and zones already well refined. This yields a significant reduction in terms of number of degrees of freedom to reach a given error level, compared to simulations with uniform mesh refinement. Besides, we have focused on the additional accuracy that both *h*- and *hp*-adapted meshes using high-polynomial degrees bring with respect to $p = 1$ simulations.

The performance and the flexibility of the proposed *hp*-algorithm have been also demonstrated in the context of turbulent jet 3D RANS simulations. The use of hybrid meshes composed of prisms in boundary layers subject to *p*-refinement and tetrahedra in the free stream regions subject to both *h*- and *p*-refinement allowed for a convergence of the DG-*hp* numerical solution using only 7 million dofs with results in close agreement with a simulation on a structured mesh using classical FV schemes with 48 million dofs.

Chapter 6

hp-adaptation for turbulent unsteady computations

In this chapter we present and apply an extension of the *hp*-adaptation strategy introduced in the previous chapter, for hybrid RANS/LES simulations of unsteady turbulent flows.

In Section 6.1 we focus on the description of the extension of the method to the unsteady hybrid RANS/LES formalism.

In Section 6.2 the *h/hp*-strategies for unsteady flows are assessed on the CoVo test case, and adaptive simulations are compared to uniformly refined meshes for the three spatial discretizations.

In Section 6.3 *hp*-adaptation is applied to the subcritical flow past a sphere at Reynolds $Re = 3700$. The resulting meshes are first evaluated from a qualitative point of view, then velocity and pressure profiles are compared to DNS and LES results in the literature.

In Section 6.4 the *hp*-adaptive algorithm is applied to the PPRIME nozzle in a ZDES mode 1 configuration, starting the adaptation from a previously adapted mesh on a steady RANS solution. Flow field results and acoustic results, obtained with a Ffowcs Williams-Hawkings (FW-H) acoustic post processing, are assessed qualitatively and quantitatively by comparison with reference results.

6.1 The *h/hp*-adaptive algorithms for turbulent unsteady computations

The steady adaptation strategy presented in Chapter 5 can be easily extended for static *h/hp*-adaptation of unsteady flows. In static adaptation, the objective of the adaptive algorithm is to identify a unique optimal spatial resolution for the simulation of a given unsteady flow. This often relies on the computation of the flow or error estimator statistics (e.g. time average/rms of the flow field, or time average/maximum/rms of the error estimator, etc). The adapted mesh is intended to be optimally representing the instantaneous flow at any moment. This is in contrast with dynamic adaptation strategies in

which the resolution is adapted over the course of the simulation to follow the instantaneous resolution requirements.

In our static approach, the adaptive algorithm is applied once a pseudo-steady state of the flow is reached. Instead of the instantaneous values, the time-average of the error estimator and the smoothness indicator accumulated for a given period of the flow is considered.

The error estimator employed to identify the regions lacking in accuracy, and improve their resolution by h or p-adaptation, is the same employed for steady flows. The smoothness indicator guiding the hp-decision is slightly modified to take into account the nature of the turbulent flows and is presented in the next paragraph.

6.1.1 Smoothness indicator

As already described in the previous chapters, a crucial point in hp-adaptive methods is the definition of a relevant strategy to choose whether to adapt an element with h-refinement or p-enrichment. In Chapter 5, we adopted the approach initially proposed for 1D by Mavriplis [202], which follows the assumption that the decay rate of the spectrum of DG modal coefficients is related to the convergence rate of the solution. This information is exploited to evaluate the smoothness of the solution in the elements in the mesh. In this chapter, we propose an alternative definition of the smoothness indicator that takes into account the underlying physics of unsteady turbulent flows. It is here assumed that for a 1D Legendre expansion of coefficients $q = 0, ..., p_K$ the power decay of the modes q can be expressed as:

$$E^{(q)} \simeq Cq^{-\sigma} \tag{6.1}$$

where $E^{(q)}$ is the energy associated with the mode q, C and σ are constants determined by a least-squares best fit of $\log(E^{(q)})$ vs. $\log(q)$. The decay coefficient σ is then used as smoothness indicator. In particular, the modal energy is proportional to the square of the momentum $E^{(q)} \propto \left(a^{(q)}\right)^2$, where $a^{(q)}$ are the polynomial coefficients associated with the norm of the momentum for the q-th mode. High decay rates imply that the solution is smooth, while the solution deviates from analytical behavior in presence of low decay rates. This particular formulation of the decay rate is interesting for turbulent flows as it correspond to the slope of spectral decay for the energy carried by turbulent scales. We can therefore draw a similarity with the classical Fourier analysis of turbulence and in particular the universality of the energy decay slope in the inertial range.

In the context of the calibration of a spectral dynamic modeling procedure for Large-Eddy Simulation, Chapelier *et al.* [82] observed that high values of σ are found in laminar or well-resolved regions, while low values are likely to appear in elements presenting an intense but poorly resolved small-scale activity. The authors used this evaluation of the energy decay as an estimation of the quality of the resolution in each element, allowing for adapting the intensity of the sub-grid dissipation locally. They evaluated *a priori* the threshold value of under-resolution σ_{thr} from Taylor-Green Vortex (TGV) direct numerical simulation data at $Re = 5000$ and they obtained values very close to the theoretical slope $-5/3$ found in Fourier space in the inertial range. For this reason they were able to link the smoothness indicator to the universal $-5/3$ power law which

uniquely describes the shape of the energy spectrum in the inertial subrange. We can exploit the same observations and use the value $\sigma_{thr} = 5/3$ as the threshold between smooth and non-smooth behavior in this chapter. Elements with $\sigma_K > 5/3$ are assumed to be smooth, and subject to p-enrichment, otherwise if this condition is not fulfilled h-refinement is used.

The coefficients of the modes are combined to obtain one single value per index q with the same approach presented in equation 5.6 for 2D and 3D computations.

Similarly to previous considerations for steady flows, we chose not to employ the coefficient $a^{(0)}$ in the log-log regression $\log(a^{(q)})^2$ vs. $\log(q)$. In fact the coefficient associated to $q = 0$ represents the cell-averaged solution, and it can severely bias the decay rate. This is problematic for $p = 1$, as removing the first coefficient prevents the estimation of the smoothness indicator. Therefore, as already done in Chapter 5, $p = 1$ elements are arbitrarily always marked as smooth elements, always requiring p-adaptation.

6.1.2 Strategy for unsteady flows

Once a statistically-steady solution (or statistically-periodic, according to the studied problem) has been reached on the initial mesh, the error estimator and the smoothness indicator are collected for a relatively short time period $T_{\epsilon\sigma}$, with respect to the total simulation time needed for data analysis and statistics collection T_{stats}. This allows the user to perform the successive adaptation steps without accumulating flow statistics for a long time. In fact the error estimator is very sensitive to poorly-resolved zones, and does not require a very long time to become representative in such zones needing an improved resolution. Both the error estimator/smoothness indicator and the flow data are sampled at a constant sampling time step Δt_{sampl}.

After each adaptation and projection of the previous solution on the newly adapted mesh, the error and the smoothness estimator and the flow data statistics are collected after a transient T_{trans}. This transient is necessary in order to let the solution adapt to the new spatial resolution.

The transient period T_{trans}, the collection period of the error estimator and the smoothness indicator $T_{\epsilon\sigma}$, and the sampling time interval Δt_{sampl} are generally case-dependent, as well as the total simulation duration where flow statistics are collected T_{stats}. Thus they must be chosen accordingly to features depending on the studied flow.

6.2 Static adaptation for the CoVo test case

In this section, the adaptive strategies extended to unsteady flows are validated on a canonical test case, the convection of an isentropic vortex test case presented in Chapter 4.

As outlined in the previous section, the error estimator and the smoothness indicator are collected and averaged for a shorter time period $T_{\epsilon\sigma} = T_{\text{CoVo}}$ than the total simulation time $10T_{\text{CoVo}}$. As the behavior of the flow, and the resolution requirements, do not vary over two successive periods, the collection of the error estimator and the smoothness indicator on only one period is sufficient to control the adaptation strategy. The error

estimator and the smoothness indicator are sampled at a constant sampling time step $\Delta t_{\text{sampl}} = T_{\text{CoVo}}/1000$.

We remark that, as this is not a turbulent flow, the smoothness indicator is the one used for steady simulations, and not the smoothness indicator adapted for scale-resolving simulations introduced in the previous section 6.1.1.

As the CoVo is initialized using an analytical expression, the projection step of the solution from the previous mesh to the new one is unnecessary. Moreover, due to the absence of transient, the collection of the smoothness and error indicators is started from the beginning of each simulation.

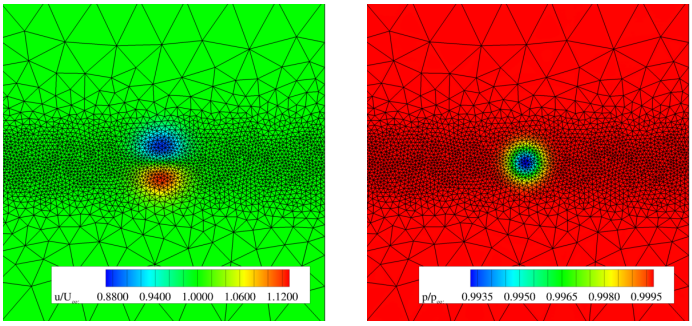


Figure 6.1: CoVo at $M_\infty = 0.5$. Velocity (left) and pressure (right) on the 11^{th} h-adaptation (4375 elements /43750 dofs) with a spatial $p = 3$ discretization.

In figure 6.1 we show the mesh obtained after 11 h-adaptations with a $p = 3$ discretization. The elements are concentrated in the zone traversed by the vortex, which is a central band of the domain. The refinement of the whole zone is due to the static adaptation approach for unsteady flows. Indeed the adaptation process needs to produce a spatial resolution which is well adapted for the full simulation time. This is in strong contrast with the results which would be produced by a dynamic adaptation approach, for which the concentration of elements would have "followed" the vortex, refining only the zone instantaneously traversed by the vortex, and coarsening the rest of the domain.

This test case might probably greatly benefit from dynamic adaptation strategies. Nonetheless, these present a number of technical and implementation issues that can impact both the accuracy and the efficiency of the resulting simulations.

A dynamic approach would require for all the tools used in the adaptive algorithm to be implemented directly in the flow solver, with direct memory access exchanges rather than file-based input/output. The implementation of dynamic load balancing would additionally be fundamental to ensure the equidistribution between different processes/threads of the computational load, and thereby ensure high parallel efficiency over the course of the simulation. This procedure has a non-negligible cost when the number of adaptations grows significantly as it happens for dynamic adaptation with respect to static adaptation. Finally we mention that dynamic adaptation in general requires accurate, conservative and efficient projection techniques to transfer the solution between two adapted meshes. In their absence, the numerical errors (dissipation, aliasing, non-conservativity) can severely impact the accuracy of the resulting simulations. In fact the projected solution should be readily exploitable, without previously undergoing

the simulation of a transient period. The difficulties mentioned above have not been analyzed in this work, and require a significant effort to produce accurate and efficient implementations. This therefore justifies the interest in static adaptation strategies, at least for flows for which the spatial resolution requirements are not expected to vary significantly over the course of the simulation.

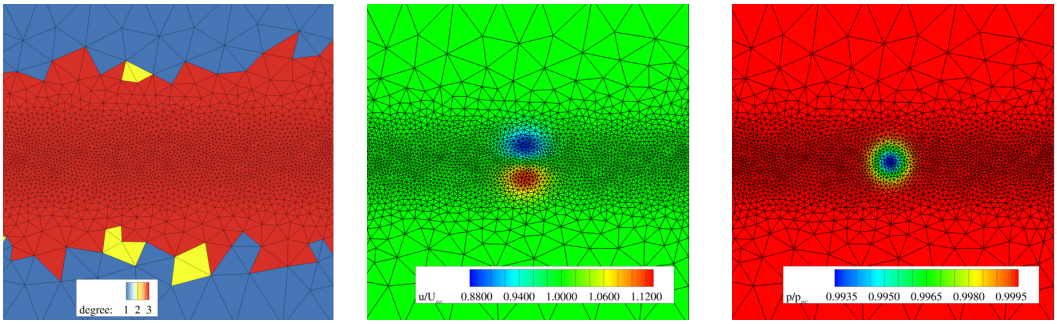


Figure 6.2: CoVo at $M_\infty = 0.5$. Polynomial degree distribution (left), velocity (middle) and pressure (right) on the 15^{th} *hp*-adaptation (45609 dofs).

In figure 6.2 the *hp*-mesh obtained at the 15^{th} *hp*-adaptation step is presented: the left panel shows the polynomial distribution, while the middle and the right panels represent the streamwise velocity and the pressure of the reconstructed high-order solution.

The size distribution on the mesh shows a very similar concentration of the dofs in the central area, to that reported in figure 6.1 for pure *h*-adaptation. Moreover this refined zone is entirely discretized with $p = 3$ elements, while the two bands closer to the top and bottom boundaries, not touched by the transported vortex, are discretized with $p = 1$ elements. Only very few $p = 2$ elements delimit the $p = 3$ zone from the $p = 1$ zone. This behavior is expected. As the CoVo does not present any non-smooth feature, the *hp*-adaptive algorithm is supposed to achieve the maximum allowed degree, before starting *h*-adaptation.

We now focus on the analysis of the error convergence properties of the adaptive algorithms, compared to uniformly refined simulations. The entropy and pressure errors for each simulation are presented in figure 6.3. Only the first four uniformly refined meshes employed in Section 4.1 are shown, and 11 *h*-adaptations are performed for $p = 1$, $p = 2$ and $p = 3$, while 15 *hp*-adaptations are performed with a minimum degree $p_{\min} = 1$ and a maximum degree $p_{\max} = 3$. At the last adaptation step, the error on the *h*-adapted meshes has reached a lower value with respect to the error of the respective most uniformly refined meshes. This is true also for the *hp*-adapted meshes as compared to $p = 3$ uniform meshes.

Figure 6.3 shows an overall very good behavior of the *h*- and *hp*-adaptive procedures. As observed already for the steady test cases in Chapter 5, the *h*-adapted meshes provide a faster convergence for both the entropy and the pressure errors, as compared to the uniformly refined meshes using the same DG spatial discretization. Despite the non-optimality of a static adaptive algorithm, a gain in dofs is always achieved with respect to uniformly adapted meshes. Analyzing the curves of the error for the *hp*-adapted meshes, we notice that after an initial gain with respect to the *h(p=3)* adapted meshes,

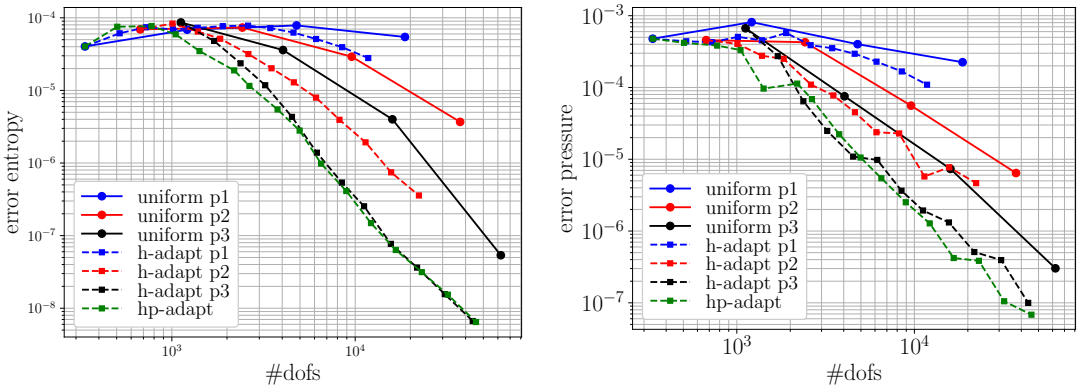


Figure 6.3: CoVo at $M_\infty = 0.5$. Convergence history of the entropy (left) and pressure (right) errors.

the curves start to superimpose (exactly for the plot of on the entropy error, and in a less regular way for the pressure error), recovering the theoretical convergence behavior.

6.3 DES of the subcritical flow past a sphere at $Re = 3700$

In this section we validate the unsteady *hp*-adaptation strategy on a canonical turbulent flow configuration, the flow past a sphere at $Re = 3700$.

The flow at this Reynolds number is subcritical, characterized by a laminar separation near the equator of the sphere. The transition to turbulence occurs in the separated shear layer. The turbulent wake is formed after transition to turbulence in the initially laminar shear layer, and the vortices created from the shear layers are shed at a characteristic large-scale shedding frequency. For a complete physical analysis of the evolution of the flow field in this configuration, we refer to the work by Rodriguez *et al.* [253].

6.3.1 Flow field computational set-up

The flow is simulated at a Reynolds number $Re = \dfrac{U_\infty D}{\nu_\infty} = 3700$, based on the freestream velocity U_∞ and the kinematic viscosity ν_∞, and the sphere diameter D. A free-stream Mach number $M_\infty = 0.2$ is chosen to simulate near incompressible flow conditions. The solid surfaces of the sphere are treated as no-slip adiabatic walls. The outer boundary conditions are set to free-stream stagnation pressure/stagnation temperature inflow at the inlet, static pressure outflow at the outlet of the domain, and they are defined using both the relevant stagnation quantities (pressure and temperature), and the static pressure in the lateral surfaces of the external cylindrical domain.

The cylindrical computational domain extends from approximately $-5D$ to $25D$ in the longitudinal direction x, and has a radius of $7D$. The sphere geometry is centered in the origin at $(x, y, z) = (0, 0, 0)$.

The initial mesh G0, uniformly discretized with $p = 1$ elements, counts 31259 elements (125036 dofs) and is generated with MMG by providing a metric and forcing the

	#*hp*-adaptation					
	G0	*hp*G1	*hp*G2	*hp*G3	*hp*G4	*hp*G5
#dofs (M)	0.125	0.222	0.535	0.877	1.375	2.087

Table 6.1: *hp*DG/DES of the subcritical flow past a sphere at $Re = 3700$. Number of dofs for the different *hp*-adapted meshes.

location of the mesh nodes on the wall. Unlike the steady cases in Chapter 5 where a trivial constant metric field was used, we chose here to set gradual sizes varying with the radius from the body to the far field. In fact a very bad resolution of the initial mesh around the body would prevent the flow around the sphere from developing coherent turbulent structures due to high dissipation, and it would require many adaptive steps to achieve a relevant solution.

The time intervals regarding the transient phase, the collection of the error estimator and the smoothness indicator, and the total duration of the simulation, introduced in Section 6.1.2 and expressed in characteristic time units $t_c = D/U_\infty$, are chosen as follows:

- $\Delta t_{\text{sampl}} = 0.0067 t_c$, which corresponds to 150 samples for each t_c,
- $T_{\text{trans}} = 10 t_c$,
- $T_{\epsilon\sigma} = 10 t_c$, which corresponds approximately to 2 shedding cycles of the wake flow,
- $T_{\text{stats}} = 350 t_c$, as proposed in the DNS study by Rodriguez *et al.* [253] to ensure a sufficient convergence of the solution statistics.

The parameters of MMG and the adaptation tool are the same as those proposed for the steady computations in Chapter 5. However, for this case, the minimum mesh size is modified to to $h_{min} = D/100$ to avoid a too strong restriction on the time step. Keeping into account the number of 1D dofs for DG introduced for the flat plate in Chapter 4, the effective DG minimum size is $h_{DG,min} = D/(100 \cdot (p+1))$, which for a $p = 3$ discretization corresponds to $h_{DGp3,min} = D/400$.

6.3.2 Computational meshes and qualitative instantaneous flow field analysis

Figure 6.5 shows the initial mesh G0 using uniform polynomial degree $p = 1$ and the 5 *hp*-adapted meshes employed in this study, *hp*G1, *hp*G2, *hp*G3, *hp*G4, *hp*G5. In table 6.1 we summarize the number of dofs employed for each *hp*-adapted mesh.

In figure 6.4 the distribution of the error estimator in the region surrounding the sphere is shown for the initial mesh (left panel), the second adapted mesh (middle panel) and the fourth adapted mesh (right panel).

In the initial mesh, the highest values of the error estimator are observed in the laminar boundary layer, while for the second and the fourth adapted meshes, relatively high values of the error estimator are found in the zone characterized by the strongest

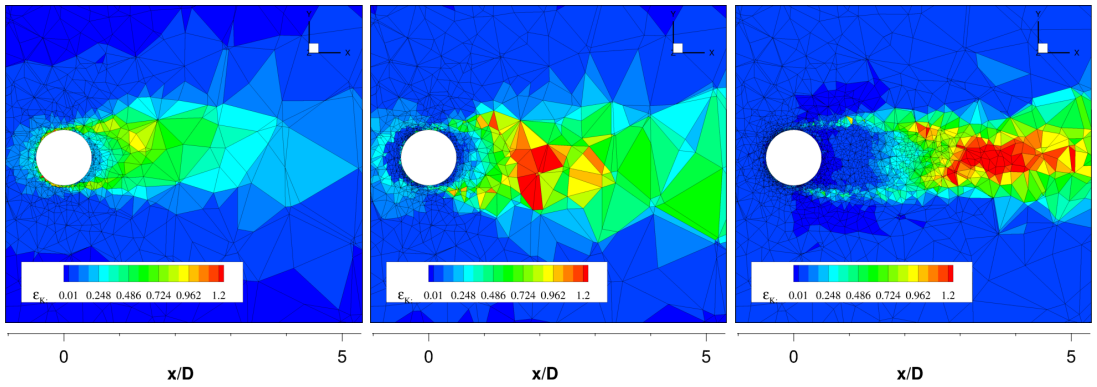


Figure 6.4: *hp*DG/DES of the subcritical flow past a sphere at $Re = 3700$. Zoom of the error estimator distribution around the body and near-wake. G0 on the left, *hp*G2 in the middle, *hp*G4 on the right.

turbulent activity with the highest values of the root mean square (rms) of the velocity fluctuations at the end of the recirculation bubble (in figure 6.10), and in the shear layers. In this flow configuration the shear layers are initially laminar (up to a variable distance between $x/D = 1.8$ and $x/D = 2.6$ according to Rodriguez *et al.* [253]). Despite their laminarity, the error estimator is found to correctly activate in shear layers until the end of the adaptation process. Due to its low speed motion, associated with low values of the rms of the velocity fluctuations characterizing this zone, in the fourth adaptation we can observe low values of the error estimator, which considers this zone resolved enough to sustain the turbulent motion in the recirculation bubble.

The meshes produced by the adaptive algorithm, with the associated polynomial degree distribution, are depicted in figure 6.5.

Due to the intrinsic global smoothness of the flow configuration, the adaptation is essentially applying p-enrichment until the maximum degree is reached, and then h-adapting the mesh. Some large elements, which are still $p = 2$, especially up to adaptation 4, can be found in the wake. This is possibly due to values of the smoothness indicator lower than $\sigma < 5/3$ in these very still coarse elements with an under-resolved solution, far from the theoretical behavior. Additionally, some small $p = 2$ elements can be found as well in the region immediately outside the shear layer. This might be caused by intermittent values of σ_K due to spurious modal coefficients with very low values. This effect might have been removed by using a longer time sampling of the smoothness indicator.

In figure 6.6 the Q-criterion is employed to visualize the vortical structures in the wake of the sphere colored by the streamwise velocity.

The initial mesh and the first three adapted meshes are clearly too coarse to correctly represent all the turbulent structures of the flow. Some large structures are already discernible in the recirculation bubble for *hp*G2 and *hp*G3. However, the Q-criterion visualization in the near wake presents strong numerical artifacts due to the high-order reprojection of a very under-resolved solution on a fine mesh. *hp*G4 is the first mesh in the adaptive process found to correctly represent the large eddies of the turbulent wake.

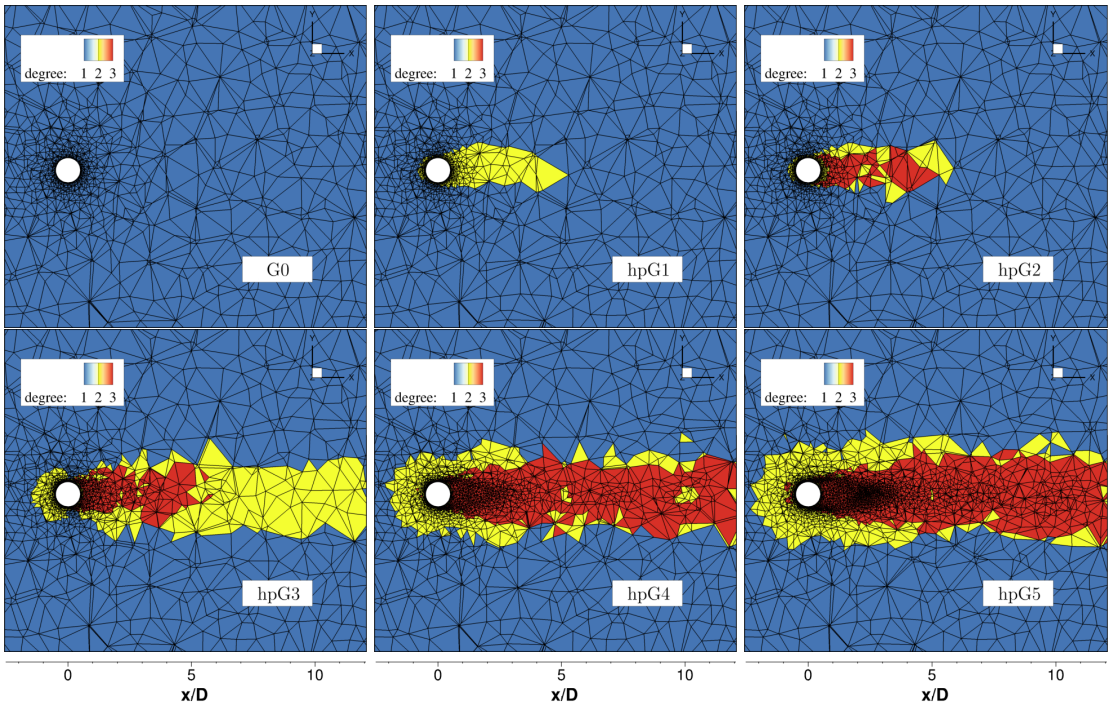


Figure 6.5: *hp*DG/DES of the subcritical flow past a sphere at $Re = 3700$. Zoom of the polynomial degree distribution around the body and near-wake. G0 to *hp*G5, from top left to bottom right.

Upon further refinement, of essentially the shear layers and the wake in the *hp*G5 case as observed in figures 6.4 and 6.5, a larger number of small turbulent structures is captured, due to the high resolution in the wake. In both *hp*G4 and *hp*G5 the typical large-scale wavy structure in the wake corresponding to the vortex shedding phenomenon, is recognizable. This is particularly evident in the case of the finest *hp*G5.

6.3.3 Quantitative data analysis

A quantitative data analysis is performed in this section for the validation of the *hp*-adaptive scale-resolving simulations.

At first we analyze the results obtained by extracting, for the six meshes, the velocity signal at two probes located in the shear layer. The two probes are located at the positions P1($x/D = 2.4, r/D = 0.6$) and P2($x/D = 3, r/D = 0.6$), and the data is extracted over a period of 350 t_c.

The Power Spectral Density (PSD) is computed based on the radial velocity v_r on the two probes, and is showed in figure 6.7. The Welch method [304] discussed in appendix A for the post-processing of an acoustic signal, is used for the computation of the PSD. An overlap of 75% is used with N=15000 sampled data per time window (the signal is windowed with a Hann approach), leading to 13 averaging windows. The spectra are azimuthally averaged on four equally azimuthally spaced probes at the same streamwise

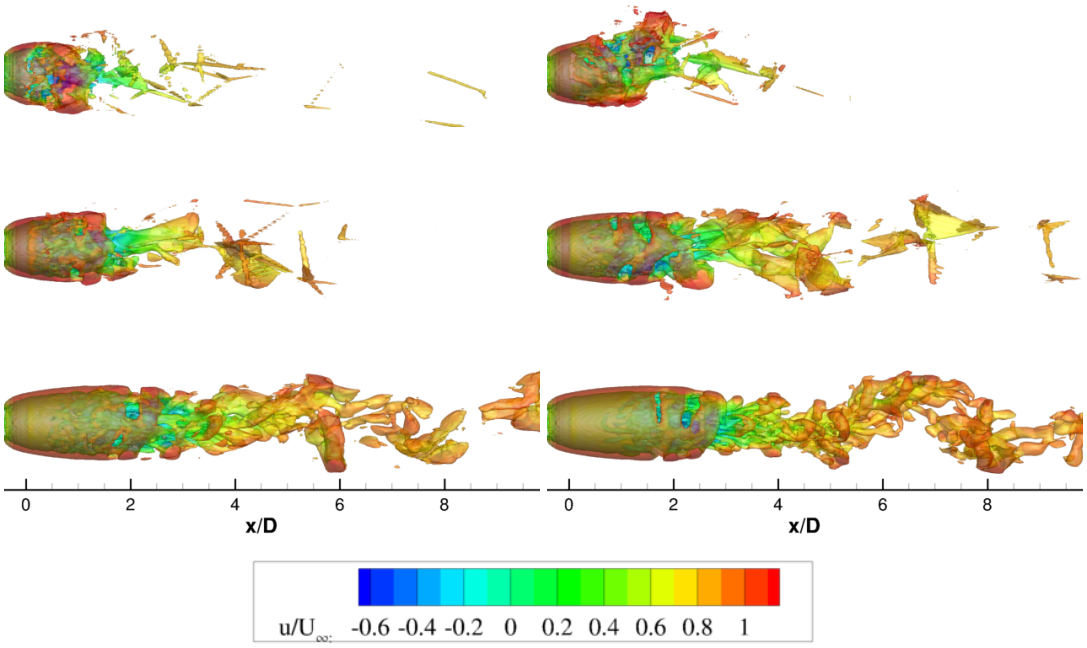


Figure 6.6: *hp*DG/DES of the subcritical flow past a sphere at $Re = 3700$. Iso-surface of $Q = 0.2(U_\infty/D)^2$, colored by the streamwise velocity component (right).

location.

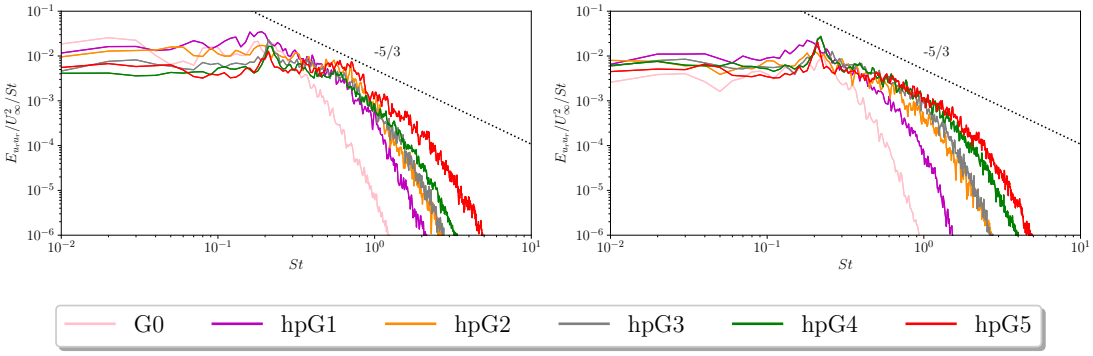


Figure 6.7: *hp*DG/DES of the subcritical flow past a sphere at $Re = 3700$. PSD of the radial velocity signal recorded at locations P1$(x/D = 2.4, r/D = 0.6)$ in the left panel and P2$(x/D = 3, r/D = 0.6)$ in the right panel

The first three meshes G0, *hp*G1 and *hp*G2 predict spectra with a very poor quality, especially in the low frequency range, where they provide strongly overestimated values of the PSD with respect to the more refined *hp*G3, *hp*G4, *hp*G5.

We focus on the curve corresponding to the adapted mesh *hp*G5. At first we remark

that the adapted mesh is able to capture the large-scale vortex frequency, with a peak in the spectra at $St = 0.21$, which is the dominant frequency of the flow. This is in line with numerical references in table 6.2. Additionally, the P1 probe identifies the presence of a secondary broadband peak in the range $St \approx 0.4 - 0.8$, present only in the closest probe to the sphere. This secondary characteristic frequency ($St_{KH} = 0.72$ in the study by Rodriguez *et al.)* is related to the Kelvin–Helmholtz instabilities generating in the shear layer.

At probe P2 the flow is fully turbulent and presents a spectra with a well defined $-5/3$ power turbulent decay in the inertial range. At the first probe P1, despite being located in an already unstable zone of the shear layer, the motion is still dominated by the Kelvin-Helmoltz instabilities, and the spectrum decay more rapidly than the turbulent slope $-5/3$.

The mean and root mean square (rms) streamwise velocity profiles are collected as well during $350t_c$, at the centerline, presented in figure 6.10, and at different locations in the wake, presented in figures 6.8 and 6.9. Velocity profiles are compared to numerical and experimental results in the literature. Available DNS data are those by Rodriguez *et al.* [253] performed on a fully unstructured mesh counting 9.48M cells with a second-order scheme, and by Bazilevs *et al.* [42] performed on a prismatic/tetrahedral grid counting 1.1M nodes/6.3M elements with a linear FEM method. Bazilevs *et al.* performed a very long simulation (1000 t_c) and observed some discrepancies according to the chosen time window. Only the results obtained on the second time window (W2), of comparable duration of the time window employed by Rodriguez *et al.* are here reported for comparison. The results from two LES computations have been as well included in our comparison to the *hp*-adaptive simulations: Yun *et al.* [315] employed a dynamic subgrid-scale model on a 3.3M cells Immersed Boundary mesh, while Eitel-Amor *et al.* [115] used a classical Smagorinsky model with a Lattice-Boltzmann Method (LBM) with local hierarchical adaptive grid refinement on a 51.7M cells mesh.

The shedding frequency St, the separation angle ϕ_s, the mean drag coefficient $\overline{C_D}$, the base pressure coefficient $\overline{C_{pb}}$ and the mean length of the recirculation bubble $\overline{L_r}$ are summarized in table 6.2 for several computations from the literature, and compared to our current computation on the *hp*G5 mesh.

The normalized mean velocity $\overline{u}/U_\infty$ profiles in figure 6.8 show the impact that *hp*-refinement has on the flow field development. DNS results by Rodriguez *et al.* are depicted in dashed black lines, DNS results by Bazilevs *et al.* in dotted blue lines, LES results by Yun *et al.* in black crosses and LES results by Eitel-Amor *et al.* in grey diamonds. The *hp*-meshes are plotted in solid lines in pink, magenta, orange, grey, green and red, respectively for G0, *hp*G1, *hp*G2, *hp*G3, *hp*G4 and *hp*G5.

The initial mesh G0 is extremely under-resolved, yielding velocity profiles very far from the numerical and experimental references. Thanks to the adaptive process, simulations on *hp*-adapted meshes lead to a progressive alignment of the *hp*-adaptive results towards the numerical reference solutions, even though the different numerical results present some discrepancies. The shape of the velocity profiles at $x/D = 1.6$ for the finest *hp*G4 and *hp*G5 is very similar to those obtained by Bazilevs *et al.,* Yun *et al.* and Eitel-Amor *et al.* The DNS profiles by Rodriguez *et al.* showed more negative values of the velocity at locations $x/D = 1.6$ and $x/D = 2$, both located in the recirculation zone, with

	Re	St	$\phi_s(°)$	$\overline{C}_D$	$\overline{C}_{pb}$	$\overline{L_r}$
Rodriguez *et al.* [253] (DNS)	3700	0.215	89.4	0.394	-0.207	2.28
Bazilev *et al.* [42] (DNS)	3700	0.221	89.4	0.392	-0.207	2.28
Pal et al. [228, 229] (DNS)	3700	0.21	91.7	0.3938	-0.215	2.22
Seidl *et al.* [265] (DNS)	5000	-	89.5	0.38	-	-
Kim *et al.* [169] (exp.)	4200	0.225	-	-	-0.224	-
Park *et al.* [235] (LES)	3700	0.21	-	0.359	-0.193	2.60
Yun *et al.* [315] (LES)	3700	0.21	90	0.355	-0.194	2.622
Eitel-Amor *et al.* [115] (LES)	3700	0.21	93.9	0.378	-0.190	2.51
Kamble *et al.* [165] (PANS)	3700	0.217	89.79	0.396	-0.22	2.24
current *hp*G5 (DES)	3700	0.21	90.3	0.382	-0.195	2.55

Table 6.2: *hp*DG/DES of the subcritical flow past a sphere at $Re = 3700$. Integral flow quantities from the literature and current results.

respect to adaptive simulations and the DNS by Bazilevs *et al.*

The position $x/D = 3$, located near the end of the recirculation zone presents the most scattered results. While the DNS from Rodriguez *et al.* and Bazilevs *et al.* predict higher (positive) velocities in the centerline, the LES by Yun *et al.* and Eitel-Amor *et al.* predict respectively slightly negative and almost zero values, meaning that the flow is at the end of the recirculating region. The velocity profiles obtained with the DES *hp*-adaptation tend to intermediate values between the two LES approaches. In locations in the wake further downstream ($x/D = 5$ and $x/D = 10$), *hp*G5 reaches very close profiles to the DNS by Rodriguez *et al.*

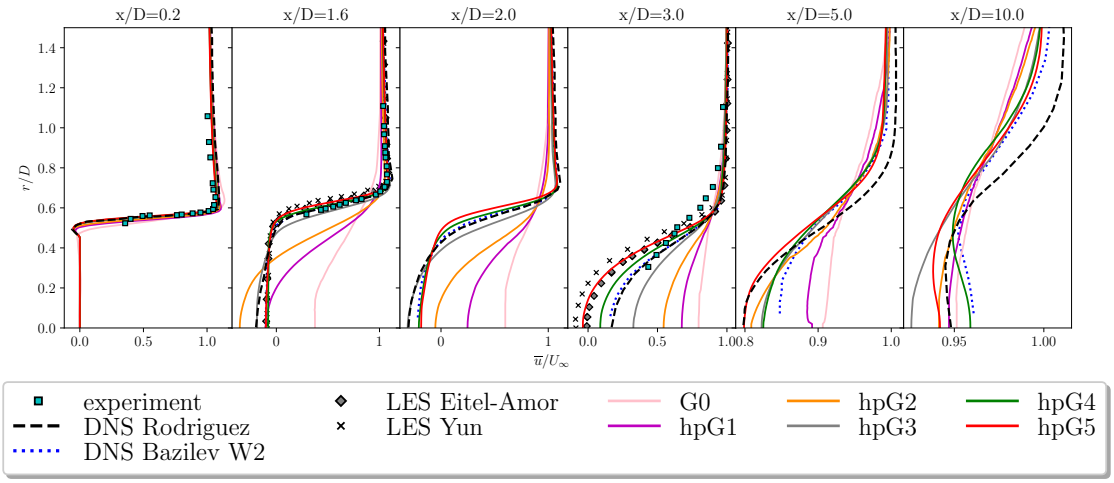


Figure 6.8: *hp*DG/DES of the subcritical flow past a sphere at $Re = 3700$. Mean streamwise velocity profiles on different locations in the wake $x/D = 0.2, 1.6, 3, 5, 10$ (from left to right). Comparison between *hp*-adaptive and experimental/numerical results.

A further insight in the flow field development is provided by the profiles of the

normalized rms of the fluctuation of the streamwise velocity $\sqrt{\overline{u'u'}}/U_\infty$ in figure 6.9. The progressive increase in resolution leads to a decrease of the rms levels in the shear layer at locations $x/D = 1.6$ and $x/D = 2$. The finest mesh *hp*G5 provides significantly lower rms values than the DNS by Rodriguez *et al.* at these locations. This trend is observed as well with respect to the results by Bazilevs *et al.,* but to a minor extent. This behavior is a clear indicator of a certain delay in the formation of instabilities in the shear layer when increasing the resolution. The rms profiles values obtained on *hp*G5 for the upstream locations $x/D = 3, 5, 10$ closely match DNS results by Rodriguez *et al.,* showing a significant improvement of the solution in the region at the end of the recirculation bubble and the wake.

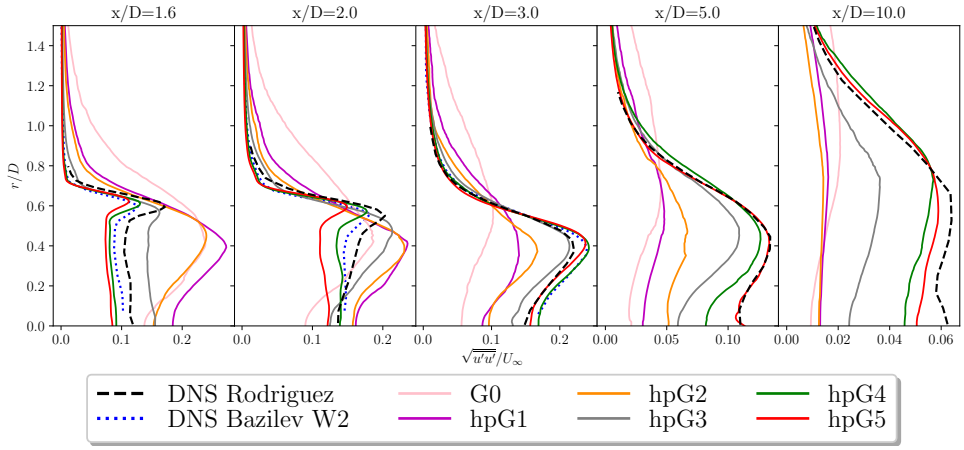


Figure 6.9: *hp*DG/DES of the subcritical flow past a sphere at $Re = 3700$. Root-mean square of the streamwise velocity fluctuations profiles in the wake $x/D = 1.6, 3, 5, 10$ (from left to right). Comparison between *hp*-adaptive and experimental/numerical results.

In figure 6.10 the axial streamwise velocity profile monotonically moves downstream when increasing the resolution, getting closer to the DNS results (LES results are not available for this plot). The velocity profile for *hp*G4 in the recirculation region is very similar to that of Bazilev*set al.,* and both the value of the minimum and its location are similarly predicted. Lower values and a more upstream position are found for the DNS computation by Rodriguez *et al.* However, the finer *hp*G5 moves the profile further downstream the DNS and overestimates the recirculation length by 10% ($L_r = 2.28$ is found by the two DNS, $L_r = 2.55$ by the DES-*hp*G5 adaptive simulation, and $L_r = 2.51$ and 2.667 is found by the two LES).

Looking at the right panel of figure 6.10 depicting the profiles of the rms of the axial velocity fluctuations, *hp*G4 closely matches profiles by Bazilevs *et al.* and the two peaks at $x \approx 2$ and $x \approx 3$ are similarly represented (the first having a lower intensity than the second). Rodriguez *et al.* report higher rms values in the recirculation bubble and two rms peaks of similar intensity at $x/D \approx 2.5$ and $x \approx 3.5$. *hp*G4 deviates from results by Bazilevs *et al.* and Rodriguez *et al.* for $x > 4$ due to a lower resolution in the wake. When increasing the resolution, *hp*G5 provides slightly lower rms in the recirculation region,

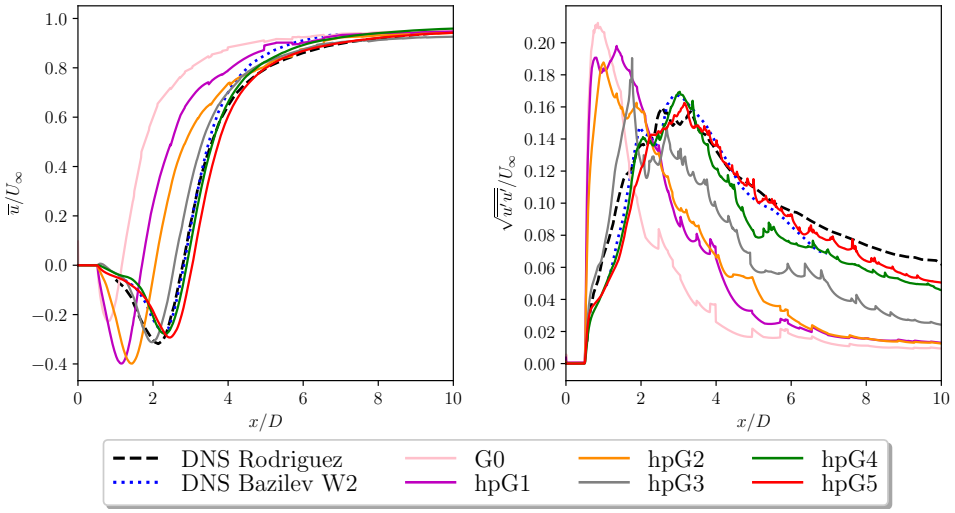


Figure 6.10: *hp*DG/DES of the subcritical flow past a sphere at $Re = 3700$. Mean (left) and rms (right) streamwise velocity profiles on the wake axis $r/D = 0$. Comparison between *hp*-adaptive and experimental/numerical results.

a lower intensity of the main peak and the secondary peak has almost been completely smoothed out. As observed in figure 6.9, the wake $x/D > 4$ is much better represented than *hp*G4, providing intermediate rms levels between the DNS by Rodriguez *et al.* and Bazilevs *et al.*

The azimuthal distribution of the mean values of the pressure coefficient $\overline{C_p}$ and the Reynolds-normalized friction coefficients $\overline{\tau_w}/(\rho_\infty U_\infty)Re^{0.5}$ on the surface of the body, are finally presented in figure 6.11. Looking at the C_p distribution in the left panel, we can notice how, when increasing resolution, the C_{pb} increases, and the *hp*G5 adapted mesh provides results in excellent agreement with experimental and numerical references, in particular with the LES approach by Eitel-Amor *et al.* The peak value in the friction coefficient obtained using *hp*G5 is in very close agreement with the results by Bazilevs *et al.,* while underestimated with respect to the DNS by Rodriguez *et al.* Looking at the integrated total drag, a lower value of the mean drag coefficient $\overline{C_D} = 0.382$ with respect to the DNS by Bazilevs *et al.* and Rodriguez *et al.* is obtained by the most refined *hp*G5 mesh. This result is closer to the LES results by Eitel-Amor*et al.* ($\overline{C_D} = 0.378$). Also the results by Yun *et al.* underestimate the $\overline{C_D}$. The underestimation of the $\overline{C_D}$ is indeed directly related to the overestimation of the recirculation length.

6.3.4 Discussion on the sensitivity of L_r

The sensitivity of the length of the recirculation zone L_r to the numerical method employed, is a very well-known topic observed in several works in the literature. In particular the delay in the formation of the turbulent instabilities, with a subsequent overestimation of the predicted length of recirculating zones, is found to be a common issue

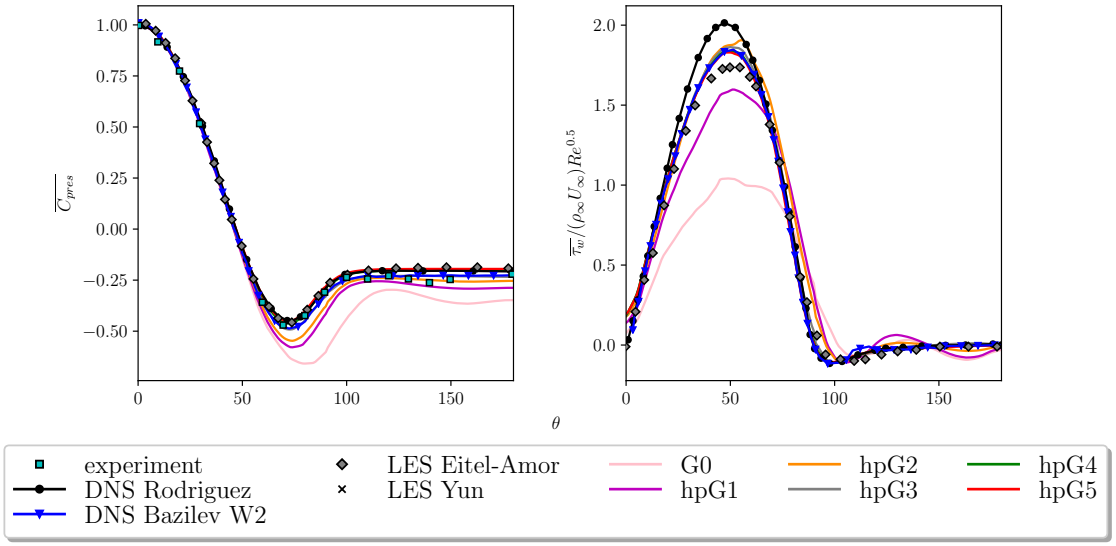


Figure 6.11: *hp*DG/DES of the subcritical flow past a sphere at $Re = 3700$. Mean pressure coefficient $\overline{C_p}$ and mean normalized skin-friction coefficient $\overline{\tau_w}/(\rho_\infty U_\infty)Re^0.5$ on the surface of the sphere. Comparison between *hp*-adaptive and experimental/numerical results.

when employing DES-like methods.

Among the several examples, Uzun *et al.* [289] observed longer recirculation bubbles in both cylinders for a supercritical tandem cylinder configuration at $Re = 166000$, attributing the issue to the high eddy viscosity values in the initial detached shear layer. The same trend was observed for both their coarse and fine grids (31M and 133M grid points).

For the subcritical flow past a cylinder at $Re = 3900$ (which has very similar flow features to our sphere at $Re = 3700$), Itam *et al.* [160] observed much longer recirculation bubbles than the experiment, when not using a subgrid scale model and when using a DDES model (with a $k - \epsilon$ closure). When switching to a dynamic VMS (Variational Multi Scale) approach, they found excellent agreement with experiments and other numerical references. In the same work, they observed an opposite behavior when considering the higher $Re = 20000$ configuration, with no model simulations and simulations with a dynamic VMS approach providing identical recirculation lengths, very close to experiments, and DDES predicting lower values.

D'Alessandro *et al.* [96] proposed the implementation of a DES model based on the $\overline{v}^2 - f$ approach, and investigated its behavior with respect to several DES-like models, on the subcritical flow past a circular cylinder at $Re = 3900$. They presented extremely scattered data, depending on both the employed DES-like model and the level of refinement of the mesh. They observed longer separating shear layers on fine grids, and early transition (and shorter shear layers) on coarse meshes when employing the proposed DES-$\overline{v}^2 - f$ model. In particular the overestimation of the recirculation length could be

observed for the finest mesh. In their work, the standard SA-DES simulations predicted significantly shorter recirculation lengths for any grid resolution, but we remarked that the implementation of SA-DES was the original by [277] without the modifications suggested by Deck presented in Section 2.2.2.

Jee *et al.* [163], although not overestimating the recirculation region, tested two different DES models on two meshes of 0.8M and 6M elements, showing in both cases significantly longer recirculation zones when employing finer meshes.

The prediction of the correct length of the recirculation zone is a sensitive issue also in a pure LES context.

For the subcritical cylinder configuration, de la Llave Plata *et al.* [100] predicted delayed instabilities with a classical WALE model (strong overprediction of the recirculation length with lower rms values in the axis) with respect to experimental measurements, using a fifth order DG discretization. The use of a VMS model provided results in very close agreement with the experiment. They observed fairly good results (though worse than VMS) also when not employing any subgrid scale model.

For the same test case, Lysenko *et al.* [198] observed a longer recirculation zone than the experiment with the most accurate dynamic k-equation eddy-viscosity SGS model model (TKE), while conventional Smagorinsky significantly underestimated this quantity. Moreover, the authors highlighted the noticeable discrepancies found in the literature when analyzing this very sensitive quantity, in experimental measurements, LES and even DNS computations. L_r from experimental measurements ranged from $L_r = 1.19 - 1.51$, LES data resulted in $L_r = 0.9 - 1.64$, and DNS computations resulted in $L_r = 1.41 - 1.59$. In the case of the DNS by Rodriguez *et al.,* the grid independence study they proposed showed a 5% variation of the length of the recirculation bubble between the most refined and the second to last most refined mesh, suggesting that further refinement could potentially further increase the length of the recirculation bubble.

The very high sensitivity of this quantity from the several employed numerical ingredients, such as subgrid scale models, spatial discretization parameters, grid resolution, appears to be therefore quite an established matter in the literature. This discussion gives some insight on why our DES *hp*-adaptation presents some discrepancies with respect to the DNS results, given the strong impact that the quality of the employed turbulence model has on modeled scale-resolving simulations. Moreover, when treating spatial resolution adaptive techniques, the quality of results has a double dependence on the chosen turbulence model. In fact, the subgrid scale model drives not only the dissipation due to the under-resolved eddies in each element, but also the distribution of the error estimator in the domain (and subsequently the whole adaptive process). The latter in fact depends on the features of the flow, which are partly dependent on the used subgrid scale model.

6.3.5 Additional simulations and further investigation

In order to assess the solution obtained on the finest (in previous Section 6.3.2 and 6.3.2) *hp*G5 mesh, we decided to perform some additional simulations (D, E, F, G), whose numerical settings are summarized in table 6.3. Cases H and I are used to investigate the sensitivity of the solution to the model.

	#case							
	B	C	D	E	F	G	H	I
#dofs (M)	1.375	2.087	2.525	4.605	1.874	3.160	2.087	2.087
turb modeling (ν_t/ν_∞)	DES 4	DES 4	DES 4	DES 4	DES 4	DES 4	DES 0.001	ILES
p	1-3	1-3	2-4	4	1-3	1-4	1-3	1-3

Table 6.3: *hp*DG/DES of the subcritical flow past a sphere at $Re = 3700$. Computational details of the additional simulations.

Cases B and C respectively correspond to the already previously analyzed *hp*G4 and *hp*G5 meshes, which employ polynomial degrees from 1 to 3.

The D case employs the *hp*G4 mesh, but a uniform increase of the polynomial degree by 1 is applied in each element of the mesh (the polynomial degree in this case is in the range $p_D = 2 - 4$). The uniform increase of the spatial resolution is supposed to provide a good guess of the trend followed by the adaptive process (that should converge to similar values but with a lower number of degrees of freedom).

Case E eliminates all the polynomial degree variations, and consists in a uniform $p = 4$ distribution in the *hp*G5 mesh ($p_E = 4$). This 4.6M dofs simulation is the most refined simulation presented in this study and is expected to be the most reliable reference for the assessment of the adaptive results.

Moreover, a separate adaptation process is performed by considering an increased maximum polynomial degree from 3 to 4. Case F is the fourth adapted mesh from such adaptive process ($p_F = 1 - 4$). Time-converged results on intermediate meshes are not available.

The adaptive process presented in previous section employing $p_{max} = 3$, is further carried on, and the sixth adapted mesh *hp*G6 corresponds to case G ($p_G = 1 - 3$).

Cases H and I consist respectively in a DES computation varying the value of ν_t/ν_∞ at the inflow of the domain from the default $\nu_t/\nu_\infty = 4$ used in previous computations to a lower value $\nu_t/\nu_\infty = 0.001$, and in an ILES computation (solving the Navier-Stokes equations without any subgrid scale models). Both cases H and I are performed on *hp*G5 with $p_{H,I} = 1 - 3$.

The meshes employed in cases D, E, F, G with their polynomial distribution are depicted in figure 6.12. Cases B, C, H and I have the same *hp* distribution as that presented on the right bottom panel in figure 6.5.

In figure 6.13 we compare profiles of the mean centerline velocity in the wake. This is the most relevant quantity for the computation of the length of the recirculation bubble, and it is found to be strictly correlated to the other main phenomena of the flow (earlier or later destabilization of the flow as well as overestimated and underestimated $\overline{C_D}$ values).

A first observation is that the solution obtained on the further adaptation *hp*G6 (case G in blue circles) provides very similar values of the recirculation length to *hp*G5 and a velocity profile superposed to that obtained on *hp*G5. The adaptation procedure has reached its converged solution, and further refinements are not expected to significantly vary the solution.

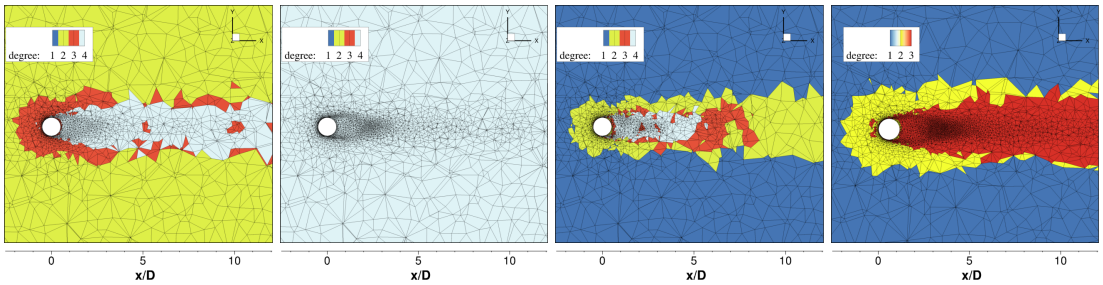


Figure 6.12: *hp*DG/DES of the subcritical flow past a sphere at $Re = 3700$. Zoom on the polynomial map distribution around the sphere for cases D, E, F, G.

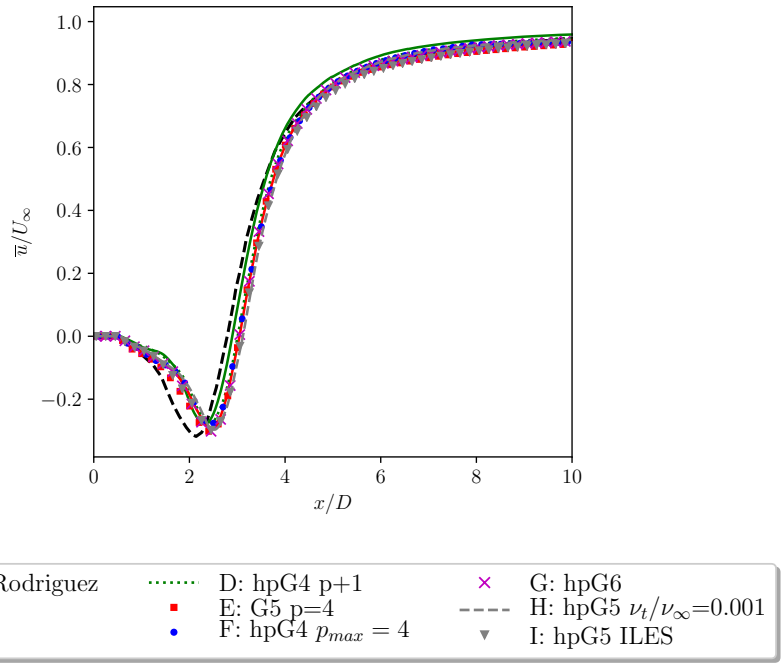


Figure 6.13: *hp*DG/DES of the subcritical flow past a sphere at $Re = 3700$. Mean streamwise velocity profiles on the wake axis $r/D = 0$. Comparison between different *hp*-DG simulations.

Moreover, we observe very similar velocity profiles when performing uniform p-enrichments on *hp*G4 (case D, in green dotted lines) and on *hp*G5 (case E in red squares), which both overestimate L_r by around 10% with respect to the DNS data. This suggests that the solution has converged, in a certain limit imposed by the mesh topology provided by the adaptation. The *hp*-adaptive process is not supposed to always converge to the DNS solution, because it is strictly linked to the subgrid scale model used. An *hp*-adaptive process using DES/LES models should converge to DNS only in the case it would be able to ensure that all elements have a size allowing the resolution of the whole range of turbulent scales. This is quite contradictory in the context of DES/LES

adaptation, where the underlying basic principle is the modeling of the non resolved small turbulent scales.

As previously mentioned, we performed as well a new *hp*-adaptive process with an increased $p_{\max} = 4$ (a higher $h_{min} \approx D/33$, $h_{DG,min} = D/165$, is employed to avoid too small time step sizes when increasing the polynomial degree in small wall elements). Also in this case the adaptation has led to an almost identical velocity profile in the axis behind the wake. This proves how the adaptation process is robust enough to lead to the same results when using different parameters. If we look at the same adaptation step when using $p_{\max} = 3$ (case B) and $p_{\max} = 4$ (case F), we can observe that 4 adaptations are sufficient to achieve converged results in the case of $p_{\max} = 4$ but not in the case of $p_{\max} = 3$. This is probably due to the use of a higher order discretization, but also to the fact that the resulting mesh F has 30% more dofs than the mesh B. In order to prove the first hypotesis, the two cases should have exactly the same resolution. It is probable that *hp*G3 with $p_{\max} = 4$ could yield similar results to case B with a lower number of dofs thanks to its higher order of accuracy, but full computations were not performed and a satisfactory answer cannot be provided.

These four test cases showed an overall agreement in the results obtained on different meshes and different spatial discretizations, suggesting that the solution obtained on *hp*G5 mesh can be actually considered the converged solution of the *hp*-adaptive process. Moreover they helped us to attribute, with a certain level of confidence, the mismatch between the results obtained on the *hp*-adaptation and the DNS results to the employed model and not to the adaptation process itself.

Finally, we decreased in case H the viscosity ratio at the inflow boundary ν_t/ν_∞, and we obtained a very similar axial velocity profile to previous simulations, with a slight increase of the length of the recirculation bubble. Some differences between results obtained with standard and zero values of ν_t/ν_∞ at the inlet were observed also by D'Alessandro *et al.* [96] for the circular cylinder at $Re = 3900$ using different DES-based methods.

The ILES computation (case I) predicts as well a slighly longer recirculation length, as observed by Itam *et al.* [160] in the $Re = 3900$ cylinder configuration (with respect to a DES-like model).

Some caution should be however advised when analyzing results obtained on the adapted meshes, varying the model or setting of parameters of the model. In fact the "grid-converged" *hp*G5 with DES using the standard viscosity ratio could be not "grid-converged" with another model, that could have differently adapted also the intermediate meshes in the adaptive process.

In general, the dependence of the adaptive process from the employed subgrid scale model could be weakened only by improving the subgrid scale model itself. In order to achieve more reliable results, the interest in employing VMS and/or dynamic LES approaches has been pointed out by de la Llave Plata *et al.* [100] and Itam *et al.* [160] in the context of low to medium Reynolds bluff body configurations. However, although being very powerful tools for the correct computation of moderately turbulent flows, pure LES approaches are still very uncommon for highly turbulent industrial flows, due to their strong requirements in term of wall resolution. The hybridization proposed by Itam *et al.,* restricting the model to Dynamic VMS in LES regions provided very

good results, while keeping the wall resolution to RANS standards. The extension and the validation of similar approaches on different high Reynolds configurations could be an interesting topic for future research, to improve the robustness of scale resolving techniques, by minimizing the dependence of the final solution to the model.

6.4 ZDES of the jet flow issued from the PPRIME nozzle at $Re_D = 10^6$

In this section, we apply the *hp*-methodology for steady flows introduced in the previous chapter and extended to unsteady flows in Section 6.1, to ZDES computations of a nozzle configuration. The nozzle test case analyzed in this work corresponds to the configuration studied with a RANS approach in Section 5.3.

First, the state of the art of jet flow scale-resolving simulations is briefly presented in Section 6.4.1, focusing on the different approaches found in the literature to reduce the computational cost and to inject turbulent instabilities. Previous jet flow simulations performed with discontinuous Galerkin methods as well as classical methods to perform acoustic analysis are briefly described.

Then the computational set-up of the *hp*-adaptive simulations using a ZDES approach, starting from RANS-adapted meshes, are presented in Section 6.4.2.

In Section 6.4.3 flow field results are presented, analyzed and compared to those obtained by other authors in the literature.

In the last part of this section, in Sections 6.4.4 and 6.4.5, an aeroacoustic analysis is performed to predict far-field noise, which is known to be a major issue in the design of aircraft. A Ffowcs Williams-Hawkings method (FW-H) [120] is employed in this work for this purpose.

6.4.1 Motivation and state of the art of jet flow scale-resolving simulations

An important issue in aircraft certification concerns the level of noise produced by an aircraft in the various phases of flight. It is well known that jet engine noise is one of the main sources of noise in civil aircraft during take-off. It is mainly for noise prediction purposes that round jets have become a recurrent subject of numerical investigation. Moreover, with the new high and ultra-high bypass ratio (HBR and UHBR) engines coming into service to ensure a lower environmental footprint of aircraft, engine installation noise also becomes an issue, due to the position of the exhausting jet closer to the wing. This leads to strong interactions between the jet flow and the deployed wing control surfaces. It becomes then more important than ever to assess the reliability of CFD computations for noise prediction.

Despite the use since the late nineties of numerical simulations to support experimental data in the study of jet aeroacoustics [214, 25, 55, 48, 290], and the strong progresses in improving jet representation [50, 310, 51, 63], the achievement of results fully matching experimental data still remains a challenge.

Numerical approaches for the simulation of jet flows First jet flow investigations were performed with DNS [62, 294] at much lower diameter-based Reynolds numbers than realistic engine jets. Despite the increase of computational means in recent years, DNS is still out of reach for realistic Reynolds numbers, and LES has become the most common approach for jet noise prediction.

Different approaches can be found in the literature, which mainly differ in the technique used to reduce the computational cost of simulations.

One approach consists in not including the nozzle geometry in the computational domain, but imposing a mean velocity profile as inflow boundary condition [52, 170], which can be disturbed to mimic a turbulent inflow. Despite being cheaper than simulations including the geometry, this approach does not ensure that the velocity profile is representative of the real flow conditions.

An explicit inclusion of the nozzle geometry with a pure LES approach has been performed by several authors [288, 58, 56]. However Reynolds numbers simulated with this approach usually range from 10^4 to $2 \cdot 10^5$ for very simple nozzle geometries, and are still too computationally expensive for complex geometries and configurations at higher Reynolds numbers. We remind that values of the diameter-based Reynolds number of jet flows considered in experiments is of the order of the million.

In order to achieve a reduction in the number of degrees of freedom inside the nozzle, WMLES approaches have been considered in the literature [64, 179, 63].

Another approach to reduce the computational effort in the computation of jet flows was introduced by Shur *et al.* [271] and followed by other authors [295, 68]. It consists in a two-step RANS-LES approach, which uses a previous complete RANS simulation including the nozzle, to provide the mean inflow to prescribe at the inlet boundaries of a successive LES computation, which simulates only the isolated jet without the nozzle geometry. However, turbulence should be generated for the LES simulation for comparisons with the experiments, since RANS provides only mean flow data. Moreover no feedback from the LES jet simulation can be brought to the interior of the nozzle simulated in RANS, preventing the formation of some fluctuations depending on the geometry further upstream, that could impact the shear-layer instability.

Various other approaches involve the use of hybrid RANS/LES methods for the simulation of jet flows. Some examples of high Reynolds flows can be found in [86, 65, 309, 132, 293], where more complex geometries are simulated. These geometries, including chevron and dual stream configurations, are more representative of jet engines with respect to classical single stream round jets, and become affordable thanks to the lower grid constraints on the boundary layer that hybrid RANS/LES methods impose, as opposed to the more strict LES requirements. A critical aspect of this kind of approach consists in a potential delay in the LES content development [102].

Turbulence tripping techniques One of the main difficulties in the simulation of turbulent jets lies in obtaining a turbulent jet from the nozzle exit, as in real jets at $Re_D > 10^5$ [317, 158]. This requires extremely fine resolutions inside the nozzle and close to the nozzle exit where the mixing layer develops, which is not computationally affordable for high Reynolds number flows.

The initial laminarity of the shear layer strongly influences the jet flow development,

leading to incorrect values of the potential core length and turbulence levels. This in turn leads to an overestimation of the acoustic field due to vortex pairing [318, 53, 57, 58, 156].

Different methods exist in the literature to inject disturbances in the boundary layer and thereby trigger the transition to a turbulent flow inside the nozzle. Some of the most common methods are the divergence-free vortex-ring method [55, 184, 290], the methods based on annular mixing layer linear stability theory [322, 49], and the methods based on geometrical tripping [261, 127, 189, 297]. Among the approaches recently employed for turbulent jet flows or general boundary layers, we find as well the Synthetic Eddy Method [161, 133], the Random Flow Generation (RFG) technique [275, 65], and the methods based on roughness elements (or tripping dots) included in the computational domain with the Immersed Boundary Method (IBM) [104, 135] or source terms [60].

Jet flow simulations with DG methods Traditionally, jet flow simulations for aeroacoustic purposes have been performed by means of high-order finite difference (FD) [214, 55, 290] due to their high accuracy, essential for capturing accurately the complex turbulent physical phenomena in jet flows. Simpler and more robust second order Finite Volume schemes are often used in industrial as well as research contexts [65, 63, 189, 297]. In this case, low-dissipation spatial schemes are fundamental for a correct representation of turbulence [133, 286, 293].

Few works have employed up to now discontinuous Galerkin methods to simulate jets [149, 201, 93, 190, 59, 84, 267].

In [149], Ham *et al.* use a second-order DG method to predict the far-field noise generated by a jet at $Re_D \sim 1.1 \cdot 10^6$. Although comparisons with experimental flow fields were reasonable, an excessive falloff in the high frequency range was observed, emphasizing the requirement of additional refinement and/or higher order discretizations.

In the works by Marek *et al.* [201] and Carton de Wiart *et al.* [72], respectively third-order incompressible and fourth-order compressible DG-LES computations of turbulent jets were performed, at relatively low Reynolds numbers $Re_D = 2 - 5 \cdot 10^4$, which are significantly lower than industrial jets.

Fourth-order DG simulations of the JEAN nozzle configuration at $Re_D = 10^6$ were performed by Lorteau *et al.* [190] on unstructured grids, and compared to the results obtained using a FV unstructured solver. Although the outcome of this comparison highlighted the potential of the DG method to accurately represent the complex physics of turbulent jets, no tripping technique was used in this study, which led to some discrepancies between the simulation results and the experimental data.

Corrigan *et al.* [93] have also reported results from third-order DG simulations of a coarse supersonic jet (6 million elements), showing the potential of the high order computation with respect to a second-order continuous Galerkin solver on a significantly finer mesh.

More recently, Chauhan *et al.* [84] compared supersonic turbulent jet simulations at Reynolds $8.5 \cdot 10^5$, using different subgrid scale models as ILES, Smagorinsky and Vreman, on meshes with over 100 million degrees of freedom. The computational mesh was divided into a near field zone discretized with a third order DG formulation which

used as well wall functions at the nozzle walls, while the intermediate zone was discretized with a second order Finite Volume method. This approach can be assimilated to a variable-order *a priori* adaptive simulation. The far-field was solved with the FW-H equations for acoustic noise prediction. They showed better performances of the Vreman model in predicting the far-field noise, yet yielding some discrepancies as compared to experimental measurements.

Results on jet simulations performed with DG methods have been usually obtained by employing classical convective schemes (LLF [190], HLLC [93] and Roe [149]).

Computational Aero Acoustics (CAA) approaches The description of both the flow field and the acoustic field with Navier-Stokes equations, referred to as Direct Noise Computations [29], would require a tremendous grid resolution, due to the need of transporting accurately short acoustic waves over long distances. This approach is usually not applicable to realistic applications, presenting a prohibitive computational cost.

For this reason, hybrid approaches are employed instead. The most common approaches rely on integral methods (as the Lighthill analogy [186], the Kirchhoff method [197] and the Ffowcs William-Hawkings (FW-H) method [120]) or on the use of the linearized Euler equations (LEE) [54, 213, 51] for the acoustic propagation.

The chosen hybrid approach for aeroacoustic predictions consists in solving the CFD field with scale-resolving simulations (DNS, LES, ILES, hybrid RANS/LES) to determine the near-field flow results, and then using these data to compute the far-field noise radiated from the jet using the FW-H surface integral formulation.

This hybrid approach has been employed for jet noise predictions of different configurations, such as in installed wing/jet or isolated jets configurations [136, 230, 287, 290, 309, 205, 238, 221, 189, 190] as well as for other aircraft components as landing gears, flaps, rotors [70, 176, 95, 168, 188].

6.4.2 Flow field computational set-up

Operating conditions The operating conditions are the same which have been presented in Section 5.3 for the RANS steady case. They are summarized in table 6.4.

p_t/p_∞	T_t/T_∞	T_j/T_∞	M_j	Re_D	M_∞
1.7	1.15	1	0.9	10^6	0.01

Table 6.4: Operating conditions of the PPRIME jet flow.

All solid surfaces of the nozzle are treated as no-slip adiabatic walls. Far-field boundary conditions are imposed on the lateral surfaces of the external cylindrical domain, the relevant stagnation pressure and temperature are defined at the inflow of the nozzle and the free-stream static pressure is specified at the outlet.

As proposed in Section 5.3, prisms and tetrahedral elements inside the nozzle are preserved over the course of the adaptation. These elements will be subject to p-adaptation only.

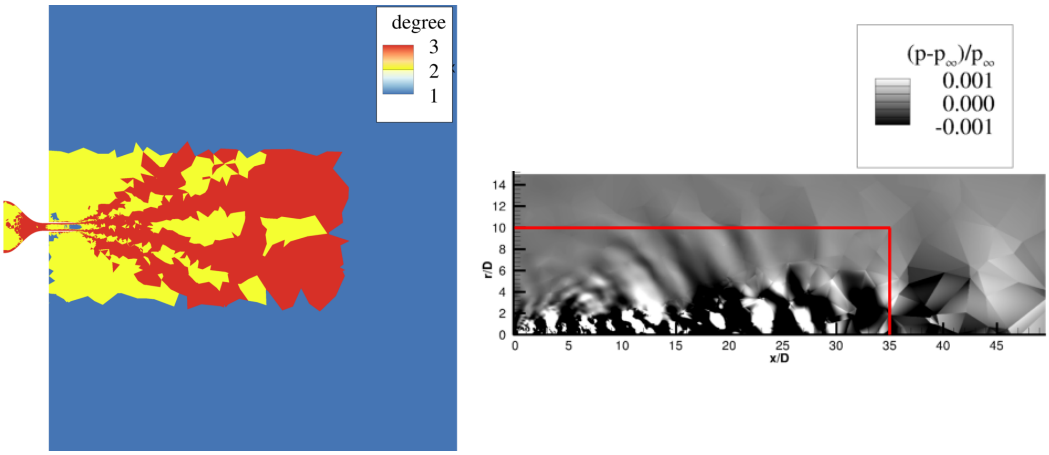


Figure 6.14: *hp*DG/ZDES simulations of the PPRIME nozzle. Slice of the computational domain for the *hp*G1 discretization with polynomial degrees (left). High-order solution interpolated on a fine post-processing mesh of the pressure fluctuations in grey scale, with a sketch of the interface of the sponge layer zone.

Resolution and computational domain The initial mesh, called *hp*G1 in the following, is the *hp*-mesh obtained from the steady *hp*-adaptation using RANS equations to solve the nozzle/jet flow presented in Section 5.3. This mesh counting 11.6 million dofs and providing a sufficient level of convergence with RANS equations, is used as starting point for ZDES mesh adaptation. Two ZDES adaptation steps are performed, yielding a discretization *hp*G2 with 18.9 million dofs, and a discretization *hp*G3 with 31.7 million dofs.

The axisymmetric computational domain extends from approximately $-10D$ (the longitudinal length of the nozzle in the x-direction) to $50D$ in the longitudinal direction x, and from $-30D$ to $30D$ in the radial direction. A sponge layer is created for elements presenting their cell centers more than 35 diameters far from the nozzle exit in the streamwise direction $x_c > 35D$, and 10 diameters from the axis of the jet $r_c > 10D$ in the radial direction. The mesh elements in these regions have a very large size which can compromise the stability of the simulations when turbulent structures or high amplitude sound waves pass through them, if a high order of accuracy is defined in those cells. An elegant way fitting to the present *hp*-methodology consists in forcing these elements to a low order $p = 1$ discretization. This creates a sponge layer which further damps the aforementioned problematic phenomena as well as reducing the overall computational burden of the simulations. This forcing is kept active through the adaptive procedure. In the left panel of figure 6.14 a slice of the domain representing the polynomial degrees, with the sponge layer in $p = 1$, is showed. In the right panel of figure 6.14, the high-order pressure fluctuations field is projected on a fine mesh. The acoustic field is briefly analyzed to assess the lack of potential acoustic reflections due to the sponge layer zone (highlighted by the red lines) or p-variations. The lack of reflections of acoustic waves across the sponge layer interface is evident, while the identification of acoustic sources due to p-interfaces in the jet would require a finer analysis. However, no critical reflec-

tions seem to be affecting the acoustic field, which correctly propagates to the far-field with a preferred direction in the range between $\theta \sim 40°$ and $\theta \sim 20°$ above the jet axis.

Adaptation parameters and characteristic time periods The parameters used by MMG and the adaptation module are:

- $h_{grad} = 1.5$, which is the default value used in this work. It provides a good compromise between excessive refinement induced by a low value of h_{grad}, and the loss of isotropy and mesh quality induced by a high value.
- $h_{min,\text{tetra}}$ is set to $h_{min} = 0.016D$ for tetrahedra, which is the length of the smallest edge of the boundary layer prisms which constitutes the interface with tetrahedra. The height of the first element of the prismatic layer is $h_{min,\text{prisms}} = 0.0003D$, and is fixed along the geometry and over the course of the adaptive procedure.
- h_{max} is set as the size of the domain.
- $r_h = 4$ such that between two steps of the adaptation process, the edge of the equilateral triangle can at most divide by four its size.
- $f_r = 2$ such that the desired number of dofs in the newly adapted mesh is twice the number of dofs present in the previous mesh. We remind that the adaptive procedure, together with the remeshing step performed by MMG, is not always able to respect the exact desired number of dofs, while however providing a good guess.
- $p_{\text{min}} = 1$ is the minimum polynomial degree of the spatial discretization allowed for *hp*-adaptation.
- $p_{\text{max}} = 3$ is the maximum polynomial degree that the spatial discretization is allowed to reach for *hp*-adaptation.

The time intervals required to pass the transient and collect the error estimator statistics, introduced in Section 6.1.2, are chosen as follows:

- $\Delta t_{\text{sampl}} = 0.0067 t_c$, which corresponds to 150 samples for each t_c;
- $T_{\text{trans}} = 50 t_c$, is sufficient for the solution to adapt to the new resolution and provide relevant error and smoothness estimates, not biased by transient phenomena;
- $T_{\epsilon\sigma} = 30 t_c$, is chosen long enough to correctly capture the main features of the flow, but short enough to avoid increasing the computational time of the adaptation process;
- $T_{\text{stats}} = 150 t_c$, providing sufficiently time-converged statistics, especially in zones relatively close to the nozzle exit and the end of the potential core. In farther regions, where the flow field is dominated by lower frequency phenomena, the simulation may need longer collection times to achieve full convergence.

Turbulence modeling approach In the current work a combination of the ZDES mode 0 (RANS) and ZDES mode 1 model is employed. In particular the model is set to RANS mode inside the nozzle, as depicted in figure 6.15.

The hybrid length in the destruction term of the SA equation is defined as:

$$\tilde{d}_{ZDES} = \begin{cases} d_w & \text{in RANS zones} \\ \min(d_w, C_{DES}\tilde{\Delta}_{DES}) & \text{in DES zones} \end{cases} \tag{6.2}$$

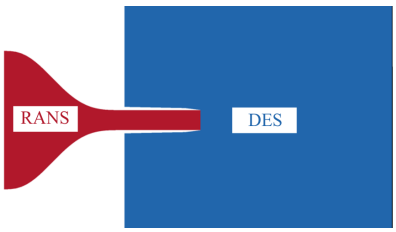


Figure 6.15: *hp*DG/ZDES simulations of the PPRIME nozzle. Forcing of the interior of the nozzle to RANS mode.

The use of RANS modeling inside the nozzle allows us to ensure the correct boundary layer evolution inside the nozzle, while employing RANS-like streamwise and azimuthal mesh spacings, and hence decreasing the computational cost that a Wall Resolved LES simulation would have required. However, no turbulent injection/tripping techniques are here employed. In fact, a RANS-like streamwise and azimuthal resolution in the nozzle, while reducing the computational cost of the adaptive simulations, would be insufficient for the development of small turbulent structures (see Chapter 2).

Adaptation strategy starting from RANS-adapted meshed The strategy employed consists in performing first a series of *hp*-adaptations using affordable steady RANS simulations which in turn provide a starting mesh for the hybrid RANS/LES adaptation that already captures some important features of the unsteady simulation. This approach, where the accurate and expensive hybrid RANS/LES adaptation is performed starting from a RANS-adapted mesh, is preferred over starting the adaptation process on a very coarse mesh with hybrid RANS/LES simulations. Such a mesh would prevent the turbulent structures of the flow from developing, and could yield numerical instabilities and very poor quality results. Moreover it would dramatically increase the computational time for the whole adaptation process, as many unsteady adaptation steps would be needed to reach accurate results.

Starting from a RANS solution is also natural in the present hybrid RANS/LES context, as the wall-attached flow (and eventually other critical zones using a Zonal DES approach) is solved in RANS mode, and a ZDES statistically-steady flow can be easily established from a RANS solution, after a transient period T_{trans}.

In this section we employ as a first mesh for the adaptation, called *hp*G1 in the following, the third adapted mesh in Section 5.3 using $f_r = 2$. We preferred to run hybrid RANS/LES simulations starting from this mesh and not the $f_r = 1.5$ grid with a similar

number of dofs. Despite the faster convergence of $f_r = 1.5$ meshes for RANS computations, the adapted meshes present some features strongly adapted to the RANS solution, as the massive refinement of the RANS high-gradient zones around the potential core. This is appropriate for RANS simulations, while from an *a priori* knowledge of the studied turbulent jet, we preferred a mesh with smoother topological features, to adapt to the new unsteady flow with a higher degree of flexibility.

Park *et al.* [234] explored the potential of hybrid RANS/LES mesh adaptation, while using RANS adapted meshes as initial meshes for DDES fixed-point adaptation. Although the method was not completely mature, they successfully applied metric-based mesh adaptation to wing and aircraft configurations. A full validation and a deeper study of scale-resolving adaptive simulations from RANS-adapted meshes could be of interest, but exceeds the scope and the computational means of the present work. Here we intend to show an application of *hp*-adaptation for industrially-relevant flow configurations, and a pragmatic approach is chosen in order to deal with realistic computational resources.

6.4.3 Flow field results

Computational meshes

Figure 6.16 shows the three *hp*-meshes employed in this study: *hp*G1, the initial *hp*-mesh of the ZDES adaptive procedure, adapted from a previous RANS adaptation; *hp*G2, the first adapted *hp*-mesh from a ZDES simulation; *hp*G3, the second adapted *hp*-mesh from a ZDES simulation.

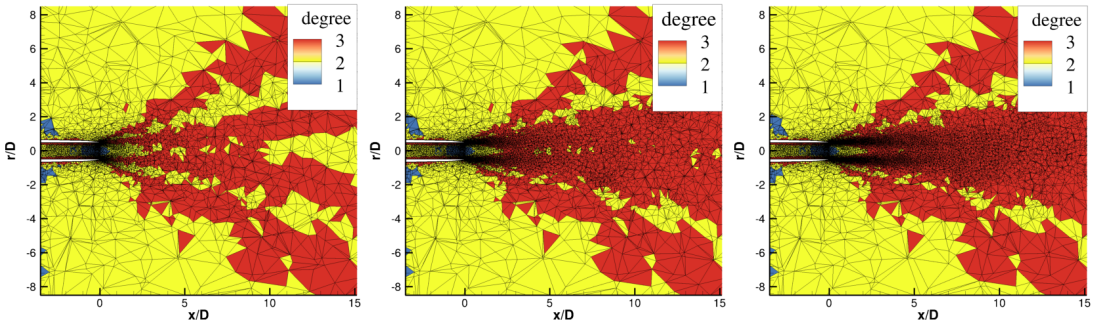


Figure 6.16: *hp*DG/ZDES simulations of the PPRIME nozzle. Zoom of the polynomial degree map in the nozzle exit region and the early jet plume for the three *hp*-adapted meshes, *hp*G1 on the left, *hp*G2 in the middle, *hp*G3 on the right.

The mesh *hp*G1 mostly presents $p = 2$ and $p = 3$ elements in the jet plume, except at the exit of the nozzle in the region close to the axis, which is discretized with $p = 1$ elements. The external part of the shear layers presents $p = 2$ elements since the smoothness estimator in previous RANS simulations had detected non-smooth features in this region, and had prevented the algorithm from increasing the polynomial degree in this zone.

A concentration of the *h*-refinement can be observed around the potential core. The zone relevant for unsteady ZDES simulations is the mixing layer zone where the vortices

develop from the lip of the nozzle. In this region the maximum polynomial degree $p = 3$ is selected by the algorithm. As seen from TGV tests in [37], this property is desirable because high-order DG discretizations describe the turbulence more accurately than low order discretizations with an equivalent number of degrees of freedom.

hp-adapted meshes *hp*G2 and *hp*G3 increase the degree of $p = 2$ elements located in the jet plume to the maximum $p = 3$, especially on the axis of the jet. Globally, the polynomial degree distribution remains the same for the three meshes *hp*G1, *hp*G2 and *hp*G3, since the RANS adaptation had already broadly identified the flow regions of interest and provided a reasonable initial discretization.

The most important difference between the three *hp*-meshes resides in the size of the elements in the jet plume. The mixing layer, presenting strong vortical structures needing very fine resolutions to be adequately captured, is progressively *h*-refined, since the maximum polynomial degree has already been reached. Higher degree computations would need *ad-hoc* load balancing techniques, which are not the subject of the present study, but will be addressed in future work. We also note that very high-order unstructured DG discretizations can present robustness or efficiency issues, which might not be compatible with applications in a challenging industrial context.

In figure 6.17 we display the distribution of the error estimator on the mesh *hp*G1 and *hp*G2, according to which *hp*G2 and *hp*G3 are built.

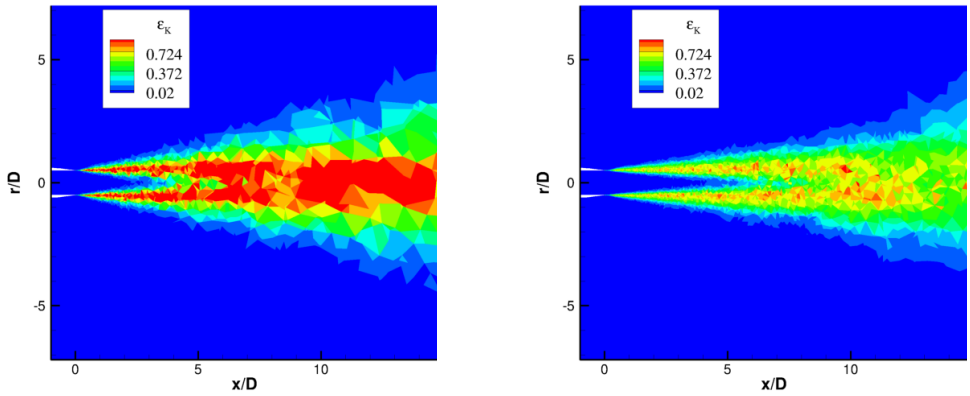


Figure 6.17: *hp*DG/ZDES simulations of the PPRIME nozzle. Zoom of the distribution of the error estimator in the nozzle exit region and the early jet plume for the *hp*-adapted meshes, *hp*G1 on the left and *hp*G2 on the right.

The error estimator is found to correctly identify the most unresolved zones of the flow, well detecting the turbulent shear layers of the jet. In particular we notice a growing intensity of the error estimator in zones corresponding to high rms values of the velocity, whose profiles will be presented in the following sections, in figures 6.22 and 6.24. These regions are mainly the lipline and the centerline at the end of the potential core. This behavior is reflected by the distribution of the degrees of freedom in figure 6.16, where the elements in the lipline are *h*-adapted in both the adaptive steps, being already at their maximum polynomial degree $p = 3$, and the elements in the centerline are first mainly *p*-adapted (*hp*G2) and then *h*-adapted (*hp*G3). A difference with respect to the steady RANS adaptations performed in Section 5.3 lies in the concentration of

the dofs obtained by the adaptation. Indeed the most refined region for scale-resolving simulations is mainly located in the turbulent shear layers, while for RANS equations it is mainly identified in the high mean velocity gradient zone of the potential core. While we believe that RANS-adapted meshes can provide an appropriate initial guess for scale-resolving adaptations, the requirement of performing proper scale-resolving adaptation is highlighted by the pronounced differences in the solved flow features by the two turbulence modeling approaches.

Qualitative instantaneous field analysis

A first qualitative assessment of the results is now conducted by the means of the visualizations of the instantaneous flow field for the three *hp*-meshes *hp*G1, *hp*G2, *hp*G3, which are presented in figure 6.18 and 6.19 and compared.

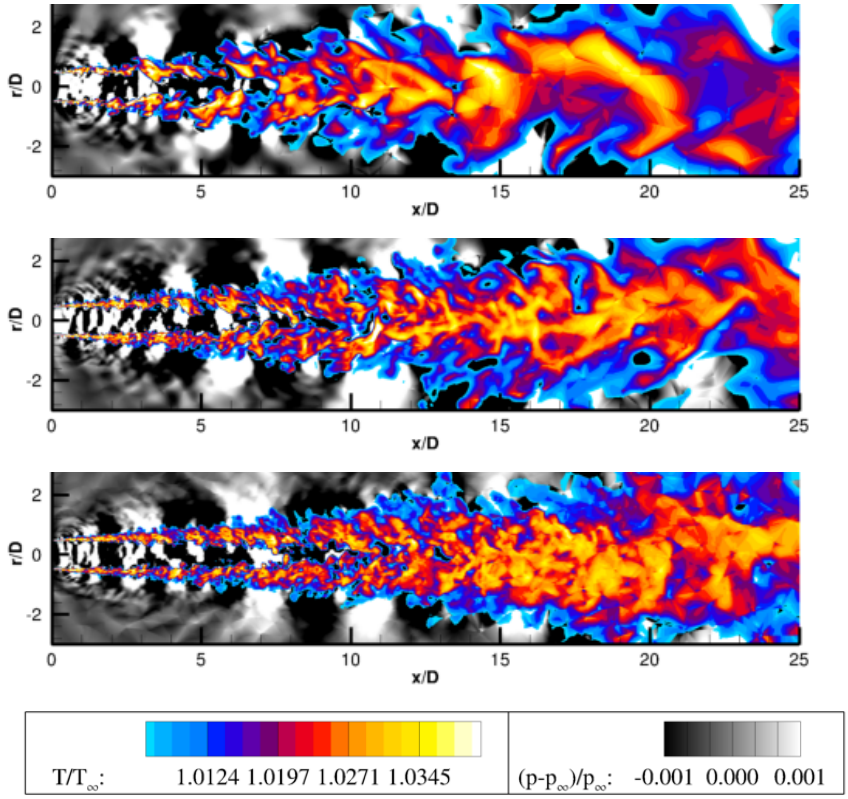


Figure 6.18: *hp*DG/ZDES simulations of the PPRIME nozzle. Iso-contours of the temperature T/T_∞ in colored-scale and the pressure fluctuations $(p - p_\infty)/p_\infty$ in greyscale. Zoom of the *hp*G1 mesh (top), the *hp*G2 mesh (center) and the *hp*G3 mesh (bottom).

Figure 6.18 describes the instantaneous fields of temperature (colour) and pressure fluctuations (grey scale), for the meshes *hp*G1 with 11.6 millions dofs (top), *hp*G2 with 18.9 millions dofs (middle) and *hp*G3 with 31.7 millions dofs (bottom). These pictures illustrate that the most resolved computation captures smaller turbulent structures, as compared to the two coarser simulations, thanks to the *hp*-adaptation in the zones of

interest of the jet. Moreover, we notice that the use of a finer grid resolution in the jet flow leads to a slower jet development (and a longer potential core). This trend will be further discussed in the next section.

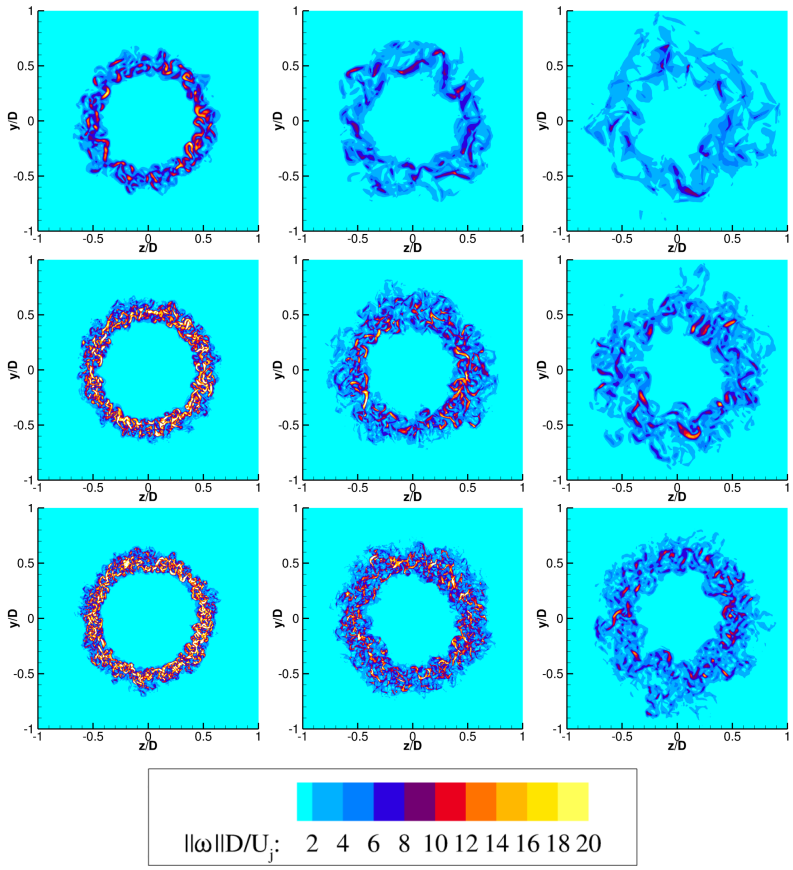


Figure 6.19: *hp*DG/ZDES simulations of the PPRIME nozzle. Iso-contours of vorticity at $x/D = 1, 2, 3$ (from left to right) for *hp*G1 (top), *hp*G2 (middle), *hp*G3 (bottom).

In figure 6.19, vorticity snapshots are shown for three different sections $x/D = 1, 2, 3$ downstream the nozzle (plots from the left to the right), for *hp*G1, *hp*G2, *hp*G3 computations (top, middle, bottom rows). *hp*G1 displays larger structures than *hp*G2 and *hp*G3 resolutions for $x/D = 1$, while the solutions obtained for *hp*G2 and *hp*G3 are almost indistinguishable. More consistent differences can be found for locations further downstream the exit nozzle. While for *hp*G1 the turbulent structures are barely captured at $x/D = 2, 3$, *hp*G2 displays a sharp representation of small structures. This is more marked for *hp*G3.

Quantitative data analysis

In this section, the simulation results are analyzed in terms of mean and fluctuating velocity profiles in the various regions of interest of the jet flow.

First we study the quality of the solution at the exit of the nozzle, and the boundary

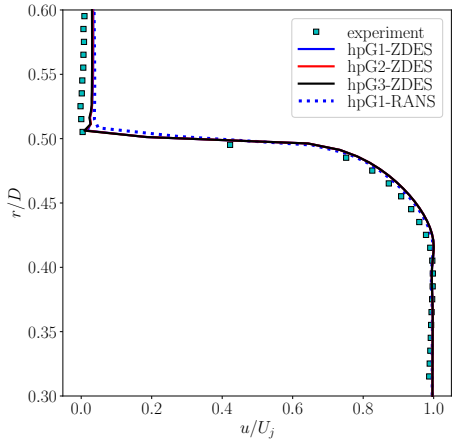


Figure 6.20: *hp*DG/ZDES simulations of the PPRIME nozzle. Boundary layer velocity profile at $x/D = 0.04$. Comparison between *hp*-adaptive and reference results.

layer velocity profile is extracted in figure 6.20 at $x/D = 0.04$. For the three simulations, the velocity profiles for *hp*G1, *hp*G2, *hp*G3, respectively in blue, red and black straight lines, are found to match experimental data in squared symbols and the RANS simulation on *hp*G1 in dotted lines. Note that in the proximity of the nozzle exit, no substantial differences can be found between *hp*G1, *hp*G2 and *hp*G3 hybrid RANS/LES results, because the flow features here mainly depend on the zones solved with RANS equations, which are similarly resolved for the three discretizations.

We note that a turbulent average velocity profile is obtained at the nozzle exit due to the RANS modeling used inside the nozzle, but turbulent injection was not considered in this work, thus some discrepancies between the present results and experimental results or numerical results with turbulence tripping inside the nozzle are expected.

	#dofs (M)	spatial discretization	mesh type
Brès *et al.* [63] G1	16	FV-2o	unstruct hexahedra
Brès *et al.* [63] G2	69	FV-2o	unstruct hexahedra
Gand & Huet [135] G1	48	FV-2o	struct hexahedra
Gand & Huet [135] G2	143	FV-2o	struct hexahedra
Shur *et al.* [272] G1	8.4	FV-ho	struct hexahedra
Shur *et al.* [272] G2	23	FV-ho	struct hexahedra
Lorteau *et al.* [190] G1	78	DG-4o	unstructured tetrahedra
Lorteau *et al.* [190] G2	165	FV-2o	unstruct tetrahedra
*hp*G1 [present]	11.6	hp-DG	unstruct tetrahedra+prisms
*hp*G2 [present]	18.9	hp-DG	unstruct tetrahedra+prisms
*hp*G3 [present]	31.7	hp-DG	unstruct tetrahedra+prisms

Table 6.5: *hp*DG/ZDES simulations of the PPRIME nozzle. Comparison between the number of dofs for simulations in the literature and the present *hp*-adapted simulations.

The mean streamwise velocity profiles on the jet axis $r/D = 0$ and on the lipline

	turbulence modelling	turbulent injection	simulated duration (D/U_j)
Brès *et al.* [63] G1	LES-Vreman	yes	2000
Brès *et al.* [63] G2	LES-Vreman	yes	1150
Gand & Huet [135] G1	ZDES mode 2	no	300
Gand & Huet [135] G2	ZDES mode 2	no	300
Shur *et al.* [272] G1	RANS profile+ILES	no	250
Shur *et al.* [272] G2	RANS profile+ILES	no	250
Lorteau *et al.* [190] G1	LES-Smagorinsky	no	180
Lorteau *et al.* [190] G2	LES-Smagorinsky	no	250
*hp*G1 [present]	ZDES mode 1	no	150
*hp*G2 [present]	ZDES mode 1	no	150
*hp*G3 [present]	ZDES mode 1	no	150

Table 6.6: *hp*DG/ZDES simulations of the PPRIME nozzle. Comparison between parameters in simulations in the literature and the present *hp*-adapted simulations.

$r/D = 0.5$, are compared in figures 6.21 and 6.22 to experimental and numerical results obtained by Brès *et al.* [63] in green lines, Gand & Huet [135] in magenta lines, Shur *et al.* [272] in orange lines, and Lorteau *et al.* [190] in cyan lines, whose parameters and numerical set-up are described in tables 6.5 and 6.6. Dotted lines represent coarse grids for each reference, while dashed lines represent fine grids. Shur *et al.* and Lorteau *et al.* show the mean velocity profile along the streamwise direction at the radial position corresponding to the peak of rms velocity instead of the lipline velocity. However the peak of the rms velocity is found very close to $r/D = 0.5$, and the two quantities can be directly compared.

Before further analyzing the results, the reference data and the related numerical techniques and modeling approaches are briefly reviewed.

Brès *et al.* used a second order Finite Volume approach on unstructured hexahedral-dominant grids, obtained with an *a priori* isotropic mesh adaptation procedure, where it is not an error estimator that guides the refinement process, but an *a priori* knowledge of the physical phenomenon. Starting from a structured mesh with a low number of elements, several embedded zones of refinement with specific target length scale were defined by the user and enforced iteratively by the adaptation tool. Moreover further mesh refinement was applied inside the nozzle in the near-wall region. Regarding their LES strategy, the Vreman subgrid model was used, coupled with an equilibrium wall model inside the nozzle, which allows for selecting element sizes significantly larger than $y^+ = 1$ at the wall. They also used synthetic-turbulence boundary conditions to model the boundary-layer trip present in the experiment on the internal nozzle surface.

Gand & Huet adopted hexahedral structured meshes with a second-order FV scheme, modeling the turbulence with ZDES mode 2 and ZDES mode 3 approaches. Here we consider only their ZDES mode 2 results, since both ZDES mode 1 employed in this work and ZDES mode 2 aim at modeling the boundary layers inside the nozzle with a

RANS approach without a tripping of turbulent structures and thus account for a similar modeling approach.

Shur *et al.* employed high-order FV schemes on structured meshes. They used a two-stage simulation procedure in which the inflow velocity profile was imposed from a RANS solution previously obtained on a coupled nozzle/jet plume RANS simulation, excluding the geometry from the ILES computation. This allowed them to reproduce the effect of the boundary layer considerably reducing the high cost of a coupled nozzle/jet plume LES. Because of this, a direct comparison between the number of dofs employed in their simulations and in our adaptive DG *hp*-meshes cannot be done in a straightforward manner. However, it is still interesting to compare our results with those obtained by Shur *et al.* as in both cases the nozzle exit profile is obtained from RANS modeling approaches.

Lorteau *et al.* adopted a fourth-order DG approach on fully tetrahedral meshes. No tripping procedure was employed, and the Smagorinsky subgrid model was used for LES. Their only DG simulation results (G1) are shown in dotted lines. The shear layer was here laminar close to the nozzle exit, as no turbulent injection technique was considered to trigger the transition to turbulence within the nozzle.

For each of the four numerical references, results obtained on two grid resolutions (except for Lorteau *et al.* for which only one DG resolution is available) are extracted and compared to the present *hp*-adaptive results. At first we observe that by refining

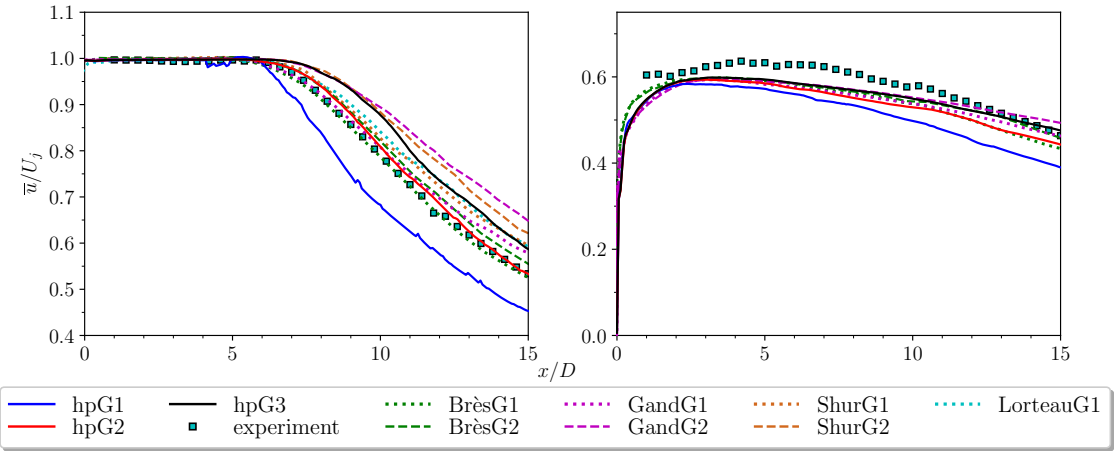


Figure 6.21: *hp*DG/ZDES simulations of the PPRIME nozzle. Mean streamwise velocity profiles on the jet axis $r/D = 0$ (left) and on the lipline $r/D = 0.5$ (right). Comparison between *hp*-adaptive and reference results.

(in h and p) the jet plume, the velocity decay on the jet axis becomes slower, leading to longer potential cores (as already observed in figure 6.18). Surprisingly, we observe that the adaptive simulation *hp*G2 seems to provide closer results to the experiment, while the more refined *hp*G3 mesh overestimates the length of the potential core. While this results might appear unexpected, this behavior has been already pointed out by Shur *et al.* [272] (in orange lines), and can be found in results from Gand & Huet [135] (in magenta lines) as well. The *hp*G3 fine mesh provides an axial mean velocity profile

in very close accordance with their most refined simulations. It should be reminded that these two modeling approached are very similar to the one employed in this work. However, a similar overestimation of the length of the potential core is also observed in pure LES simulations on the fine grid by Brès *et al.* [63] (in dashed green lines) in a less marked manner, and by Lorteau *et al.* (in cyan lines). The reasons for this overestimation of the jet potential core length are still not clear. Shur *et al.* hypothesized two reasons for this behavior: one is that it might be caused by the use of ILES, while the other is that it might be caused by the feedback loop between jet turbulence and nozzle not accurately reproduced by the system of boundary conditions. The second observation would be valid also for our ZDES mode 1 model, because the small-scale variations of the turbulent flow cannot travel upstream and interact with the interior of the nozzle, which is modeled in RANS. Lorteau *et al.* attributed this overestimation to the lack of an appropriate turbulent injection technique to trigger the transition to turbulence within the nozzle, as it happens in the experiment.

Even though a formal answer has not been found, we think that a combination of several effects could be the cause of this behavior. In particular the lack of turbulent injection: the turbulent transition and formation of jet coherent structures here occurs in the shear layer, outside the nozzle. This consists in the process of vortex rolling-up and pairing, after which three-dimensional turbulence appears. This may affect both the flow and the acoustic solution in different ways [63, 190, 132]. We remind here that the experiment presents a fully turbulent boundary layer, after triggering the turbulence inside the nozzle with a strip. In this case, the boundary layer turbulent structures formed inside the nozzle are likely to influence the dynamics of shear-layer structures formed in the vicinity of the nozzle exit. A similar tripping technique could be considered as future work in order to elucidate its effect on the jet plume flow dynamics and verify if the simulations converge towards the experimental results when the mesh is refined or adapted. It is also possible that the resolution is still insufficient. In this case, a fourth more refined simulation (not considered in this thesis due to limited computational resources) should be performed in order to assess the flow properties in terms of mean and fluctuating profiles and verify the trends observed in the previous three *hp*-simulations.

Compared to reference simulations, the axial mean velocity is in very close accordance with the most refined simulations by Gand & Huet and Shur *et al.,* while for $x/D > 10$ the decay of the velocity is faster for *hp*G3. This is due to a lower resolution in this zone with respect to the two authors, suggesting that a further adaptation step providing additional refinement for $x/D > 10$ might eventually slow the decay of the centerline velocity.

Analyzing the mean velocity profile on the lipline in the right panel of figure 6.21, reveals the impact that grid refinement has in the development of the shear layer, especially for $x/D > 2$. Here grid refinement yields an increased velocity along the lipline, which tends towards the experimental profile.

In figure 6.22 we report the rms of the streamwise velocity profiles in the centerline (left) and in the lipline (right). Looking at the centerline velocity profiles, we notice that the profile is not smooth, but presents spurious peaks. These peaks are pronounced for the coarse simulation *hp*G1, while they tend to decrease in intensity thanks to the adaptive grid refinement. This aspect can originate from the DG scheme, where the

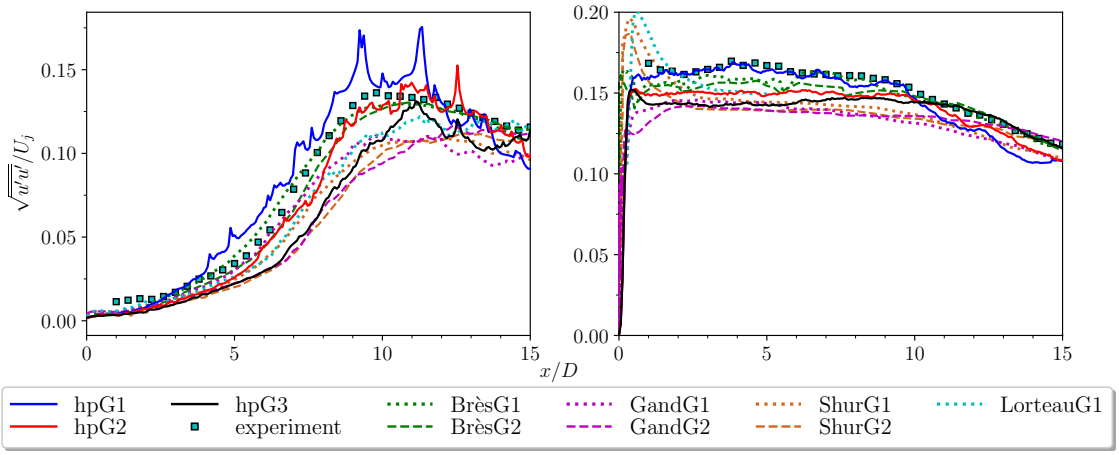


Figure 6.22: *hp*DG/ZDES simulations of the PPRIME nozzle. Root-mean square of the streamwise velocity fluctuation profiles on the jet axis $r/D = 0$ (left) and on the lipline $r/D = 0.5$ (right). Comparison between *hp*-adaptive and reference results.

solution is discontinuous across elements and the polynomial approximating the solution can have very sharp shapes near the interfaces of very coarse elements, but also possibly from the statistical sampling which is low at the centerline due to the inability of performing a spatial azimuthal averaging at this location. This behavior is improved thanks to the mesh adaptation, and peaks in the solution gradually disappear for *hp*G2 and *hp*G3. These numerical artifacts in the solution are not apparent for the lipline plots in the right panels of figures 6.21 and 6.22, as well as the velocity profiles in figures 6.23 and 6.24, due to the better sampling of the solution which is averaged in the azimuthal direction and/or a better resolution in the lipline area.

The small peak seen in the lipline velocity rms profile around $x/D = 0.5$ is very similar for *hp*G2 and *hp*G3, indicating that in the region close to the nozzle exit the two meshes display the same flow features, probably due to the similar size of the elements in this region. A lack of resolution just downstream of the jet exit appears to accelerate the mixing-layer transition, resulting in higher axial (in the centerline) and peak values (in the lipline) of turbulence intensities for the *hp*G1 simulation as compared to the more refined simulations *hp*G2 and *hp*G3. This can be seen both in the right and the left panel of figure 6.22 for $x/D > 0.5$. For the finer simulations we observe an underestimation of the turbulence intensities with respect to the experiments in the early phases of the jet development for $x/D < 10$. This behavior is observed as well for the reference simulations. Dotted curves, representing coarser meshes, show higher rms levels than dashed curves, representing the finest mesh for each reference. This is particularly evident for simulations by Shur *et al.* and Gand & Huet. The decrease in axial/peak rms velocities for $x/D < 10$ can be directly associated with the longer potential core length for the fine simulations [51]. In contrast, a lower resolution leads to lower rms axial/peak velocities downstream of $x/D = 10$.

For $x/D > 10$ peak rms levels in the lipline match more closely the experiment when the mesh is adaptively refined, which indicates that the adaptation strategy is able to

refine far wake regions and yield a better representation of the turbulent structures in these regions. The irregular axial rms profiles for $x/D > 10$ suggest that statistics could be collected for a longer sampling period to provide smoother plots. In fact, from $x/D > 10$ the flow has a longer eddy turnover time than the region close to the exit nozzle. Moreover the rms quantities need more time to converge as compared to average quantities, and the axial quantities cannot be azimuthally averaged.

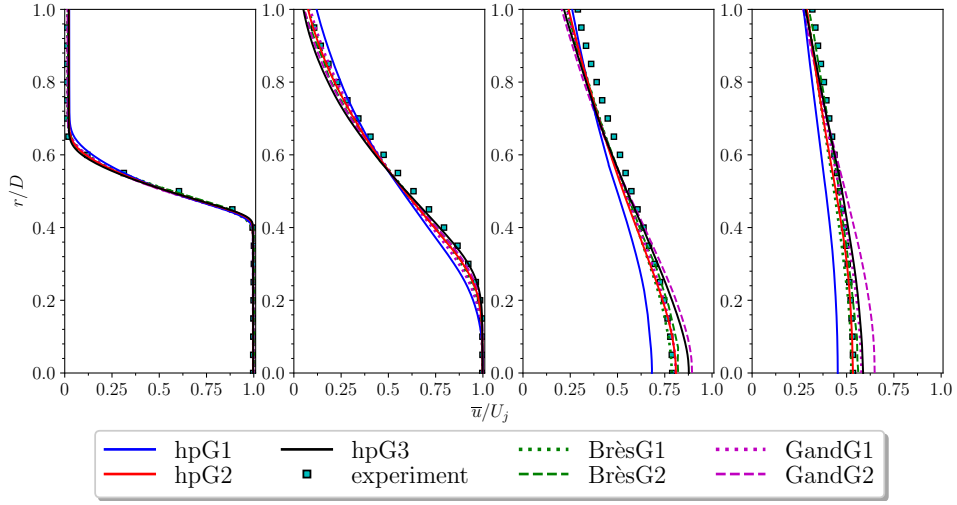


Figure 6.23: *hp*DG/ZDES simulations of the PPRIME nozzle. Mean streamwise velocity profiles at different locations in the jet plume $x/D = 1, 5, 10, 15$ (from left to right). Comparison between *hp*-adaptive and reference results.

Figures 6.23 and 6.24 show the mean and rms streamwise velocity radial profiles at different axial locations downstream of the nozzle exit. For the locations $x/D = 1, 5$ closer to the nozzle exit, we observe an improvement in the prediction of the mean velocity profile during the mesh adaptation process, and a lower shear layer thickness of the jet is observed for *hp*G2 and *hp*G3 simulations. At locations $x/D = 10, 15$, similar observations to those made for the centerline profile in figure 6.21 can be made. *hp*G1 clearly underestimates the mean velocity level close to the centerline. *hp*G2 seems to provide results in close agreement with experimental results, while the *hp*G3 simulation overestimates the velocity for $r/D < 0.5$ at both the locations, especially at $x/D = 15$.

However, the *hp*G3 profile at $x/D = 10$ closely matches the finest simulation by Gand & Huet, and displays intermediate values between their coarse G1 and their fine G2 for $x/D = 15$. This confirms once again the trend of overestimation of the axial velocity when refining the mesh, already observed by other authors who performed numerical simulations on two different grid resolutions.

From the profiles of the rms of the velocity fluctuations in figure 6.24, we see that *hp*G2 and *hp*G3 provide lower levels of turbulent intensities with respect to the *hp*G1 results, for $x = 1, 5, 10$. Since the simulation does not present explicit turbulent structures at the nozzle exit, lower levels of the rms of velocity are expected in the vicinity of the nozzle. The higher levels of the *hp*G1 simulation, apparently closer to the experimental results than *hp*G2 and *hp*G3, are attributed to a lack of resolution, which fictitiously pro-

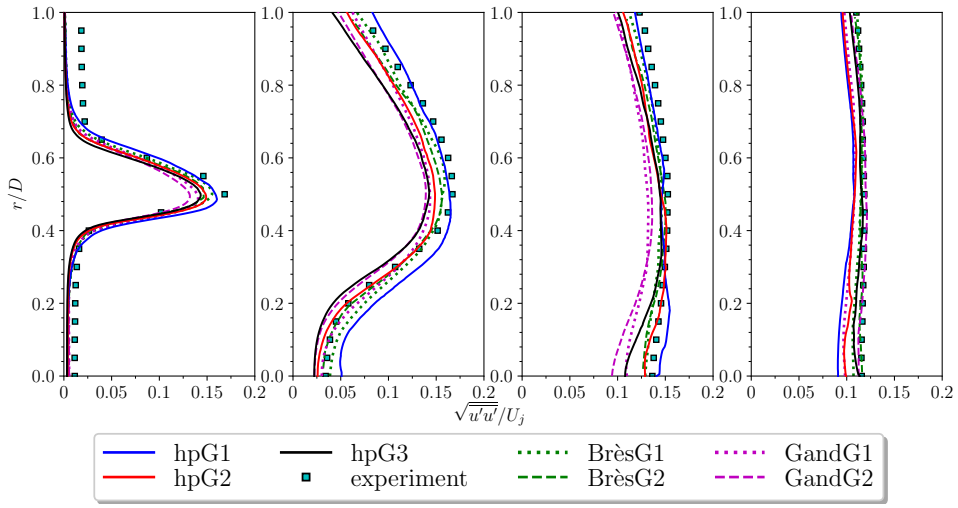


Figure 6.24: *hp*DG/ZDES simulations of the PPRIME nozzle. Root-mean square of the streamwise velocity fluctuation profiles at different locations in the jet plume $x/D = 1, 5, 10, 15$ (from left to right). Comparison between *hp*-adaptive and reference results.

vides a better solution (as already observed in the right panel of figure 6.22). The same behavior is observed for $x/D = 5$ and $x/D = 10$, yet the *hp*G1 simulation shows a substantial overestimation of rms levels when getting closer to the axis. This is related to the coarse grid-induced rapid growth of the shear layer due to the RANS-to-LES transition. This produces larger vortices, leading to a higher kinetic energy.

All these observations agree with the results by Brès *et al.* and Gand & Huet, respectively displayed with green and magenta lines, comparing the coarse (dotted lines) and the fine (dashed lines) simulations. The rms velocity profile at $x/D = 15$ matches more closely the experiment as the mesh is refined, as can be observed also for the simulations by Brès *et al.* and Gand & Huet. This emphasizes the ability of the *hp*-adaptation strategy to detect and refine the turbulent structures in the far wake and improve in turn the estimation of the turbulent fluctuations.

Overall good results, in agreement with numerical references, have been obtained with the present unstructured *hp*-adaptive strategy with a reasonable amount of degrees of freedom. The most refined *hp*-adapted mesh employed here presents a lower number of dofs than both the unstructured DG-$p3$ simulations by Lorteau *et al.* and the structured FV simulations by Gand & Huet, while achieving very close results.

Observations and comparisons with Brès *et al.* and Shur *et al.* in terms of number of dofs employed and achieved accuracy should be analyzed with caution. This is because the former employed both turbulent injection which explicitly guaranties explicit turbulent structures inside the nozzle, and a wall model for LES which allows for defining a coarser mesh inside the nozzle compared to our approach. The latter did not mesh the interior of the nozzle, yielding an overall reduction of dofs compared to approaches with an explicit meshing of the nozzle interior.

To conclude the assessment of the present simulations, the near field pressure fluctuations are analyzed. We directly record the pressure time history at four probes

P1($x/D = 2.62, r/D = 1.07$), P2($x/D = 3.42, r/D = 1.18$), P3($x/D = 4.47, r/D = 1.33$), P4($x/D = 5.85, r/D = 1.52$), downstream the nozzle exit for 36 equally spaced locations in the azimuthal direction. The Power Spectral Density (PSD) is then computed from the pressure fluctuations signal and the resulting spectra are azimuthally averaged.

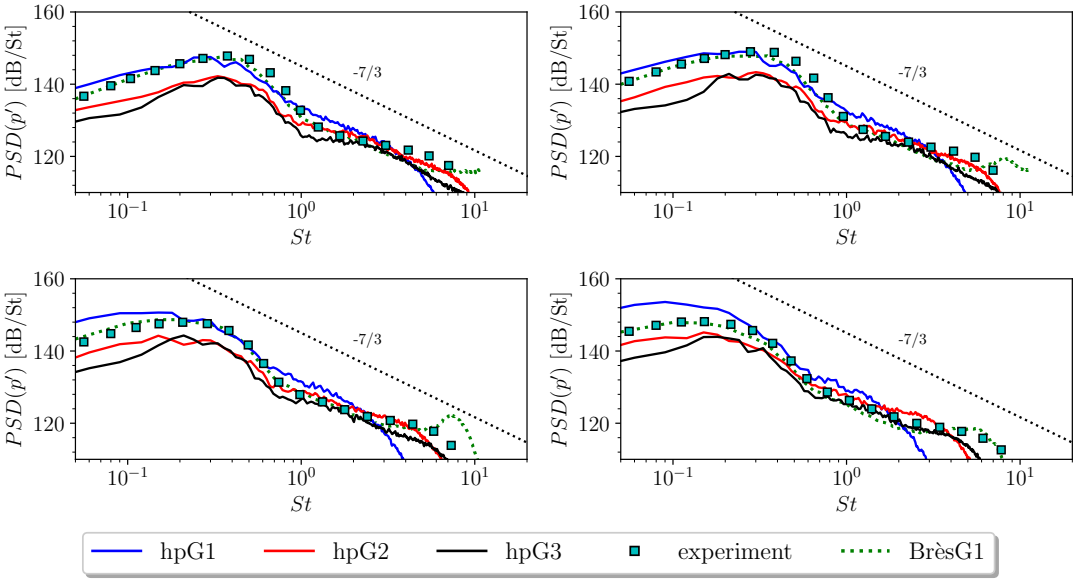


Figure 6.25: *hp*DG/ZDES simulations of the PPRIME nozzle. PSD of the pressure signal recorded at locations P1($x/D = 2.62, r/D = 1.07$), P2($x/D = 3.42, r/D = 1.18$), P3($x/D = 4.47, r/D = 1.33$), P4($x/D = 5.85, r/D = 1.52$) from top left to bottom right.

In figure 6.25 we compare the *hp*G1, the *hp*G2 and the *hp*G3 computations to the experiment and the coarse G1 simulation from Brès *et al.* for which the results are immediately available.

In the inertial subrange, the spectra follow the expected -7/3 slope for the pressure, up to the grid cut-off Strouhal number St_{co}. The St_{co} number depends on the resolution of the mesh in each sampled location, and the more refined grids can resolve the PSD spectra up to increasing St_{co} numbers. This is observed in figure 6.25, where the spectrum of the coarse *hp*G1 starts to deviate from the theoretical behavior at lower St numbers than *hp*G2 and *hp*G3 in all the four locations. In particular we can observe, for example at the location P4 in the right panel of the bottom row of figure 6.25, that the *hp*G1 mesh presents $St_{co} \approx 1.5$, which increases for *hp*G2 ($St_{co} \approx 3$) and *hp*G3 ($St_{co} \approx 4$).

Moving to the medium frequency range of the spectra, we notice a global overestimation of the near field noise levels for the discretization *hp*G1 with respect to the experiment. While the discretization *hp*G2 provides spectra in better agreement with experimental results in the low-frequency range, a slight overestimation of the noise level is still present in the Strouhal range $1 < St < 3$. This overestimation is more marked for the probe located closer to the nozzle exit.

Regarding the finer *hp*G3 discretization, the shear-layer development is characterized by lower pressure spectra levels, in accordance to the lower turbulence intensities

already pointed out in figure 6.22. This results in an overall good prediction of the near-field noise levels in the medium frequency range, and an underestimation in the low-frequency range of the spectra. It should be remarked that, as already discussed, some mismatch between adaptive numerical results and numerical/experimental results are most likely caused by the lack of turbulence injection method, as well as the relatively short duration of the adaptive simulations with respect to reference simulations.

6.4.4 Far-field acoustics computational set-up

For the three *hp*-meshes, we performed computations of the far-field noise on two arrays of microphones: a cylindrical array located at $14.3D$ from the jet axis and a polar array located at $50D$ from the nozzle exit. For each angle of observation, 36 azimuthal microphones have been employed. The cylindrical and the polar arrays are represented respectively in the left and in the right panel of figure 6.27, together with the complete Ffowcs Williams-Hawkings (FW-H) surface chosen for the study.

The FW-H surface integral method is used to estimate the far-field noise implemented in the KIM software developed at ONERA [238]. This method allows for computing the acoustic radiation of the acoustic sources generated by a turbulent flow: starting from the fluid flow field computed with LES on a surface enclosing the noise sources, the noise is propagated to the observer located far from the noise source.

The high-order solution is extracted on a cylindrical-conical-cylindrical surface enclosing the nozzle geometry and the noise surfaces, as shown in figure 6.26, and stored with a sampling period $\Delta t_{\text{FW-H}} = 0.033 t_c$, which corresponds to once every five flow statistics collections ($\Delta t_{\text{sampl}} = 0.0067 t_c$). This surface is closed at the downstream extremity, and the additional flux terms proposed by Rahier *et al.* [238] are used in the FW-H formulation, in order to improve the stability of the noise computed from the different surfaces, especially in the low frequency part of the spectra.

The instantaneous flow field is extracted on four different surfaces of revolution (S1, S2, S3, S4) in figure 6.26, which differ by their radial extent, in order to assess the effects of the position of the surfaces on the acoustic results. The radii of the upstream cylinder-shaped control surfaces S1, S2, S3, S4 are respectively $r/D = 0.68, 1, 1.35, 1.7$. Truncated cones with spreading rate $\alpha = 0.15$ (final radii $r/D = 3.68, 4, 4.35, 4.72$ respectively for S1, S2, S3, S4) connect the upstream cylinders with downstream cylinders. The closing surfaces D1, D2, D3 are located at $x/D = 20, 25, 30$.

Only the results obtained by signals extrapolated from the surface S1-D3 are presented in this section. The independence of acoustic results from the choice of the surface employed to perform the integration is assessed in appendix A on the finest mesh *hp*G3.

The total sampling time for the pressure on the FW-H surfaces is the same as for the accumulated statistics $T_{\text{stats}} = 150 t_c$. However, due to geometrical transients at the beginning and end of the acoustic extrapolation, the exploitable interval reduces to around $130 t_c$. Although we are aware of the relatively short time of collection of the pressure signals, the overall behavior of the *hp*-adaptation on acoustic results is still representative. Some examples of relevant simulations performed over similar amounts of non-dimensional units can be found in [51] for the finest grids and in [190] for similar nozzle

configurations.

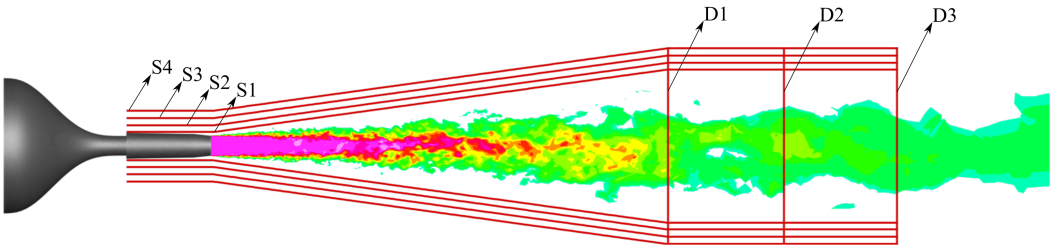


Figure 6.26: *hp*DG/ZDES simulations of the PPRIME nozzle. Schematics of the acoustical set-up. Location of the FW-H surfaces.

6.4.5 Far-field acoustic results

The Power Spectral Density (PSD) is computed on the microphones at different angles of observation, from $\theta = 20°$ to $\theta = 90°$, and is azimuthally averaged on the 36 azimuthal microphones for each angle of observation to compensate for the shorter time signal in the simulations with respect to experiments and reference LES and ZDES results. The Welch method is used to estimate the PSD [304]. A Hann window is applied to each block composed of 1000 data samples and an overlap of 75% is imposed. We have chosen to employ this set of parameters for the spectral post processing, which provides strongly smoothed PSD spectra, in order to help their readability and the comparisons among the different *hp*-meshes. Different parameters for the Welch PSD estimation are tested in appendix A providing very similar results. A low sensitivity of the Overall Sound Pressure Level (OASPL) to the number of overlapping segments chosen for the post processing is also shown in appendix A.

The PSD is reported in [dB/St], as a function of the frequency expressed in terms of the corresponding Strouhal number $St = fD/U_j$, following the same non-dimensionalization used in the experiment and the LES reference results by Brès *et al.*

The resulting spectra obtained on the three *hp*-adapted meshes are shown in figure 6.28, *hp*G1 in blue, *hp*G2 in red, *hp*G3 in black, and compared with experimental results in squared symbols and LES reference results by Brès *et al.* on the G1 mesh in dotted green lines.

In the first place we observe that the noise level is overestimated for the coarse *hp*G1 and the medium mesh *hp*G2 for all the observation angles. The use of more refined meshes globally leads to lower levels of the PSD, especially in the low frequency range. This behavior is in accordance with the lower turbulence intensities pointed out in figure 6.22, when increasing the resolution. This results in a very good prediction of the farfield noise levels by the finest *hp*G3 mesh in the medium frequency range, and a slight under-estimation in the low-frequency range of the spectra for low angles of observations. This behavior is expected, as the turbulent intensities predicted by the *hpG3* mesh in the axis just downstream the potential core, which are correlated to the strong low frequency noise emitted at low angles, are slightly underestimated as compared to experimental measurements.

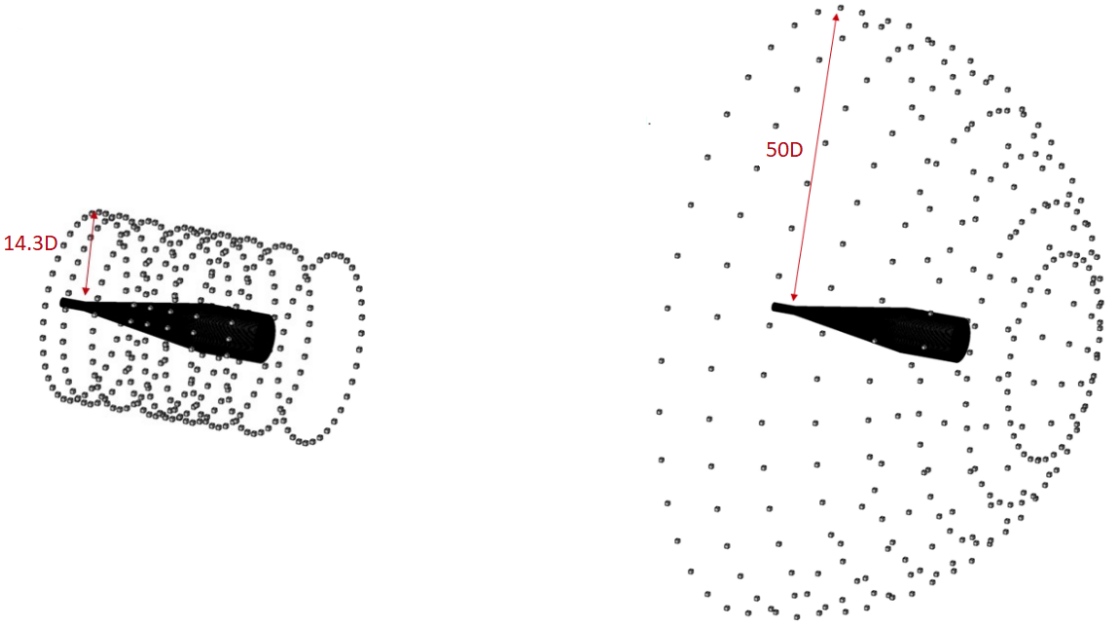


Figure 6.27: *hp*DG/ZDES simulations of the PPRIME nozzle. Representation of the cylindrical (left) and polar array (right) with the chosen FW-H surface S1-D3.

Moreover, for high angles of observation, especially for $\theta > 70°$, the PSD is overestimated in the medium frequency range for all the three resolutions. This is due to the vortex rolling and pairing which in our simulations, performed without turbulent injection or tripping techniques in the nozzle, occurs outside the nozzle and causes increased noise levels with respect to the experiment. However, we can observe that globally *hp*-adaptation leads to spectra which are in closer agreement with the experiment, and a less pronounced bump in the medium/high frequency range.

Finally, the Overall Sound Pressure Level (OASPL) in [dB] is computed in the frequency range $0.05 < St < 3$, for both the cylindrical and the polar array (figure 6.29). Very similar observations to those made for the spectra can be made for the OASPL levels. Globally, the OASPL is overestimated for both the coarse and the medium meshes, especially for high angles. While still overestimating the noise levels of around 1 dB in the polar array and 2 dB in the cylindrical array for high angles of observation, due to the vortex pairing phenomenon, the finest *hp*G3 mesh provides global results in good agreement with the experimental measurements. For low angles of observation $\theta < 40°$, the noise levels are underestimated of around 1-2 dB, due to the underestimation of the low frequencies in the PSD spectrum. Similar results can be observed in the simulations by Shur *et al.* [272], who observed acoustic levels in closer agreement with the experimental measurements, despite a longer potential core, when refining the mesh. This is probably linked to the correct levels of turbulent intensities obtained far from the nozzle on the lipline for $x/D > 10$, observable in the right panel in figure 6.22, and the comparable maximum turbulent fluctuations level on the jet axis in the left panel in figure 6.22, that might compensate for the weaker rms levels observed in the potential core.

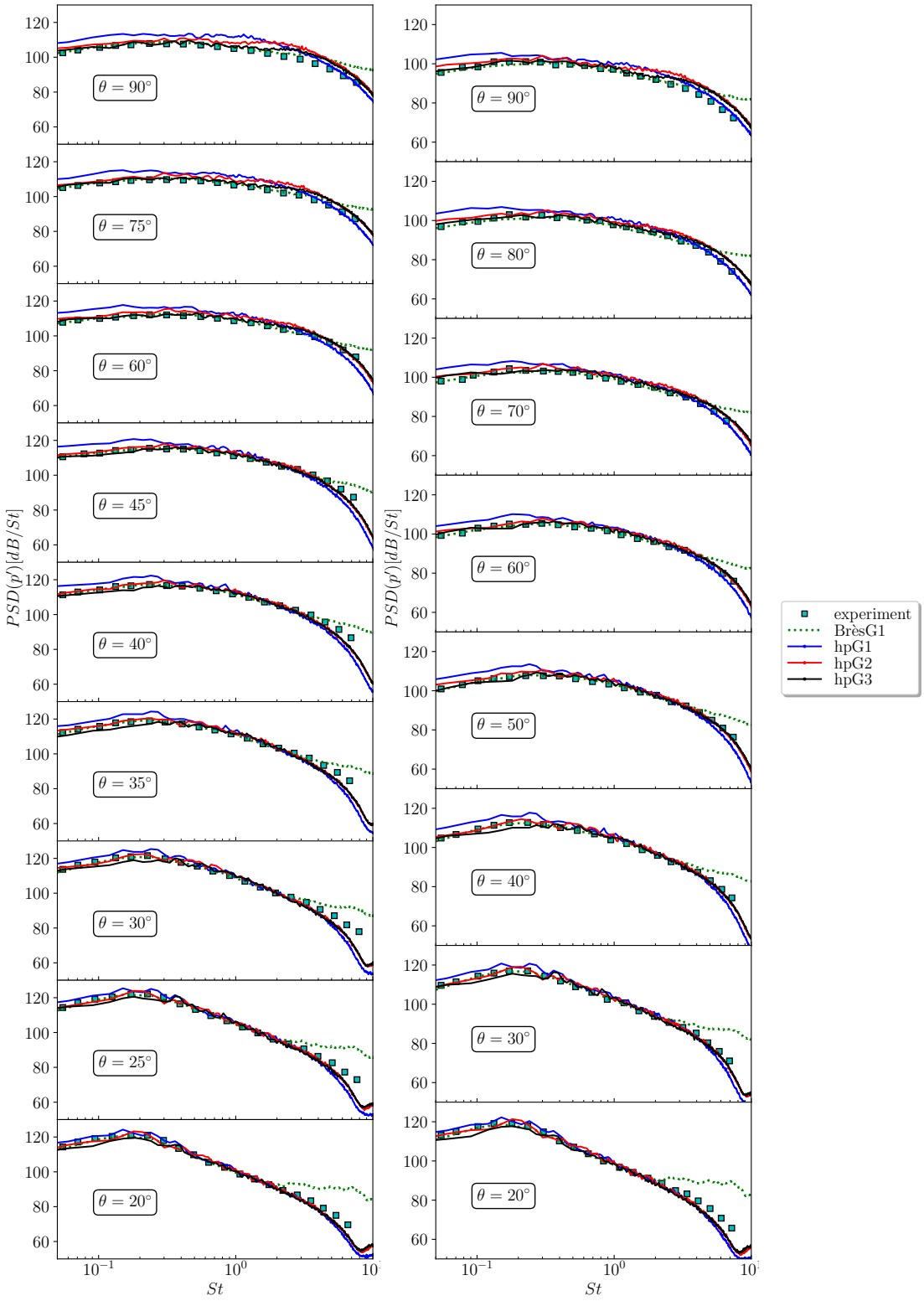


Figure 6.28: *hp*DG/ZDES simulations of the PPRIME nozzle. PSD of the pressure on the cylindrical microphone array of radius r = 14.3D (left) and on the polar microphone array at 50D from the nozzle exit (right) for different angles of observation.

Overall, the adaptation process has significantly improved the estimation of the far-field noise. These results are providing a reliable prediction of the trend that the acoustic levels follow when increasing the resolution. However, as observed by Gand & Huet [135] for the fluid flow results, longer simulation times ($T_{\text{stats}} > 300D/U_j$) should be performed in order to decrease the uncertainties, especially due to the low frequency phenomena characterizing the jet evolution.

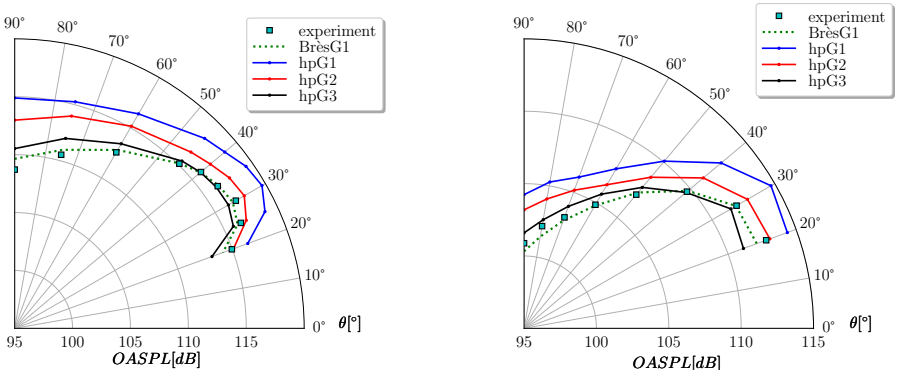


Figure 6.29: *hp*DG/ZDES simulations of the PPRIME nozzle. OASPL on the cylindrical microphone array of radius r = 14.3D (left) and on the polar microphone array at 50D from the nozzle exit (right) for different angles of observation.

6.5 Conclusions

In this chapter, the *hp*-adaptation strategy on hybrid prismatic/tetrahedral meshes has been proposed for scale-resolving simulations of turbulent flows based on hybrid RANS/LES approaches.

Adaptive simulations of the convection of a vortex test case have been first performed to validate the implementation of the static unsteady adaptive strategy. The usual error estimator based on both the measure of the energy contained in the highest order polynomial modes and the jumps of the solution at the element interfaces, and the smoothness indicator based on the decay of the polynomial modes have been averaged over one characteristic period of the flow, before performing the adaptation step. The integral error on the entropy and on the pressure with respect to the analytical solution obtained after 10 rounds of the vortex, has demonstrated a gain in terms of number of dofs of the adaptive strategies as compared to uniform refinement.

For scale resolving simulations, the error estimator has been found to correctly identify the flow regions of interest in a hybrid RANS/LES context. This error estimator has been used in the *hp*-adaptive algorithm in combination with a slightly different formulation of the smoothness estimator employed for steady configurations, better suited for scale-resolving simulations.

The *hp*-adaptive algorithm has been tested on the subcritical flow past a sphere, at Reynolds number $Re = 3700$ using Detached Eddy Simulation. The flow solution obtained on the resulting final *hp*-adapted mesh matches LES numerical results from the

literature, while employing a lower number of degrees of freedom. Some discrepancies with the DNS data show the strong dependency of the *hp*-adaptive process on the employed turbulence model. The use of dynamic and/or VMS models should be preferred when a very high level of accuracy is sought. Although achieving mesh convergence for a modeled scale-resolving simulation is made a difficult task due to the dependency of the model itself on the mesh size, additional simulations have been performed in order to assess the convergence and the quality of the results obtained with the adaptive process. Very similar results are found when uniformly p-enriching the discretization of the adapted meshes, when employing a higher maximum degree in the adaptive process, and when performing a further adaptation step.

hp-adapted DG computations of the PPRIME nozzle at $Re_D = 10^6$ have been carried out in a ZDES mode 1 configuration, and the impact of the increased resolution has been analyzed for three adapted meshes *hp*G1, *hp*G2, *hp*G3 presenting respectively 11.6, 18.9 and 31.7 million dofs. Quantitative results compared fairly well to the numerical references, obtaining overall close results to classical FV schemes on structured meshes and non-adaptive DG methods, with a reduced number of degrees of freedom. The flow solver CODA has been interfaced with the acoustic solver KIM, and the acoustic analysis of the far-field propagated jet noise has been performed on the *hp*-adapted meshes using a FW-H method. The acoustic spectra and OASPL computed from the pressure signal on the microphones located in the far-field present a good agreement between the adaptive simulations and the numerical and experimental references.

However, some discrepancies can be found in the flow field obtained on the finest adapted meshes, with respect to the results obtained by the experimental measurements. In particular, longer potential cores, associated with lower turbulence intensities, are found when increasing the resolution. This appears to be a recurrent issue in the literature, especially when dealing with simulations in which the turbulence is not explicitly triggered inside the nozzle. The impact that the lack of turbulent injection, together with increased spatial resolutions, has on the statistics of the jets, should be carefully validated in order to find an unequivocal answer.

Chapter 7

Towards mesh adaptation for transonic flows over complex geometries: application to industrial configurations

In this chapter, we present the most recent improvements of the *h*-adaptation strategy that we have introduced to account for complex geometries and flows presenting physical discontinuities such as shocks. The aim of these developments is to progress towards *hp*-adaptation for transonic applications of industrial interest. The study presented in this chapter focuses on inviscid flows. An extension towards viscous turbulent flows could be the subject of future research.

An essential feature for the application of the developed adaptive strategies to industrial applications is the correct representation of the geometry at each adaptation step. For relatively simple geometries such as the ones considered in the previous chapters, an initial fine enough discretization of the surface is sufficient to preserve the correct geometry over the course of the adaptive process. However, for industrial design purposes on complex configurations, even a small misrepresentation of the surface can significantly bias the results.

In Section 7.1.1 we describe the implementation in our adaptation tool of the possibility to reproject the surface of the body to the CAD employed to generate the initial mesh. A modification to the current error estimator, tested on subsonic configurations in previous chapters, is proposed in Section 7.1.2 for transonic simulations to improve shock capturing features.

The implementation of the reprojection on the CAD geometry has allowed us to test the *h*-adaptive strategy in Sections 7.2 and 7.3 on two configurations of industrial interest, the transonic ONERA M6 wing and the wing/body configuration of the Common Research Model. Due to the limited efficiency in handling high Reynolds viscous flows by the current isotropic adaptive algorithm, only inviscid flows are considered.

7.1 Modifications to the adaptive algorithm

7.1.1 Reprojection algorithm

The adaptive strategies presented up to this point lack an essential feature which is required for the simulation of wall-bounded flows past complex geometries.

The default MMG surface remeshing capabilities are driven by the h_{Hausd} parameter. This parameter, despite being useful to ensure a proper representation of curved boundaries, is case-dependent. Moreover, even choosing an adequate h_{Hausd} value, given the lack of intrinsic knowledge of the CAD geometry, the initial surface mesh needs to be sufficiently fine to avoid a progressive degradation of the surface representation over the adaptation steps.

In the present work, in order to remove these constraints, we opt for implementing the possibility to reproject the newly adapted surface mesh output from MMG, on the CAD. In figure 7.1 we show the algorithm for *h*-adaptation, with the two new bricks depicted in orange. The different software used in the adaptation are reported in parenthesis for each block.

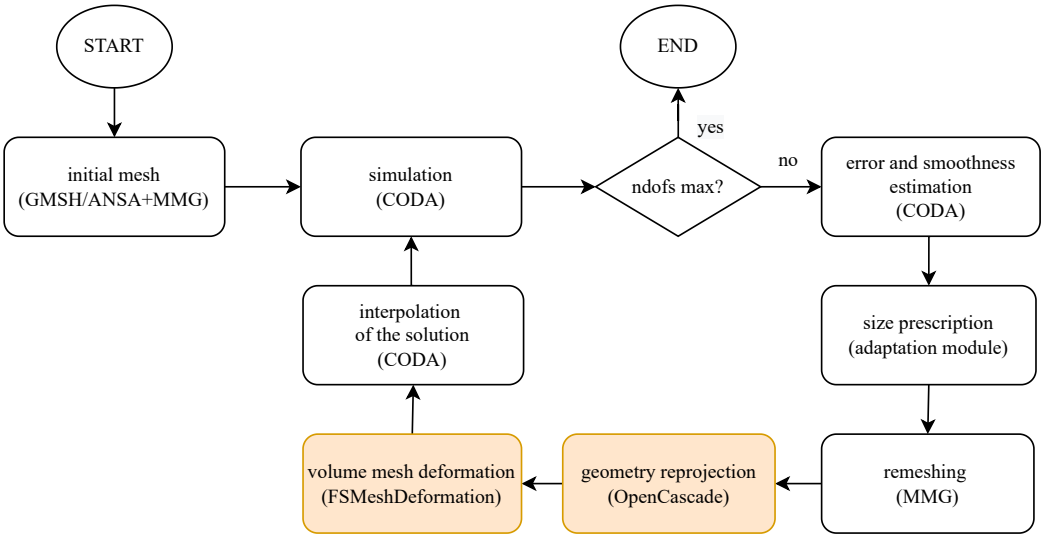


Figure 7.1: Adaptation algorithm for *h*-adaptation modified to handle complex geometries.

For this purpose the *OpenCascade* library [10] with its python wrapper *pythonOCC* [11] is employed. The BREP format is used to store the CAD model (in IGES or STEP format), mainly consisting of vertices, edges, faces. The class `GeomAPI_ProjectPointOnSurf` is employed to compute the orthogonal projection of each point of the body geometry onto the BREP faces from the CAD.

However, this step is not sufficient to obtain good quality meshes. In fact in case of a strong misrepresentation of the geometry and/or very small elements, it can lead to the intersection of 3D elements with the geometry and the onset of mesh elements with a negative volume. A possible solution consists in deforming the produced volume

mesh while preserving unchanged the correct surface mesh. It uses the linearized theory of elastic deformations on the volume mesh, and solves a numerical approximation of the corresponding partial differential equations using the Finite Element Method. The degrees of freedom are in this case the displacements at each node [114]. The solution of the linear system relies on the PETSc solver, in particular in this work we used an ILU-preconditioned GMRES iterative solver.

The developed reprojection capability becomes an integral part of the adaptation module, but can be used also as stand-alone module to fix low quality meshes. An example of misrepresentation of the geometry is presented in the left panel of figure 7.2, while in the right panel we report its corrected counterpart.

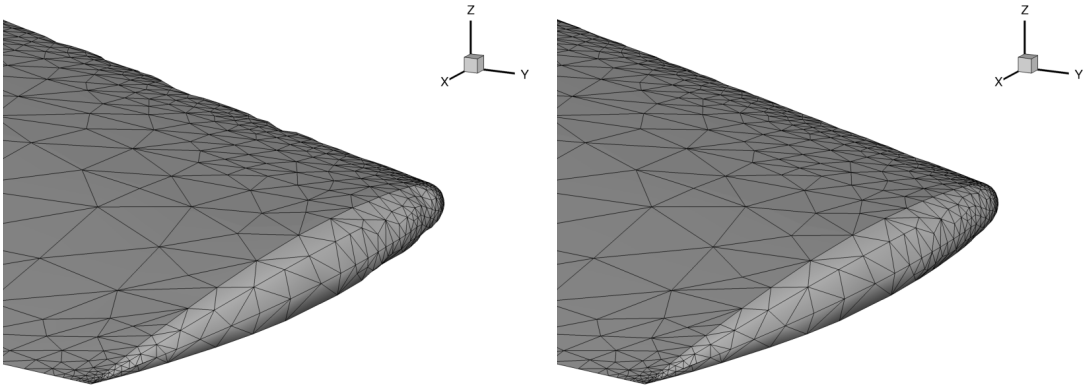


Figure 7.2: Inviscid flow past the ONERA M6 wing at $M_\infty = 0.84$. Zoom on the tip of the wing. The surface obtained from MMG on the left and the reprojected surface on the right.

The h_{Hausd} parameter used to generate the meshes reported in this chapter is the default value $h_{Hausd} = 0.01$. This is suboptimal for the test case of the ONERA M6 wing and the CRM wing/body configuration, for which a lower value should be used. This value has been employed in order to partly avoid the dependence of the mesh adaptation algorithm on this parameter, and let only the error estimator act in highly curved zones. This will be of particular interest for future developments of the present adaptive algorithms, including the use of high-order curved elements. In fact, for high-order meshes it is expected that the number of elements in the proximity of highly curved geometries will decrease as opposed to cases employing linear meshes. Under these conditions, the error estimator should take into account only the physical features of the flow, since the geometric representation of the boundaries would be exactly reproduced by the curved mesh. On the contrary, in linear meshes, the estimator is expected to activate also where very coarse elements are used to discretize highly curved regions. The use in this study of a lower h_{Hausd} parameter would partially bias the assessment of the ability of the error estimator to mark such under-resolved regions.

In figure 7.3 a detail of the nose of the wing/body CRM aircraft configuration, pre-

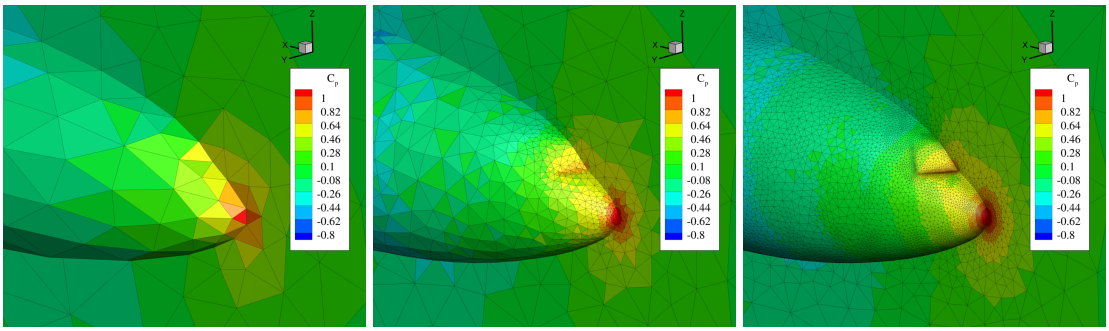


Figure 7.3: Inviscid flow past the CRM wing/body configuration at $M_\infty = 0.85$. Zoom on the wind-shield. Left: intial mesh, center: mesh after 5 $h(p = 1)$ adaptations, right: mesh after 10 $h(p = 1)$ adaptations.

sented with further details in Section 7.3, is shown for the initial mesh and the mesh adapted 5 and 10 times for a $p = 1$ discretization. In the initial mesh, the very coarse linear elements are not sufficiently small to capture the detail of the wind shield. Thanks to the mesh adaptation strategy including CAD reprojection capabilities, we are able to recover the initially hidden geometric details. In contrast, this information would be completely lost and the geometry representation further deteriorated in the absence of the geometry reprojection.

We remark however that the implementation of the geometry reprojection as a correction step, as done in this work, is suboptimal. While it provides good results with an acceptable CPU cost, the optimal implementation would require the interaction between the CAD geometry and the surface mesh directly in the remeshing step.

7.1.2 Error estimator for transonic simulations

The estimator used up to this point has been shown to be accurate for laminar subsonic and scale-resolving simulations, but was not designed for transonic simulations. In particular, the estimator lacks information from thermodynamic quantities that are representative of the shock features. In fact the presence of shocks in the analyzed flow can severely change the effectiveness of the current estimator, requiring the inclusion of shock capturing features.

In this section, we introduce a modification to the error estimator presented in Section 5.1.1 in equation (5.3), and used in previous chapters, in order to efficiently capture shocks. In particular, a contribution of the pressure field is added to both the high-order based term and the jump-based term of the existing estimator.

In the following, the high-order-based contribution of the error estimator, computed from the pressure, will be called ϵ_{SSP} for Small Scale Pressure, and the contribution based on the jump of the pressure $\epsilon_{\text{JUMPpres}}$.

$$\epsilon_{\text{mom+pres},K} = \epsilon_{\text{SSED},K,norm} + \epsilon_{\text{JUMPmom},K,norm} + \epsilon_{\text{SSP},K,norm} + \epsilon_{\text{JUMPpres},K,norm} \tag{7.1}$$

As done for equation (5.3), each single estimator in equation (7.1) is normalized over their respective maximum and minimum values in the domain.

In particular the term $\epsilon_{\text{SSP},K}$ is computed as:

$$\epsilon^2_{\text{SSP},K} = \frac{||\mathrm{p}_{h,p} - \mathrm{p}_{h,p-1}||^2_{L^2(K)}}{|K|} \tag{7.2}$$

and the term $\epsilon^2_{\text{JUMPpres}}$ is computed as:

$$\epsilon^2_{\text{JUMPpres},K} = \frac{1}{N_e} \sum_{e=1}^{N_e} \frac{||\mathrm{p}^+_h - \mathrm{p}^-_h||^2_{L^2(\partial K_e)}}{4|\partial K_e|} \ . \tag{7.3}$$

In the previous equations, $\mathrm{p}_{h,p}$ is the solution computed from the conservative variables, and $\mathrm{p}_{h,p-1}$ is the projection of $\mathrm{p}_{h,p}$ to the element-wise polynomial space $\mathcal{V}^{p-1}$. We remind that in our compressible formulation of the Euler equations, the pressure is not a conservative variable, for which this projection operation would have been trivial. Then the projection of the pressure on the reduced polynomial space $\mathcal{V}^{p-1}$ is not readily available, and must be explicitly computed.

7.2 *h*-adaptation of the inviscid transonic flow past the ONERA M6 wing

In this section, we apply the h-adaptation strategy to the ONERA M6 wing [8, 263]. The ONERA M6 wing is a swept, semi-span wing without twist, which uses a symmetric airfoil. The geometry is modified from its original description to display a sharp trailing edge [9]. This test case has been considered for previous mesh adaptation studies, both in its turbulent RANS [233, 30] and inviscid version [128, 32]. The inviscid configuration has been used for validation of Euler solvers [89] and for shape optimization [196, 262].

The flow conditions are taken from Test 2308 [263], with $M_\infty = 0.84$ and angle of attack $\alpha = 3.06°$. The initial mesh counts 15253 elements, which corresponds to 61012 dofs for a $p = 1$ discretization and 152530 dofs for a $p = 2$ discretization, and is depicted in figure 7.4.

The domain is a half three dimensional C-shaped mesh, with the symmetry plane located at $y = 0$. The root leading edge is centered in the origin in the symmetry plane, and the far-field is located at a distance of approximately 87 mean aerodynamic chords c ($c = 0.64607m$, $b = 1.1963m$ is the semi-span of the wing).

h-adaptation is performed on the ONERA M6 wing for a $p = 1$ and $p = 2$ discretization, using both the standard error estimator used in previous chapters ϵ_{mom} and the estimator $\epsilon_{\text{mom+pres}}$ presented in equation (7.1), and the performance of the newly introduced error estimator is assessed over uniform refinement.

h(p=1) adaptive simulations

Starting with a qualitative analysis, we report in figure 7.5 the meshes obtained after ten adaptation steps by using the two error estimators, ϵ_{mom} on the left panel and $\epsilon_{\text{mom+pres}}$ on the right panel.

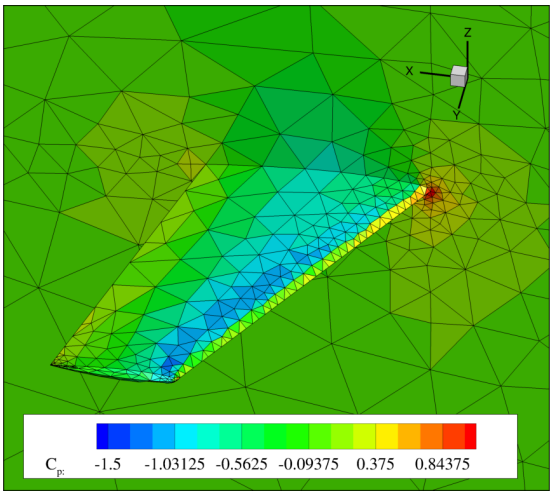


Figure 7.4: Inviscid flow past the ONERA M6 wing at $M_\infty = 0.84$. Pressure coefficient contour on the initial mesh with a $p = 1$ discretization.

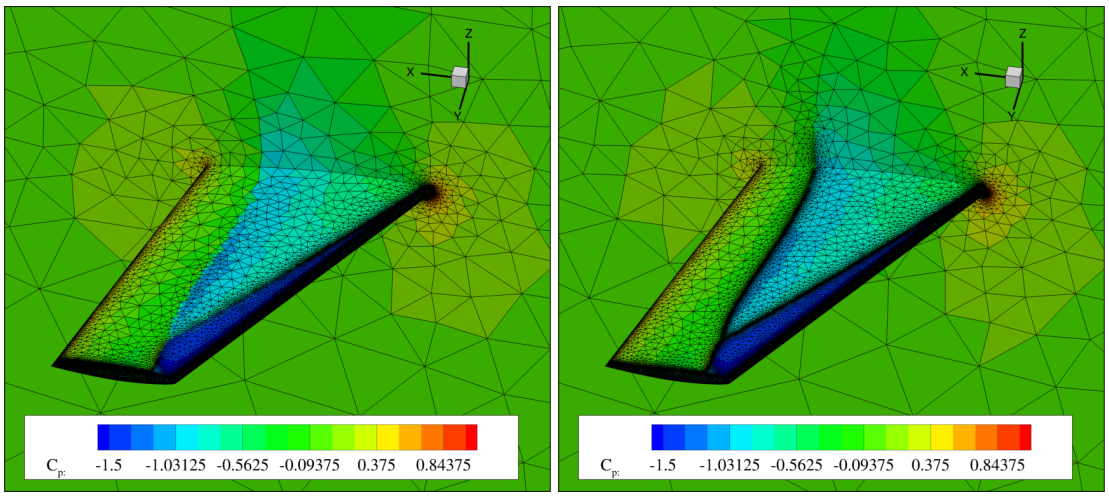


Figure 7.5: Inviscid flow past the ONERA M6 wing at $M_\infty = 0.84$. Pressure coefficient contour on the 10^{th} $h(p = 1)$ adapted meshes with ϵ_{mom} in the left panel and $\epsilon_{\text{mom+pres}}$ in the right panel.

We can observe that the former indicator ϵ_{mom} is unable to fully capture the discontinuities, which characterize the flow. The leading and the trailing edges are well refined, while only one side of the lambda-shock is correctly refined. On the other hand, when employing a combination of error estimators on the momentum and on the pressure, the shock is clearly well resolved, as well as the high-velocity gradient zones at the trailing and the leading edges.

For a more quantitative analysis, we report the convergence history of the two integral quantities C_D and C_L in figure 7.6 obtained with the adaptive procedure guided by the two error estimators. The results are compared to uniformly refined meshes, and to

the C_D and C_L values obtained by Balan *et al.* [32] from adapted meshes as reference. In particular we use the values from their anisotropic Hessian-based mesh adaptation with the finest available mesh counting around 6 million dofs. Given that the anisotropic mesh adaptation is supposed to perform significantly better than isotropic adaptation for transonic cases, we believe that their computations are sufficiently reliable to serve as numerical reference.

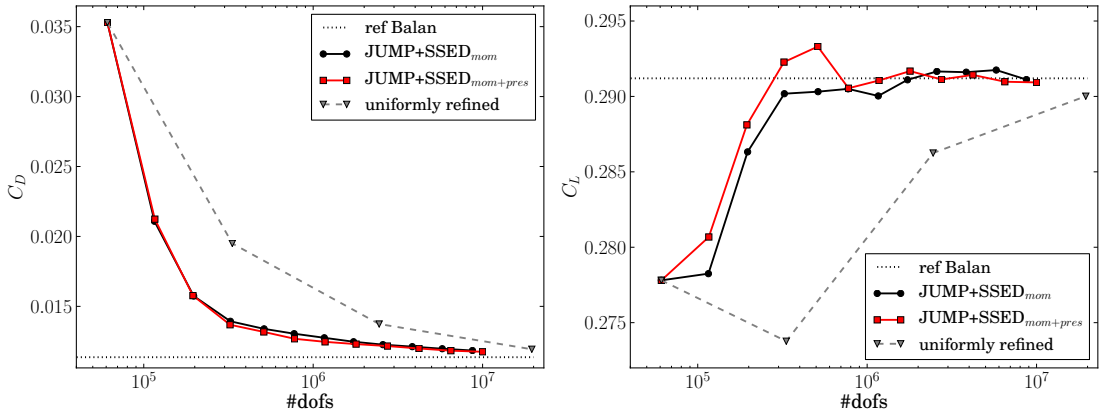


Figure 7.6: Inviscid flow past the ONERA M6 wing at $M_\infty = 0.84$. Convergence history of the integral quantities C_D and C_L vs. number of dofs for $h(p = 1)$ simulations for the two error estimators, compared to $p = 1$ uniformly refined simulations and the reference simulation by Balan *et al.* [32].

From the plots in figure 7.6, no significant differences can be found when using the two error estimators, and they both provide a faster convergence of the integral quantities with respect to the uniformly refined meshes. They appear to converge to the same values of C_D and C_L, in close agreement with the results by Balan *et al.,* and approximately at the same rate. Given the very different topology of the meshes obtained with the two error estimators, it appears that a certain compensation of the effects might be present in the computation of the two integral quantities.

For this reason we analyze the pressure coefficient profiles at 6 spanwise stations over the wing after 10 adaptations (the corresponding meshes are presented in figure 7.5), and we compare them to the results from the experiment and a numerical result from the literature.

In particular we can find Euler computations in the paper from Choi *et al.* [89], performed with the SU2 DG-FEM flow solver. They showed comparisons of the pressure coefficient distribution at different locations of the wing with experimental data. The results included in figure 7.10 are obtained on a mesh counting around 2.2 million elements, with a DG-$p2$ discretization, yielding around 58M dofs (their polynomial basis is different from the one used in this thesis, and involves $(p + 1)^3$ dofs per element).

As expected, inviscid numerical results cannot be directly compared to experimental results, which present much smoother shocks due to the presence of viscosity, but are still representative of the main characteristics of flow, as we can observe in figure 7.7.

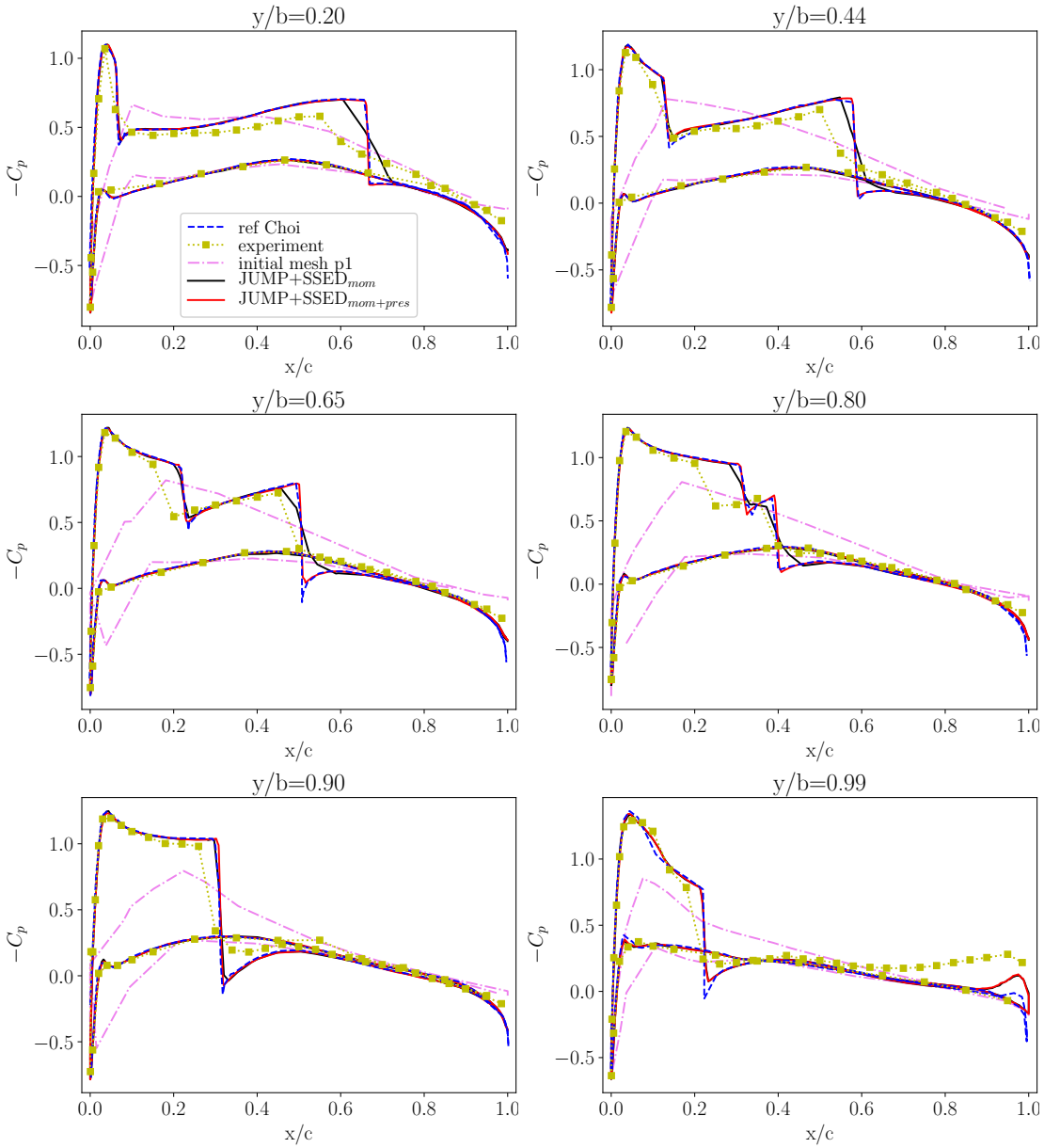


Figure 7.7: Inviscid flow past the ONERA M6 wing at $M_\infty = 0.84$. Pressure coefficient profiles at $y/b = 0.2, 0.44, 0.65, 0.8, 0.9, 0.99$ for the two error estimators on the 10^{th} $h(p = 1)$ adaptation, compared to the reference simulation by Choi *et al.* [89] and the experiment.

The Euler computations presented in this chapter are not meant to be predictive of the real flow, but are a demonstration of the shock capturing capabilities of the adaptive procedure, combined with a more complex geometric representation of the body than in previous chapters.

The adaptive process is capable to recover the physical features of the flow start-

ing from a very poorly accurate initial solution (violet dotted lines in figure 7.7). This is significant as it demonstrates the robustness of the method for its application in an industrial environment.

Comparing now the curves obtained with the two estimators, we observe from the plots in figure 7.7, that the adaptive simulations obtained with the combined error estimator on the momentum and on the pressure in red, match with a very good agreement the results from Choi *et al.* in dashed blue lines. The insufficient resolution of the left shock provided by the meshes adapted using the ϵ_{mom} estimator is highlighted in figure 7.7, especially at the first four stations, from $y/b = 0.2$ to $y/b = 0.8$. The representation of the shock is clearly much smoother than in the numerical reference and the numerical solution from $\epsilon_{\text{mom+pres}}$.

An aspect requiring attention is that for transonic simulations, and even more for inviscid cases, the error estimator always presents high values along the shock. This leads to infinite refinement of the shock if a minimal size is not imposed. While in a viscous simulation the shock wave presents a finite thickness depending on the viscosity and the velocity values, in inviscid simulations the shock should physically have zero thickness, if the discretization errors were not taken into account.

In the adaptation process presented in figures 7.5, 7.6 and 7.7, the value $h_{min} = 3 \cdot 10^{-3}$ is used to limit the refinement. All the other parameters are not varied from Chapter 5, with the exception of the default $h_{Hausd} = 0.01$. Considering the number of 1D degrees of freedom in a DG discretization, we can assume that for a given DG discretization, the effective DG size is $h_{min,DG} = h_{min}/(p+1)$ (similarly to what done in Section 4.2 for the computation of the effective DG $\overline{y^+}_{DG}$ value or in Section 2.2.2 for the computation of the 1D subgrid filter).

For a $p = 1$ discretization, the effective minimum DG size is therefore $h_{min,DG} = 3 \cdot 10^{-3}/2 = 1.5 \cdot 10^{-3} \simeq 1/430c$. We can then assess the dependence of the adaptive strategy on this parameter, by performing computations using a lower $h_{min} = 1 \cdot 10^{-3}$ ($h_{min,DG} \simeq 1/1290c$) and a higher $h_{min} = 5 \cdot 10^{-3}$ ($h_{min,DG} \simeq 1/258c$) as compared to the previously used $h_{min} = 3 \cdot 10^{-3}$.

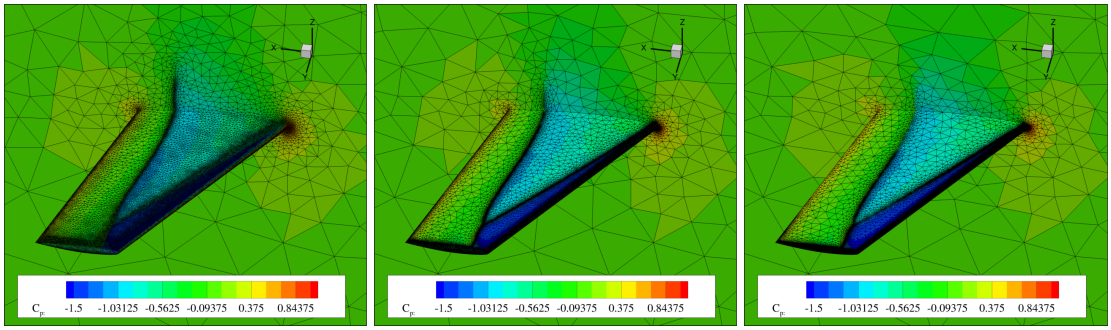


Figure 7.8: Inviscid flow past the ONERA M6 wing at $M_\infty = 0.84$. Pressure coefficient contour on the 10^{th} $h(p=1)$ adapted meshes obtained with different h_{min} values. $h_{min} = 5 \cdot 10^{-3}$ in the left panel, $h_{min} = 3 \cdot 10^{-3}$ in the middle panel (same as the right panel in figure 7.7) and $h_{min} = 1 \cdot 10^{-3}$ in the right panel.

In figure 7.8 the mesh obtained after 10 adaptations with $h_{min} = 5 \cdot 10^{-3}$, shown in

the left panel, has already reached the maximum refinement in the shock, in the leading and in the trailing edge. Once this occurs, the refinement, obeying to the increment of the number of dofs prescribed by the adaptive algorithm, acts almost uniformly on the wing, and in particular in the region between the leading edge and the shock, leading to a large uniformly refined region around the leading edge. This leads also to a refinement of the shock in the volume at a larger distance from the wall than the two meshes with smaller h_{min} values. The middle panel reports the same mesh already shown in the right panel of figure 7.5. The right panel in figure 7.8, corresponding to the lowest value of h_{min}, shows a stronger refinement in the more intense shock region closer to the tip, delaying the refinement of the shock region closer to the root.

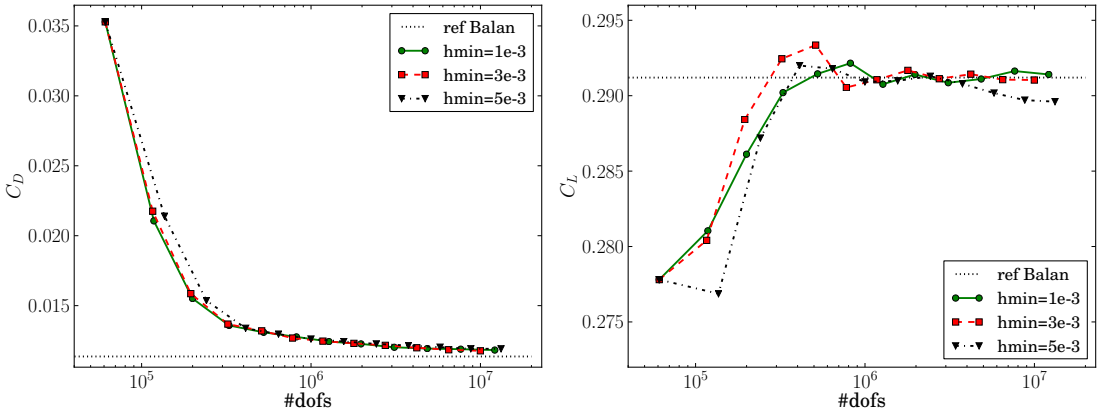


Figure 7.9: Inviscid flow past the ONERA M6 wing at $M_\infty = 0.84$. Convergence history of the integral quantities C_D and C_L vs. number of dofs for $h(p = 1)$ simulations using different h_{min} values.

Looking at the convergence history of the integral quantities in figure 7.9, despite some oscillations in the C_L, the simulations with $h_{min} = 1{\cdot}10^{-3}$ and $h_{min} = 3{\cdot}10^{-3}$ provide very similar values of the C_L and the C_D, while the adaptation with $h_{min} = 5 \cdot 10^{-3}$ tends to significantly lower values of C_L.

The trend highlighted in figure 7.8 is reflected in figure 7.10, where sharper profiles in the shock are found with $h_{min} = 1 \cdot 10^{-3}$ for $y/b \geq 0.8$ with respect to the other two meshes, while the C_p profiles on the lower surface still present some differences with the other profiles, since in this case the adaptive algorithm has concentrated the refinement mainly in the shock region.

The sharper discontinuities that can be seen employing the lowest h_{min} for the $y/b = 0.99$ are explained by a lower thickness of the resolved shock, due to the use of smaller elements. However, we remark that the visualization relies on the element-averaged pressure coefficient extracted in each surface element. Since several dofs are present per element in a DG discretization, we cannot exclude that slightly sharper peaks are present also in the solution employing $h_{min} = 3 \cdot 10^{-3}$, but the size of the element is not sufficiently small to correctly represent it by its mean value.

This study highlights the importance of the definition of a minimal size h_{min} to avoid

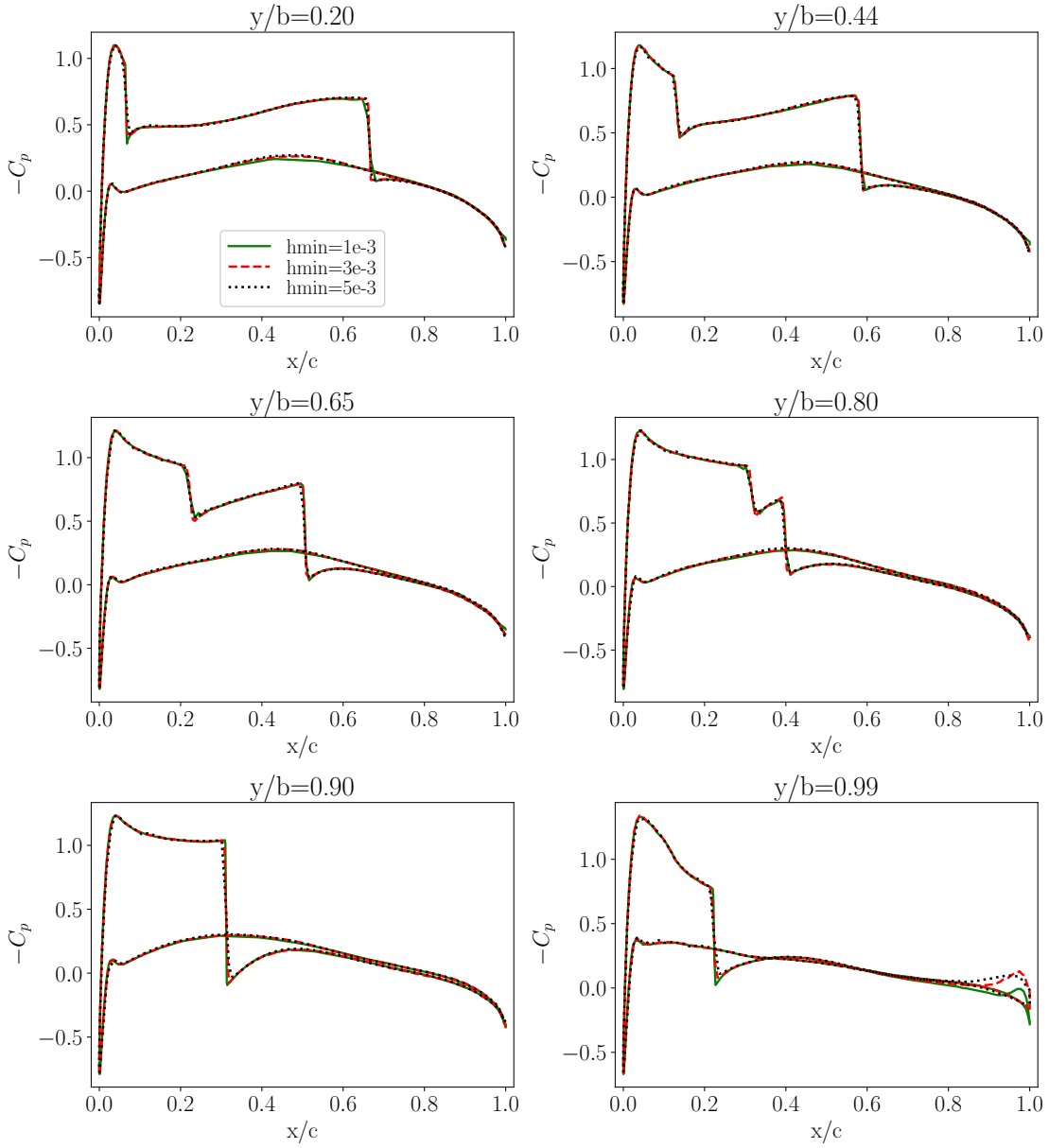


Figure 7.10: Inviscid flow past the ONERA M6 wing at $M_\infty = 0.84$. Pressure coefficient profiles at $y/b = 0.2, 0.44, 0.65, 0.8, 0.9, 0.99$ using different h_{min} values on the 10^{th} $h(p = 1)$ adaptation.

an overrefinement of the shock. At the same time, it has shown that h_{min} does not affect significantly the adaptation process, provided that this value is selected within a range of acceptable values, which is related to the numerical shock thickness. For viscous simulations it is expected that the adaptive process will present a lower dependency on this value, due to the finite shock thickness in the presence of viscous effects.

h(p=2) adaptive simulations

We now consider the results of *hp*-adaptive simulations using a $p = 2$ discretization. The objective of this study is to highlight potential differences in the error estimator behavior between $p = 1$ and high-order discretizations. The value of h_{min} is chosen again as $3 \cdot 10^{-3}$.

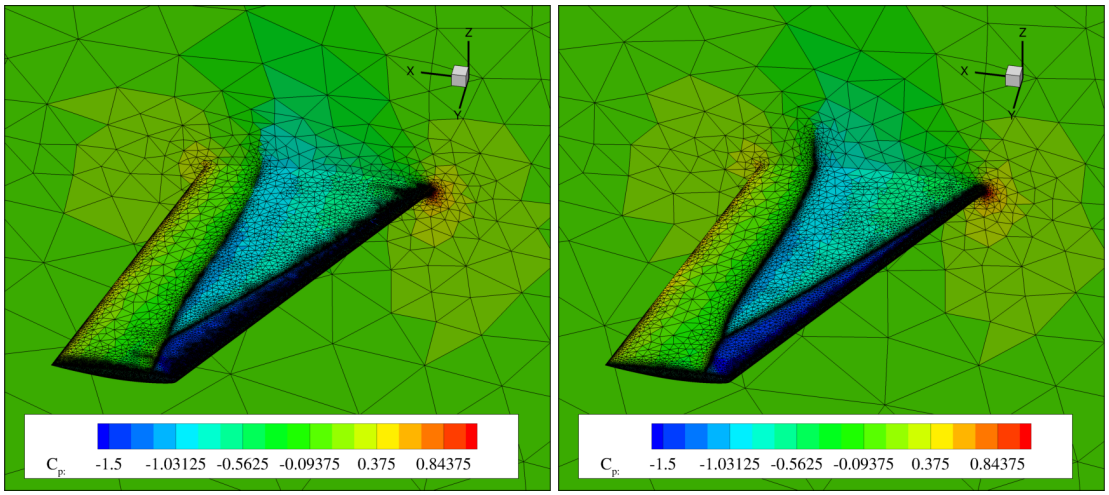


Figure 7.11: Inviscid flow past the ONERA M6 wing at $M_\infty = 0.84$. Pressure coefficient contour on the 7$^{\text{th}}$ $h(p = 2)$ adapted meshes with ϵ_{mom} in the left panel and $\epsilon_{\text{mom+pres}}$ in the right panel.

The meshes obtained on a $p = 2$ discretization using the two error estimators ϵ_{mom} and $\epsilon_{\text{mom+pres}}$ are presented in figure 7.11. The differences between the two meshes are less pronounced as compared to the $p = 1$ simulations in figure 7.5. This is mainly due to two reasons. The first is that the error estimator is more accurate when computed using a high-order discretization as compared to a second order discretization, especially thanks to the increased amount of information provided by the highest-order modes of the solution. The second reason lays in the different exponent in the size reduction formulation in equation 5.9 for $p = 1$ and $p = 2$ simulations. In fact, since the smoothness indicator is not available for $p = 1$ simulations, the exponent is always set to $m_n = p+1 = 2$ in this case. On the other hand, $p = 2$ simulations benefit from a decreased m_n in non smooth zones, which additionally contributes to the reduction of the sizes of the elements over the course of the adaptations.

Despite the smaller differences with respect to $h(p = 1)$ meshes, the mesh obtained with $\epsilon_{\text{mom+pres}}$ in the right panel of figure 7.11 well captures both the shock and the trailing/leading edge, while the mesh obtained with ϵ_{mom} in the right panel has a lower resolution of the shock (we can observe larger elements both on the wing surface and in the symmetry plane).

In figure 7.12 the convergence history of the C_D and C_L value obtained with the estimators ϵ_{mom} and $\epsilon_{\text{mom+pres}}$ on $p = 2$ discretizations are compared to the $h(p = 1)$-adaptation using $\epsilon_{\text{mom+pres}}$.

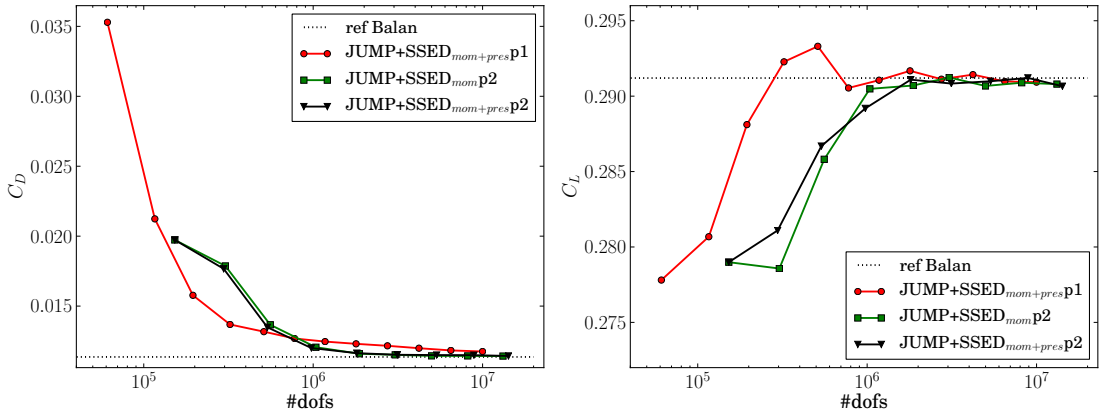


Figure 7.12: Inviscid flow past the ONERA M6 wing at $M_\infty = 0.84$. Convergence history of the integral quantities C_D and C_L vs. number of dofs for $h(p = 2)$ simulations using different error estimators, compared to $h(p = 1)$ simulations and the reference simulation by Balan *et al.* [32].

Both the C_D and the C_L values seem to converge to the same values as $p = 1$ adaptations. A faster convergence than $p = 1$ can be noticed especially for the C_D value after the third adaptation, while this is harder to claim for the C_L, which has an oscillating behavior for the $p = 1$ case and is more sensitive to the representation of the geometry. Large $p = 2$ elements in the leading edge in initial steps are still insufficient to provide the correct solution if the geometry is represented by coarse linear elements. In order to provide as fair as possible comparisons between the two discretizations, quadratic elements should be employed for the $p = 2$ discretization.

The distribution of the pressure coefficient is shown in figure 7.13 in order to further validate the solution obtained on h-adapted meshes.

As expected from the quantitative analysis of the mesh visualization in figure 7.11, the difference in the resolution of the shock between ϵ_{mom} in green lines and $\epsilon_{\text{mom+pres}}$ in black lines is not as pronounced as it is for $h(p = 1)$. Nonetheless slightly improved results are obtained for $\epsilon_{\text{mom+pres}}$ in the two positions closer to the root.

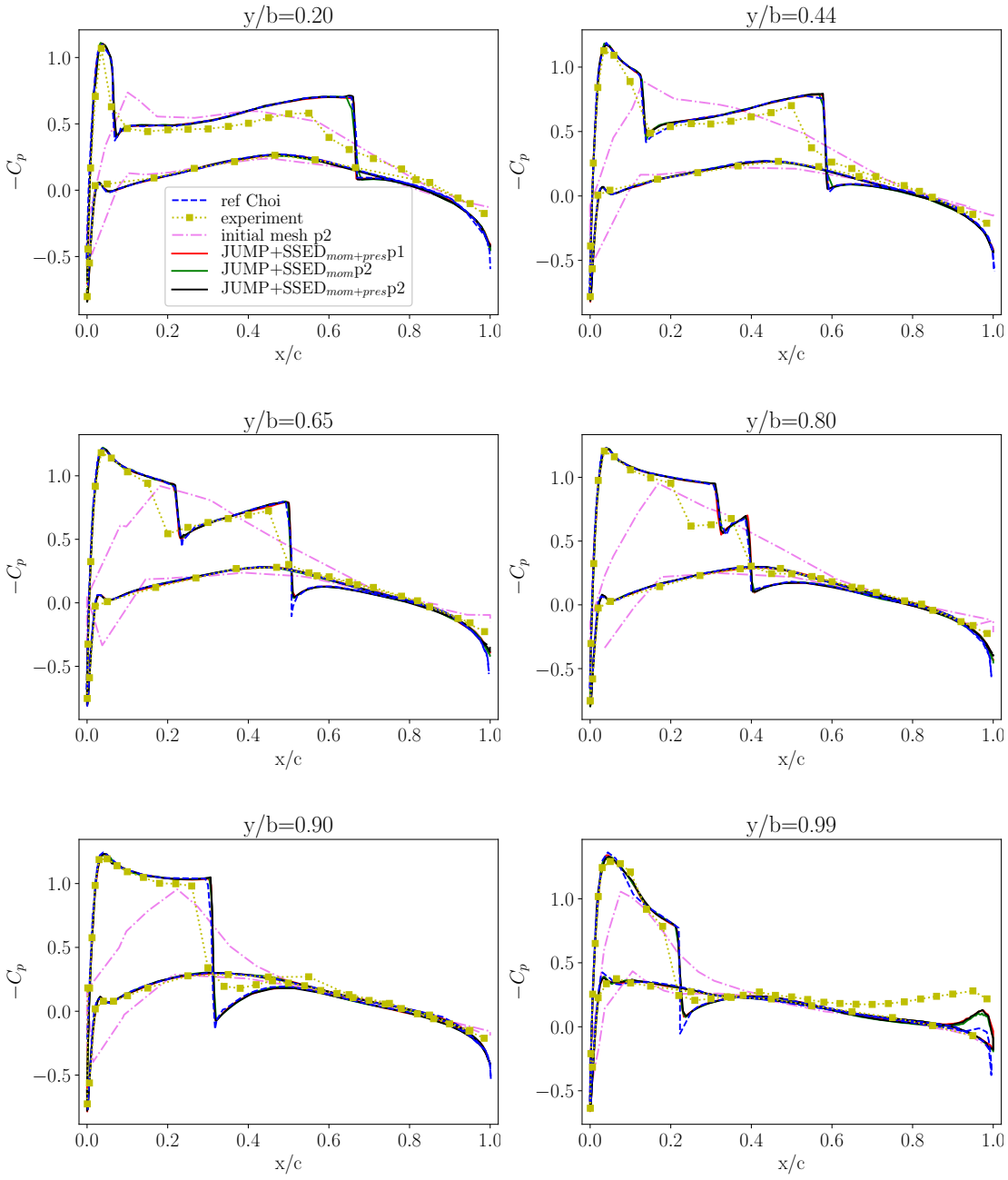


Figure 7.13: Inviscid flow past the ONERA M6 wing at $M_\infty = 0.84$. Pressure coefficient profiles at $y/b = 0.2, 0.44, 0.65, 0.8, 0.9, 0.99$ for two error estimators on the 7^{th} $h(p = 2)$ adaptation, compared to the reference simulation by Choi *et al.* [89], the experiment, and the 10^{th} $h(p = 1)$ adapted mesh.

7.3 h-adaptation of the inviscid transonic flow past the CRM wing/body configuration

In this last section, the h-adaptive procedure is tested on the Common Research Model (CRM) [7]. The CRM is an open geometry configuration designed by NASA and Boeing in consultation with other aerospace industries, to provide a representative geometry of a contemporary transonic commercial transport for the validation of CFD tools. This transonic transport aircraft model with a design cruise Mach number of 0.85 has been used in the last editions of the AIAA Drag Prediction Workshop (DPW) [4] and High Order Workshop (HOW) [5].

The transonic flow at $M_\infty = 0.85$ is considered. The geometry employed here is built from the CAD with an aeroelastic deflection at the angle-of-attack $\alpha = 2.75°$, from the HOW5 (and DPW6). A lower $\alpha = 1.5°$ is here employed to provide C_L values obtained with the inviscid simulation in a range of values comparable to those obtained by experiment and viscous simulations ($C_L = 0.5 - 0.55$).

The CRM configuration has been recently subject to RANS mesh adaptation in the high lift configuration proposed in the Third High Lift Prediction Workshop (HiLiftPW-3) [211, 17, 31]. Inviscid configurations have been mainly employed for validation purposes [279] or shape optimization [24, 268, 217].

The initial mesh in figure 7.14 employed to start the h-adaptive procedure counts 37829 elements, which means 151316 dofs for a $p = 1$ discretization, and 379290 dofs for a $p = 2$ discretization. The domain is half a cubic box, with the nose of the aircraft close to $x = 0$, in the symmetry plane located at $y = 0$, and the far-field located at approximately 417 mean aerodynamic chords c from the body ($c = 0.2758$ in the mesh).

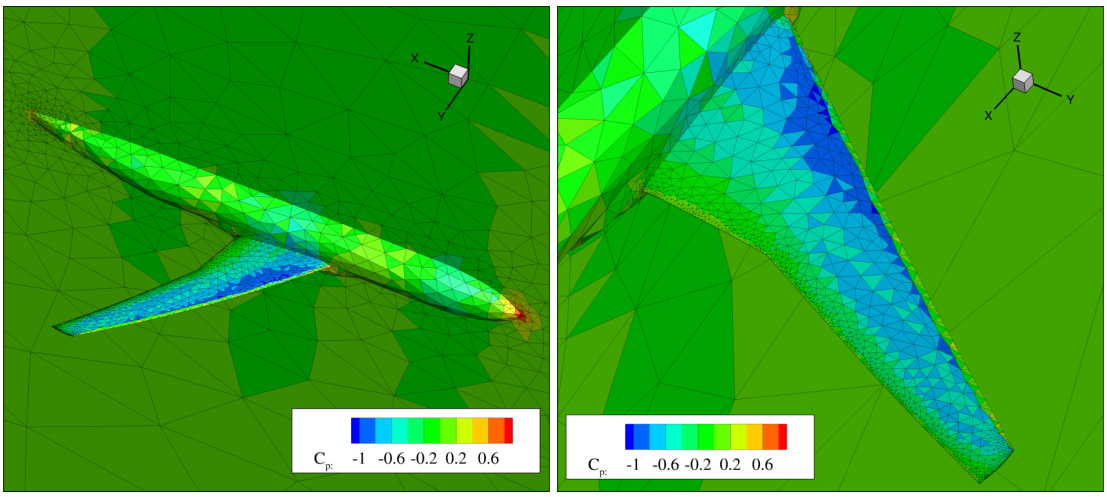


Figure 7.14: Inviscid flow past the CRM wing/body configuration at $M_\infty = 0.85$. Pressure coefficient contour on the initial mesh with a $p = 1$ discretization. Detail of the wing in the right panel.

The error estimator introduced in Section 7.1 is employed to control the h-adaptive simulations for the $p = 1$ and $p = 2$ discretizations. The minimum size is here $h_{min} =$

$1 \cdot 10^{-3} \simeq 1/276c$, with an effective DG spacing $h_{min,DG} = 5 \cdot 10^{-4} \simeq 1/552c$ for $p = 1$ computations and $h_{min,DG} \simeq 3.3 \cdot 10^{-4} \simeq 1/827c$ for $p = 2$ computations.

The surface meshes output from the $h(p = 1)$ adaptive procedure are shown in figures 7.15 and 7.16 for the 7$^{\text{th}}$ and the 10$^{\text{th}}$ adaptation steps, with a contour of the element-averaged pressure coefficient.

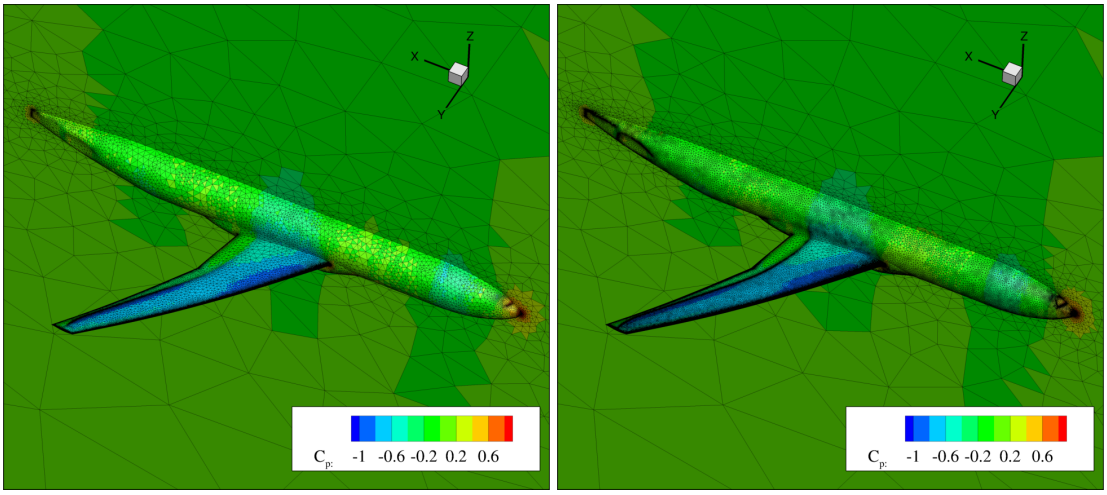


Figure 7.15: Inviscid flow past the CRM wing/body configuration at $M_\infty = 0.85$. Pressure coefficient contour on the 7$^{\text{th}}$ (left) and the 10$^{\text{th}}$ (right) $h(p = 1)$ adapted mesh.

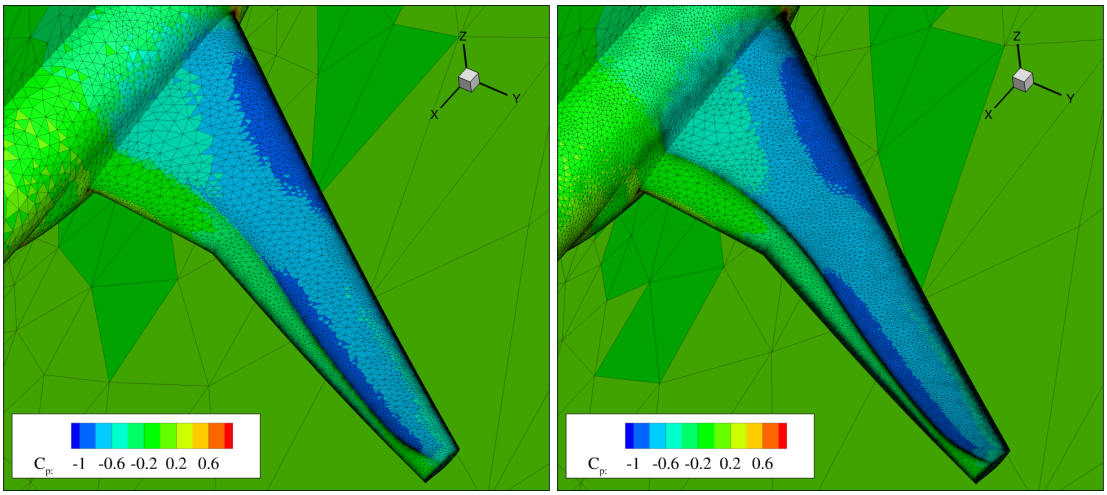


Figure 7.16: Inviscid flow past the CRM wing/body configuration at $M_\infty = 0.85$. Pressure coefficient contour on the 7$^{\text{th}}$ (left) and the 10$^{\text{th}}$ (right) $h(p = 1)$ adapted mesh. Detail of the wing from figure 7.15.

The shock is refined starting from the region presenting the strongest discontinuity, closer to the tip. Then, progressively, the refined shock zone extends towards the root, similarly to what observed for the ONERA M6 wing. The leading edge and trailing

edge are discretized with increasingly smaller elements, and the fuselage is progressively refined as well in a more uniform manner.

These are not the only features that the adaptive algorithm is capable to identify. We take a deeper look by extracting a volume slice on the progression of meshes at $x = 2$ ($x \simeq 7c$) in the left panels of figure 7.17, in particular we report the initial mesh and the meshes after 2, 5 and 9 adaptations. In the right panels of figure 7.17 we post-process the $p = 1$ solution obtained on the four meshes, by interpolating it on a finer mesh, and we show the contours of the vorticity on different planes in the tip region.

Thanks to the adaptive procedure, the trailing edge vortex is progressively refined as seen from figure 7.17. While the initial mesh provides a very limited representation of the trailing edge vortex, the quality of its representation by the vorticity contours is increased over the adaptive process, as clearly observable in the right panels of figure 7.17.

The visualization of the artificial viscosity, designed to track discontinuities and stabilize the numerical scheme, highlights the capability of the adaptive strategy to remove spurious numerical artifacts, while refining the description of the relevant flow features. We show in figure 7.18 the progression of meshes from the initial to the 7^{th} adapted mesh, with contours of the artificial viscosity, with a zoom on the tip region of the wing. The addition of artificial viscosity, aimed at stabilizing DG computations, is supposed to be active only in regions presenting discontinuous features, while vanishing in the rest of the domain.

Initially, the artificial viscosity is mainly active in both the shock and in a large region close to the leading edge, up to the third adapted mesh in figure 7.18. The adaptive algorithm first concentrates some efforts on the leading and on the trailing edge regions (first and second adaptations), which are too poorly discretized to allow the refinement of other flow features. Here high values of the artificial viscosity are present not only in the elements directly connected to the leading edge, but also in a larger region. This is an indication that the very coarse discretization with linear elements of the curved leading edge is leading to high-order oscillations which are identified by the artificial viscosity shock sensor. When the leading edge is refined sufficiently and the curvature is more representative of the real geometry, the artificial viscosity is of lower intensity and is limited to very few elements in the leading edge, in addition to the elements in the shock.

From the third adaptation, the shock starts to be refined, and its position moves backwards up to the fifth adaptation. From this point on, the shock has a stable position and continues to be refined until h_{min} is achieved. The transient positioning of the shock in the first four adapted meshes, does not significantly affect the adaptive procedure, even if no coarsening is employed. In fact, by limiting the reduction in size of each element to be at most two ($r_h = 2$), the refinement produced during these first steps is similar to the refinement due to the h_{grad} parameter, which simply avoids abrupt size variation from one element to the other.

The plots in figure 7.18 have served as well to assess the artificial viscosity techniques implemented in the flow solver CODA on both 3D aircraft configuration and very coarse linear meshes. The possibility to exploit the Persson-Peraire sensor for the choice between h- and p-adaptation for transonic applications, similarly to what was proposed by

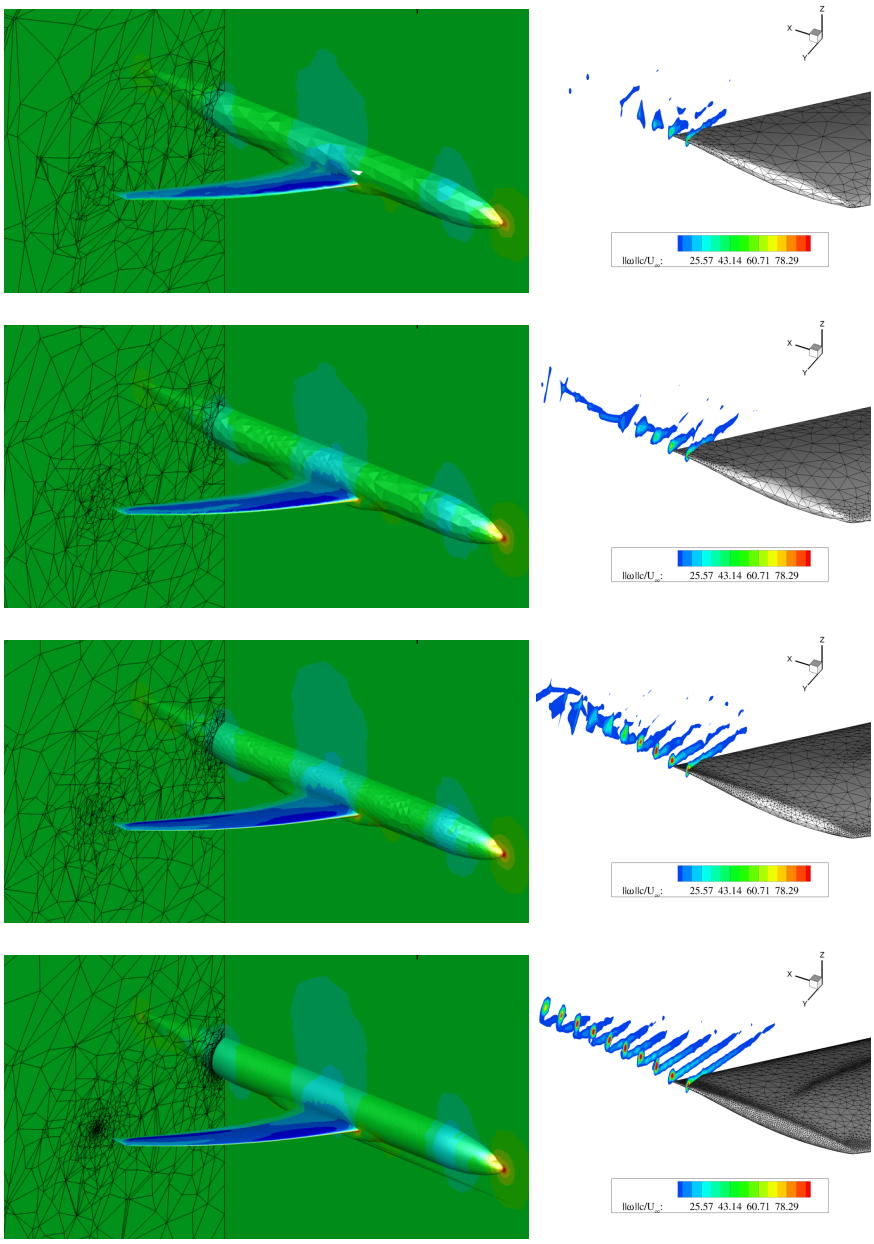


Figure 7.17: Inviscid flow past the CRM wing/body configuration at $M_\infty = 0.85$. Initial mesh, 2^{nd}, 5^{th} and 9^{th} $h(p = 1)$ adapted meshes from top to bottom panels. Volume slice in the $y - z$ plane at $x = 2$ ($x \simeq 7c$) in left panels with pressure coefficient C_p contour, vorticity $||\omega||$ contour on planes $1.88 < x < 1.97$ in right panels.

Wang [301], can be realistically explored in future research also for $p = 1$ elements for this kind of configurations. Although the artificial viscosity distribution is governed by a ramp-up function from smooth to irregular regions, it should be possible to exploit the same threshold values that are used to calibrate this function for each polynomial degree

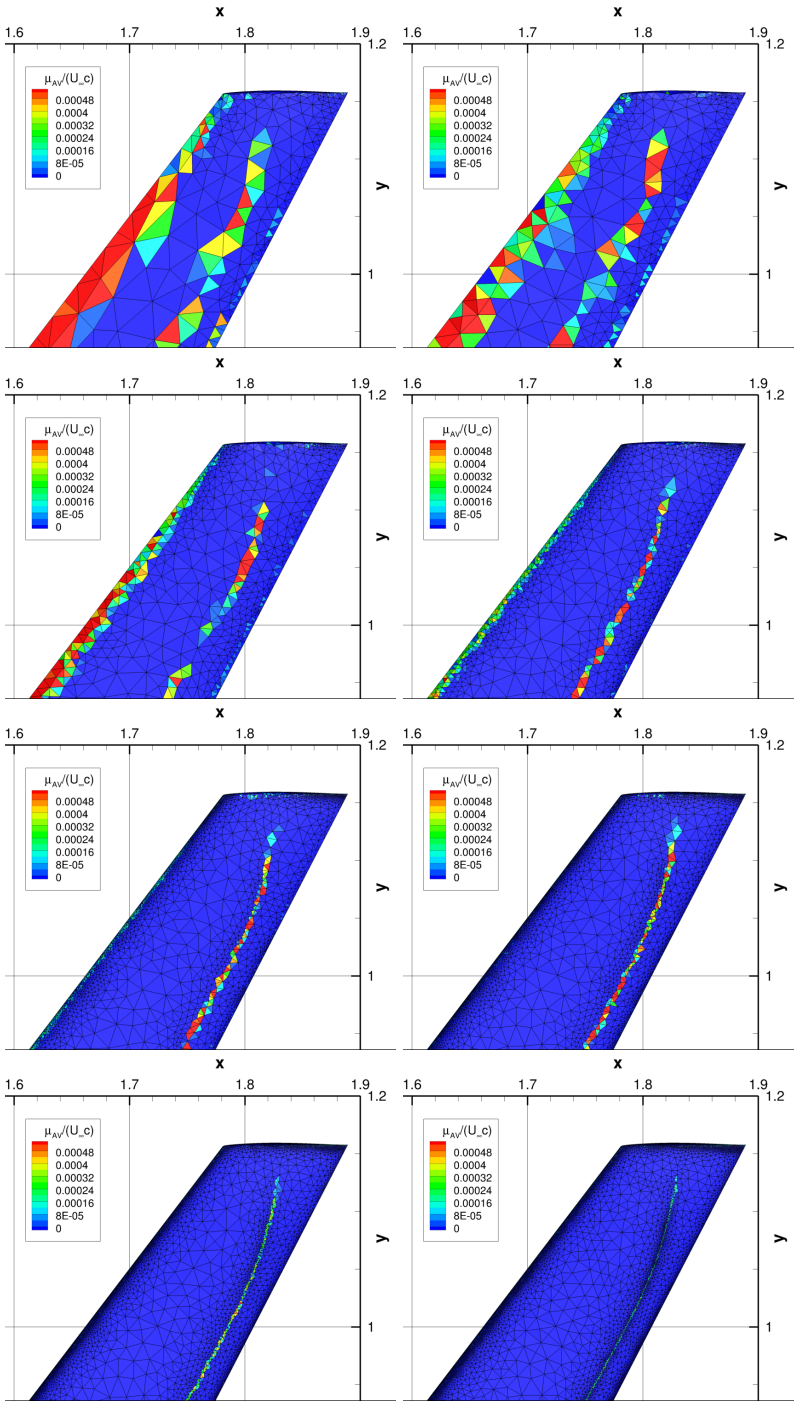


Figure 7.18: Inviscid flow past the CRM wing/body configuration at $M_\infty = 0.85$. Artificial viscosity μ_{AV} contour. Initial mesh and $h(p = 1)$ adaptations 1,2,3,4,5,6,7 from top left to bottom right.

considered.

The convergence history of the C_D and the C_L values is presented in figure 7.19 for uniform refinement with $p = 1$ in grey lines, $h(p = 1)$ adaptation in red lines, $h(p = 2)$ adaption with $\sigma_{thr} = 1$ in green lines, and $h(p = 2)$ adaption with $\sigma_{thr} = 0.5$ in orange lines. The inclusion in this study of two different values of the smoothness indicator threshold will be motivated later.

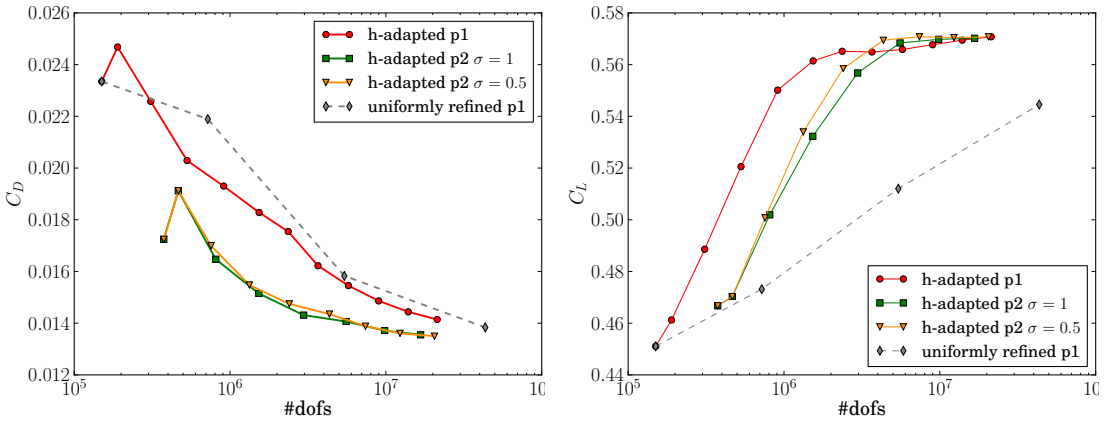


Figure 7.19: Inviscid flow past the CRM wing/body configuration at $M_\infty = 0.85$. Convergence history of the integral quantities C_D and C_L vs. number of dofs for $h(p = 1)$ and $h(p = 2)$ adapted simulations compared to $p = 1$ uniformly refined simulations.

The h-adapted meshes with $p = 1$ provide a faster convergence of both the C_D and the C_L values with respect to the $p = 1$ uniformly refined meshes. The gain in dofs provided by h-adaptation with respect to uniform refinement is however less obvious for the C_D. This is probably due to the interaction between the different components of the drag in an inviscid flow (induced, wave and spurious) that are varying. A far field drag decomposition analysis should be performed [204] to clarify the different contributions to the global drag. The C_D value obtained performing $h(p = 2)$ adaptation for both the smoothness indicator thresholds, converges significantly faster than the $h(p = 1)$ simulations, as expected by the higher order of accuracy, and similarly to what observed for the ONERA M6 wing in figure 7.12. The C_L, after the first adaptations mainly affected by the coarse discretization of the leading edge with linear elements, achieves a converged value with a lower number of dofs than the adaptive $h(p = 1)$ discretization.

We report the C_p profiles in figure 7.20 obtained from the initial mesh, from the last $h(p = 2)$ meshes for $\sigma_{thr} = 0.5$ (7^{th} and 8^{th}) and $\sigma_{thr} = 1$ (7^{th}) and the last four adapted $h(p = 1)$ meshes (8^{th}, 9^{th}, 10^{th} and 11^{th}) in order to assess the effective convergence of the solution. Only this type of convergence study can be performed, given that no consistent numerical reference has been found in the literature for this test case.

For $h(p = 1)$ simulations, the position of the shock in the 8^{th} adapted mesh in solid red lines is already correctly captured, but the solution is still varying between the 8^{th} and the 9^{th} adaptation step in blue solid lines, especially in the locations closer to the root and the last location at $y/b = 0.97$. The 10^{th} adaptation in solid yellow lines and the

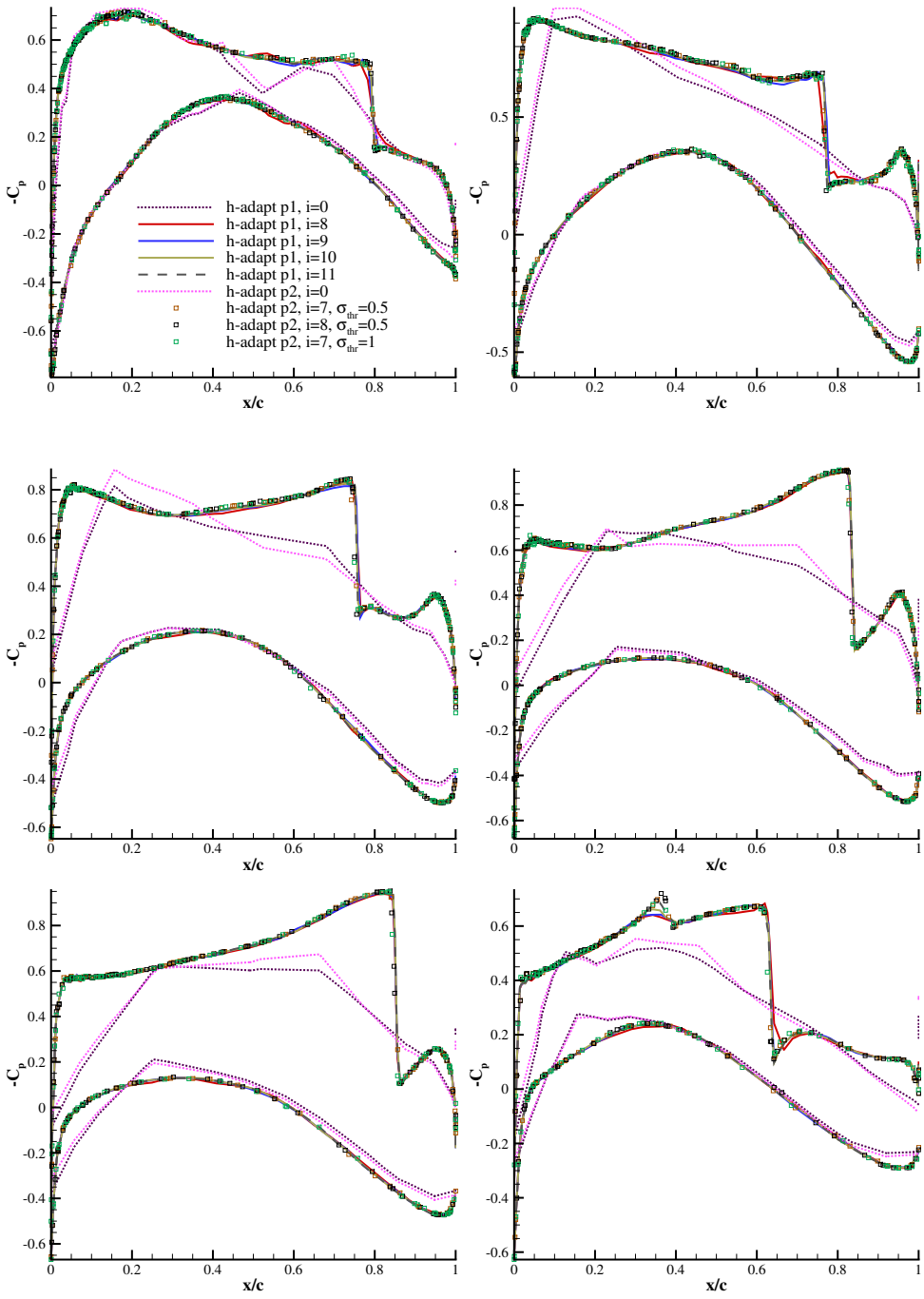


Figure 7.20: Inviscid flow past the CRM wing/body configuration at $M_\infty = 0.85$. Pressure coefficient profiles at $y/b = 0.125, 0.37, 0.5024, 0.7268, 0.8456, 0.97$ for $h(p = 1)$ and $h(p = 2)$ adaptations.

11^{th} adaptation in black dashed lines are superimposed almost everywhere, proving that our results are very close to mesh convergence. One exception is the secondary shock

at $y = 0.99$, where the finest mesh provides a sharper representation. This secondary shock structure is characterized by a forward-swept lambda shape in the wing tip region, visible in the right panel of figure 7.15 and in both panels of figure 7.22 presented in the following for $h(p = 2)$ adaptations. The solution on $h(p = 2)$ adapted meshes in squared symbols is very close to $h(p = 1)$ results, detecting the same position for the primary shock and the small secondary shock in the tip region.

We now focus on the influence of the smoothness indicator threshold σ_{thr} on the adaptation process. We recall that the smoothness indicator threshold is used to distinguish within the mesh adaptation process, the elements which are characterized by smooth solutions with a convergence rate $m = p + 1$, and those which are characterized by non-smooth solution and are assumed to present a reduced convergence rate $m = 1$.

The smoothness indicator is not defined for $p = 1$ discretizations, therefore we analyze $p = 2$ simulations.

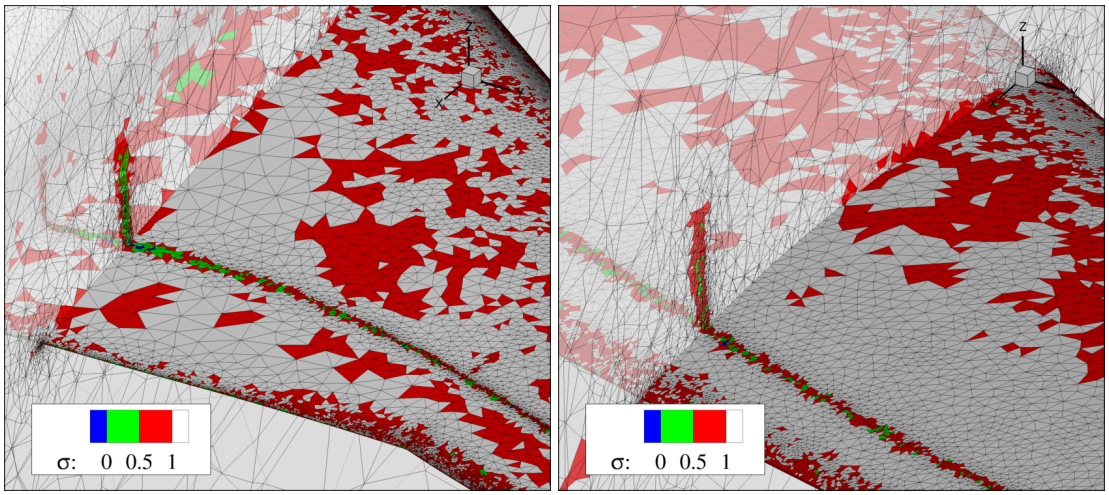


Figure 7.21: Inviscid flow past the CRM wing/body configuration at $M_\infty = 0.85$. Smoothness contour on the 8$^{\text{th}}$ $h(p = 2)$ adapted mesh with $\sigma_{thr} = 0.5$. Volume slices at $y/b = 0.15$ and $y/b = 0.78$.

Figure 7.21 illustrates the smoothness indicator distribution on the 8$^{\text{th}}$ adapted mesh using $\sigma_{thr} = 0.5$. The blue color marks elements with $\sigma_K < 0$, which constitute a very small percentage of the elements marked as non smooth. Elements in green present values of the smoothness indicator $0 < \sigma_K < 0.5$, while elements in red have $0.5 < \sigma_K < 1$. Grey elements are considered smooth for both values of σ_K, verifying $\sigma_K > 1$.

In our standard approach described in Chapter 5, the value $\sigma_{thr} = 1$ (as in the original study by Mavriplis [202]) is used to differentiate between smooth and non smooth elements. From figure 7.22 the threshold $\sigma_{thr} = 1$ appears however to be too high, unexpectedly marking as non smooth some elements which do not appear to be located in a problematic region of the flow. For this reason we have chosen to test the h-adaptation strategy by employing $\sigma_{thr} = 0.5$, as proposed by Leicht *et al.* [182] in their hp-adaptation of the DLR-F6 wing-body RANS configuration. The elements colored in green or blue, present almost exclusively in the shock, would be the only elements

marked as non smooth by the adaptation employing $\sigma_{thr} = 0.5$.

The meshes obtained after 7 adaptations using the standard $\sigma_{thr} = 1$ and the lower threshold $\sigma_{thr} = 0.5$ are shown in figure 7.22.

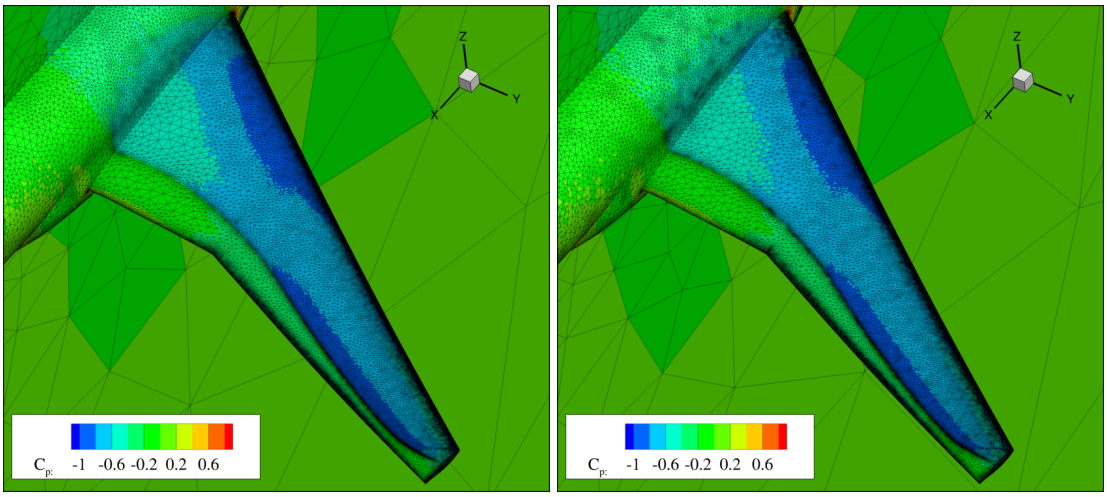


Figure 7.22: Inviscid flow past the CRM wing/body configuration at $M_\infty = 0.85$. Pressure coefficient contour on the 7^{th} $h(p = 2)$ adapted mesh using $\sigma_{thr} = 0.5$ and $\sigma_{thr} = 1$. Detail of the wing.

The use of different threshold values does not introduce significant differences in the adapted meshes. One minor difference that can be identified is in the size distribution outside the shocks (as for example in the fuselage), which in the case of $\sigma_{thr} = 1$ presents "spots" of refined zones. This is due to the interaction between a checkerboard-like pattern of the error estimator in the fuselage (visible also for the $h(p = 1)$ adaptation in figure 7.15), and values of smoothness indicator $\sigma < \sigma_{thr}$ which yield a stronger refinement. Besides, the mesh obtained using $\sigma_{thr} = 0.5$ presents a slightly coarser discretization of the shock next to the root with respect to the $\sigma_{thr} = 1$ mesh, which is however not visibly affecting the pressure coefficient profiles in figure 7.20.

The influence of σ_{thr} on the h-adaptive algorithm appears therefore to be limited. However, we point out that the use of different values might lead to more evident differences if used as hp-choice in the context of hp-adaptation. For such cases a more detailed study might be of interest.

7.4 Conclusion

We have implemented in our adaptation tool the possibility to reproject the surface mesh of the body onto the respective CAD employed to generate the initial mesh. This was one of the main bottlenecks of the adaptive algorithm towards its application for realistic aircraft configurations. The implementation of the CAD reprojection capabilities has allowed us to carry on the mesh adaptation procedure, starting from very coarse initial meshes. The algorithm has been capable to preserve the correct geometry representation over the course of the adaptation process, and avoid a progressive deterioration of the

geometry that could have occurred without the interaction of the adapted meshes with the CAD of the geometric representation of the body. Moreover, the algorithm has been able to gradually take into account the geometric details, initially hidden in the early coarse stages of the mesh adaptation process.

The error estimator has been modified for transonic simulations in order to improve the shock capturing feature of the error estimator, by taking into account the error on the pressure, in addition to the one computed on the momentum employed in previous chapters.

The adaptive algorithm has been tested on two test cases representative of Airbus aerodynamic design activities: the ONERA M6 wing and the CRM wing/body configuration. In this chapter the simulated flows have been solved with the Euler equations, due to the limited efficiency in handling high Reynolds viscous flows by the current isotropic adaptive algorithm. Despite the simplified configuration, as compared the RANS flows of more significant interest for industrial applications, these test cases have allowed us to both validate the reprojection algorithm and assess the performance of the error estimator.

The modified error estimator has been applied to second and third order DG simulations of the M6 wing. While providing results for third order simulations globally similar to the ones provided by the previous estimator, the new error estimator significantly improves the behavior of the $h(p = 1)$ adaptive process. In the case of $p = 1$ in fact, the addition of the contribution of the pressure clearly compensates for both the lower performance of the error estimator for low order discretizations, and the lack of smoothness indicator that would help to further reduce the sizes of under resolved elements in non-smooth zones. The performance of the h-adaptive algorithm has been assessed with respect to uniformly refined meshes, demonstrating a faster convergence of the integral quantities. The impact of the limiting of the minimum size h_{min} in the adaptation process has also been assessed for $h(p = 1)$ simulations.

Even though fully DG $p = 1$ simulations cannot be considered as high-order simulations, the development of well established error estimators and adaptation techniques also for $p = 1$ discretizations is important essentially for two reasons. First, industrial aerodynamic design activities still rely on second order schemes. Second, a correct error estimation for second order discretizations is essential for an efficient hp-adaptation for transonic flows which is a subject of future research.

The h-adaptive algorithm has been finally applied to the CRM wing/body configuration by performing DG $h(p = 1)$ and $h(p = 2)$ adaptive simulations. The adaptation has correctly identified the zones of most interest of the flow (shock, leading edge, trailing edge, trailing edge vortex). A global gain in terms of degrees of freedom of the h-adapted meshes with respect to second-order uniformly refined meshes, has shown the potential of mesh adaptation and high-order discretizations for the simulation of such flow configurations, over classical second-order approaches. The results obtained on the ONERA M6 wing and the CRM highlighted the relevance of the developed adaptive techniques for industrial aircraft computations. The development of two important features such as the handling of anisotropic elements and curved geometries will be mandatory in future work to fully exploit the potential of high-order adaptive techniques for steady RANS aircraft configurations.

Chapter 8

Conclusions and perspectives

8.1 Conclusions

The objectives of this PhD thesis have been the development of *hp*-adaptive techniques based on discontinuous Galerkin methods in the framework of the new generation flow solver CODA (CFD ONERA DLR Airbus), and their application to test cases relevant in an industrial aeronautical context. A particular interest towards hybrid RANS/LES simulations for acoustic applications by employing adaptive DG methods, has motivated the largest part of this work. This research program supports the initiative to develop an automatic mesh-solver process for aerodynamic design activities.

After a brief introduction of the flow equations, the most common approaches for scale resolving simulations have been presented in Chapter 2, with a focus on classical Detached Eddy Simulation and Zonal Detached Eddy Simulation, using the Spalart-Allmaras model. The modal high-order discontinuous Galerkin discretization of the Navier-Stokes equations employed in this work has been presented as well in Chapter 2.

An overview of resolution adaptation techniques has been then presented in Chapter 3, highlighting the different approaches in the literature to perfom *h*-, *p*- and *hp*-adaptation, focusing on *a posteriori* error estimators relevant in the approach employed in this work. This class of error estimators, directly computed exploiting information only from the DG solution, constitutes an efficient, local, simple and low-computational cost choice.

In Chapter 4, devoted to the validation and the assessment of DG techniques for unsteady and turbulent applications, the order of convergence of the flow solver CODA has been first validated for an unsteady test case with an analytical solution to assess the numerical errors of the DG scheme using unstructured simplicial meshes. Preliminary analyses on the influence of the y^+ value in RANS boundary layers, and on the C_{DES} in homogeneous turbulence, have also been carried out. These analyses have illustrated that the use of high-order polynomials is beneficial in the sense that it both relaxes the near-wall resolution requirements, and accurately represents the turbulent structures in fully-developed turbulent regions.

In Chapter 5, *h*- and *hp*-adaptive strategies suitable to discontinuous Galerkin methods have been developed for solving steady fluid flow problems on unstructured simplicial meshes. An *a posteriori* error estimator based on both the measure of the energy

contained in the highest order polynomial modes and the jumps at the element interfaces has been assessed in the framework of isotropic metric-based *h*- and *hp*-adaptation. These methodologies have been combined with a smoothness indicator guiding both the choice between *h*- and *p*-adaptation and the reduction of the size of elements marked for *h*-adaptation. Adaptive DG computations of three configurations of 2D/3D laminar steady flows on triangular/tetrahedral meshes have been performed based on polynomial degrees $p = 1$, $p = 2$ and $p = 3$. Overall, the adaptation strategies have been found to capture accurately the zones in which the solution needs higher refinement and zones already well refined. This has yielded a significant reduction in terms of number of degrees of freedom to reach a given error level, as compared to simulations with uniform mesh refinement. One limitation of the *hp*-choice employed in this *hp*-adaptive method, is the inability to compute the smoothness indicator for low order elements, which, if marked for refinement, are always *p*-enriched. This can be acceptable for intrinsically smooth subsonic flows, while it might be not suitable for transonic flows. In this case the use of high-order discretizations in shocks would increase the number of dofs without increasing the local order of convergence, and at the same time it would yield stability issues, which are of increasing severity for increasing polynomial degrees.

The proposed *hp*-adaptive algorithm has been then extended to hybrid meshes, as anisotropic structured or pseudo-structured mesh elements are necessary to capture adequately boundary layer dynamics in RANS and LES simulations. For this reason we have introduced in our simulations a fixed prismatic near-wall mesh for capturing the boundary layer, whose elements cannot be remeshed but only *p*-enriched. The performance of the *hp*-algorithm for hybrid meshes has been demonstrated in the context of turbulent jet 3D RANS simulations, on the PPRIME nozzle test case. The DG *hp*-adapted numerical solution has provided converged results in close agreement with a simulation on a structured mesh using classical FV schemes with a much larger number of dofs. The considered adaptive strategy represents therefore a suitable approach for similar configurations, in which the most important features in the boundary layer can be effectively resolved by applying only *p*-refinement on the initial frozen boundary layer mesh. However, this might not be the case for other types of applications, such as transonic aircraft configurations. For such applications, for example, the shock should be *h*-refined also on the surface and in the boundary layer, and not only in the isotropic fully tetrahedral region above the prismatic boundary layer.

In Chapter 6 the adaptive algorithms has been extended as well to unsteady flows, with the final aim of performing hybrid RANS/LES simulations of the jet issuing from the PPRIME nozzle.

The same error estimator based on both the measure of the energy contained in the highest order polynomial modes and the jumps of the solution at the elements interfaces, has been employed for the simulation of scale-resolving flows, with a slightly modified smoothness indicator for scale-resolving simulations, based on the decay of the polynomial modes. The strategy for each adaptation step has consisted in first simulating a transient phase required for the solution to adapt to the new discretization, then collecting and averaging the error estimator and the smoothness indicator for a given characteristic period of the flow, which is a fraction of the time necessary for the first and second order flow statistics to converge. After these steps, specific to the algorithm

for unsteady flows, the same adaptive strategy adopted for steady flows has been applied by using the time-averaged error estimator and smoothness indicator for the prescription of the new sizes and polynomial degrees.

The unsteady adaptive strategies have been first assessed on the transport of a vortex by uniform flow, proving a gain in terms of number of dofs of the adaptive strategies against uniform refinement. This has proven as well that a static adaptation approach can be efficiently applied when the technological tools are not yet adapted for dynamic approaches, also on such spatially evolving test cases. The *hp*-adaptation unsteady procedure has been then applied to a hybrid RANS/LES configuration of a sphere at $Re = 3700$ using a DES approach. The error estimator has been found to correctly identify the flow regions of interest in a hybrid RANS/LES context, and the *hp*-adapted meshes have provided results in line with the references in the literature using similar subgrid turbulence models, with a decreased number of dofs.

In the last section of Chapter 6, the *hp*-adaptive algorithm has been applied to the PPRIME nozzle test case at $Re_D = 10^6$ in a hybrid RANS/LES configuration. ZDES mode 1 combined to mode 0 has been used to separate the interior part of the nozzle, modeled using RANS equations, and the free jet area, modeled using modified DES equations. The unsteady hybrid RANS/LES adaptation has been here initialized from an initial *hp*-mesh provided by a series of steady RANS adaptations. In fact starting from a very coarse initial mesh would have here dramatically increased the computational time for the whole adaptation process, as many unsteady adaptation steps would have been needed to reach accurate results with properly developed turbulent structures. The flow solver CODA has been then interfaced with the acoustic solver KIM developed at ONERA, in order to perform Ffowcs Williams-Hawking aeroacoustic far-field computations on the adapted meshes. The impact of increased resolution has been analyzed for three adapted meshes presenting respectively 11.6, 18.9, 31.7 million dofs. Quantitative fluid flow results have compared fairly well to numerical references, obtaining overall close results to classical FV schemes on structured meshes and non-adaptive DG methods, with a reduced number of degrees of freedom. Within an uncertainty range due to the short simulation time, the adapted simulation shows a very good agreement with reference acoustic spectra and OASPL in the far-field. However, some questions remain open on the impact of turbulent injection and strong mesh refinement on the statistics of the jets. In fact, although results have compared well with numerical data, it appears that refining the mesh could lead to longer potential cores and lower levels of velocity fluctuations in the potential core with respect to experimental measurements.

These results have shown nonetheless the potential of high-order DG *hp*-adaptation in performing scale resolving simulations of turbulent flows, for both aerodynamic and aeroacoustic prediction purposes. They have provided a motivation to continue exploring LES models for high-order methods and turbulence generation mechanisms to provide results sufficiently in agreement with experiments in the simulation of jets.

In the last Chapter 7, concerning the work mostly performed in the last period of the thesis carried out in Airbus, the main objective has been the demonstration of the complete adaptive chain on transonic wall-bounded flows past complex geometries. For this purpose, we have implemented in our adaptation tool the possibility to reproject the surface mesh of the body onto the respective CAD employed to generate the initial mesh,

by exploiting external and FlowSimulator libraries. We focused in this chapter on the inclusion of shock capturing in the adaptive procedure, by modifying the error estimator to take into account also the pressure as a representative variable on which computing the error estimator in each element. As previously mentioned, the current strategy, relying on isotropic error estimators and size definition approach, is not yet well adapted for the efficient adaptation of the boundary layers of high Reynolds numbers flows. For this reason, we have focused in this chapter only on inviscid flow conditions. Despite the simplified configurations employed, these test cases have still allowed the validation of the reprojection algorithm and the assessment of the error estimator behavior.

Second order DG $h(p = 1)$ and third order DG $h(p = 2)$ adaptive computations have been performed on the ONERA M6 wing and on the CRM wing/body test cases. The estimator has been found to correctly identify the critical zones of the transonic flow past wings and full aircraft configurations, such as shocks in the first instance, but also the leading edge, the trailing edge and the trailing vortex region.

The C_D and C_L resulting from the h-adaptive procedures have demonstrated a faster convergence rate as compared to second-order uniformly refined discretizations, leading to a consistent reduction of dofs with respect to the classical refinement approach. The suitability of the adopted artificial viscosity approach for the stabilization of DG schemes has been as well verified in the framework of aircraft configurations. Inviscid simulations have provided very sharp shocks, not deteriorated by the addition of the artificial viscosity, which has been observed to activate at the expected locations. This study has been intended both to directly demonstrate in the context of CODA a preliminary metric-based adaptive capability for Airbus applications, and to serve as a validation basis of the numerical and technological ingredients for future high-order h- and hp-adaptation to transonic aircraft configurations. The current restriction to isotropic elements is suboptimal with respect to anisotropic mesh adaptation for transonic viscous flows. In particular, the lack of anisotropic features represents a significant limitation to capturing efficiently the boundary layers, as well as testing efficiently the adaptive procedure on realistic test cases.

We believe that the present work significantly contributes towards the application of high-order metric-based spatial resolution adaptation to industrial configurations. The developed strategy and the obtained results provide the fundamental building blocks for future research and developments. The full integration of such adaptive techniques in the industrial aerodynamic design process will however require a number of further improvements, which are outlined in the next section.

8.2 Perspectives

We list here some possible paths to follow in future work to improve the developed adaptive algorithms.

- *Anisotropic mesh adaptation for high order DG methods.* The development of an anisotropic h-adaptive method, aimed at adapting the mesh following the directional features of the flow, would significantly improve the adaptation procedure. Transonic simulations would benefit from the discretization of the shocks with

anisotropic elements, allowing for a consistent reduction of the number of degrees of freedom to represent anisotropic flow features with respect to the use of isotropic elements. Moreover an automatic anisotropic meshing of the turbulent boundary layers would be very beneficial as well for both RANS and scale-resolving simulations. Different strategies can be targeted for handling boundary layers. One possibility would consist in employing a fully tetrahedral mesh, where the boundary layers would be discretized with highly anisotropic tetrahedra. A second strategy would rely on pseudo-structured prismatic boundary layers with automatic adaptation of the surface and of the height of the boundary layer. The choice between the two strategies would strictly depend on the available tools (mainly the remesher in the fully tetrahedral case and the extruder in the pseudo-structured boundary layer approach) and the accuracy of the employed numerical methods on the two types of boundary layer meshes.

- *Load balancing techniques.* A robust load-balancing strategy should be developed in order to make *hp*-adaptive techniques competitive from a CPU time point of view. The results reported in this work show the potential of *hp*-methods with respect to uniformly refined meshes and *h*-adaptive techniques, in terms of number of dofs. The parallel implementation employed for computations in this work has consisted in the partitioning of the mesh according to a criterion based on the number of elements, and not of the number of dofs. This means that, without a robust load-balanced implementation of the *hp*-DG solver, simulations performed on *hp*-adaptive meshes might have a comparable cost to a simulation performed on the same *h*-mesh, but discretized with the uniform maximum polynomial degree. If dynamic *h*- and *hp*-adaptation are targeted, these load balancing techniques should be able to efficiently balance the work load at run time.

- *Dynamic unsteady h- and hp-adaptation*. Although static unsteady adaptation has provided satisfying results on a statistically steady flow configuration such the studied subsonic jet, a dynamic unsteady adaptation procedure would be necessary for time evolving flows without statistically steady or periodic features. This would require the implementation of a dynamic load balancing technique for the *p*-adaptation part, as well as efficient run time mesh redistribution. More accurate techniques of reprojection of the solution from a mesh to the new finer one should also be implemented. In fact in dynamic mesh adaptation, the reprojected solution should be immediately exploitable on the subsequent time step. The reprojection strategy should therefore be conservative and avoid the introduction of spurious oscillations. However, the essential feature for the efficiency of dynamic unsteady *h*- and *hp*-adaptation is the implementation of the adaptive process directly as part of the flow solver rather than in an external, file based tool. This is the only way to avoid expensive exchanges of information during the simulation, which should be fully coupled to the adaptation process.

- *Adaptations with coarsening.* One possible drawback of the currently implemented adaptive algorithm is the lack of a coarsening mechanism. This is justified by the fact that in the present work, very coarse meshes have been employed as initial

meshes, and the algorithm aims at increasing the number of dofs at each adaptation step. It means that even though in early stages of adaptation the error estimator might mark for adaptation some cells in zones presenting high values of the error in these intermediate meshes providing not converged solutions, this would not lead to extremely overrefined zones, given the initially very coarse meshes. However, if adaptation is to be used on arbitrarily refined initial meshes for the optimization of the distribution of the dofs in the mesh, without having their number increased at each adaptation, coarsening would be mandatory. Fixed-point iterations would assure the "optimality" of the distribution of the dofs, overcoming a possible transient behavior of the error estimator, however at the expense of a higher number of adaptation steps.

- *Extension to high-order elements.* The exploitation of the full potential of high-order adaptive methods can be possible if the adaptive algorithms are extended to high-order elements. This would allow the discretization of the surfaces with a lower number of elements, while keeping a high order of accuracy on the correct geometry. This would lead to a faster convergence of the adaptive algorithms, which are now constrained by the presence of a large number of boundary elements in order to correctly represent the geometry.

- *Mesh convergence studies, subgrid models and turbulence injection for turbulent jets at high Reynolds numbers.* In future work, turbulent injection/tripping techniques should be considered, in order to rigorously assess the impact of the developed turbulent structures in the interior of the nozzle, on both flow field and far-field acoustic results. Moreover, a deeper insight on the dependence of jet flow statistics on the mesh resolution and on subgrid turbulence models should be performed in order to finally clarify the recurrent issue in the literature concerning the solution on refined meshes diverging from experimental results.

- *hp-adaptation for transonic flows*. The interest of high-order DG methods in transonic simulations should be further explored by implementing *hp*-adaptive strategies well suited for transonic simulations. The shock should be discretized with low order elements (preferably with polynomial degrees not higher than $p = 1$), while the current smoothness indicator is not implemented for $p = 0$ or $p = 1$ elements. A possible path could be the use of the Persson-Peraire indicator used for the computation of the artificial viscosity, imposing a threshold value between *h*- and *p*-adaptation consistent with the value used in the artificial viscosity definition to mark elements needing additional dissipation. Even though the aim of such an *hp*-adaptation strategy should be preserving a low-order discretization in shocks, we cannot completely exclude some high-order elements to be present in shocks. If employing very high-order polynomials, the flow solver should be equipped as well with more robust stabilization/shock capturing techniques for high-order simulations, other than the Persson-Peraire artificial viscosity, such as the entropy residual based method by Guermond *et al.* [147]. Moreover, the reduction of the local order of accuracy is known as being an important limitation of DG methods in the simulation of flows featuring strong discontinuities, such as in transonic industrial

aeronautical configurations. The development of an effective *hp*-strategy would aid the investigation of the global convergence behavior of high-order adaptive DG methods in the presence of shocks. A very careful distribution of h and p over the computational domain, lowering p and increasing h near discontinuities, and maximizing p in smooth regions, would allow us to benefit the most from high order methods applied to complex transonic configurations.

Appendix A

Validation of the acoustic analysis

A.1 Influence of the averaging procedure

The PSD spectra of the acoustic signals presented in section 6.4 have been estimated using the Welch method [304]. The Welch procedure is carried out by dividing the time signal into successive blocks, computing the periodogram for each block using a windowing sequence, and then averaging the PSD estimates. Each block can overlap the adjacent block by a certain factor. This method allows for reducing the typical oscillations encountered when evaluating the raw spectrum of a given signal using the classical Discrete Fourier Transform (DFT) and Fast Fourier Transform (FFT) methods. However, the signal size must be sufficiently long for the block decomposition to be efficient. In our case, the signal length ($130t_c$) is relatively short, but obtaining clean spectra is important to be able to compare to the other data sets. We apply a Hann window to each block composed of N=1000 sampled data, with an overlap of 75%. This leads to 11 averaging blocks. In order to validate the PSD levels provided by the employed settings, the spectra for the fine *hp*G3 mesh in black in figure 6.28 are compared with the spectra obtained by using only one block of data, and with those obtained by varying the overlap factor and the number of sampled data N for each averaging block.

In figure A.1, the spectra obtained using one averaging block are depicted in orange, while those employing segments of data samples of length N=1000 and a 75% overlapping (and 11 averaging blocks) are depicted in black. In magenta lines we report the spectra obtained by increasing the number of sampling data for each block to N=2000 while fixing the overlapping to 75% (with 4 averaging blocks), and in cyan lines the results obtained by decreasing the overlap to 50% with N=1000 (and 6 averaging blocks). The same colors are employed for the OASPL levels in figure A.4. The PSD spectra obtained by employing the three different settings for the averaging blocks appear to be correctly averaging the spectra computed over the entire length of the signal. The spectra obtained using the three sets of parameters in black, magenta and cyan lines, are almost completely superposed. However, it is the integrated noise levels that can confirm the insensitivity of the computed noise levels to the employed averaging procedure.

In the OASPL plot in figure A.4, the levels obtained averaging over different numbers of blocks of sample data and those obtained employing only one block of data are almost indistinguishable, indicating that the chosen set of parameters is reliable for the acoustic

analysis.

We remark that the spectra presented here have always been spatially averaged on the 36 azimuthal microphones for each angle of observation. Without this spatial averaging procedure, the spectra would have displayed strong oscillations, complicating comparisons between the different curves.

A.2 Sensitivity study on the FW-H surfaces

In this section we study the influence of the FW-H surface employed for the acoustic post processing, on the PSD spectra and, consequently, on the integrated OASPL levels obtained from the computation on the finest hpG3 adapted mesh.

We remind that the instantaneous flow field has been extracted at each sampling time step on four different surfaces of revolution (S1, S2, S3, S4) in figure 6.26, which differ by their radial distance.

At first, a comparison is performed by setting the closing surface to D3 (the farthest), and varying the surface of revolution S1, S2, S3, S4. For both the cylindrical and the polar array, the spectra obtained for different angles of observation and the OASPL in [dB] computed in the frequency range $0.05 < St < 3$ are shown respectively in figure A.2 and A.5.

Looking at the spectra, we observe that the results obtained using the four surfaces of revolution are almost indistinguishable at the low/medium frequency range, which has a strong impact on the OASPL values. The curves start to differ at a Strouhal number between 1 and 2, depending on the observation angle. This is an expected behavior, due to the fact that the farther the surface from the jet region, the lower the grid resolution, and the lower the cut-off Strouhal number. This means that short-wavelength acoustic waves are not well resolved near the farther surfaces, leading to a rapid drop of the PSD levels.

As can be noticed from the OASPL plots in figure A.5, this affects the overall computed noise only to a minor extent. Some small differences can be seen especially for high angles of observation $\theta > 50°$, while low angles present very similar results. In fact sound levels at low angles of observation are mostly affected by the more energetic vortices at the end of the potential core, which are essentially captured in the same way by the four surfaces. On the other hand, the smaller turbulent structures in the jet shear layer, closer to the nozzle exit, which contribute the most to the noise at high angles of observation, might dissipate while reaching the farthest surface S4 and yield lower noise levels.

As the results from S1 and S2 are very close to each other, we can assume the two surfaces as reliable enough to predict the correct noise levels, and we have decided to employ the S1 surface.

The second part of the sensitivity study consists in a the comparison of the results obtained by varying the closing surface while keeping fixed the surface of revolution. For this purpose, S1 is kept fixed and the results obtained using D1, D2, D3 are shown in figures A.3 and A.6, respectively showing the spectra at different angles of observation and the corresponding OASPL levels.

Despite very similar results observed for the spectra displayed in figure A.3, figure A.6 displays the highest levels of the OASPL for every angle of observation, when employing the D1 surface, especially for $\theta < 40°$. The computations performed with D2 and D3 provide indistinguishable noise levels from $\theta = 90°$ up to $\theta \sim 30° - 25°$. At the lowest angle $\theta = 20°$, D2 provides a higher noise level as compared to D3.

This behavior has been observed by Rahier *et al.* [238] using the same FW-H formulation with the additional flux terms employed in this work, employing different closing surfaces. They motivated the discrepancies observed for low angles of observation with the wave refraction due to the velocity gradients in the jet, which leads to a cone of silence around the jet axis and noise reductions on the jet axis. They observed as well that the shorter the control surface the less these refraction effects are taken into account, and highlighted the need of control surfaces long enough to compute the jet acoustic radiation at observer points located near the jet axis. Following these observations, and the typical length of the control surfaces which range from $x/D = 25$ to $x/D = 30$ [63, 205, 134], we chose to employ the farthest D3 surface for the comparisons between the three *hp*-adapted meshes presented in section 6.4.

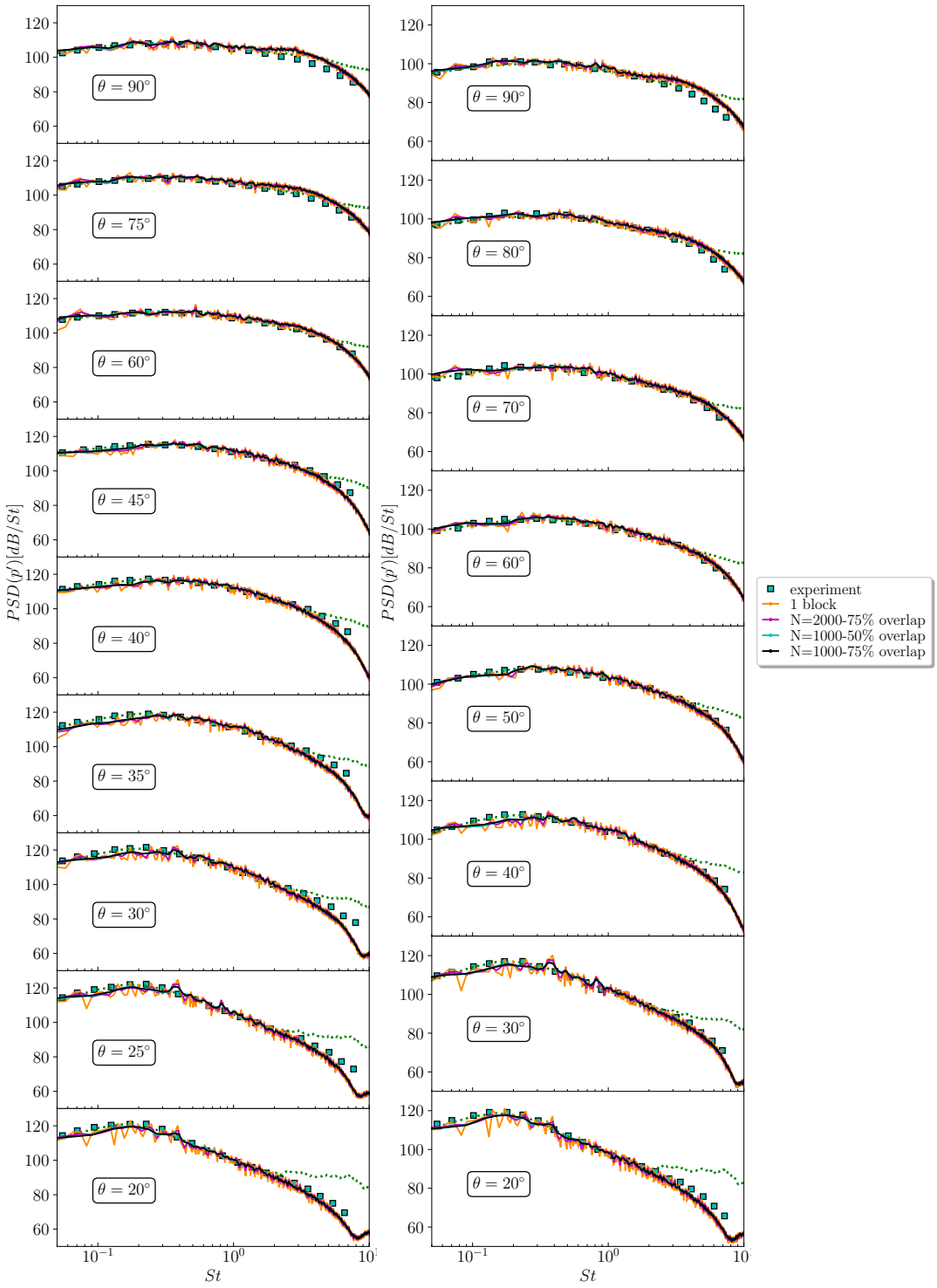


Figure A.1: hpDG/ZDES simulations of the PPRIME nozzle. PSD of the pressure on the cylindrical microphone array of radius r = 14.3D (left) and on the polar microphone array at 50D from the nozzle exit (right) for different angles of observation. Different parameters for the Welch PSD estimation.

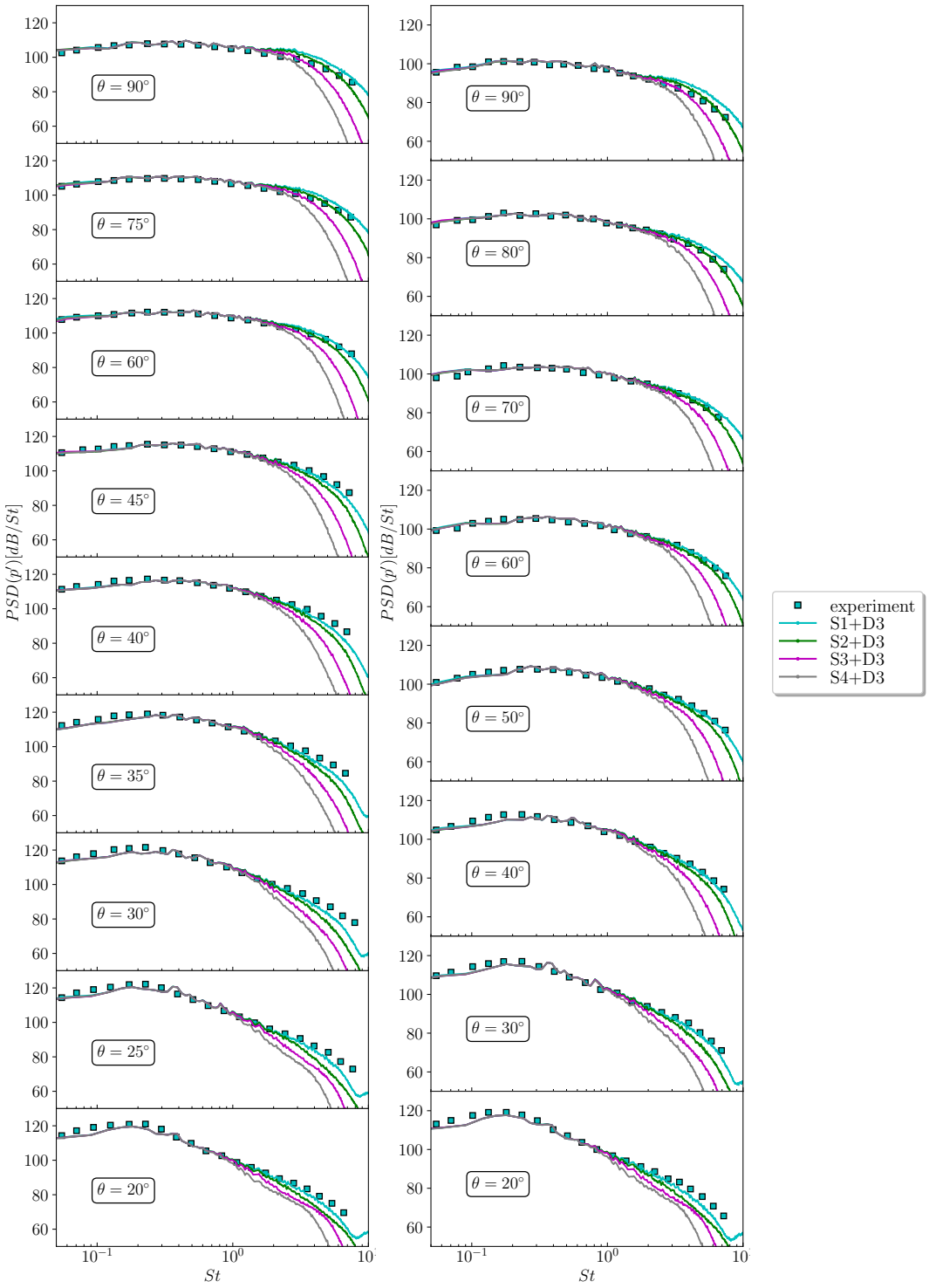


Figure A.2: hpDG/ZDES simulations of the PPRIME nozzle. PSD of the pressure on the cylindrical microphone array of radius r = 14.3D (left) and on the polar microphone array at 50D from the nozzle exit (right) for different angles of observation. Different FW-H surfaces of revolution.

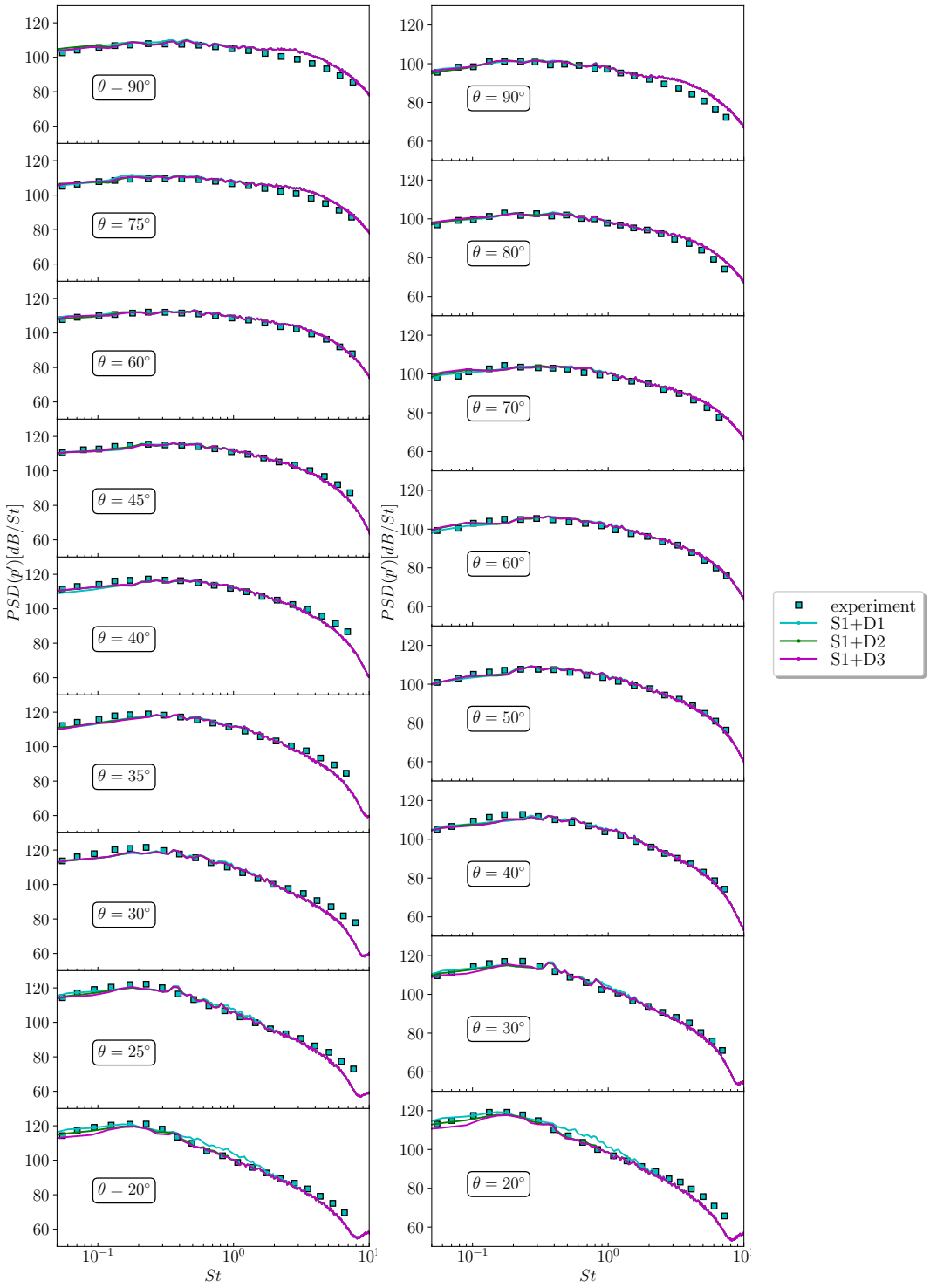


Figure A.3: hpDG/ZDES simulations of the PPRIME nozzle. PSD of the pressure on the cylindrical microphone array of radius r = 14.3D (left) and on the polar microphone array at 50D from the nozzle exit (right) for different angles of observation. Different closing FW-H surfaces.

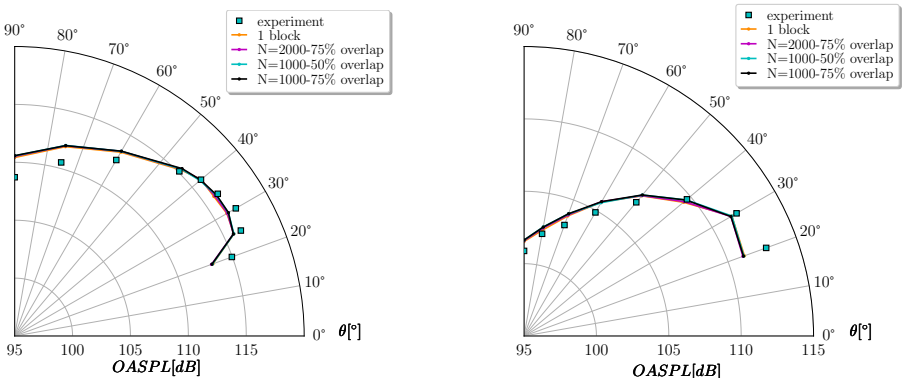


Figure A.4: hpDG/ZDES simulations of the PPRIME nozzle. OASPL on the cylindrical microphone array of radius r = 14.3D (left) and on the polar microphone array at 50D from the nozzle exit (right) for different angles of observation using different number of overlapping blocks for the Welch PSD estimation and different parameters for the Welch PSD estimation.

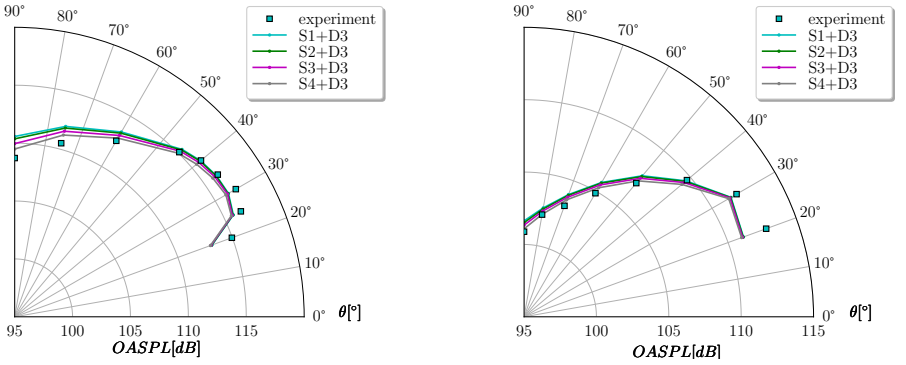


Figure A.5: hpDG/ZDES simulations of the PPRIME nozzle. OASPL on the cylindrical microphone array of radius r = 14.3D (left) and on the polar microphone array at 50D from the nozzle exit (right) for different angles of observation using different FW-H surfaces of revolution.

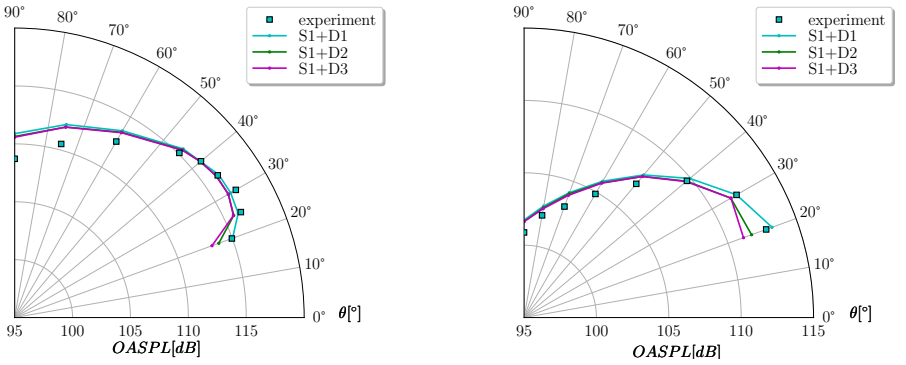


Figure A.6: hpDG/ZDES simulations of the PPRIME nozzle. OASPL on the cylindrical microphone array of radius r = 14.3D (left) and on the polar microphone array at 50D from the nozzle exit (right) for different angles of observation using different closing FW-H surfaces.

References

[1] 2DN00: 2D NACA 0012 Airfoil Validation Case. https://turbmodels.larc.nasa.gov/naca0012_val.html.

[2] Acoustic Reference Nozzle with Mach 0.97, Unheated Jet Flow. https://www.grc.nasa.gov/www/wind/valid/arn/index.html.

[3] ANSA. The advanced CAE pre-processing software for complete model build up. https://www.beta-cae.com/ansa.html.

[4] Drag Prediction Workshop - NASA. https://https://aiaa-dpw.larc.nasa.gov/.

[5] International Workshop on High-Order CFD Methods. https://www1.grc.nasa.gov/research-and-engineering/hiocfd/.

[6] MMG Platform - Upgrade your meshes. https://www.mmgtools.org.

[7] NASA Common Research Model. https://commonresearchmodel.larc.nasa.gov/.

[8] ONERA M6 Wing. https://www.grc.nasa.gov/www/wind/valid/m6wing/m6wing.html.

[9] ONERA M6 Wing. https://turbmodels.larc.nasa.gov/onerawingnumerics_val.html.

[10] OpenCascade. Backing your path to digital future. https://www.opencascade.com/.

[11] pythonOCC. https://dev.opencascade.org/project/pythonocc.

[12] VI1 Vortex transport by uniform flow. https://how5.cenaero.be/content/vi1-vortex-transport-uniform-flow.

[13] 2D Zero Pressure Gradient Flat Plate Verification Case. https://turbmodels.larc.nasa.gov/flatplate.html, Accessed: 2021-10-15.

[14] A Abbà, A Recanati, M Tugnoli, and L Bonaventura. Dynamical p- adaptivity for LES of compressible flows in a high order DG framework. *J Comput Phys*, 420:109720, 2020.

[15] S Adjerid, M Aiffa, and JE Flaherty. Computational methods for singularly perturbed systems. In *Proceedings of Symposia in Applied Mathematics*, volume 56, pages 47–83. AMS, 1998.

[16] PW Agostinelli, B Rochette, D Laera, J Dombard, B Cuenot, and L Gicquel. Static mesh adaptation for reliable large eddy simulation of turbulent reacting flows. *Phys Fluids*, 33(3):035141, 2021.

[17] F Alauzet and L Frazza. 3D RANS anisotropic mesh adaptation on the high-lift version of NASA's Common Research Model (HL-CRM). In *AIAA Aviation 2019 Forum*, page 2947, 2019.

[18] F Alauzet, X Li, ES Seol, and MS Shephard. Parallel anisotropic 3D mesh adaptation by mesh modification. *Eng Comput*, 21(3):247–258, 2006.

[19] F Alauzet and A Loseille. Metrix user guide. error estimates and mesh control for anisotropic mesh adaptation. Technical report, INRIA, Rocquencourt, 2009. https://hal.inria.fr/inria-00363007.

[20] F Alauzet and A Loseille. A decade of progress on anisotropic mesh adaptation for computational fluid dynamics. *Comput Aided Des*, 72:13–39, 2016.

[21] F Alauzet, A Loseille, and G Olivier. Time-accurate multi-scale anisotropic mesh adaptation for unsteady flows in CFD. *J Comput Phys*, 373:28–63, 2018.

[22] A Alla, Z Mghazli, M Fortin, and F Hecht. R-adaptation par l'estimateur d'erreur hiérarchique. 2006.

[23] SR Allmaras and FT Johnson. Modifications and clarifications for the implementation of the Spalart-Allmaras turbulence model. In *Seventh International Conference on Computational Fluid Dynamics (ICCFD7), 9-13 July 2012, Big Island, Hawaii*, 2012. https://www.iccfd.org/iccfd7/assets/pdf/papers/ICCFD7-1902_paper.pdf.

[24] O Amoignon, J Hradil, and J Navratil. Study of parameterizations in the project CEDESA. In *52nd Aerospace Sciences Meeting*, page 0570, 2014.

[25] N Andersson, L-E Eriksson, and L Davidson. Large-eddy simulation of subsonic turbulent jets and their radiated sound. *AIAA J*, 43(9):1899–1912, 2005.

[26] O Antepara, O Lehmkuhl, R Borrell, J Chiva, and A Oliva. Parallel adaptive mesh refinement for large-eddy simulations of turbulent flows. *Comput Fluids*, 110:48–61, 2015.

[27] AF Antoniadis, P Tsoutsanis, and D Drikakis. Assessment of high-order finite volume methods on unstructured meshes for RANS solutions of aeronautical configurations. *Comput Fluids*, 146:86–104, 2017.

[28] DN Arnold, A Mukherjee, and L Pouly. Locally adapted tetrahedral meshes using bisection. *SIAM J Sci Comput*, 22(2):431–448, 2000.

[29] C Bailly, C Bogey, and O Marsden. Progress in direct noise computation. *Int J Aeroacoustics*, 9(1-2):123–143, 2010.

[30] A Balan, MA Park, WK Anderson, DS Kamenetskiy, JA Krakos, T Michal, and F Alauzet. Verification of anisotropic mesh adaptation for turbulent simulations over ONERA M6 wing. *AIAA J*, 58(4):1550–1565, 2020.

[31] A Balan, MA Park, SL Wood, WK Anderson, and K Jacobson. Angle-of-Attack Sweep with Mesh Adaptation for High-Lift Configurations. In *AIAA Scitech 2022 Forum*, page 0218, 2022.

[32] A Balan, MA Park, SL Wood, WK Anderson, A Rangarajan, DP Sanjaya, and G May. A review and comparison of error estimators for anisotropic mesh adaptation for flow simulations. *Comput Fluids*, page 105259, 2021.

[33] A Balan, M Woopen, and G May. Adjoint-Based hp-Adaptation for a Class of High-Order Hybridized Finite Element Schemes for Compressible Flows. In *21st AIAA Computational Fluid Dynamics Conference, 24-27 June 2013, San Diego, CA, USA*, 2013. AIAA Paper 2013-2938.

[34] A Balan, M Woopen, and G May. Adjoint-based hp-adaptivity on anisotropic meshes for high-order compressible flow simulations. *Comput Fluids*, 139:47–67, 2016.

[35] N Barral, G Olivier, and F Alauzet. Time-accurate anisotropic mesh adaptation for three-dimensional time-dependent problems with body-fitted moving geometries. *J Comput Phys*, 331:157–187, 2017.

[36] F Basile, J-B Chapelier, M de la Llave Plata, R Laraufie, and P Frey. A high-order h-adaptive discontinuous galerkin method for unstructured grids based on a posteriori error estimation. In *AIAA Scitech 2021 Forum, 11–15 & 19–21 January 2021, VIRTUAL EVENT*, 2021. AIAA Paper 2021-1696.

[37] F Basile, J B Chapelier, R Laraufie, and R Frey. hp-adaptive hybrid RANS/LES simulations for unstructured meshes with the discontinuous Galerkin method. In *AIAA SciTech 2022 Forum*, page 1207, 2022.

[38] F Bassi, L Botti, A Colombo, A Crivellini, M Franciolini, A Ghidoni, and G Noventa. A p-adaptive matrix-free discontinuous Galerkin method for the implicit LES of incompressible transitional flows. *Flow Turbul Combust*, 105(2):437–470, 2020.

[39] F Bassi, L Botti, A Colombo, DA Di Pietro, and P Tesini. On the flexibility of agglomeration based physical space discontinuous Galerkin discretizations. *J Comput Phys*, 231(1):45–65, 2012.

[40] F Bassi, A Colombo, A Crivellini, KJ Fidkowski, M Franciolini, A Ghidoni, and G Noventa. Entropy-adjoint p-adaptive discontinuous Galerkin method for the under-resolved simulation of turbulent flows. *AIAA J*, 58(9):3963–3977, 2020.

[41] F Bassi and S Rebay. A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier–Stokes equations. *J Comput Phys*, 131(2):267–279, 1997.

[42] Y Bazilevs, J Yan, M De Stadler, and S Sarkar. Computation of the flow over a sphere at Re= 3700: A comparison of uniform and turbulent inflow conditions. *J Appl Mech*, 81(12):121003, 2014.

[43] AD Beck, T Bolemann, D Flad, H Frank, GJ Gassner, F Hindenlang, and C-D Munz. High-order discontinuous Galerkin spectral element methods for transitional and turbulent flow simulations. *Int J Numer Methods Fluids*, 76(8):522–548, 2014.

[44] A Belme, A Dervieux, and F Alauzet. Time accurate anisotropic goal-oriented mesh adaptation for unsteady flows. *J Comput Phys*, 231(19):6323–6348, 2012.

[45] P Benard, G Balarac, V Moureau, C Dobrzynski, G Lartigue, and Y D'Angelo. Mesh adaptation for large-eddy simulations in complex geometries. *Int J Numer Methods Fluids*, 81(12):719–740, 2016.

[46] M Berggren, S-E Ekström, and J Nordström. A discontinuous Galerkin extension of the vertex-centered edge-based finite volume method. *Commun Comput Phys*, 5:456–468, 2009.

[47] P-E Bernard, N Chevaugeon, V Legat, E Deleersnijder, and J-F Remacle. High-order h-adaptive discontinuous Galerkin methods for ocean modelling. *Ocean Dyn*, 57(2):109–121, 2007.
[48] DJ Bodony and SK Lele. Large eddy simulation of turbulent jets and progress towards a subgrid scale turbulence model. In *International Workshop on LES for Acoustics*, pages 7–8, 2002.
[49] DJ Bodony and SK Lele. On using large-eddy simulation for the prediction of noise from cold and heated turbulent jets. *Phys Fluids*, 17(8):085103, 2005.
[50] DJ Bodony and SK Lele. Current status of jet noise predictions using large-eddy simulation. *AIAA J*, 46(2):364–380, 2008.
[51] C Bogey. Grid sensitivity of flow field and noise of high-Reynolds-number jets computed by large-eddy simulation. *Int J Aeroacoustics*, 17(4-5):399–424, 2018.
[52] C Bogey and C Bailly. Effects of inflow conditions and forcing on subsonic jet flows and noise. *AIAA J*, 43(5):1000–1007, 2005.
[53] C Bogey and C Bailly. Influence of nozzle-exit boundary-layer conditions on the flow and acoustic fields of initially laminar jets. *J Fluid Mech*, 663:507–538, 2010.
[54] C Bogey, C Bailly, and D Juvé. Computation of flow noise using source terms in linearized Euler's equations. *AIAA J*, 40(2):235–243, 2002.
[55] C Bogey, C Bailly, and D Juvé. Noise investigation of a high subsonic, moderate Reynolds number jet using a compressible large eddy simulation. *Theor Comput Fluid Dyn*, 16(4):273–297, 2003.
[56] C Bogey and O Marsden. Simulations of initially highly disturbed jets with experiment-like exit boundary layers. *AIAA J*, 54(4):1299–1312, 2016.
[57] C Bogey, O Marsden, and C Bailly. Large-eddy simulation of the flow and acoustic fields of a Reynolds number 10 5 subsonic jet with tripped exit boundary layers. *Phys Fluids*, 23(3):035104, 2011.
[58] C Bogey, O Marsden, and C Bailly. Influence of initial turbulence level on the flow and sound fields of a subsonic jet at a diameter-based Reynolds number of 10 5. *J Fluid Mech*, 701:352–385, 2012.
[59] SM Bosniakov, AV Wolkov, AP Duben, VI Zapryagarev, TK Kozubskaya, SV Mikhaylov, AI Troshin, and VO Tsvetkova. Comparison of two higher accuracy unstructured scale-resolving approaches applied to dual-stream nozzle jet simulation. *Math Models Comput Simul*, 12(3):368–377, 2020.
[60] J Boudet, J-F Monier, and F Gao. Implementation of a roughness element to trip transition in large-eddy simulation. *J Therm Sci*, 24(1):30–36, 2015.
[61] S Bourasseau. *Contribution to a mesh refinement method based on the adjoint vector for the computation of aerodynamic outputs*. PhD thesis, Université Nice Sophia Antipolis, JAD - Laboratoire Jean Alexandre Dieudonné, 2015. `https://hal.archives-ouvertes.fr/tel-01279567v2`.
[62] P Brancher, JM Chomaz, and P Huerre. Direct numerical simulations of round jets: vortex induction and side jets. *Phys Fluids*, 6(5):1768–1774, 1994.
[63] GA Brès, P Jordan, V Jaunet, M Le Rallic, AVG Cavalieri, A Towne, SK Lele, T Colonius, and OT Schmidt. Importance of the nozzle-exit boundary-layer state in subsonic turbulent jets. *J Fluid Mech*, 851:83–124, 2018.
[64] GA Brès, ST Bose, F Ham, and SK Lele. Unstructured large eddy simulations for nozzle interior flow modeling and jet noise predictions. In *20th AIAA/CEAS Aeroacoustics Conference*, page 2601, 2014.
[65] V Brunet. Random flow generation technique for civil aircraft jet simulations with the ZDES approach. In *Progress in Hybrid RANS-LES Modelling*, pages 193–204. Springer, 2012.
[66] N Burgess and D Mavriplis. An hp-adaptive discontinuous Galerkin solver for Aerodynamic flows on Mixed-Element meshes. In *49th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition, 4-7 January 2011, Orlando, Florida, USA*, 2011. AIAA Paper 2011-490.
[67] JC Butcher. Coefficients for the study of Runge-Kutta integration processes. *J Aust Math Soc*, 3(2):185–201, 1963.
[68] X Cai and F Ladeinde. Hybrid RANS/LES Calculations of High Speed Jet Noise. In *37th AIAA Fluid Dynamics Conference and Exhibit*, page 3870, 2007.
[69] A Carabias, A Belme, A Loseille, and A Dervieux. Anisotropic Goal-oriented error analysis for a third-order accurate CENO Euler discretization. *Int J Numer Methods Fluids*, 86(6):392–413, 2018.
[70] M Caraeni, Y Dai, and D Caraeni. Acoustic investigation of rod airfoil configuration with des and fwh. In *37th AIAA Fluid Dynamics Conference and Exhibit*, page 4106, 2007.
[71] HA Carson, AC Huang, MC Galbraith, SR Allmaras, and DL Darmofal. Mesh optimization via error sampling and synthesis: An update. In *AIAA Scitech 2020 Forum*, page 0087, 2020.

[72] C Carton de Wiart and K Hillewaert. A discontinuous Galerkin method for implicit LES of moderate Reynolds number flows. In *53rd AIAA Aerospace Sciences Meeting*, page 0055, 2015.
[73] MJ Castro-Díaz, F Hecht, B Mohammadi, and O Pironneau. Anisotropic unstructured mesh adaption for flow simulations. *Int J Numer Methods Fluids*, 25(4):475–491, 1997.
[74] M Ceze and K Fidkowski. Output-driven anisotropic mesh adaptation for viscous flows using discrete choice optimization. In *48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, page 170, 2010.
[75] M Ceze and KJ Fidkowski. Anisotropic hp-adaptation framework for functional prediction. *AIAA J*, 51(2):492–509, 2013.
[76] MA Ceze and KJ Fidkowski. High-order output-based adaptive simulations of turbulent flow in two dimensions. *AIAA J*, 54(9):2611–2625, 2016.
[77] Neal M Chaderjian. Advances in rotor performance and turbulent wake simulation using DES and adaptive mesh refinement. In *Seventh International Conference on Computational Fluid Dynamics (ICCFD7)*, number ARC-E-DAA-TN5574, 2012.
[78] N Chalmers, G Agbaglah, M Chrust, and C Mavriplis. A parallel hp-adaptive high order discontinuous Galerkin method for the incompressible Navier-Stokes equations. *J Comput Phys: X*, 2:100023, 2019.
[79] B Chaouat. The state of the art of hybrid RANS/LES modeling for the simulation of turbulent flows. *Flow, Turbulence and Combustion*, 99(2):279–327, 2017.
[80] J-B Chapelier, M De La Llave Plata, and E Lamballais. Development of a multiscale LES model in the context of a modal discontinuous Galerkin method. *Comput Methods Appl Mech Eng*, 307:275–299, 2016.
[81] J-B Chapelier, M De La Llave Plata, F Renac, and E Lamballais. Evaluation of a high-order discontinuous Galerkin method for the DNS of turbulent flows. *Comput Fluids*, 95:210–226, 2014.
[82] J-B Chapelier and G Lodato. A spectral-element dynamic model for the Large-Eddy simulation of turbulent flows. *J Comput Phys*, 321:279–302, 2016.
[83] DR Chapman. Computational aerodynamics development and outlook. *AIAA J*, 17(12):1293–1313, 1979.
[84] M Chauhan and L Massa. Large-Eddy Simulation of Supersonic Jet Noise with Discontinuous Galerkin Methods. *AIAA J*, pages 1–20, 2021.
[85] N Chauvet, S Deck, and L Jacquin. Numerical study of mixing enhancement in a supersonic round jet. *AIAA J*, 45(7):1675–1687, 2007.
[86] N Chauvet, S Deck, and L Jacquin. Zonal Detached Eddy Simulation of a controlled propulsive jet. *AIAA J*, 45(10):2458–2473, 2007.
[87] K Chitale, O Sahni, S Tendulkar, R Nastasia, M Shephard, and K Jansen. Boundary layer adaptivity for transonic turbulent flows. In *21st AIAA Computational Fluid Dynamics Conference*, page 2445, 2013.
[88] H Choi and P Moin. Grid-point requirements for large eddy simulation: Chapman's estimates revisited. *Phys Fluids*, 24(1):011702, 2012.
[89] JH Choi, JJ Alonso, and E van der Weide. An analysis of inviscid transonic flows over three-dimensional wings using the discontinuous Galerkin solver in SU2. In *AIAA Scitech 2020 Forum*, page 1564, 2020.
[90] L Cirrottola, A Froehly, A Guardone, G Quaranta, B Re, M Ricchiuto, et al. R-adaptation for unsteady compressible flow simulations in three dimensions. In *International Conference on Adaptive Modeling and Simulation (ADMOS 2019)-ECCOMAS Thematic Conference*, pages 1–4, 2019.
[91] Bernardo Cockburn. Discontinuous galerkin methods for convection-dominated problems. In *High-Order Methods for Computational Physics*, pages 69–224. Springer, 1999.
[92] A Colombo, G Manzinali, A Ghidoni, G Noventa, M Franciolini, A Crivellini, and F Bassi. A p-adaptive implicit discontinuous Galerkin method for the under-resolved simulation of compressible turbulent flows. In *7th European Conference on Computational Fluid Dynamics (ECFD 7), 11-15 June 2018, Glasgow, UK*, 2018. `http://congress.cimne.com/eccm_ecfd2018/admin/files/filePaper/p1714.pdf`.
[93] AT Corrigan, A Kercher, J Liu, and K Kailasanath. Jet noise simulation using a higher-order discontinuous galerkin method. In *2018 AIAA Aerospace Sciences Meeting*, page 1247, 2018.
[94] O Coulaud and A Loseille. Very high order anisotropic metric-based mesh adaptation in 3D. *Procedia Eng*, 163:353–365, 2016.

[95] J Dahan, C O'Reilly, and G Efraimsson. Numerical investigation of a realistic nose landing gear. In *20th AIAA/CEAS Aeroacoustics Conference*, page 2077, 2014.

[96] V D'Alessandro, S Montelpare, and R Ricci. Detached–eddy simulations of the flow over a cylinder at Re= 3900 using OpenFOAM. *Comput Fluids*, 136:152–169, 2016.

[97] C Dapogny, C Dobrzynski, and P Frey. Three-dimensional adaptive domain remeshing, implicit domain meshing, and applications to free and moving boundary problems. *J Comput Phys*, 262:358–378, 2014.

[98] G Daviller, M Brebion, P Xavier, G Staffelbach, J-D Müller, and T Poinsot. A mesh adaptation strategy to predict pressure losses in LES of swirled flows. *Flow Turbul Combust*, 99(1):93–118, 2017.

[99] M de la Llave Plata, V Couaillier, and M-C Le Pape. On the use of a high-order discontinuous Galerkin method for DNS and LES of wall-bounded turbulence. *Comput Fluids*, 176:320–337, 2018.

[100] M de la Llave Plata, E Lamballais, and F Naddei. On the performance of a high-order multiscale DG approach to LES at increasing Reynolds number. *Comput Fluids*, 194:104306, 2019.

[101] S Deck. Zonal-detached-eddy simulation of the flow around a high-lift configuration. *AIAA J*, 43(11):2372–2384, 2005.

[102] S Deck. Recent improvements in the Zonal Detached Eddy Simulation (ZDES) formulation. *Theor Comput Fluid Dyn*, 26(6):523–550, 2012.

[103] S Deck and N Renard. Towards an enhanced protection of attached boundary layers in hybrid RANS/LES methods. *J Comput Phys*, 400:108970, 2020.

[104] S Deck, P-E Weiss, and N Renard. A rapid and low noise switch from RANS to WMLES on curvilinear grids with compressible flow solvers. *J Comput Phys*, 363:231–255, 2018.

[105] L Demkowicz, W Rachowicz, and P Devloo. A fully automatic hp-adaptivity. *J Sci Comput*, 17(1):117–142, 2002.

[106] VF Dippold. Generating a grid for unstructured rans simulations of jet flows. In *2018 Fluid Dynamics Conference, 25-29 June 2018, Atlanta, Georgia, USA*, 2018. AIAA Paper 2018-3223.

[107] C Dobrzynski and P Frey. Anisotropic Delaunay Mesh Adaptation for Unsteady Simulations. In RV Garimella, editor, *Proceedings of the 17th International Meshing Roundtable*, pages 177–194. Springer, 2008.

[108] KT Doetsch and K Fidkowski. Unsteady Combined Entropy and Output-based Adjoint Approach for Mesh Refinement and Error Estimation. In *AIAA Aviation 2019 Forum*, page 2951, 2019.

[109] V Dolejší. hp-DGFEM for nonlinear convection-diffusion problems. *Math Comput Simul*, 87:87–118, 2013.

[110] V Dolejší. Anisotropic hp-adaptive method based on interpolation error estimates in the Lq-norm. *Appl Numer Math*, 82:80–114, 2014.

[111] V Dolejší and M Feistauer. *Discontinuous Galerkin method: analysis and applications to compressible flow. Springer Series in Computational Mathematics*, volume 48. Springer, 2015.

[112] V Dolejší, G May, and A Rangarajan. A continuous hp-mesh model for adaptive discontinuous Galerkin schemes. *Appl Numer Math*, 124:1–21, 2018.

[113] V Dolejší and P Solin. hp-discontinuous galerkin method based on local higher order reconstruction. *Appl Math Comput*, 279:219–235, 2016.

[114] RP Dwight. Robust mesh deformation using the linear elasticity equations. In *Computational fluid dynamics 2006*, pages 401–406. Springer, 2009.

[115] G Eitel-Amor, M Meinke, and W Schröder. A lattice-Boltzmann method with hierarchically refined meshes. *Comput Fluids*, 75:127–139, 2013.

[116] D Ekelschot, D Moxey, SJ Sherwin, and J Peiró. A p-adaptation method for compressible flow problems using a goal-based error indicator. *Comput Struct*, 181:55–69, 2017.

[117] C Eskilsson. An hp-adaptive discontinuous galerkin method for shallow water flows. *Int J Numer Methods Fluids*, 67(11):1605–1623, 2011.

[118] EA Fadlun, R Verzicco, P Orlandi, and J Mohd-Yusof. Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations. *J Comput Phys*, 161(1):35–60, 2000.

[119] R Feuillet, O Coulaud, and A Loseille. Anisotropic Error Estimate for High-order Parametric Surface Mesh Generation. In *IMR 2019-28th International Meshing Roundtable*, 2019.

[120] JE Ffowcs Williams and DL Hawkings. Sound generation by turbulence and surfaces in arbitrary motion. *Philosophical Transactions of the Royal Society of London. Series A, Mathematical and Physical Sciences*, 264(1151):321–342, 1969.

[121] KJ Fidkowski. Output-based space–time mesh optimization for unsteady flows using continuous-in-time adjoints. *J Comput Phys*, 341:258–277, 2017.
[122] KJ Fidkowski and DL Darmofal. A triangular cut-cell adaptive method for high-order discretizations of the compressible Navier–Stokes equations. *J Comput Phys*, 225(2):1653–1672, 2007.
[123] KJ Fidkowski and DL Darmofal. Review of output-based error estimation and mesh adaptation in computational fluid dynamics. *AIAA J*, 49(4):673–694, 2011.
[124] D Flad, Andrea B, and C-D Munz. Simulation of underresolved turbulent flows by adaptive filtering using the high order discontinuous Galerkin spectral element method. *J Comput Phys*, 313:1–12, 2016.
[125] L Formaggia and S Perotto. New anisotropic a priori error estimates. *Numer Math*, 89(4):641–667, 2001.
[126] B Fornberg. Steady viscous flow past a sphere at high Reynolds numbers. *J Fluid Mech*, 190:471–489, 1988.
[127] A Fosso-Pouangué, M Sanjosé, and S Moreau. Jet noise simulation with realistic nozzle geometries using fully unstructured LES solver. In *18th AIAA/CEAS aeroacoustics conference (33rd AIAA aeroacoustics conference)*, page 2190, 2012.
[128] F Fraysse, E Valero, and J Ponsin. Comparison of mesh adaptation using the adjoint methodology and truncation error estimates. *AIAA J*, 50(9):1920–1932, 2012.
[129] P-J Frey and F Alauzet. Anisotropic mesh adaptation for CFD computations. *Comput Methods Appl Mech Eng*, 194(48-49):5068–5082, 2005.
[130] PJ Frey and P-L George. *Mesh Generation: Application to Finite Elements, Second Edition*. Wiley Online Library, 2008.
[131] J Fröhlich and D Von Terzi. Hybrid LES/RANS methods for the simulation of turbulent flows. *Progress in Aerospace Sciences*, 44(5):349–377, 2008.
[132] F Gand. Investigation of turbulence development in incompressible jets with zonal detached eddy simulation (ZDES) and synthetic turbulent inflow. *Int J Heat Fluid Flow*, 61:425–437, 2016.
[133] F Gand, V Brunet, and G Mancel. Zonal Detached Eddy Simulation (ZDES) using turbulent inflow and high order schemes: Application to jet flows. In *Progress in Hybrid RANS-LES Modelling*, pages 141–152. Springer, 2015.
[134] F Gand and M Huet. Improvement of jet flow simulations using ZDES mode 3 and silent turbulence generation. In *25th AIAA/CEAS Aeroacoustics Conference*, page 2687, 2019.
[135] F Gand and M Huet. On the generation of turbulent inflow for hybrid RANS/LES jet flow simulations. *Comput Fluids*, 216:104816, 2021.
[136] F Gand, M Huet, T Le Garrec, and F Cléro. Jet noise of a UHBR nozzle using ZDES: external boundary layer thickness and installation effects. In *23rd AIAA/CEAS Aeroacoustics Conference*, page 3526, 2017.
[137] H Gao and ZJ Wang. A residual-based procedure for hp-adaptation on 2-d hybrid meshes. In *49th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, page 492, 2011.
[138] R Garimella. Conformal refinement of unstructured quadrilateral meshes. In *Proceedings of the 18th International Meshing Roundtable*, pages 31–44. Springer, 2009.
[139] G Gassner, C Altmann, F Hindenlang, M Staudenmeier, and CD Munz. Explicit Discontinuous Galerkin Schemes with Adaptation in Space and Time. In *36th CFD/ADIGMA course on hp-adaptive and hp-multigrid methods, VKI LS 2009*, 2009.
[140] G Gassner, M Staudenmaier, F Hindenlang, M Atak, and C-D Munz. A space–time adaptive discontinuous Galerkin scheme. *Comput Fluids*, 117:247–261, 2015.
[141] P-L George, H Borouchaki, F Alauzet, P Laug, A Loseille, D Marcum, and L Maréchal. Mesh generation and mesh adaptivity: theory and techniques. In *Encyclopedia of Computational Mechanics Second Edition*. Wiley Online Library, 2018.
[142] E H Georgoulis, E Hall, and P Houston. Discontinuous Galerkin methods for advection-diffusion-reaction problems on anisotropically refined meshes. *SIAM J Sci Comput*, 30(1):246–271, 2008.
[143] EH Georgoulis, E Hall, and P Houston. Discontinuous Galerkin methods on hp-anisotropic meshes II: A posteriori error analysis and adaptivity. *Appl Numer Math*, 59(9):2179–2194, 2009.
[144] M Germano, U Piomelli, P Moin, and WH Cabot. A dynamic subgrid-scale eddy viscosity model. *Phys Fluids A: Fluid Dyn*, 3(7):1760–1765, 1991.

[145] J Glaubitz, AC Nogueira, JLS Almeida, RF Cantão, and CAC Silva. Smooth and compactly supported viscous sub-cell shock capturing for discontinuous Galerkin methods. *J Sci Comput*, 79(1):249–272, 2019.

[146] GJ Gorman, G Rokos, J Southern, and PHJ Kelly. Thread-Parallel Anisotropic Mesh Adaptation. In *New Challenges in Grid Generation and Adaptivity for Scientific Computing*, pages 113–137. Springer, 2015.

[147] J-L Guermond and R Pasquetti. Entropy-based nonlinear viscosity for Fourier approximations of conservation laws. *Comptes Rendus Mathematique*, 346(13-14):801–806, 2008.

[148] W Gui and I Babuška. The h, p and h-p versions of the finite element method in 1 dimension. Part 3: The adaptive h-p version. *Numer Math*, 49:659–683, 1986.

[149] F Ham, A Sharma, M Shoeybi, S Lele, P Moin, and E van der Weide. Noise prediction from cold high-speed turbulent jets using large-eddy simulation. In *47th AIAA Aerospace Sciences Meeting including The New Horizons Forum and Aerospace Exposition*, page 9, 2009.

[150] NJ Harris, SE Benzley, and SJ Owen. Conformal Refinement of All-Hexahedral Element Meshes Based on Multiple Twist Plane Insertion. In *IMR*, pages 157–168, 2004.

[151] A Harten and JM Hyman. Self adjusting grid methods for one-dimensional hyperbolic conservation laws. *J Comput Phys*, 50(2):235–269, 1983.

[152] R Hartmann, J Held, T Leicht, and F Prill. Error estimation and adaptive mesh refinement for aerodynamic flows. In *ADIGMA-A European Initiative on the Development of Adaptive Higher-Order Variational Methods for Aerospace Applications*, pages 339–353. Springer, 2010.

[153] F Hecht. BAMG: Bidimensional Anisotropic Mesh Generator, 1998. `https://www.ljll.math.upmc.fr/hecht/ftp/bamg/bamg.pdf`.

[154] S Heinz. A review of hybrid RANS-LES methods for turbulent flows: Concepts and applications. *Progress in Aerospace Sciences*, 114:100597, 2020.

[155] P Houston, B Senior, and E Süli. Sobolev regularity estimation for hp-adaptive finite element methods. In F Brezzi, A Buffa, S Corsaro, and A Murli, editors, *Numer Math Adv Appl*, pages 631–656. Springer, 2003.

[156] M Huet. Influence of boundary layers resolution on heated, subsonic, high Reynolds number jet flow and noise. In *19th AIAA/CEAS Aeroacoustics Conference*, page 2141, 2013.

[157] T JR Hughes, L Mazzei, and KE Jansen. Large eddy simulation and the variational multiscale method. *Computing and visualization in science*, 3(1):47–59, 2000.

[158] ZD Husain and AKMF Hussain. Axisymmetric mixing layer: influence of the initial and boundary conditions. *AIAA J*, 17(1):48–55, 1979.

[159] D Ibanez and M Shephard. Mesh adaptation for moving objects on shared memory hardware. *25th International Meshing Roundtable (IMR25), 27-30 September 2016, Washington, DC, USA*, 2016. `https://imr.sandia.gov/papers/imr25/2007_imr25RN_Ibanez.pdf`.

[160] E Itam, S Wornom, B Koobus, and A Dervieux. Hybrid simulation of high-Reynolds number flows relying on a variational multiscale model. In *Symposium on Hybrid RANS-LES Methods*, pages 207–217. Springer, 2016.

[161] N Jarrin, R Prosser, J-C Uribe, S Benhamadouche, and D Laurence. Reconstruction of turbulent fluctuations for hybrid RANS/LES simulations using a synthetic-eddy method. *Int J Heat Fluid Flow*, 30(3):435–442, 2009.

[162] H Jasak and AD Gosman. Automatic resolution control for the finite-volume method, part 2: Adaptive mesh refinement and coarsening. *Numer Heat Transf B: Fundam*, 38(3):257–271, 2000.

[163] S Jee and K Shariff. Detached-eddy simulation based on the v2-f model. *Int J Heat Fluid Flow*, 46:84–101, 2014.

[164] S Kaennakham, A Holdø, and C Lambert. A new simple h-mesh adaptation algorithm for standard Smagorinsky LES: a first step of Taylor scale as a refinement variable. *Int J Multiphys*, 4(1):33–50, 2010.

[165] C Kamble and S Girimaji. Characterization of coherent structures in turbulent wake of a sphere using partially averaged Navier–Stokes (PANS) simulations. *Phys Fluids*, 32(10):105110, 2020.

[166] S Kamkar, A Jameson, and A Wissink. Automated Grid Refinement Using Feature Detection. In *47th AIAA Aerospace Sciences Meeting including The New Horizons Forum and Aerospace Exposition*, page 1496, 2009.

[167] G Karniadakis and S Sherwin. *Spectral/hp Element Methods for Computational Fluid Dynamics*. Oxford University Press, 2013.

[168] P Kholodov and S Moreau. Identification of Noise Sources in a Realistic Turbofan Rotor Using Large Eddy Simulation. In *Acoustics*, volume 2, pages 691–706. Multidisciplinary Digital Publishing Institute, 2020.

[169] HJ Kim and PA Durbin. Observations of the frequencies in a sphere wake and of drag increase by acoustic excitation. *The Physics of fluids*, 31(11):3260–3265, 1988.

[170] J Kim and H Choi. Large eddy simulation of a circular jet: effect of inflow conditions on the near field. *J Fluid Mech*, 620:383–411, 2009.

[171] AN Kolmogorov. Equations of turbulent motion in an incompressible fluid. In *Dokl. Akad. Nauk SSSR*, volume 30, pages 299–303, 1941.

[172] M Kompenhans, G Rubio, E Ferrer, and E Valero. Comparisons of p-adaptation strategies based on truncation-and discretisation-errors for high order discontinuous Galerkin methods. *Comput Fluids*, 139:36–46, 2016.

[173] JA Krakos and DL Darmofal. Anisotropic output-based mesh optimization for unsteady flows. In *21st AIAA Computational Fluid Dynamics Conference*, page 3083, 2013.

[174] L Krivodonova, J Xin, J-F Remacle, N Chevaugeon, and JE Flaherty. Shock detection and limiting with discontinuous Galerkin methods for hyperbolic conservation laws. *Appl Numer Math*, 48(3-4):323–338, 2004.

[175] G Kuru, M de la Llave Plata, V Couaillier, R Abgrall, and F Coquel. An adaptive variational multiscale discontinuous Galerkin method for Large Eddy Simulation. In *54th AIAA Aerospace Sciences Meeting, 4-8 January 2016, San Diego, California, USA*, 2016. AIAA Paper 2016-0584.

[176] R Langtry, E Gren, J Larssen, and P Spalart. Evaluation of structured and unstrucutred grids for detached eddy simulation of flap edge noise. In *15th AIAA/CEAS Aeroacoustics Conference (30th AIAA Aeroacoustics Conference)*, page 3102, 2009.

[177] J Larsson, S Kawai, J Bodart, and I Bermejo-Moreno. Large eddy simulation with modeled wall-stress: recent progress and future directions. *Mech Eng Rev*, 3(1):15–00418, 2016.

[178] BE Launder and DB Spalding. The numerical computation of turbulent flows. In *Numerical prediction of flow, heat transfer, turbulence and combustion*, pages 96–116. Elsevier, 1983.

[179] S Le Bras, H Deniau, C Bogey, and G Daviller. Development of compressible large-eddy simulations combining high-order schemes and wall modeling. *AIAA J*, 55(4):1152–1163, 2017.

[180] T Leicht and R Hartmann. Anisotropic mesh refinement for discontinuous Galerkin methods in two-dimensional aerodynamic flow simulations. *Int J Numer Methods Fluids*, 56:2111–2138, 2008.

[181] T Leicht and R Hartmann. Error estimation and anisotropic mesh refinement for 3d laminar aerodynamic flow simulations. *J Comput Phys*, 229(19):7344–7360, 2010.

[182] T Leicht and R Hartmann. Error estimation and hp-adaptive mesh refinement for discontinuous Galerkin methods. In ZJ Wang, editor, *Adaptive High-Order Methods in Computational Fluid Dynamics*, pages 67–94. World Scientific, 2011.

[183] T Leicht, J Jägersküpper, D Vollmer, A Schwöppe, R Hartmann, J Fiedler, and T Schlauch. DLR-project Digital-X-Next Generation CFD Solver 'Flucs'. In *Deutscher Luft- und Raumfahrtkongress, 13-15 September 2016*, 2016. http://www.dglr.de/publikationen/2017/420027.pdf.

[184] P-T Lew, G Blaisdell, and A Lyrintzis. Recent progress of hot jet aeroacoustics using 3-d large-eddy simulation. In *11th AIAA/CEAS Aeroacoustics Conference*, page 3084, 2005.

[185] X Li, MS Shephard, and MW Beall. 3D anisotropic mesh adaptation by mesh modification. *Comput Methods Appl Mech Eng*, 194(48-49):4915–4950, 2005.

[186] MJ Lighthill. On sound generated aerodynamically I. General theory. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 211(1107):564–587, 1952.

[187] DK Lilly. The representation of small-scale turbulence in numerical simulation experiments. *IBM Form*, pages 195–210, 1967.

[188] W Liu, JW Kim, X Zhang, D Angland, and B Caruelle. Landing-gear noise prediction using high-order finite difference schemes. *J Sound Vib*, 332(14):3517–3534, 2013.

[189] M Lorteau, F Cléro, and F Vuillot. Analysis of noise radiation mechanisms in hot subsonic jet from a validated large eddy simulation solution. *Phys Fluids*, 27(7):075108, 2015.

[190] M Lorteau, M De La Llave Plata, and V Couaillier. Turbulent jet simulation using high-order DG methods for aeroacoustic analysis. *Int J Heat Fluid Flow*, 70:380–390, 2018.

[191] A Loseille and F Alauzet. Continuous mesh framework part I: well-posed continuous interpolation error. *SIAM J Numer Anal*, 49(1):38–60, 2011.
[192] A Loseille and F Alauzet. Continuous mesh framework part II: validations and applications. *SIAM J Numer Anal*, 49(1):61–86, 2011.
[193] A Loseille, A Dervieux, and F Alauzet. Fully anisotropic goal-oriented mesh adaptation for 3D steady Euler equations. *J Comput Phys*, 229(8):2866–2897, 2010.
[194] A Loseille, L Frazza, and F Alauzet. Comparing anisotropic adaptive strategies on the second AIAA sonic boom workshop geometry. *J Aircr*, 56(3):938–952, 2019.
[195] A Loseille and R Lohner. Anisotropic adaptive simulations in aerodynamics. In *48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition, 4-7 January 2010, Orlando, Florida, USA*, 2010. AIAA Paper 2010-169.
[196] C Lozano. Discrete surprises in the computation of sensitivities from boundary integrals in the continuous adjoint approach to inviscid aerodynamic shape optimization. *Comput Fluids*, 56:118–127, 2012.
[197] AS Lyrintzis. The use of Kirchhoff's method in computational aeroacoustics. 1994.
[198] DA Lysenko, IS Ertesvåg, and KE Rian. Large-eddy simulation of the flow over a circular cylinder at Reynolds number 3900 using the OpenFOAM toolbox. *Flow, turbulence and combustion*, 89(4):491–518, 2012.
[199] K Mani and D Mavriplis. Discrete adjoint based time-step adaptation and error reduction in unsteady flow problems. In *18th AIAA Computational Fluid Dynamics Conference*, page 3944, 2007.
[200] DL Marcum and F Alauzet. Unstructured mesh generation using advancing layers and metric-based transition for viscous flowfields. In *21st AIAA Computational Fluid Dynamics Conference, 24-27 June 2013, San Diego, CA, USA*, 2013. AIAA Paper 2013-2710.
[201] M Marek, A Tyliszczak, and A Bogusławski. Large eddy simulation of incompressible free round jet with discontinuous galerkin method. *Int J Numer Methods Fluids*, 79(4):164–182, 2015.
[202] C Mavriplis. A Posteriori Error Estimators for Adaptive Spectral Element Techniques. In P Wesseling, editor, *Proceedings of the Eighth GAMM-Conference on Numerical Methods in Fluid Mechanics. Notes on Numerical Fluid Mechanics (NNFM)*, volume 29, pages 333–342, 1990.
[203] M Meinel and GO Einarsson. The FlowSimulator framework for massively parallel CFD applications. *PARA 2010*, 2010.
[204] B Mele, M Ostieri, and R Tognaccini. Aircraft lift and drag decomposition in transonic flows. *J Aircr*, 54(5):1933–1944, 2017.
[205] S Mendez, M Shoeybi, SK Lele, and P Moin. On the use of the Ffowcs Williams-Hawkings equation to predict far-field jet noise from large-eddy simulations. *Int J Aeroacoustics*, 12(1-2):1–20, 2013.
[206] F Menter. Zonal two equation kw turbulence models for aerodynamic flows. In *23rd fluid dynamics, plasmadynamics, and lasers conference*, page 2906, 1993.
[207] F Menter, A Hüppe, A Matyushenko, and D Kolmogorov. An overview of hybrid RANS–LES models developed for industrial CFD. *Applied Sciences*, 11(6):2459, 2021.
[208] F Menter, M Kuntz, and R Bender. A scale-adaptive simulation model for turbulent flow predictions. In *41st aerospace sciences meeting and exhibit*, page 767, 2003.
[209] T Michal, D Babcock, D Kamenetskiy, J Krakos, M Mani, R Glasby, T Erwin, and DL Stefanski. Comparison of fixed and adaptive unstructured grid results for drag prediction workshop 6. *J Aircr*, 55(4):1420–1432, 2018.
[210] T Michal and J Krakos. Anisotropic mesh adaptation through edge primitive operations. In *50th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition, 9-12 January 2012, Nashville, Tennessee, USA*, 2012. AIAA Paper 2012-159.
[211] TR Michal, DS Kamenetskiy, and J Krakos. Anisotropic Adaptive Mesh Results for the Third High Lift Prediction Workshop (HiLiftPW-3). In *2018 AIAA Aerospace Sciences Meeting*, page 1257, 2018.
[212] WF Mitchell and MA McClain. A survey of hp-adaptive strategies for elliptic partial differential equations. In T Simos, editor, *Recent Advances in Computational and Applied Mathematics*, pages 227–258. Springer, 2011.
[213] P Morris and S Boluriaan. The prediction of jet noise from cfd data. In *10th AIAA/CEAS Aeroacoustics Conference*, page 2977, 2004.
[214] PJ Morris, LN Long, TE Scheidegger, and S Boluriaan. Simulations of supersonic jet noise. *Int J Aeroacoustics*, 1(1):17–41, 2002.

[215] V Moureau, G Lartigue, P Bénard, and R Mercier. Parallel and dynamic mesh adaptation of thetrahedral-based meshes for propagating fronts and interfaces: application to premixed combustion. In *32nd International Conference on Parallel Computational Fluid Dynamics (ParCFD2021)*, 2021.

[216] S Mozaffari, E Guilmineau, M Visonneau, and J Wackers. Average-based mesh adaptation for hybrid RANS/LES simulation of complex flows. *Comput Fluids*, page 105202, 2021.

[217] O Mykhaskiv, P Mohanamuraly, J-D Mueller, S Xu, and S Timme. CAD-based shape optimisation of the NASA CRM wing-body intersection using differentiated CAD-kernel. In *35th AIAA Applied Aerodynamics Conference*, page 4080, 2017.

[218] F Naddei. *Adaptive Large Eddy Simulations based on discontinuous Galerkin methods*. PhD thesis, Université Paris-Saclay, CMAP - Centre de Mathématiques Appliquées - École Polytechnique, 2019. https://hal.archives-ouvertes.fr/tel-02380883/.

[219] F Naddei, M de la Llave Plata, V Couaillier, and F Coquel. A comparison of refinement indicators for p-adaptive simulations of steady and unsteady flows using discontinuous Galerkin methods. *J Comput Phys*, 376:508–533, 2019.

[220] F Naddei, M de la Llave Plata, and E Lamballais. Spectral and modal energy transfer analyses of LES using the discontinuous Galerkin method and their application to the Variational Multiscale approach. *J Comput Phys*, 427:110031, 2021.

[221] IZ Naqavi, Z-N Wang, PG Tucker, M Mahak, and P Strange. Far-field noise prediction for jets using large-eddy simulation and Ffowcs Williams–Hawkings method. *Int J Aeroacoustics*, 15(8):757–780, 2016.

[222] A Neifeld, D Boenke, J Dierke, and R Ewert. Jet noise prediction with Eddy relaxation source model. In *21st AIAA/CEAS Aeroacoustics Conference, 22-26 June 2015, Dallas, TX, USA*, 2015. AIAA Paper 2015-2370.

[223] G Nicolas, T Fouquet, S Geniaut, and S Cuvilliez. Improved adaptive mesh refinement for conformal hexahedral meshes. *Adv Eng Softw*, 102:14–28, 2016.

[224] F Nicoud and F Ducros. Subgrid-scale stress modelling based on the square of the velocity gradient tensor. *Flow, turbulence and Combustion*, 62(3):183–200, 1999.

[225] N Odier, A Thacker, M Harnieh, G Staffelbach, L Gicquel, F Duchaine, Nicolás G Rosa, and J-D Müller. A mesh adaptation strategy for complex wall-modeled turbomachinery LES. *Comput Fluids*, 214:104766, 2021.

[226] N Offermans, A Peplinski, O Marin, and P Schlatter. Adaptive mesh refinement for steady flows in Nek5000. *Comput Fluids*, 197:104352, 2020.

[227] D Pagnutti and C Ollivier-Gooch. A generalized framework for high order anisotropic mesh adaptation. *Comput Struct*, 87(11-12):670–679, 2009.

[228] A Pal, S Sarkar, A Posa, and E Balaras. Regeneration of turbulent fluctuations in low-Froude-number flow over a sphere at a Reynolds number of 3700. *J Fluid Mech*, 804, 2016.

[229] A Pal, S Sarkar, A Posa, and E Balaras. Direct numerical simulation of stratified flow past a sphere at a subcritical Reynolds number of 3700 and moderate Froude number. *J Fluid Mech*, 826:5–31, 2017.

[230] U Paliath and S Premasuthan. Large eddy simulation for jet installation effects. In *19th AIAA/CEAS aeroacoustics conference*, page 2137, 2013.

[231] J Palm, C Eskilsson, and L Bergdahl. An hp-adaptive discontinuous Galerkin method for modelling snap loads in mooring cables. *Ocean Eng*, 144:266–276, 2017.

[232] MA Park. Adjoint-based, three-dimensional error prediction and grid adaptation. *AIAA J*, 42(9):1854–1862, 2004.

[233] MA Park, A Balan, WK Anderson, MC Galbraith, P Caplan, HA Carson, TR Michal, JA Krakos, DS Kamenetskiy, A Loseille, et al. Verification of unstructured grid adaptation components. In *AIAA Scitech 2019 Forum*, page 1723, 2019.

[234] MA Park, WL Kleb, WK Anderson, SL Wood, A Balan, BY Zhou, and NR Gauger. Exploring unstructured mesh adaptation for hybrid reynolds-averaged navier–stokes/large eddy simulation. In *AIAA SciTech 2020 Forum*, page 1139, 2020.

[235] N Park, S Lee, J Lee, and H Choi. A dynamic subgrid-scale eddy viscosity model with a global model coefficient. *Phys Fluids*, 18(12):125109, 2006.

[236] P-O Persson and J Peraire. Sub-cell shock capturing for discontinuous Galerkin methods. In *44th AIAA Aerospace Sciences Meeting and Exhibit, 9-12 January 2006, Reno, Nevada, USA*, 2006. AIAA Paper 2006-112.

[237] U Piomelli and E Balaras. Wall-layer models for large-eddy simulations. *Annu Rev Fluid Mech*, 34(1):349–374, 2002.

[238] G Rahier, M Huet, and J Prieur. Additional terms for the use of Ffowcs Williams and Hawkings surface integrals in turbulent flows. *Comput Fluids*, 120:158–172, 2015.

[239] A Rangarajan, A Balan, and G May. Mesh optimization for discontinuous Galerkin methods using a continuous mesh model. *AIAA J*, 56(10):4060–4073, 2018.

[240] A Rangarajan, G May, and V Dolejší. Adjoint-based anisotropic hp-adaptation for discontinuous Galerkin methods using a continuous mesh model. *J Comput Phys*, 409:109321, 2020.

[241] Ajay Mandyam Rangarajan. Metric Based hp-Adaptation Using a Continuous Mesh Model for Higher Order Schemes.

[242] AM Rangarajan, A Chakraborthy, and G May. A goal oriented optimization technique for tetrahedral grids using a continuous-mesh model. In *AIAA Scitech 2019 Forum*, page 0349, 2019.

[243] AM Rangarajan, A Chakraborty, G May, and V Dolejší. A continuous-mesh optimization technique for piecewise polynomial approximation on tetrahedral grids. In *2018 Fluid Dynamics Conference*, page 3246, 2018.

[244] AM Rangarajan, G May, and V Dolejší. Adjoint-based anisotropic mesh adaptation for discontinuous galerkin methods using a continuous mesh model. In *23rd AIAA Computational Fluid Dynamics Conference*, page 3100, 2017.

[245] J-F Remacle, JE Flaherty, and MS Shephard. An adaptive discontinuous Galerkin technique with an orthogonal basis applied to compressible flow problems. *SIAM Review*, 45(1):53–72, 2003.

[246] J-F Remacle, C Geuzaine, G Compère, and BT Helenbrook. Adaptive mesh generation and visualization. In R Blockley and W Shyy, editors, *Encyclopedia of Aerospace Engineering*. John Wiley & Sons, Ltd., 2010.

[247] J-F Remacle, X Li, N Chevaugeon, and MS Shephard. Transient mesh adaptation using conforming and non conforming mesh modifications. In *Proceedings of the 11th International Meshing Roundtable (IMR11), 15-18 September 2002, Ithaca, New York, USA*, pages 261–272, 2002. `https://imr.sandia.gov/papers/abstracts/Re263.html`.

[248] J-F Remacle, X Li, MS Shephard, and JE Flaherty. Anisotropic adaptive simulation of transient flows using discontinuous Galerkin methods. *Int J Numer Methods Eng*, 62(7):899–923, 2005.

[249] F Renac. Development of Discontinuous Galerkin Method for RANS Equations on Multibloc Hexahedral Meshes. In *ADIGMA-A European Initiative on the Development of Adaptive Higher-Order Variational Methods for Aerospace Applications*, pages 95–106. Springer, 2010.

[250] N Renard and S Deck. Recent improvements in the formulation of mode iii of zdes (zonal detached eddy simulation) for wmles use at re_θ>10^4. In *53rd AIAA Aerospace Sciences Meeting*, page 1981, 2015.

[251] M-C Rivara and G Iribarren. The 4-triangles longest-side partition of triangles and linear refinement algorithms. *Math Comput*, 65(216):1485–1502, 1996.

[252] L Rochery and A Loseille. P2 cavity operator and Riemannian curved edge length optimization: a path to high-order mesh adaptation. In *AIAA Scitech 2021 Forum*, page 1781, 2021.

[253] I Rodriguez, R Borell, O Lehmkuhl, CD Perez Segarra, and A Oliva. Direct numerical simulation of the flow over a sphere at Re= 3700. *J Fluid Mech*, 679:263–287, 2011.

[254] PL Roe. Approximate Riemann solvers, parameter vectors, and difference schemes. *J Comput Phys*, 43(2):357–372, 1981.

[255] C Roy. Review of discretization error estimators in scientific computing. In *48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, page 126, 2010.

[256] AM Rueda-Ramírez, J Manzanero, E Ferrer, G Rubio, and E Valero. A p-multigrid strategy with anisotropic p-adaptation based on truncation errors for high-order discontinuous Galerkin methods. *J Comput Phys*, 378:209–233, 2019.

[257] CL Rumsey, JP Slotnick, and AJ Sclafani. Overview and summary of the third AIAA high lift prediction workshop. *J Aircr*, 56(2):621–644, 2019.

[258] Y Saad. *Iterative methods for sparse linear systems*. SIAM, 2003.

[259] G Sadaka, A Rakotondrandisa, P-H Tournier, F Luddens, C Lothodé, and I Danaila. Parallel finite-element codes for the simulation of two-dimensional and three-dimensional solid–liquid phase-change systems with natural convection. *Comput Phys Commun*, 257:107492, 2020.

[260] P Sagaut, M Terracol, and S Deck. *Multiscale and multiresolution approaches in turbulence-LES, DES and Hybrid RANS/LES Methods: Applications and Guidelines*. World Scientific, 2013.

[261] M Sanjosé, A Fosso Pouangué, S Moreau, G Wang, and T Padois. Unstructured les of the baseline exejet dual-stream jet. In *20th AIAA/CEAS aeroacoustics conference*, page 3037, 2014.

[262] S Schmidt, C Ilic, V Schulz, and NR Gauger. Three-dimensional large-scale aerodynamic shape optimization based on shape calculus. *AIAA J*, 51(11):2615–2627, 2013.

[263] V Schmitt and F Charpin. Pressure Distribution on the ONERA M6 wing at Transonic Mach Numbers. Technical report, Experimental Data Base for Computer Program Assessment, AGARD Rept., 1979.

[264] SM Schnepp. Error-driven dynamical hp-meshes with the Discontinuous Galerkin Method for three-dimensional wave propagation problems. *J Comput Appl Math*, 270:353–368, 2014.

[265] V Seidl, S Muzaferija, and M Perić. Parallel DNS with local grid refinement. *Applied Scientific Research*, 59(4):379–394, 1997.

[266] M Shakoor and CH Park. A higher-order finite element method with unstructured anisotropic mesh adaption for two phase flows with surface tension. *Comput Fluids*, 230:105154, 2021.

[267] W Shen and SAE Miller. Validation of a high-order large eddy simulation solver for acoustic prediction of supersonic jet flow. *J Theor Comput Acoust*, 28(03):1950023, 2020.

[268] D Shi-Dong, C-H Chen, and S Nadarajah. Adjoint-based aerodynamic optimization of benchmark CRM wing. In *35th AIAA Applied Aerodynamics Conference*, page 3755, 2017.

[269] C-W Shu. A survey of strong stability preserving high order time discretizations. *Collected lectures on the preservation of stability under discretization*, 109:51–65, 2002.

[270] M Shur, PR Spalart, M Strelets, and A Travin. Detached-eddy simulation of an airfoil at high angle of attack. In *Engineering turbulence modelling and experiments 4*, pages 669–678. Elsevier, 1999.

[271] ML Shur, PR Spalart, and MK Strelets. Noise prediction for increasingly complex jets. Part I: Methods and tests. *Int J Aeroacoustics*, 4(3):213–245, 2005.

[272] ML Shur, PR Spalart, and MK Strelets. LES-based evaluation of a microjet noise reduction concept in static and flight conditions. *J Sound Vib*, 330(17):4083–4097, 2011.

[273] ML Shur, PR Spalart, MK Strelets, and AK Travin. A hybrid RANS-LES approach with delayed-DES and wall-modelled LES capabilities. *Int J Heat Fluid Flow*, 29(6):1638–1649, 2008.

[274] J Smagorinsky. General circulation experiments with the primitive equations: I. The basic experiment. *Monthly weather review*, 91(3):99–164, 1963.

[275] A Smirnov, S Shi, and I Celik. Random flow generation technique for large eddy simulations and particle-dynamics modeling. *J. Fluids Eng.*, 123(2):359–371, 2001.

[276] P Spalart and S Allmaras. A one-equation turbulence model for aerodynamic flows. In *30th Aerospace Sciences Meeting and Exhibit, 6-9 January 1992, Reno, Nevada, USA*, 1992. AIAA Paper 1992-439.

[277] PR Spalart. Comments on the feasibility of LES for wings, and on a hybrid RANS/LES approach. In *Proceedings of first AFOSR international conference on DNS/LES*. Greyden Press, 1997.

[278] PR Spalart, S Deck, M L Shur, KD Squires, MK Strelets, and A Travin. A new version of detached-eddy simulation, resistant to ambiguous grid densities. *Theor Comput Fluid Dyn*, 20(3):181–195, 2006.

[279] VK Suman and JK Puttam. Verification of 3D Unstructured Grid Based Euler Solver Developed Using Common Framework For NASA CRM Configuration.

[280] RC Swanson and S Langer. Comparison of NACA 0012 Laminar Flow Solutions: Structured and Unstructured Grid Methods. Technical report, NASA Langley Research Center Hampton, VA, United States & Deutsches Zentrum fuer Luft- und Raumfahrt e.V. Cologne, Germany, 2016. `https://ntrs.nasa.gov/api/citations/20160003623/downloads/20160003623.pdf`.

[281] M Tomac and D Eller. Steps towards automated robust RANS meshing. In *4th CEAS Air & Space Conference, 16-19 September 2013, Linköping, Sweden*, pages 114–123, 2013. `http://www.diva-portal.org/smash/get/diva2:653052/FULLTEXT01.pdf`.

[282] S Toosi and J Larsson. Anisotropic grid-adaptation in large eddy simulations. *Comput Fluids*, 156:146–161, 2017.

[283] C Tsolakis, N Chrisochoides, MA Park, A Loseille, and T Michal. Parallel anisotropic unstructured grid adaptation. *AIAA Journal*, 59(11):4764–4776, 2021.

[284] M Tugnoli, A Abbà, L Bonaventura, and M Restelli. A locally p-adaptive approach for Large Eddy Simulation of compressible flows in a DG framework. *J Comput Phys*, 349:33–58, 2017.
[285] G Tumolo, L Bonaventura, and M Restelli. A semi-implicit, semi-Lagrangian, p-adaptive discontinuous Galerkin method for the shallow water equations. *J Comput Phys*, 232(1):46–67, 2013.
[286] JC Tyacke and PG Tucker. LES of jet flow and noise with internal and external geometry features. In *53rd AIAA Aerospace Sciences Meeting*, page 0503, 2015.
[287] JC Tyacke, Z-N Wang, and PG Tucker. LES–RANS of installed ultra-high-bypass-ratio coaxial jet aeroacoustics with flight stream. *AIAA J*, 57(3):1215–1236, 2019.
[288] A Uzun and MY Hussaini. Investigation of high frequency noise generation in the near-nozzle region of a jet using large eddy simulation. *Theor Comput Fluid Dyn*, 21(4):291–321, 2007.
[289] A Uzun and MY Hussaini. An application of delayed detached eddy simulation to tandem cylinder flow field prediction. *Comput Fluids*, 60:71–85, 2012.
[290] A Uzun, AS Lyrintzis, and GA Blaisdell. Coupling of integral acoustics methods with LES for jet noise prediction. *Int J Aeroacoustics*, 3(4):297–346, 2004.
[291] DA Venditti and DL Darmofal. Adjoint error estimation and grid adaptation for functional outputs: Application to quasi-one-dimensional flow. *J Comput Phys*, 164(1):204–227, 2000.
[292] DA Venditti and DL Darmofal. Anisotropic grid adaptation for functional outputs: application to two-dimensional viscous flows. *J Comput Phys*, 187(1):22–46, 2003.
[293] J Verrière, F Gand, and S Deck. Zonal detached-eddy simulations of a dual-stream jet. *AIAA J*, 54(10):3176–3190, 2016.
[294] R Verzicco and P Orlandi. Direct simulations of the transitional regime of a circular jet. *Phys Fluids*, 6(2):751–759, 1994.
[295] K Viswanathan, ML Shur, PR Spalart, and MK Strelets. Comparisons between experiment and large-eddy simulation for jet noise. *AIAA J*, 45(8):1952–1966, 2007.
[296] AW Vreman. An eddy-viscosity subgrid-scale model for turbulent shear flow: Algebraic theory and applications. *Phys Fluids*, 16(10):3670–3681, 2004.
[297] F Vuillot, N Lupoglazoff, M Lorteau, and F Clero. Large eddy simulation of jet noise from unstructured grids with turbulent nozzle boundary layer. In *22nd AIAA/CEAS Aeroacoustics Conference*, page 3046, 2016.
[298] J Wackers, G Deng, A Leroyer, P Queutey, and M Visonneau. Adaptive grid refinement for hydrodynamic flows. *Comput Fluids*, 55:85–100, 2012.
[299] M Wallraff and T Leicht. Higher order multigrid algorithms for a discontinuous Galerkin RANS solver. In *52nd Aerospace Sciences Meeting, 13-17 January 2014, National Harbor, Maryland, USA*, 2014. AIAA Paper 2014-0936.
[300] L Wang, MK Gobbert, and M Yu. A dynamically load-balanced parallel p-adaptive implicit high-order flux reconstruction method for under-resolved turbulence simulation. *J Comput Phys*, 417:109581, 2020.
[301] L Wang and DJ Mavriplis. Adjoint-based h–p adaptive discontinuous Galerkin methods for the 2d compressible Euler equations. *J Comput Phys*, 228(20):7643–7661, 2009.
[302] L Wang and M Yu. Comparison of ROW, ESDIRK, and BDF2 for unsteady flows with the high-order flux reconstruction formulation. *J Sci Comput*, 83(2):1–27, 2020.
[303] J Waters, D Carrington, and DW Pepper. An adaptive finite element method with dynamic LES for turbulent reactive flows. *Comput Thermal Sci: Int J*, 8(1), 2016.
[304] P Welch. The use of fast Fourier transform for the estimation of power spectra: a method based on time averaging over short, modified periodograms. *IEEE Trans Audio Electroacoust*, 15(2):70–73, 1967.
[305] DC Wilcox. Reassessment of the scale-determining equation for advanced turbulence models. *AIAA J*, 26(11):1299–1310, 1988.
[306] A Wissink. An overset dual-mesh solver for computational fluid dynamics. In *7th International conference on computational fluid dynamics (ICCFD7)*, 2012.
[307] FD Witherden and PE Vincent. On the identification of symmetric quadrature rules for finite element methods. *Comput Math Appl*, 69(10):1232–1241, 2015.
[308] M Woopen, A Balan, G May, and J Schütz. A comparison of hybridized and standard DG methods for target-based hp-adaptive simulation of compressible flow. *Comput Fluids*, 98:3–16, 2014.

[309] H Xia, S Karabasov, P Tucker, A Dowling, and T Hynes. Hybrid RANS-LES Modelling of Chevron Nozzles with Prediction of Far Field Sound. In *49th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, page 21, 2011.

[310] H Xia, PG Tucker, and S Eastwood. Large-eddy simulations of chevron jet flows with noise predictions. *Int J Heat Fluid Flow*, 30(6):1067–1079, 2009.

[311] M Yano and DL Darmofal. Case C1. 3: Flow over the NACA 0012 airfoil: Subsonic inviscid, transonic inviscid, and subsonic laminar flows. In *First International Workshop on High-Order CFD Methods*, 2012. `http://people.ku.edu/~z651w035/hiocfd/abstracts/c13_naca_MIT.pdf`.

[312] M Yano and DL Darmofal. An optimization-based framework for anisotropic simplex mesh adaptation. *J Comput Phys*, 231(22):7626–7649, 2012.

[313] M Yano et al. *An optimization framework for adaptive higher-order discretizations of partial differential equations on anisotropic simplex meshes*. PhD thesis, Massachusetts Institute of Technology, 2012.

[314] M Yano, J Modisette, and D Darmofal. The importance of mesh adaptation for higher-order discretizations of aerodynamic flows. In *20th AIAA Computational Fluid Dynamics Conference, 27-30 June 2011, Honolulu, Hawaii, USA*, 2011. AIAA Paper 2011-3852.

[315] G Yun, D Kim, and H Choi. Vortical structures behind a sphere at subcritical Reynolds numbers. *Phys Fluids*, 18(1):015102, 2006.

[316] MJ Zahr and P-O Persson. An r-adaptive, high-order discontinuous Galerkin method for flows with attached shocks. In *AIAA Scitech 2020 Forum*, page 0537, 2020.

[317] KBMQ Zaman. Effect of initial condition on subsonic jet noise. *AIAA J*, 23(9):1370–1373, 1985.

[318] KBMQ Zaman. Far-field noise of a subsonic jet under controlled excitation. *J Fluid Mech*, 152:83–111, 1985.

[319] S Zeoli, G Balarac, P Bénard, G Georis, F Houtin-Mongrolle, and L Bricteux. Large eddy simulation of wind turbine wakes using adaptative mesh refinement. In *Journal of Physics: Conference Series*, volume 1618, page 062056, 2020.

[320] L-B Zhang. A parallel algorithm for adaptive local refinement of tetrahedral meshes using bisection. *Numer Math Theory Methods Appl*, 2:65–89, 2009.

[321] Z Zhang and C Groth. Parallel high-order anisotropic block-based adaptive mesh refinement finite-volume scheme. In *20th AIAA Computational Fluid Dynamics Conference*, page 3695, 2011.

[322] W Zhao, SH Frankel, and L Mongeau. Large eddy simulations of sound radiation from subsonic turbulent jets. *AIAA J*, 39(8):1469–1477, 2001.

[323] X Zhao, R Conley, N Ray, VS Mahadevan, and X Jiao. Conformal and non-conformal adaptive mesh refinement with hierarchical array-based half-facet data structures. *Procedia Eng*, 124:304–316, 2015.

[324] M Zhu, C Pérez Arroyo, A Fosso Pouangué, M Sanjosé, and S Moreau. Isothermal and heated subsonic jet noise using large eddy simulations on unstructured grids. *Comput Fluids*, 171:166–192, 2018.