Development of a high-order residual distribution method for Navier-Stokes and RANS equations

Dante de Santis

THÈSE

présentée à

L'UNIVERSITÉ BORDEAUX 1

ÉCOLE DOCTORALE DE MATHÉMATIQUES ET INFORMATIQUE

Par **Dante De Santis**

POUR OBTENIR LE GRADE DE

DOCTEUR

SPECIALITÉ: **Mathématiques Appliquées**

DEVELOPMENT OF A HIGH-ORDER RESIDUAL DISTRIBUTION METHOD FOR NAVIER-STOKES AND RANS EQUATIONS

SCHÉMAS D'ORDRE ÈLEVÈ DISTRIBUANT LE RÉSIDU POUR LA RÉSOLUTION DES ÉQUATIONS DE NAVIER-STOKES ET NAVIER-STOKES MOYENNÉES

Préparée à INRIA Bordeaux Sud-Ouest (équipe BACCHUS) - 2013

Directeur de thèse: Rémi Abgrall

Co-directeur de thèse: Mario Ricchiuto

Soutenue le: 3 Décembre 2013

Après avis des rapporteurs:

- Herman Deconinck Professeur, von Karman Institute
- Charbel Farhat Professeur, Stanford University

Devant la commission d'examen composeèe de:

• Rémi Abgrall	Professeur, Université Bordeaux 1 et INRIA	Examinateur
• Mejdi Azaiez	Professeur, ENSCPB,	Président
• Vincent Couailler	Ingénieur-chercheur, ONERA	Examinateur
• Herman Deconinck	Professeur, von Karman Institute	Rapporteur
• Bernhard Müller	Professeur, Norvegian University of Science and Technology	Examinateur
• Mario Ricchiuto	Chargé de recherche, INRIA	Examinateur

a Nilüfer

Abstract

The construction of compact high-order Residual Distribution schemes for the discretization of steady multidimensional advection-diffusion problems on unstructured grids is presented. Linear and non-linear scheme are considered. A piecewise continuous polynomial approximation of the solution is adopted and a gradient reconstruction procedure is used in order to have a continuous representation of both the numerical solution and its gradient. It is shown that the gradient must be reconstructed with the same accuracy of the solution, otherwise the formal accuracy of the numerical scheme is lost in applications in which diffusive effects prevail over the advective ones, and when advection and diffusion are equally important. Then the method is extended to systems of equations, with particular emphasis on the Navier-Stokes and RANS equations. The accuracy, efficiency, and robustness of the implicit RD solver is demonstrated using a variety of challenging aerodynamic test problems.

Keywords: Residual Distribution schemes, High-order methods, Gradient reconstruction, Advection-diffusion problems, Compressible flows, RANS equations, Spalart-Allmaras equation, Implicit methods.

Résumé

Cette thèse présente la construction de schémas distribuant le résidu (RD) d'ordre très élevés, pour la discrétisation d'équations d'advection-diffusion multidimensionnelles et stationnaires sur maillages non structurés. Des schémas linéaires ainsi que des schémas non linéaires sont considérés.

Une approximation de la solution polynomiale par morceaux et continue sur chaque élément est adoptée, de plus une procédure de reconstruction du gradient que celle de la solution numérique est utilisée afin d'avoir une représentation continue de la solution numérique et de son gradient. Il est montré que le gradient doit être reconstruit avec la même précision de la solution, sans quoi la précision formel du schéma numérique est perdue dans les cas où les effets de diffusion prévalent sur les effets d'advection, et aussi quand l'advection et la diffusion sont également importants.

Ensuite, la méthode est étendue à des systèmes d'équations, en particulier aux équations de Navier-Stokes et aux équations RANS. La précision, l'efficacité et la robustesse du solveur RD implicite sont démontrées sur plusieurs cas tests.

Mots-clés: Schéma aux Résidus Distribués, Schéma d'ordre Très Élevé, Reconstruction du gradient, Problèmes d'advection-diffusion, Écoulements compressibles, Équations RANS, Équations de Spalart-Allmaras, Méthodes implicites

Résumé

La dynamique des fluides joue un rôle important dans la conception d'un large spectre d'applications industrielles, qui vont des appareils de cuisine aux sous-marins nucléaires. En particulier, l'industrie aéronautique a toujours profitée des progrès réalisés dans le domaine de la recherche de la dynamique des fluides.

À cause d'un marché compétitif, l'industrie aéronautique doit maintenant livrer des produits qui ont de meilleurs performances avec un coût de conception et réalisation réduit. Cela a produit un changement dans le paradigme de la conception industrielle: la dynamique des fluides computationnelle, ou Computational Fluid Dynamics (CFD), est devenu l'élément principal dans le processus de conception de l'avion en raison de sa flexibilité et moindre coût quand il faut explorer la faisabilité de plusieurs designs alternatifs. Cela peut être intégré aussi dans un environnement multidisciplinaire d'analyse et optimisation.

La complexité de l'écoulement autour d'un avion commercial est telle que réaliser une simulation numérique prédictive est encore très difficile. Afin de simuler de façon fiable les caractéristiques de l'écoulement, il est nécessaire d'utiliser des modèles complexes et calculer des solutions très précises.

Malgré les progrès réalisés en CFD, en terme de temps de calcul et modélisation, des simulations aérodynamiques de l'écoulement turbulent autour des configurations complexes sont encore très coûteux et difficiles. L'obligation à obtenir des résultats avec un niveau de précision suffisant dans un temps de calcul court est toujours une contrainte sévère pour exploiter complètement les avantages de la CFD pour des applications industrielles et pour l'intégration des simulations dans des procédures d'optimisation.

La majorité des outils utilisés pour des simulations CFD dans l'industrie aéronautique est basée sur des méthodes de type volume finis (FV) au deuxième ordre. Lorsque des applications complexes sont considérées, la précision de ces méthodes est dégradée, en obtenant entre un premier ou deuxième ordre en fonction du maillage. Pour obtenir des solutions indépendantes des maillages utilisés, il est nécessaire d'utiliser des mailles très fines avec un grand nombre de degrés de liberté, ce qui génère un temps de calcul trop long.

Les méthodes d'ordre élevé peuvent produire une réduction plus rapide des erreurs de discrétisation avec une augmentation modérée du nombre des degrés de liberté. En fait, en supposant que la méthode numérique est d'ordre $k + 1$, la

diminution d'erreur avec la taille du maillage (h) est $e \propto h^{k+1}$. Cela signifie que si, par exemple, la taille du maillage se réduit, la réduction de l'erreur avec les méthodes d'ordre élevé est plus grande par rapport à des méthodes du premier ou du deuxième ordre. D'autre part, le coût en terme de temps de calcul augmente avec l'ordre des méthodes, mais seulement avec un facteur fixé. Si on veut calculer la solution avec un niveau de précision donné, il y aura un ordre optimal pour lequel les méthodes d'ordre élevé sont plus précises et plus rapides que les méthodes d'ordre inférieures, pour un niveau d'erreur de discrétisation suffisamment faible.

L'ordre optimal de précision est en fait inconnu et dépend des problèmes considérés, mais pour un calcul CFD classique, il est typiquement trois ou quatre.

Une approche innovante pour la solution des écoulements compressibles est représentée par les schémas distribuant le résidu (RD). Dans le cadre de la méthode RD, la solution numérique est supposée être continue, en ayant ainsi des similitudes avec la méthode des éléments finis (FE). L'utilisation d'une approximation continue de la solution garantit que le nombre des degrés de liberté est plus petit par rapport à une approche discontinue. En même temps, la possibilité d'utiliser une approximation d'ordre élevé de la solution est gardée sans la nécessité de recourir à une procédure de reconstruction. En pratique, puisque la solution approchée ne satisfait pas exactement les équations qui régissent l'écoulement, une quantité intégrale (le résidu total) est définie sur chaque élément et est ensuite distribuée à chaque degré de liberté. Le résidu distribué est ensuite utilisé pour faire évoluer la solution discrète.

Malgré les potentialités de l'approche RD, ces méthodes ont été rarement utilisées pour des applications aéronautiques complexes. Les applications ont été limitées à la discrétisation des équations d'Euler au deuxième ordre, sur des maillages avec des triangles (2D) et des tétraèdres (3D). La difficulté majeure, qui a limité l'extension des méthodes RD vient du fait que les schémas RD les plus populaires s'appuient sur des interprétations géométriques. De plus, leur extension aux éléments généraux et aux ordres arbitraires d'approximation de la solution n'est pas claire.

Dans ce travail, le paradigme traditionnel de l'approche multidimensionnelle est abandonné en faveur de méthodes centrales qui sont plus flexibles que les schémas classiques RD multidimensionnelles. L'idée est de récupérer la flexibilité typique des méthodes FE, mais en même temps de construire une classe de systèmes non-linéaires qui permettent une discrétisation simple et sans paramètres des équations avec une approximation stable et non-oscillatoire des discontinuités. L'idée n'est pas nouvelle et a déjà été utilisée avec succès pour la discrétisation d'ordre élevé des équations d'Euler sur des maillages quelconque, mais l'utilisation pour les écoulements visqueux n'était pas claire encore. L'objectif de ce travail est la construction d'une méthode RD robuste, efficace et d'ordre élevé pour la discrétisation des écoulement laminaires et turbulents, sur des maillages arbitraires, en deux et trois dimensions spatiales. Même si l'accent sera mis sur les fluides visqueux, des problèmes non visqueux sont également examinés dans l'hypothèse de gaz parfaits et aussi avec des lois thermodynamiques complexes.

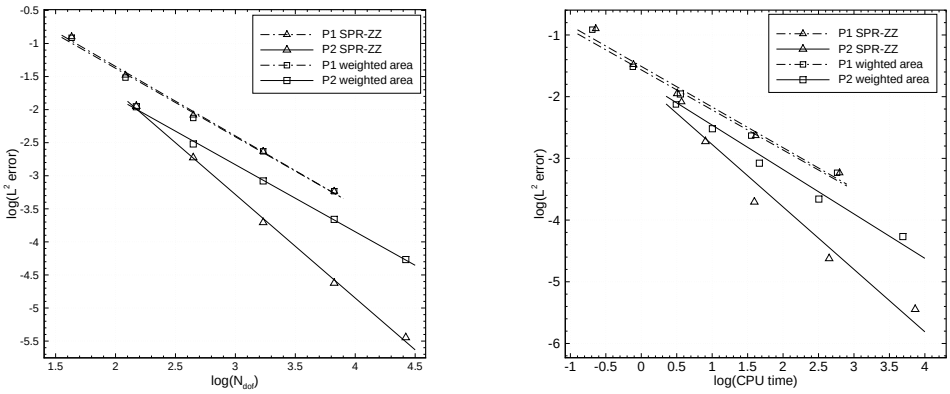


Figure 1: Norme L^2 des erreurs pour la solution du problème d'advection-diffusion linéaire, sur des maillages triangulaires, en fonction du nombre de degrés de liberté et du temps de calcul.

Même si les méthodes RD ont été utilisées pendant longtemps avec succès pour la discrétisation des problèmes d'advection, leur extension à des problèmes d'advection-diffusion n'a pas atteint un niveau complet de compréhension. Le simple couplage d'une méthode RD pour les termes d'advection avec une approche différente pour ceux de diffusion se traduit généralement par un schéma global qui n'est pas optimal, même si les deux schémas séparément sont bons. Dans ce travail, l'advection et la diffusion sont discrétisés avec la même méthode afin de construire un schéma RD efficace et flexible qui conserve une précision optimale pour de différents cas, qui vont de problèmes où l'advection domine la diffusion à des problèmes où la diffusion domine l'advection. Le point clé de cette approche est la reconstruction d'une valeur unique du gradient de la solution numérique pour chaque degré de liberté du maillage, donc l'approximation continue de la solution et de son gradient peut être utilisée pour calculer le résidu total sur chaque élément. La précision du schéma mis en place est illustrée à l'aide de cas test pertinents, voir Figure 1.

En se basant sur l'expérience acquise dans le cas des équations scalaires, la méthode numérique est appliquée aux systèmes d'équations de Navier-Stokes. En plus de la formulation scalaire, la solution du système d'équations nécessite un traitement particulier pour les conditions aux limites. De plus, un solveur implicite qui repose sur l'approche matrix-free est adopté afin d'accélérer la convergence de la solution à l'état stationnaire. La robustesse et la précision du schéma mis en place sont illustrées à l'aide d'un grand nombre de cas tests pertinents, voir Figure 2.

L'approche RD est ensuite appliquée à la solution des équations RANS. Le modèle de turbulence Spalart-Allmaras entièrement couplé avec les équations moyennées de l'écoulement est adopté. L'approche entièrement couplée permet une extension directe de la méthode RD des équations de Navier-Stokes aux équations RANS. De plus, leséquations moyennées de l'écoulement et de la turbulence sont discrétisées avec le même degré de précision. Pour rendre la méthode robuste pour

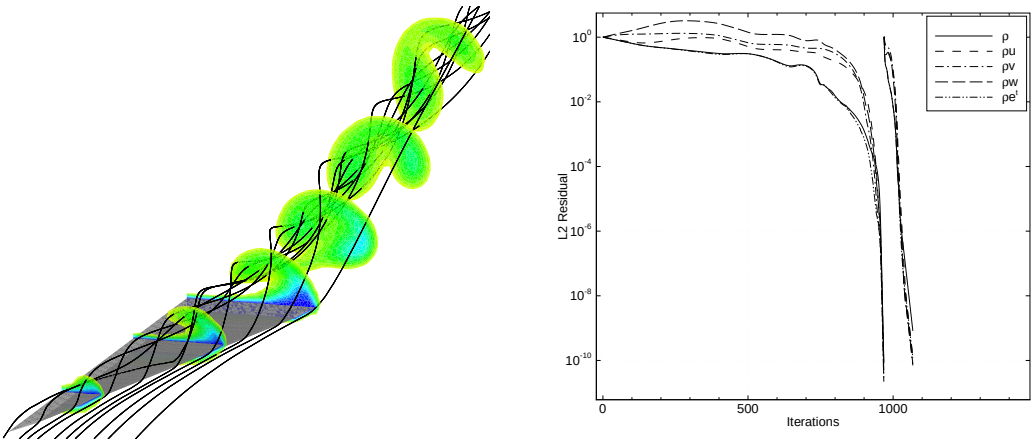
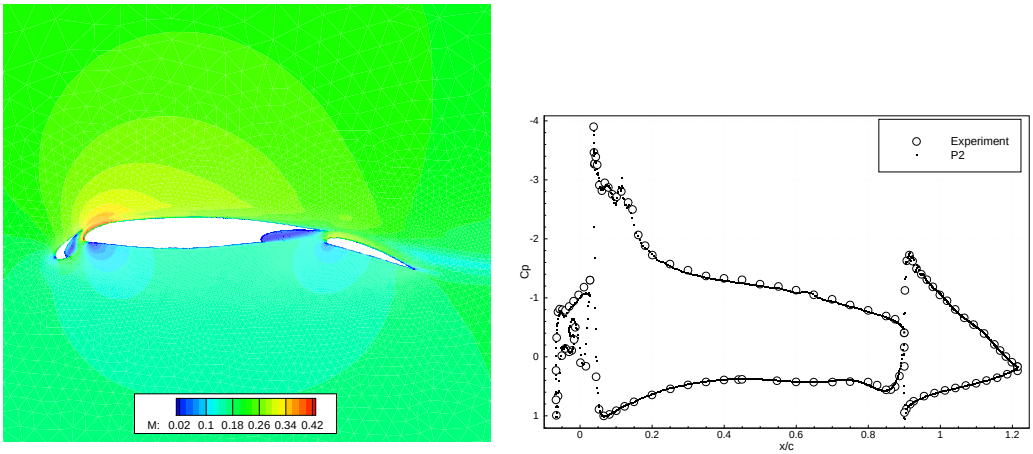


Figure 2: Gauche: Ligne de courant et des isosurface du nombre de Mach le long et derrière l'aile, pour une simulation du troisième ordre. Droite: histoire de convergence pour les simulations du deuxième et troisième ordre.

des applications pratiques, la forme originale de l'équation de Spalart-Allmaras a été modifiée et une méthode LU-SGS non-linéaire est utilisée pour faire converger la solution vers l'état stationnaire. Les résultats numériques montrent que le solveur final est suffisamment robuste pour la simulation transsonique des écoulements turbulents. De plus, malgré la mauvaise régularité du champ de l'écoulement turbulent, il y a des avantages à utiliser des méthodes RD d'ordre élevé pour la simulation des écoulements turbulents. La robustesse et la précision du schéma mis en place sont illustrées à l'aide d'un grand nombre de cas tests pertinents, voir Figure 3.



(a) (b)

Figure 3: Nombre de Mach (a) et coefficient de pression (b) pour le profil multi-élément L1T2, avec des éléments quadratiques.

Acknowledgements

I Would like to acknowledge the support of colleagues, friends and family.

First and foremost, I would like to express my gratitude to my supervisor Prof. Remi Abgrall for guiding me to these studies, for providing me with the most interesting topics, for his invaluable support and consistent consideration. Throughout these years, he has been a consistent and reliable source of guidance and inspiration, he provided me with a truly unique opportunity by letting me space to develop my own ideas and helping me to refine them. I am also grateful to him for the many opportunities I had during my thesis to attend meetings and conferences and interact with other researchers.

I am also grateful to Dr. Mario Ricchiuto who was always available for helpful discussions. I cannot forget the great help of Dr. Pietro M. Congedo, especially during the last part of the thesis. I have particularly appreciated his way to tackle down problems and his unceasing spirit of doing scientific research. I have learned, for sure, a lot from him.

I would like to thank Prof. Herman Deconinck and Prof. Charbel Farhat who agreed to be part of my reading committee, as well as the members of my thesis oral exam committee. This work has been financially supported by the FP7 STREP IDIHOM (contract #265780).

I also wish to thank Dr. Alberto Guardone, my advisor during the master thesis at Politecnico di Milano. I couldn't have started the Ph.D in Bordeaux without his help. I am pleased to acknowledge my friends and colleagues at INRIA, Cédric, Damien, Dragan, Kun Kun and in particular Gianluca, Maria Giovanna, Pietro and Mario with whom I spent a great deal of time outside the lab. Together with Gianluca, we have faced all the good and bad aspects of our job, since the years of the master degree in Italy. I have been lucky to work with him.

I wish to thank my parents for their constant faith in me. They always supported me in my decisions and choices and I greatly appreciate their support. Last but not least, I would like to thank Nilüfer, who has borne the weight of a long distance relationship. I cannot express how much I appreciate the sacrifices you have made for me. I hope the damage is not irreversible.

Contents

Chapter 1

Introduction

1.1 Background

Fluid dynamics plays an important role in the design of a large variety of industrial products, ranging from kitchen appliances to nuclear submarines. In particular, the aeronautical industry has always exploited the progresses made in the fluid dynamic research field.

Due to global competition and ecological pressures, the aeronautical industry now needs to deliver products which are better, faster and cheaper to produce. This has produced a shift in the design paradigm, in which the numerical solution of the governing equations, or Computational Fluid Dynamics (CFD), is the principal element in the aircraft design process because of the flexibility it provides for rapid and comparatively inexpensive evaluation of alternative designs, and because it can be integrated in a simulation environment treating concurrently both multidisciplinary analysis and optimization. Together with the increasing fidelity of mathematical modeling and numerical methods within the fruitful research field of CFD, the role of the wind tunnel is currently changing: it is moving from being a design tool to becoming a tool that validates the designs obtained through the use of CFD. This shift is also favorable because it yields significant savings in terms of both time and budget for wind tunnel experimental campaigns and because it allows the engineer to go into even more detail, *e.g.*, to analyze local features of the flow solution.

The complexity of the flow regime around a commercial airplane is such that performing an accurate and predictive numerical simulation is still a very challenging process. In order to reliably simulate the features of the flow, complicated models and accurate solutions are needed.

The accuracy of numerical solutions is determined from the correspondence between numerical approximation and the exact solution, which is usually unavailable. Numerical solutions of fluid flows generally include several groups of error [75]:

Modeling errors are defined as the difference between the actual flow field and the exact solution of the mathematical model that describes the system in term

of coupled partial differential equations. For example, in the case of laminar flows, modeling errors may be considered negligible for practical purposes, but for turbulent flows, some modeling equations for the turbulence have to be introduce, since the direct simulation of the underlying physic is not affordable yet. The additional models do not always describe the physic of the problem accurately, resulting in modeling errors.

Discretization errors are defined as the difference between the exact (unknown) solution of the system of the governing equations and the approximated solution obtained with a numerical method. Since discretization errors depend on the accuracy of the numerical method and the discretization of the solution domain, they should be reduced considering more accurate numerical schemes and more resolved discretization.

Iteration convergence errors are defined as the difference between the exact and the approximated solution of the discrete system of equations. In fact, the discretization of the original non-linear system of partial equations results in an algebraic system of non-linear equations that must be solved numerically via an iterative method. By computing the residual of the discrete set of equations, it is possible to monitor the evolution of the convergence errors with the number of the iterations. Obviously, in order to be sure that the numerical solution is the correct solution that the numerical method should provide, the iteration convergence errors must be zero or reasonably small.

The use of the CFD for industrial aircraft design started in the early 1970s, and since then has undergone tremendous development which made it a new technology giving important contributions to all stages of the design of an aircraft. The progress made in these years is enormous. Three factors were instrumental in this: a) the increase in available computer resources, b) the progress in development of efficient numerical methods, and c) the progress in physical modeling [129].

Typical applications of CFD, currently underway or planned in the near future are: a) design of the aircraft external shape, e.g. airfoil and wing design, nacelles and body-fairings optimization, b) evaluation of the performance and control database, e.g. stability and control derivatives together with the analysis of stability, control and handling characteristics, c) evaluation of the loads database for the structural design, d) systems integration, e.g. interference between weapons and the airframe or evaluation of the performances of the air conditioning system in the cabin and e) many more, among which aero-servo-elastic analysis. A graphical survey of the typical applications of CFD within today's aeronautical industry is presented in Figure 1.1.

Despite the progress made in CFD, in terms of user time and computational resources, large aerodynamic simulations of turbulent flow around complex aircraft configurations are still very expensive and time consuming. The requirement to achieve results at a sufficient level of accuracy in a short computational time is still

Figure 1.1: Use of CFD in aircraft development in Airbus [63].

a severe constrain to completely exploit the benefits of the CFD in the whole design of industrial aircraft applications and in the integration of the numerical methods in multidisciplinary simulations and optimization procedures.

1.2 The Need for High-Order Methods

The majority of the CFD tools used for aerodynamic simulations in the aeronautical industry are based on second-order Finite Volume (FV) methods [48]. When complex applications are considered the accuracy of these methods is degraded, ranging between first and second-order due to the irregular and highly stretched meshes. As emerged in different editions of the AIAA Drag Prediction Workshop [43, 62, 68], FV methods may not produce sufficiently accurate results on meshes with typical grid sizes used in the industry. High fidelity solutions are also required to separate discretization errors from modeling errors making possible to check the deficits of physical modeling. In order to obtain mesh independent solutions very fine meshes with a very large number of degrees of freedom are required, which lead to enormous computational times.

High-order methods have the potential to achieve a faster reduction of the discretization errors with a moderate increase of the number of the degrees of freedom. In fact, assuming that a numerical method has order $k+1$, the error decrease with the mesh size h according to $e \propto h^{k+1}$. This means that if, for example, the grid size halves, the reduction of the error with high-order methods is larger compared to first or second-order schemes. On the other hand, the cost in term of computational time

increase with the order of the methods, but only with a fixed factor. If one wants to compute the solution with a prescribed level of accuracy, there will be an optimal order for which high-order methods are more accurate and faster than lower-order ones, *for a sufficiently low level of error discretization*. The optimal order of accuracy is actually unknown and depends on the specific problems considered, but for typical CFD computations is in the range of three or four.

Solutions of compressible flows are not globally smooth and this spoils the higher order characteristics in non-smooth regions like shocks or sharp edges, but still high-order methods will be overall more accurate than low-order schemes. Furthermore this problem can be tackled by local mesh adaptation instead of globally finer meshes. Selecting an appropriate order of the method, combined with local mesh adaptation, a higher accuracy can be achieved with similar computational effort compared to classical second-order finite volume schemes.

1.3 High-Order Methods for Unstructured Grids

Although numerical schemes for structured grids may be very simple and extremely efficient, real life applications involves complex geometries that make impossible or extremely difficult the use of structured grids. For this reasons, only high order methods formulated on unstructured grids have the potential to be applied to test cases relevant for industrial applications. In the following a short review of the most common high order methods is given, in addition some very recent schemes are also included.

ENO/WENO Methods

This type of FV methods has its root to the first-order Godunov method [52], which was later extended to second-order by van Leer [125, 126], and to arbitrary order of accuracy by Harten et al. [53], all on structured grids. The basic ideas where extended later by many researcher for ENO [1, 46, 87, 115] and WENO [24, 49, 66, 76] schemes. For a review of ENO and WENO schemes, see [113].

This methods are based on the ideas adopted in the FV approach, but a high order polynomial representation of the solution on each cell is adopted instead of the piecewise constant approximation, used in standard first order FV methods. The higher-order solution is constructed out of the piecewise constant element data of adjacent cells using a reconstruction procedure. The cells, included in the reconstruction, represent the reconstruction stencil of the method. The problem with this approach is that the high-order solution reconstruction involves very large stencils for unstructured grids, thus the resulting scheme is extremely complex to program and very expensive in term of computational time. Moreover, real high-order is achieved only on relatively smooth grids. A further drawback of this methods is that they are not suited for efficient parallelization, due to the large stencil for the

reconstruction.

Continuous Finite Elements Methods

Differently from FV approach, the Finite Element (FE) method is based on the so-called residual discretization. In these schemes the discrete solution is expressed as a linear combination of basis functions which are continuous piecewise polynomials. The basic FE formulation bring to the well known Galerkin scheme, which is however not stable for wave equations and a stabilization mechanism must be added. Many types of stabilizing techniques have been developed to remedy to the stability problem. Examples include the streamline upwind Petrov–Galerkin (SUPG) [57], Galerkin/least squares [58, 65], Taylor Galerkin method [78] amongst many others. Moreover, when discontinuities are present in the solution an additional shock capturing term must be added to guaranties the monotonicity of the solution.

High-order versions of the FE method are easy to construct, since the accuracy of the method depends on the degree of the polynomial basis functions. Consequently, the discretization stencil is compact, because no reconstruction procedure is necessary.

Discontinuous Galerkin Methods

The Discontinuous Galerkin (DG) method combines features of the FV method as well as of the discretization with standard FE. The DG method is based on the Galerkin formulation of the governing equations, but the solution is assumed to be discontinuous between two adjacent elements. The discontinuities of the solution at the faces of the elements are taking into account by numerical fluxes, as done in the FV approach. For a review of the DG method, see [33].

High-order representation of the solution is obtained by using high-order polynomials within elements, thus the degrees of freedom associated with any element are coupled only with those of the neighboring elements sharing a face, making the resulting scheme extremely compact and flexible. This feature makes the DG method ideally suited for parallel computers and it also results in highly sparse matrices in a linearized implicit time integration scheme.

The DG schemes have shown impressive results in the last years, however one of the major drawbacks of this approach is the extremely high number of degrees of freedom introduced and consequently the high computational cost. Moreover, in presence of discontinuities, the high order polynomial approximation produces spurious oscillations in the numerical solution that reduce the benefit of using a high-order method. Some stabilization technique is required to prevent spurious oscillations. One way consists in supplementing the numerical scheme with an artificial viscosity term [54, 95], another way is concerned with the elaboration of a local projection method or slope limiter to enforce the nonlinear stability [22, 34]. However, the accurate and effective discretization of discontinuous solutions with the DG framework is not completely clear and it is still an active area of research.

Spectral Volume, Spectral Difference and Flux Reconstruction Methods

Both Spectral Volume (SV) and Spectral Difference (SD) employ a element-wise discontinuous polynomials as for DG methods. The SV method is similar to a FV method, while the SD method is close to a finite difference method.

In the SV method, the high order representation of the solution is obtained by partitioning each element of the grid into sub-cells, the unknown are the cell-average values on each sub-cell that are used to construct a continuous polynomial representation of the solution within the main element. In this way, SV methods represents an attempt to overcome the problem of the solution reconstruction that characterize standard FV methods. [131–133]. Despite they potential, SV have some of the drawback of the DG method, *i.e.*, high number of degrees of freedom and high computational cost. Furthermore, Fourier analysis revealed the presence of weak instability in several SV partitions of the main elements [44, 45].

The SD method relies on the differential formulation of the governing equations. In this approach two sets of grid points, *i.e.*, the solution points and flux points are defined in each element. [77, 134] The solution points are used to construct a polynomial representation of the solution within the element. Since the solution is discontinuous across the element boundaries, numerical fluxes are computed at flux points along the interface, while the continuous flux function is evaluated at flux points at the interior of the element. Once the fluxes at all the flux points are recomputed, they are used to form polynomial flux representation which is used to express the divergence of the flux at the solution points. The SD method is easier to implement than the DG and SV methods because it does not involve surface or volume integrals, Moreover, the SD method in 1D and for 2D quadrilateral mesh is similar to the staggered-grid multi-domain spectral method by Kopriva et al. [70], therefore the SD method can be viewed as the extension of the staggered-grid multi-domain spectral method to simplexes.

The Flux Reconstruction (FR) approach, originally proposed by Huynh [61] and then extended by Jameson and coworkers [27, 28], share some similarity with the SD method. In the FR method, as in the SD method, the solution points and flux points are defined in each element. The solution points are used to locally construct a polynomial representation of the solution in the element, which is used to evaluate a common interface numerical flux at the element boundaries. Then, correction functions are introduced and added to the discontinuous flux representation in the element. An approximate continuous flux is now available, which is used to evaluate the divergence of the flux at the solution points.

It has been demonstrated that several high-order schemes, including nodal DG schemes and various SD schemes can be cast within the single unifying FR framework [13]. Like the SD method, the FR approach uses the differential form of the governing equations and does not require the calculation of surface or volumes integrals, making it cheaper than the classical DG methods. Although both SD and FR methods have showed to reach high-order accuracy, that make them particularly

suited for aeroacoustic or LES applications, it is not clear what the behavior of these scheme is for non smooth solutions, since they relies on the differential form of the governing equations.

1.4 Motivation and Objectives

An alternative approach to the numerical methods introduced in the previous section is represented by Residual Distribution (RD) schemes. In the RD framework, the numerical solution is assumed to be continuous, thus sharing similarities with the FE method. The use of a continuous approximation of the solution guarantees that the number of degrees of freedom is smaller with respect to a discontinuous approach, but at the same time the possibility to use a high order approximation of the solution is kept without the necessity to use a reconstruction procedure. In short, since the approximated solution does not satisfies the governing equation exactly, an integral quantity, the total residual, is defined on each element and is then distributed to each degree of freedom of elements. The distributed residual is then used to make evolve the discrete solution.

The development of RD methods dates back to pioneering work of Roe [101, 103] in the early eighties, during the attempt to construct truly multidimensional upwind scheme, potentially more accurate than standard FV methods. Since the very first multidimensional upwind schemes developed by Roe [102] and Roe and Sidilkover [104, 114], several research groups contributed to the development of this class of schemes, above all Deconinck and coworkers at von Karman Institute for Fluid Dynamics and Abgrall and coworkers at Université de Bordeaux I, see [99] for a detailed review. The higher accuracy and compact character of RD schemes makes them very efficient when compared to FV schemes [138]. Moreover, the RD approach offers a natural possibility to construct parameter-free, non-oscillatory schemes, leading to a more robust and reliable scheme with respect to FE schemes.

Despite the potentiality of the RD approach, such class of schemes has been rarely used for complex aeronautical applications. Most of the applications are limited to the discretization of the Euler equations up to the second order accuracy, on gird of triangles and tetrahedra, for two and three dimensional simulations, respectively. The main difficulty which limited the extension of RD methods comes from the fact that the most popular RD schemes deeply rely on geometrical interpretations, thus their extension to general elements and to arbitrary orders of approximation of the solution is unclear.

In this work, the traditional paradigm of the multidimensional upwind approach is abandoned if favor of central schemes which are more flexible than the classical multidimensional RD schemes. The idea is to recover the flexibility typical of FE schemes, but at the same time to construct a class of non-linear schemes which allow a simpler and parameter-free discretization of the governing equations with a stable

and non-oscillatory approximation of discontinuities. The idea is not new and has been successfully used for the high order discretization of the Euler equations on arbitrary grids [3], but their use for viscous flows need to be investigated.

The objective of this work is the construction of a robust, efficient and accurate high-order RD solver for the discretization of steady laminar and turbulent flows, on arbitrary grids, in two and three spatial dimensions. Although much of the work will be focus on viscous fluids, simpler inviscid problems are also considered under the hypothesis of ideal gas and complex thermodynamic laws, as well.

In order to apply RD schemes to complex problems that could be relevant for industrial applications, some important issues which have been only marginally considered on previous works need to be analyzed with care. A discussion of the main issues faced in this work is given in the following subsections.

1.4.1 Extension to Advection-Diffusion Problems

Real life applications involve the solution of systems of conservations law containing viscous dissipation terms. The use of RD methods for the discretization of problems with viscous effects has been always very limited. This is due to the fact that is not clear how to handle viscous terms, which have an isotropic behavior, within an upwind mechanism.

The first attempts to discretize advection-diffusion problems with the RD framework used a simple coupling of the RD method for the advective part with a Galerkin discretization of the viscous terms [90, 124]. This approach seems quite logical and is still a common practice for the discretization of the Navier-Stokes equation with RD methods. However, an error analysis [85] reveled that the resulting scheme is only first order accurate in the case in which advection and diffusion have the same importance, and the formal order of accuracy is recovered only in the pure advection or the pure diffusion limits.

Consider, for example, the flow of a viscous fluid over an airfoil. The problem can be considered almost inviscid everywhere except in a small boundary layer over the airfoil, where viscous effects are not negligible but the problem here is non pure diffusive. Since the boundary layer resolution affects directly the values of the friction forces on the wall and the drag force generated, it is extremely important that the accuracy of numerical scheme does not spoil in this region.

A first attempt to remedy to the loss of accuracy for advection-diffusion problems was addressed in [98] where a proper blending of the multidimensional RD method with Galerkin scheme was adopted. However the approach in not very flexible and it is difficult to define a proper blending function for system of equations.

The approach used in this work has been highlighted by Nishikawa and Roe [85], who showed that in order to preserve the accuracy of the numerical scheme, it is important that advection and diffusion should not be discretized using two different schemes, but they have to be discretized within the same RD method. This

approach brings to the definition of a total residual for both the advective and viscous terms, which is then distributed to the degrees of freedom with a RD method. One disadvantage of using this approach is that a gradient reconstruction procedure must be used in order to define an unique value of the gradient of the numerical solution along the face of two adjacent elements. The quality of the gradient reconstruction will in general influence the overall accuracy of the numerical scheme but, on the other hand, the cost and the compactness of the gradient reconstruction procedure plays also an important role in the global efficiency of the numerical scheme. For this reason different gradient reconstruction procedures are considered in this work and their effect on the accuracy of the numerical scheme is studied in detail.

1.4.2 Turbulence Modeling

Turbulent flows represent a complex multi-scale phenomenon which requires extremely small space and time resolutions. The direct simulation of turbulent flows for practical applications is still out of range. A widely use approach for aerodynamic applications in bases on the use of Reynolds Average Navier-Stokes (RANS) equation, in which the small turbulent scales are modeled instead of being solved. RANS equations represent a very challenging application area for high-order methods, due to reduced regularity of the solution and because of the increment of the stiffness of the governing equations.

In this work the single-equation Spalart-Allmaras model is used as a closure model for the turbulent eddy viscosity. Despite the fact that the Spalart-Allmaras model is largely used in the aerodynamic field, this model is know to have spurious oscillations of the turbulent working variable that occur at the edges of the boundary layers and wakes. These oscillations has been usually ignored in the classical FV schemes, however, the presence of oscillations can lead to negative values of the turbulent working variable which impact the stability of high-order numerical schemes.

During the last years, several modification to the original form of the Spalart-Allmaras equations have been introduced in order to alleviate the problem of the spurious oscillations and to increase the robustness of the numerical form. Most of this modifications have been introduced for DG methods, but the same problem has been encounter also in the present work with high order RD schemes. In reference [86] are introduced some modifications to the Spalart-Allmaras equation intended to stabilize the model for negative values of the turbulence model working variable. However, the robustness of the high order discretization is not addressed in reference [86] and some additional modifications to improve the robustness and the iterative convergence of the numerical scheme has been considered by other researchers [23, 40, 79]. In this work some ideas introduced to ameliorate the behavior of the Spalart-Allmaras model for DG schemes have been adopted in the context of the RD methods. These modifications has been found of the paramount importance

in order to construct a robust and reliable numerical solver.

1.4.3 Implicit Schemes

When applying a RD method to the solution of steady problems, the approach leads to the solution of a non-linear system of equations. Usually, a pseudo-time approach is used to make the system evolve to the steady state, with explicit or implicit schemes. Explicit schemes are simpler and less expensive than implicit ones, but they have severe time step limitations due to stability reasons, that make them unattractive for steady state simulations. In fact, since the time dependence is purely artificial, one is not interested to solve accurately the transients but need just to converge as fast as possible to the steady state solution. For this reason the backward Euler method is generally adopted, due to the great margin of stability of this method. At each non-linear iteration, the solution of the resulting linear system is obtained with an approximated method, which in most cases is the preconditioned GMRES, thus a family of inexact Newton-Krylov method is obtained [127].

A key aspect in the construction of implicit schemes is the evaluation of the Jacobian matrix of the residual of the numerical scheme. This matrix, for real life applications, is very large and sparse and its exact calculation may be very complex, if not impossible. The convergence of the scheme to the steady state is largely influenced by the accuracy with which the entries of the Jacobian matrix are computed, as well as the type of preconditioned used. An alternative approach, which avoid the explicit calculation and assembly of the Jacobian matrix, is the so called matrix-free approach that exploits the fact that in the GMRES algorithm the Jacobian matrix is involved only in matrix-vector products, thus it is possible to approximate this operation with a finite difference-like approach. However, for practical applications a preconditioner has to be used, otherwise the scheme will be not able to converge to the steady state. This implies that an approximated version of the Jacobian must be computed anyway, but the convergence of the numerical scheme is expected to be marginally affected by the accuracy of the computed Jacobian since the preconditioning itself is already an approximation of the inverse of the Jacobian matrix. The resulting approach is called Jacobian-free approach [69].

In this work the Jacobian-free approach has been successfully used for the simulation of Euler and Navier-Stokes equations with RD schemes; this approach has showed to be more efficient than the matrix-based GMRES approach in reducing the residual of the numerical scheme to the machine-zero value. Furthermore, the use of the Jacobian-free method is almost mandatory for non-linear RD schemes, for which is impossible to compute the Jacobian of the scheme due to use of non-linear mappings and non-differentiable operations.

Another very attractive implicit scheme is based on the lower-upper symmetric Gauss–Seidel (LU-SGS) approximate factorization scheme, originally developed for structured grids by Jameson and Yoon [64], has been extended and applied to

unstructured grids [112, 112]. The method has been subsequently improved [29] to make its performance comparable with a fully implicit scheme. More recently the approach has been further developed bringing to the so called non-linear LU-SGS method [120]. The advantage of this approach lies in the fact that the method ultimately solve for the original non-linear problem instead of the linearized one, making it very effective, especially for very stiff problems like the RANS equations. Another interesting feature of the non-linear LU-SGS is that only the diagonal blocks of the Jacobian matrix are needed, this results in a huge reduction of memory and computational effort in the calculation of the Jacobian matrix. The non-linear LU-SGS has been used in this wok for the simulations of turbulent flows, this approach resulted to be more robust and effective than Krylov based methods, making possible to reduce the residual of the numerical scheme down to the machine-zero value, even for transonic turbulent flows.

1.4.4 The Contribution of the Thesis

The thesis is an attempt to deal with the issues described in the previous section, by combining some ideas already present in literature, and at the same time proposing new concepts. The overall goal of this work is the extension of high-order RD methods to advection-diffusion scalar problems, and the application of the resulting scheme to the discretization of the steady Navier-Stokes and the RANS equations. Part of the work focuses on the construction of linear and non-linear RD schemes for the high-order discretization of scalar advection-diffusion equations, with particular emphasis on the issue to construct a numerical scheme with an uniform order of accuracy for a wide range of applications, ranging from advection dominated to diffusion dominated problems, including pure advection and pure diffusion problems, as well. The final objective is the construction of an implicit RD method for the numerical simulation of steady laminar and turbulent flows on unstructured grids, in two and three spatial dimensions. In order to test the robustness and the accuracy of the proposed RD solver, several test problems that span a range of complexity and difficulty, from simple inviscid flows to turbulent flows, are examined. The goal of the dissertation is to demonstrate that, with the described strategy, high-order solutions for practical problems can be obtained.

The work presented in the thesis has lead to the following main contributions

- Extension of high-order RD linear and non-linear schemes to the discretization of steady-state advection diffusion problems, with the second and third order spatial accuracy.
- Analysis of accurate gradient reconstruction techniques for the computation of viscous terms in the RD framework, with emphasis on the study of the influence of the gradient reconstruction quality on the accuracy of the numerical scheme. The so called Super-convergent Patch Recovery (SPR) method has been identified as the method of choice for the gradient reconstruction.

In particular the present work indicates that the SPR is recommended for the construction of RD methods applied to the discretization of advection-diffusion problems, with an uniform order of accuracy for both advection dominated and diffusion dominated problems. To the author's knowledge, this is the first time that the SPR approach is used in the discretization of the partial differential equations in the field of the fluid dynamics.

- Extension of the proposed numerical scheme to the discretization of the steady Navier-Stokes and RANS equations on unstructured curved grids, in two and three spatial dimensions, with the second and third order accurate spatial discretization. In particular the issue of the construction of an efficient and robust implicit scheme has been addressed with particular care.

The implementation of the numerical scheme resulted in a parallel numerical code for the discretization of the steady Euler, Navier-Stokes and RANS equations with the Spalart-Allmaras turbulence equation, at the second and third order of accuracy, on unstructured grids in two and three spatial dimensions. The code uses the Jacobian-free or the non-linear LU-SGS methods as implicit solver to accelerate the converged to the steady state. In the most recent development, complex thermodynamic laws have been implemented to simulated also flows with real gas effects.

An embryonic version of the code for the discretization of the Euler equations, with the non-linear RD scheme using a GMRES solver, was already available at the beginning of the thesis. However, due to the necessity to use general high-order grids, implement different non-linear solvers and efficiently implement gradient reconstruction procedures, of the original code only some routines for the partition of the grid and data exchange in parallel simulations have been retained, all the other parts of the code have been written from scratch.

1.5 Thesis Outline

The thesis is organized as follows. The first part of the thesis deals with scalar equations. In particular, Chapter 2 describes the properties of an abstract prototype RD scheme for solving scalar first-order PDEs. The basic steps needed to discretize the continuous problem are presented and the notation used throughout the manuscript is introduces. In Chapter 3 are described in detail the different steps involved in the construction of a RD method for the discretization of scalar advection problems. Moreover, different RD schemes are presented, including both multidimensional upwind schemes and central ones. In Chapter 4, the extension of the RD method to the discretization of second-order PDEs is described, with particular emphasis on the gradient reconstruction procedures for the discretization of the diffusive terms. The effectiveness of the proposed approach is evaluated by the means of numerical

experiments for advection-diffusion and pure diffusion problems.

In the second part of the thesis systems of conservation laws are considered. In particular, in Chapter 5, the numerical method presented for the scalar equation is extended to systems of equations and the problem of the construction of an efficient implicit scheme is addressed. Several results for the Euler and the Navier-Stokes equations are presented. In Chapter 6 the discretization of the RANS equations coupled with the Spalart-Allmaras turbulence model is discussed and the robustness and the accuracy of the numerical scheme is assess against several test cases. In Chapter 7 results obtained in inviscid flows with complex thermodynamic laws are shown. Finally, in Chapter 8 conclusions are drawn.

Related Publications

- R. Abgrall, G. Baurin, A. Krust, D. De Santis and M. Ricchiuto. Numerical approximation of parabolic problems by residual distribution schemes. *International Journal for Numerical Methods in Fluids*, 71(9), 1191–1206, 2013.

- R. Abgrall, D. De Santis and M. Ricchiuto. High-order preserving residual distribution schemes for advection-diffusion scalar problems on arbitrary grids. *SIAM Journal on Scientific Computing*. Accepted, 2013.

- R. Abgrall and D. De Santis. Linear and non-linear high-order accurate residual distribution schemes for the discretization of the steady compressible Navier-Stokes equations. *Journal of Computational Physics*. Under review, 2013.

- R. Abgrall, P.M. Congedo, D. De Santis and N. Razaaly. Residual distribution scheme for real-gas computations. *Computer & Fluids*. Under review, 2013.

- R. Abgrall and D. De Santis. Robust high-order residual distribution schemes for compressible RANS equations on unstructured grids. To be submitted

Related Conference Proceedings

- R. Abgrall and D. De Santis. Very High Order Residual Distribution Schemes for Turbulent Compressible Flows. *19th IMACS World Congress*. San Lorenzo del Escorial, Spain, 26-30 Augustus, 2013.

- R. Abgrall and D. De Santis. High Order Preserving Residual Distribution Schemes for the Laminar and Turbulent Navier-Stokes on Arbitrary Grids.

21st AIAA Computational Fluid Dynamics Conference: 40 Years of CFD. San Diego, California, June 24-27, 2013.

- R. Abgrall, D. De Santis and M. Ricchiuto. High-Order Residual Distribution Scheme for the RANS Equations. ICCFD7-2802, *7th International Conference on Computational Fluid Dynamics (ICCFD7)*. Hawaii, 2012.

- R. Abgrall and D. De Santis. High-Order Residual Distribution Scheme for Navier-Stokes Equations. AIAA Paper 2011-3231, *20th AIAA Computational Fluid Dynamics Conference*. Hawaii, 2011.

Related Conference/Workshop Presentations

- R. Abgrall and D. De Santis. Contribution to *The 2nd International Workshop for High-Order CFD Methods*, Cologne, Germany, May 27-29, 2013.

- R. Abgrall, D. De Santis and M. Ricchiuto. High order residual distribution schemes for laminar and turbulent compressible flows. *European workshop on High Order Nonlinear Numerical Methods for Evolutionary PDEs (HONOM 2013)*. Bordeaux, France, March 18-22, 2013.

- R. Abgrall, D. De Santis and M. Ricchiuto. Very high order residual distribution schemes for laminar and turbulent compressible flows. *Advances in Computational Mechanics*. San Diego, California, February 24-27, 2013.

- R. Abgrall, D. De Santis and M. Ricchiuto. High-Order Residual Distribution Scheme for the RANS Equations. *6th European Congress on Computational Methods in Applied Sciences and Engineering*. Vienna, Austria, 2012.

- R. Abgrall, D. De Santis and M. Ricchiuto. High order residual distribution schemes for laminar and turbulent viscous flows. *International Conference on Spectral and High Order Methods (ICOSAHOM)*, mini-symposium "Towards the application of higher-order unstructured methods in industrial CFD", Tunisia, 2012.

- R. Abgrall, D. De Santis and M. Ricchiuto. Contribution to *The 1st International Workshop for High-Order CFD Methods*, Nashville, TN, January 7-8, 2012.

Part I

Scalar Conservation Laws

Chapter 2

The Residual Distribution Approach

This chapter introduces the RD approach for solving scalar first-order PDEs, as for example scalar advection problems, considering a generic abstract prototype scheme. The objective is to describe the key ideas of the RD approach, and also to introduce the notation used through the whole manuscript. The basic properties of RD schemes are presented and the theoretical accuracy of the method is analyzed here.

2.1 Preliminaries

Consider solving the following steady, scalar conservation law in d space dimensions

$$\boldsymbol{\nabla}\cdot \boldsymbol{f}(u) = 0, \tag{2.1}$$

within an arbitrary domain Ω, where $u(\boldsymbol{x}) \in \mathbb{R}$ is the scalar unknown and $\boldsymbol{f}(u) \in \mathbb{R}^d$ is a given vector flux function, with $\boldsymbol{x} \in \Omega \subset \mathbb{R}^d$ the spatial coordinate. The Eq. (2.1) must be supplemented with the proper boundary conditions on the inflow portion of the boundary $\partial\Omega$

$$u|_{\partial\Omega^-} = g(s), \quad s \in \partial\Omega^-, \tag{2.2}$$

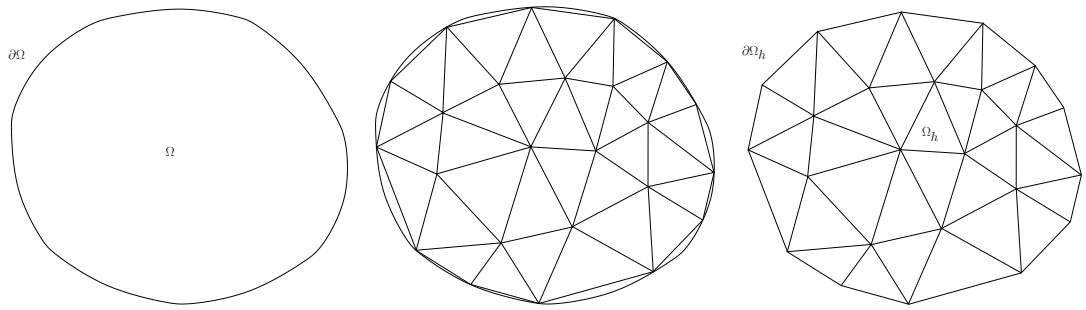
where the given function g represents the boundary condition of the problem on the inflow boundary and $\partial\Omega^- = \{\boldsymbol{x} \in \partial\Omega \,|\, \boldsymbol{a}\cdot\boldsymbol{n} < 0\}$, with $\boldsymbol{n}$ the outward normal vector to the boundary of the domain and $\boldsymbol{a}$ the advection velocity defined as

$$\boldsymbol{a} = \frac{\mathrm{d}\boldsymbol{f}}{\mathrm{d}u}. \tag{2.3}$$

The domain Ω is approximated by the computational domain Ω_h, this is the space filling tessellation composed of the union of E non-overlapping elements Ω_e with characteristic size h. Hence (see Figure 2.1)

$$\Omega \simeq \Omega_h = \bigcup_{e=1}^{E} \Omega_e, \tag{2.4}$$

and the set of all elements is denoted by $\mathcal{E}_h$. Here the elements are assumed to be polyhedra.



(a) (b) (c)

Figure 2.1: An Example of a numerical domain (a), tessellation with triangular elements (b) and the final discrete domain (c).

2.2 Prototype of a RD Scheme

To move to the discrete form of the governing equation, the solution is assumed to be approximated by the continuous piecewise k-th order polynomial $u_h(\boldsymbol{x}) \in V_h$, with

$$V_h = \text{span}\{\psi_i(\boldsymbol{x})\}_{i\in\mathcal{N}_h}, \quad \boldsymbol{x} \in \Omega_h, \tag{2.5}$$

where $\mathcal{N}_h$ is the set of all the degrees of freedom in the numerical domain, and $\psi_i \in \mathbb{P}^k$ is the i-th basis function. Noting that the solution of the discrete problem can be written as a linear combination of the basis functions, one obtains

$$u_h(\boldsymbol{x}) = \sum_{i\in\mathcal{N}_h} \psi_i(\boldsymbol{x})\, u_i, \tag{2.6}$$

with u_i the, still unknown, nodal value of the numerical solution at the generic degree of freedom i. The support of the approximated solution is local, being restricted to few elements that share a common degree of freedom. Furthermore, restricting on the element the same expansion used for the global approximation of solution, the variation of the solution on each element can be expressed as

$$u_h(\boldsymbol{x})|_{\Omega_e} = \sum_{i\in\mathcal{N}_h^e} \psi_i(\boldsymbol{x})\, u_i, \tag{2.7}$$

where $\mathcal{N}_h^e$ is the set of the degrees of freedom of the element e.

It is well known that the numerical solution, generally, does not exactly satisfy the governing equation. This means that if the discrete solution, expressed in the form of the Eq. (2.6), is injected into the governing equation, the right-hand side

of the Eq. (2.1) will be a quantity different from zero. This quantity represents the residual associated with the numerical discretization. In particular, for each element e of the numerical domain, the following integral quantity

$$\Phi^e(u_h) = \int_{\Omega_e} \boldsymbol{\nabla} \cdot \boldsymbol{f}(u_h)\, \mathrm{d}\Omega, \tag{2.8}$$

can be defined, which is called total residual in the RD jargon. The previous expression is nothing but the integral over the element of the governing equation evaluated with the numerical solution, but it assumes an important role in the construction of the RD methods. Obviously, if the exact solution is used in the Eq. (2.8), the total residual is zero by definition.

The total residual is an integral quantity defined for each element, but it cannot be used to write an enough number of equations for all the unknowns of the numerical problem. The next step in the construction of a RD method is the definition of nodal quantities associated with the degrees of freedom of the element. This task is accomplished by redistributing the total residual to all the degrees of freedom of the element. The distribution step might be generically written as follows

$$\Phi_i^e = \beta_i^e(u_h)\Phi^e, \quad \forall\, i \in \mathcal{N}_h^e, \tag{2.9}$$

where $\beta_i^e = \beta_i^e(u_h)$ are called distribution coefficients. It is obvious that the consistency constrain imposes that

$$\sum_{i \in \mathcal{N}_h^e} \Phi_i^e(u_h) = \Phi^e(u_h), \tag{2.10}$$

which implies that the distributions coefficients on each element must sum up to one, namely

$$\sum_{i \in \mathcal{N}_h^e} \beta_i^e(u_h) = 1. \tag{2.11}$$

In this part of the work, the expression of the distribution coefficients is left undetermined to keep the description more general, but it is important to underline already now that the definition of the distribution coefficients is the most important aspect of RD methods. Indeed, it is the distribution process which diversifies the different schemes among the RD methods, and determines the good or the bad properties of the numerical schemes.

The last step in the RD methods consists in requiring that, for each degree of freedom, all the distributed residuals coming from the elements that share that degree of freedom sum up to zero

$$\sum_{e \in \mathcal{E}_h^i} \Phi_i^e(u_h) = 0, \quad \forall\, i \in \mathcal{N}_h, \tag{2.12}$$

where $\mathcal{E}_h^i$ is the set of all the elements that have the degree of freedom i in common. In this way the numerical scheme is conservative in the sense that a variation of

u^h within Ω_h only depends on the fluxes through the boundary of the domain $\partial\Omega_h$. Indeed, summing up Eq. (2.12) for all the degrees of freedom one obtains

$$\sum_{i\in\mathcal{N}_h^e}\sum_{e\in\mathcal{E}_h^i}\Phi_i^e(u_h) = \oint_{\partial\Omega_h}\boldsymbol{f}(u_h)\cdot\hat{\boldsymbol{n}}\,\mathrm{d}\partial\Omega. \tag{2.13}$$

Note that Eq. (2.12) does not take into account the terms arising from the imposition of the boundary conditions, which are neglected in the analysis done in this chapter.

The Eq. (2.12) represents a non-linear system of equations that must be solved for the nodal unknowns u_i, in practice the non-linear system is relaxed as follows

$$|\mathcal{C}_i|\frac{\mathrm{d}u_i}{\mathrm{d}t} + \sum_{e\in\mathcal{E}_h^i}\Phi_i^e(u_h) = 0, \quad \forall\, i\in\mathcal{N}_h, \tag{2.14}$$

where t is a pseudo-time and $|\mathcal{C}_i|$ is the volume of the dual cell around i. The full discrete form of the previous equation can be obtained using the forward Euler method, for example. The final form of the discrete RD scheme reads

$$|\mathcal{C}_i|\frac{u_i^{n+1}-u_i^n}{\Delta t^n} + \sum_{e\in\mathcal{E}_h^i}\Phi_i^e(u_h) = 0, \quad \forall\, i\in\mathcal{N}_h. \tag{2.15}$$

The change of the nodal values of the solution during the iterative process is driven by only non-zero total residuals on the elements; for $n\to\infty$ the total residual on each element vanishes and the steady state solution is obtained.

2.3 Properties of the Numerical Method

In the previous section, only the conservation constrain has been used in the construction of a prototype of the RD method, nothing has been said about accuracy and stability of the numerical scheme. This section is devoted to the analysis of basic properties of the numerical scheme, in order to find the conditions under which the scheme is stable and converges to the correct solution. Furthermore, here, a proof of the theoretical accuracy of the numerical method is given.

2.3.1 Basic Properties

Lax-Wendroff theorem

The classical Lax-Wendroff theorem [74] assures that if a numerical scheme for hyperbolic conservation laws is conservative and it converges, then it converges towards a weak solution of the governing equation. In particular, for RD methods it has been proved that

Theorem 1 *Given a bounded initial data $u_0 \in L^\infty(\mathbb{R}^d)$, a square integrable function $u \in L^2(\mathbb{R}^d \times \mathbb{R}^+)$, and a constant C depending on u_0 and u, such that the approximated solution $u_h(\boldsymbol{x}, t)$ obtained with the numerical scheme (2.14) verifies*

$$\sup_h \sup_{(\boldsymbol{x},t)} |u_h| \leq C \quad \text{and} \quad \lim_{h\to 0} \|u_h - u\|_{L^2_{\text{loc}}(\mathbb{R}^d\times\mathbb{R}^+)} = 0, \tag{2.16}$$

then u is a weak solution of the problem.

See [9] for the proof.

Positivity

Consider now, the properties that must be satisfied by the numerical method in order to have a non-oscillatory solution. To this purpose, the scheme (2.14) is first re-written in the following form

$$|\mathcal{C}_i|\frac{\mathrm{d}u_i}{\mathrm{d}t} + \sum_{e\in\mathcal{E}_h^i} \sum_{\substack{j\in\mathcal{N}_h^e \\ j\neq i}} c_{ij}^e (u_i - u_j) = 0, \quad \forall\, i \in \mathcal{N}_h. \tag{2.17}$$

The first property considered is the so-called Local Extrema Diminishing property, which guarantees that local extrema are kept bounded by the numerical scheme. Rearranging Eq. (2.17) as follows

$$\begin{aligned} |\mathcal{C}_i|\frac{\mathrm{d}u_i}{\mathrm{d}t} &= -\sum_{e\in\mathcal{E}_h^i} \sum_{\substack{j\in\mathcal{N}_h^e \\ j\neq i}} c_{ij}^e (u_i - u_j) \\ &= -\sum_{\substack{j\in\mathcal{N}_h^i \\ j\neq i}} \left(\sum_{e\in\mathcal{E}_h^i\cap\mathcal{E}_h^j} c_{ij}^e \right) (u_i - u_j) = -\sum_{\substack{j\in\mathcal{N}_h^i \\ j\neq i}} \tilde{c}_{ij}(u_i - u_j), \end{aligned} \tag{2.18}$$

with $\mathcal{N}_h^i$ the set of the degrees of freedom contained in the stencil of i, it is easy to see that, if

$$\tilde{c}_{ij} = \sum_{e\in\mathcal{E}_h^i\cap\mathcal{E}_h^j} c_{ij}^e \geq 0, \quad \forall\, i \in \mathcal{N}_h,\ \forall\, j \in \mathcal{N}_h^i,\ j \neq i, \tag{2.19}$$

the variation of u_i is non-positive if u_i is a local maximum ($u_i \geq u_j$), and is non-negative if u_i is a local minimum ($u_i \leq u_j$). A stronger constrain can obtained by requiring that each c_{ij}^e is positive on every element e (local positivity).

Maximum principle

Consider, now, the fully discrete version of the Eq. (2.17), namely

$$|\mathcal{C}_i|\frac{u_i^{n+1} - u_i^n}{\Delta t} + \sum_{e\in\mathcal{E}_h^i} \sum_{\substack{j\in\mathcal{N}_h^e \\ j\neq i}} c_{ij}^e (u_i^n - u_j^n) = 0, \quad \forall\, i \in \mathcal{N}_h, \tag{2.20}$$

where an explicit scheme has been used for the time discretization. It easy to verify that the previous scheme satisfies the following maximum principle

$$\min_{j\in\mathcal{N}_h^i} u_j^n \leq u_i^{n+1} \leq \max_{j\in\mathcal{N}_h^i} u_j^n \tag{2.21}$$

if the Local Extrema Diminishing property (2.19) holds and the following time step restriction is kept

$$\Delta t \leq \frac{|\mathcal{C}_i|}{\sum\limits_{\substack{j\in\mathcal{N}_h^i \\ j\neq i}} \tilde{c}_{ij}}, \quad \forall i \in \mathcal{N}_h. \tag{2.22}$$

Indeed, by writing

$$\begin{aligned} u_i^{n+1} &= u_i^n - \frac{\Delta t}{|\mathcal{C}_i|} \sum_{\substack{j\in\mathcal{N}_h^i \\ j\neq i}} \tilde{c}_{ij}(u_i^n - u_j^n) \\ &= \left(1 - \frac{\Delta t}{|\mathcal{C}_i|} \sum_{\substack{j\in\mathcal{N}_h^i \\ j\neq i}} \tilde{c}_{ij}\right) u_i^n + \frac{\Delta t}{|\mathcal{C}_i|} \sum_{\substack{j\in\mathcal{N}_h^i \\ j\neq i}} \tilde{c}_{ij} u_j^n \\ &= \bar{c}_{ii} u_i^n + \sum_{\substack{j\in\mathcal{N}_h^i \\ j\neq i}} \bar{c}_{ij} u_j^n = \sum_{j\in\mathcal{N}_h^i} \bar{c}_{ij} u_j^n, \end{aligned} \tag{2.23}$$

since, property (2.19), together with the condition (2.22), guarantees that $\bar{c}_{ij} \geq 0,\ \forall\, i, j$, hence

$$\left(\sum_{j\in\mathcal{N}_h^i} \bar{c}_{ij}\right) \min_{j\in\mathcal{N}_h^i} u_j^n \leq u_i^{n+1} \leq \left(\sum_{j\in\mathcal{N}_h^i} \bar{c}_{ij}\right) \max_{j\in\mathcal{N}_h^i} u_j^n. \tag{2.24}$$

By using the trivial relation $\sum_{j\in\mathcal{N}_h^i} \bar{c}_{ij} = 1$, one obtains the condition (2.21). For steady problems the following version of the maximum principle is more appropriate

$$\min_{j\in\mathcal{N}_h^i} \left(\lim_{n\to\infty} u_j^n\right) \leq \lim_{n\to\infty} u_i^n \leq \max_{j\in\mathcal{N}_h^i} \left(\lim_{n\to\infty} u_j^n\right). \tag{2.25}$$

A numerical scheme which satisfies the maximum principle is said to be positive.

A consequence of the maximum principle is that the numerical scheme is L^∞-stable and the following bounds hold for the numerical solution

$$\min_{i\in\mathcal{N}_h} u_i^0 \leq u_j^n \leq \max_{i\in\mathcal{N}_h} u_i^0, \quad \forall\, i \in \mathcal{N}_n,\ \forall\, n. \tag{2.26}$$

2.3.2 Accuracy and Order Preservation

In this section an error estimation for a generic RD scheme is given. It is important to note that the following results are valid only for a solution at the steady state

and they are not generalizable to unsteady problems. In addition, it is also given the condition that must be fulfilled by a RD scheme in order to get a numerical method which is able to maintain the theoretical order of accuracy.

Consider the numerical scheme (2.14) at the steady state and analyze the following expression

$$\sum_{i\in\mathcal{N}_h} \varphi_i(\boldsymbol{x}) \sum_{e\in\mathcal{E}_h^i} \Phi_i^e(u_h) = 0, \tag{2.27}$$

for any smooth, compact, function $\varphi(\boldsymbol{x})$, with $\varphi_i = \varphi(\boldsymbol{x}_i)$. The continuous piecewise polynomial approximation of φ is given by

$$\varphi_h(\boldsymbol{x}) = \sum_{i\in\mathcal{N}_h} \varphi_i\, \psi_i(\boldsymbol{x}), \tag{2.28}$$

with

$$\|\varphi_h\|_{L^\infty(\Omega)} \leq \|\varphi\|_{L^\infty(\Omega)} < \infty. \tag{2.29}$$

The Eq. (2.27) can be rewritten as follows

$$\begin{aligned} 0 &= \sum_{i\in\mathcal{N}_h} \varphi_i \sum_{e\in\mathcal{E}_h^i} \Phi_i^e(u_h) = \sum_{e\in\mathcal{E}_h} \sum_{i\in\mathcal{N}_h^e} \varphi_i \Phi_i^e(u_h) \\ &= \int_\Omega \varphi_h \boldsymbol{\nabla\cdot}\, \boldsymbol{f}(u_h)\, \mathrm{d}\Omega + \sum_{e\in\mathcal{E}_h} \sum_{i\in\mathcal{N}_h^e} \varphi_i \left(\Phi_i^e(u_h) - \Phi_i^{e,G}(u_h)\right), \end{aligned} \tag{2.30}$$

where $\Phi_i^{e,G}(u_h)$ is the so-called Galerkin residual

$$\Phi_i^{e,G}(u_h) = \int_{\Omega_e} \psi_i \boldsymbol{\nabla\cdot}\, \boldsymbol{f}(u_h)\, \mathrm{d}\Omega. \tag{2.31}$$

By using the basic relation of the basis functions

$$\sum_{i\in\mathcal{N}_h^e} \psi_i = 1, \tag{2.32}$$

one can easily obtain the following identity

$$\sum_{i\in\mathcal{N}_h^e} \Phi_i^e(u_h) - \sum_{i\in\mathcal{N}_h^e} \Phi_i^{e,G}(u_h) = \Phi^e(u_h) - \Phi^e(u_h) = 0 \tag{2.33}$$

and hence $\forall j \in \mathcal{N}_h^e$

$$\sum_{e\in\mathcal{E}_h} \sum_{i\in\mathcal{N}_h^e} \varphi_i \left(\Phi_i^e(u_h) - \Phi_i^{e,G}(u_h)\right) = \sum_{e\in\mathcal{E}_h} \sum_{i\in\mathcal{N}_h^e} (\varphi_i - \varphi_j) \left(\Phi_i^e(u_h) - \Phi_i^{e,G}(u_h)\right). \tag{2.34}$$

Thanks to the previous relation, it is possible to rewrite Eq. (2.30) as follows

$$\begin{aligned} \sum_{i\in\mathcal{N}_h} \varphi_i \sum_{e\in\mathcal{E}_h^i} \Phi_i^e(u_h) = &\int_\Omega \varphi_h \boldsymbol{\nabla\cdot}\, \boldsymbol{f}(u_h)\, \mathrm{d}\Omega \\ &+ \frac{1}{C} \sum_{e\in\mathcal{E}_h} \sum_{i,j\in\mathcal{N}_h^e} (\varphi_i - \varphi_j) \left(\Phi_i^e(u_h) - \Phi_i^{e,G}(u_h)\right), \end{aligned} \tag{2.35}$$

where C is the number of possible combinations of i and j in the expression (2.34). If $\boldsymbol{f}(u_h)$ is a smooth function and a $(k+1)$-th order accurate approximation of $\boldsymbol{f}(u)$, and using the fact that $\varphi \in C_0^1(\Omega)$, one gets

$$\int_{\Omega_e} \varphi_h \boldsymbol{\nabla\cdot}\, \boldsymbol{f}(u_h)\,\mathrm{d}\Omega = \int_{\Omega_e} \varphi \boldsymbol{\nabla\cdot}\, \boldsymbol{f}(u)\,\mathrm{d}\Omega + \mathcal{O}\big(h^{k+1}\big), \tag{2.36}$$

and hence

$$\begin{aligned}\sum_{i\in\mathcal{N}_h} \varphi_i \sum_{e\in\mathcal{E}_h^i} \Phi_i^e(u_h) = &\int_\Omega \varphi \boldsymbol{\nabla\cdot}\, \boldsymbol{f}(u)\,\mathrm{d}\Omega \\ &+ \frac{1}{C}\sum_{e\in\mathcal{E}_h}\sum_{i,j\in\mathcal{N}_h^e} (\varphi_i-\varphi_j)\left(\Phi_i^e(u_h) - \Phi_i^{e,\,G}(u_h)\right) + \mathcal{O}\big(h^{k+1}\big).\end{aligned} \tag{2.37}$$

On the other hand, for a regular solution

$$\begin{aligned}\Phi_i^{e,\,G}(u_h) &= \oint_{\partial\Omega_e} \psi\big(\boldsymbol{f}(u_h)-\boldsymbol{f}(u)\big)\cdot\hat{\boldsymbol{n}}\,\mathrm{d}\partial\Omega - \int_{\Omega_e} \boldsymbol{\nabla}\psi\cdot\big(\boldsymbol{f}(u_h)-\boldsymbol{f}(u)\big)\,\mathrm{d}\Omega \\ &= \mathcal{O}(h^{k+d}) + \mathcal{O}(h^{k+d}) = \mathcal{O}(h^{k+d}),\end{aligned} \tag{2.38}$$

having used the fact that $|\partial\Omega_e| = \mathcal{O}(h^{d-1})$ and $|\Omega_e| = \mathcal{O}(h^d)$.

Since $\Phi_i^{e,\,G}(u_h) = \mathcal{O}(h^{k+d})$, the number of elements is $\mathcal{O}(h^{-d})$, and remembering that $\varphi \in C_0^1(\Omega)$, with $\varphi_i - \varphi_j = \mathcal{O}(h)$, one can prove that, if $\Phi_i^e(u_h) = \mathcal{O}(h^{k+d})$, then the following error estimation is true

$$\left|\sum_{i\in\mathcal{N}_h} \varphi_i \sum_{e\in\mathcal{E}_h^i} \Phi_i^e(u_h) - \int_\Omega \varphi \boldsymbol{\nabla\cdot}\, \boldsymbol{f}(u)\,\mathrm{d}\Omega\right| \le \tilde{C}\,\|\boldsymbol{\nabla}\varphi\|_{L^\infty(\Omega)}\,h^{k+1}, \tag{2.39}$$

see [11] for more details. The previous error estimation leads to the following important condition

Definition 1 *Given a continuous $(k+1)$-th order accurate approximation of the flux function, $\boldsymbol{f}(u)$, in the case of d spatial dimensions, a necessary condition for a RD scheme of the form (2.17) to be $(k+1)$-th order accurate, at the steady state, is that*

$$\Phi_i^e(u_h) = \mathcal{O}(h^{k+d}). \tag{2.40}$$

Note that, for a continuous k-th order approximation of the solution, at the steady limit, one has the following approximation

$$\Phi^e(u_h) = \int_{\Omega_e} \boldsymbol{\nabla\cdot}\, \boldsymbol{f}(u_h)\,\mathrm{d}\Omega = \mathcal{O}(h^{k+d}). \tag{2.41}$$

This allows to introduce the following property

Definition 2 *A RD scheme of the type (2.17), with the distributed residuals expressed as* $\Phi_i^e(u_h) = \beta_i^e \Phi^e(u_h)$*, such that*

$$\sum_{i \in \mathcal{N}_h^e} \beta_i^e(u_h) = 1, \tag{2.42}$$

and that verifies the Definition 1, is said to be order preserving if the distribution coefficients are bounded independently on the solution, the mesh and the initial data. Namely

$$\max_{e \in \mathcal{E}_h} \max_{i \in \mathcal{N}_h^e} |\beta_i^e| < \infty. \tag{2.43}$$

Historically, this condition is also called Linearity Preserving.

2.3.3 Godunov's Theorem

A RD scheme of the type (2.17) is said to be linear if all the coefficients c_{ij}^e are independent on the numerical solution, on the other hand a RD scheme for which the coefficients c_{ij}^e, and so the distribution coefficients β_i^e, depend on the solution, is said non-linear.

RD schemes obey the classical Godunov's theorem, that means that a linear scheme cannot be more than first order accurate, remaining at the same time monotone near discontinuities. The construction of a high-order RD scheme which is at the same time positive and Linearity Preserving, requires that the distribution process must depend itself on the numerical solution, making the scheme non-linear.

Chapter 3

RD Schemes for Multidimensional First-Order PDEs on Unstructured Grids

In this chapter the RD method is described from a practical point of view, with particular emphasis on the way how actually performing the distribution process. Details about the interpolation of the numerical solution and the calculation of the total residual over the elements are given first, then different procedures to distribute the residual to the degrees of freedom are described.

Two family of schemes are presented: multidimensional upwind and central schemes. The first group includes schemes developed in order to extend to more spatial dimensions the upwind property; while in one spatial dimension the upwind mechanism can be easily construct, for two or three dimensional problems it is more complicated to include a truly upwind property in the numerical scheme. The second group includes schemes for which the truly upwind property is relaxed in favor of a more flexibility. Finally, a class of non-linear schemes is introduced with the aim to construct high-order and monotone schemes, overcoming the barrier of the Godunov's theorem for linear schemes.

3.1 Basic Elements of the Numerical Scheme

In the RD method here formulated, the solution is assumed to be continuous and is expressed as a linear combination of basis functions, as in Eq. (2.6). This implies that the basis functions, used to express the numerical solution, must be continuous over the computational domain.

In general, the basis is associated to a reference element of standard, simple shape, and then transformed to the basis of the real elements of the computational domain by a suitable coordinate transformation. To enforce the continuity of the approximation, the basis functions can be chosen such that the continuity condition

is automatically satisfied. For instance, using Lagrangian functions [1] with the degrees of freedom on the element suitably located, the continuity of the solution is always satisfied, as it is well know from the finite element theory.

3.1.1 Basis Functions

The expressions of the basis functions on the reference elements are here recalled, for sake of completeness, for different types of elements considered in this work. As standard practice, the basis functions are first formulated on standard, reference elements, and then transformed into the physical element.

Segments

The reference segment is defined as $\{\xi : 0 \leq \xi \leq 1\}$. For a segment of degree k, two degrees of freedom are located at the extremes of the element and for the remaining $k-1$ degrees of freedom, a set of equally-spaced nodes is used, for $k > 1$, (Figure 3.1). The expression of a basis function, associated with the degree of freedom i, reads

$$\psi_i^k(\xi) = \frac{\prod_{\substack{j=1 \\ j\neq i}}^{k+1}(\xi - \xi_j)}{\prod_{\substack{j=1 \\ j\neq i}}^{k+1}(\xi_i - \xi_j)}, \quad i = 1, \ldots, k+1, \tag{3.1}$$

where ξ_j is the coordinate of the degree of freedom j.

(a) (b)

Figure 3.1: Reference segment element for linear (a) and quadratic (b) approximations.

Triangles

The reference triangle is defined as $\{\xi, \eta : 0 \leq \xi \leq 1, 0 \leq \eta \leq 1 - \xi\}$, and for a basis of degree k, the number of degrees of freedom of the element is

$$N = \frac{(k+1)(k+2)}{2}. \tag{3.2}$$

The first three degrees of freedom are located at the vertices of the triangles, the other $3(k-1)$ degrees of freedom are located on the faces of the element with an equally-spaces distribution, and the remaining degrees of freedom are symmetrically distributed around the barycenter of the element, (Figure 3.2).

[1] In this work, only standard Lagrangian elements are used; an example of RD method using non-Lagrangian elements is reported in [12].

For this element the expression of basis functions becomes easier if barycentric coordinates are used instead of the classical Cartesian ones. The barycentric coordinates for the reference triangle are defined as follows

$$\lambda_1 = 1 - \xi - \eta, \quad \lambda_2 = \xi, \quad \lambda_3 = \eta, \tag{3.3}$$

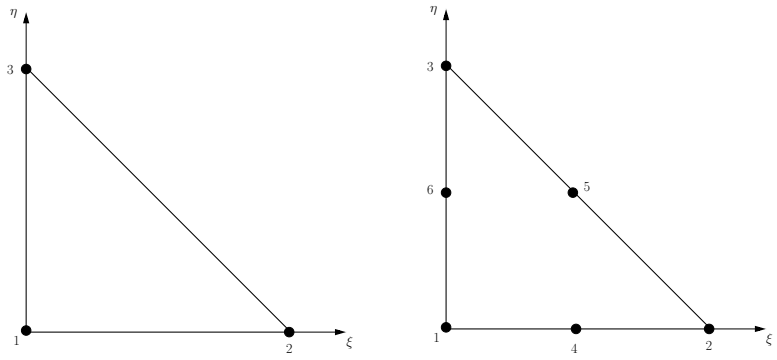
and the linear basis functions on the reference triangle simply read

$$\psi_i^1(\lambda_1, \lambda_2, \lambda_3) = \lambda_i, \quad i = 1, 2, 3. \tag{3.4}$$

For quadratic triangles, it is easy to verify that the basis functions for the reference element are given by

$$\begin{aligned} \psi_i^2(\lambda_1, \lambda_2, \lambda_3) &= (2\lambda_i - 1)\lambda_i, \quad && i = 1, 2, 3 \\ \psi_i^2(\lambda_1, \lambda_2, \lambda_3) &= 4\lambda_m \lambda_n, && i = 4, 5, 6 \end{aligned} \tag{3.5}$$

with m and n the extreme nodes of the segment which contains the internal node i.



(a) (b)

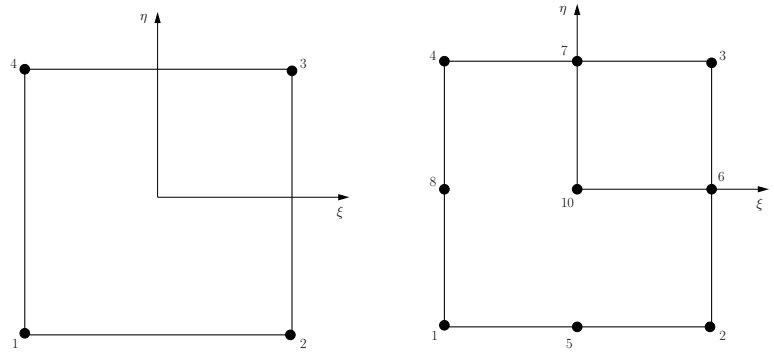
Figure 3.2: Reference triangular element for linear (a) and quadratic (b) approximations.

Quadrangles

The nodes and the basis functions for quadrilateral elements are obtained by the tensor product of the one-dimensional ones, thus the total number of degrees of freedom, for a basis of degree k, is $N = (k+1)^2$. The reference element is defined as $\{\xi, \eta : -1 \leq \xi, \eta \leq 1\}$ (Figure 3.3), and the basis functions are given by the following expressions

$$\begin{aligned} \psi_i^k(\xi, \eta) = \frac{1}{4} \bar{\psi}_m^k(\xi) \bar{\psi}_n^k(\eta), \qquad & i = 1, \ldots, (k+1)^2, \\ & m, n = 1, \ldots, k+1, \end{aligned} \tag{3.6}$$

with $\bar{\psi}^k$ the basis functions of the one-dimensional segment, as defined previously.



(a) (b)

Figure 3.3: Reference triangular element for linear (a) and quadratic (b) cubic approximations.

Tetrahedra

The reference tetrahedron is defined as an element in the three dimensional space such that $\{\xi, \eta, \zeta : 0 \leq \xi \leq 1, 0 \leq \eta \leq 1 - \xi, 0 \leq \zeta \leq 1 - \xi - \eta\}$, and for a basis of degree k, the number of degrees of freedom of the element is

$$N = \frac{(k+1)(k+2)(k+3)}{6}. \tag{3.7}$$

The first four degrees of freedom are located at the vertices of the tetrahedron, the degrees of freedom on the faces are located in the same way seen for triangles, and the remaining degrees of freedom are symmetrically distributed around the barycenter of the element, (Figure 3.4) Introducing the barycentric coordinates for the reference tetrahedron

$$\lambda_1 = 1 - \xi - \eta - \zeta, \quad \lambda_2 = \xi, \quad \lambda_3 = \eta \quad \lambda_4 = \zeta, \tag{3.8}$$

the basis functions become analogous to the expressions shown for the triangles. In the case of a linear element, the basis functions simply read

$$\psi_i^1(\lambda_1, \lambda_2, \lambda_3, \lambda_4) = \lambda_i, \quad i = 1, \ldots, 4. \tag{3.9}$$

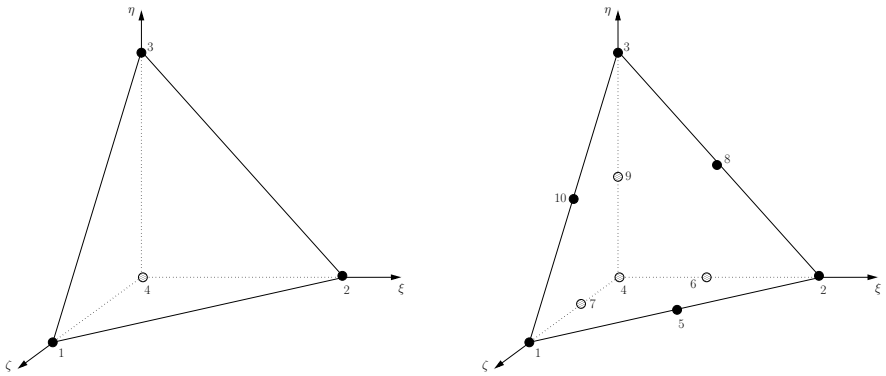
For a quadratic tetrahedron, the basis functions on the reference element are given by

$$\begin{aligned} \psi_i^2(\lambda_1, \lambda_2, \lambda_3, \lambda_4) &= (2\lambda_i - 1)\lambda_i, \quad && i = 1, \ldots, 4 \\ \psi_i^2(\lambda_1, \lambda_2, \lambda_3, \lambda_4) &= 4\lambda_m \lambda_n, && i = 5, \ldots, 10 \end{aligned} \tag{3.10}$$

with m, n the extreme nodes of the segment which contains the node i.

Hexahedra

The nodes, and the basis functions, for hexahedral elements are obtained by tensor product of the one-dimensional ones, thus the total number of degrees of freedom,



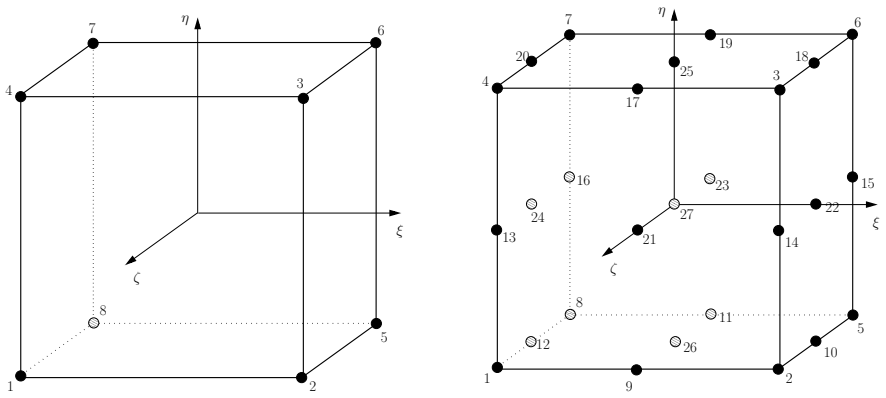
(a) (b)

Figure 3.4: Reference tetrahedral element for linear (a) and quadratic (b) approximations.

for a basis of degree k, is $N = (k+1)^3$. The reference element is defined as $\{\xi, \eta, \zeta : -1 \leq \xi, \eta, \zeta \leq 1\}$, (Figure 3.5), and the basis functions are given by the following expression

$$\psi_i^k(\xi,\eta,\zeta) = \frac{1}{8}\bar{\psi}_l^k(\xi)\bar{\psi}_m^k(\eta)\bar{\psi}_n^k(\zeta), \qquad \begin{matrix} i = 1,\ldots,(k+1)^3, \\ l,m,n = 1,\ldots,k+1, \end{matrix} \tag{3.11}$$

with $\bar{\psi}^k$ the basis function of the one-dimensional segment.



(a) (b)

Figure 3.5: Reference hexahedral element for linear (a) and quadratic (b) approximations.

3.1.2 Element Mappings

The connection between the reference element, $\hat{\Omega}$, and the physical one, Ω_e, is done by means of a mapping, $\mathcal{J} : \hat{\Omega} \to \Omega_e$. Depending on the choice of $\mathcal{J}$, the map can account for elements with curved edges/faces, Figure 3.6. It is assumed that i) the

Jacobian of $\mathcal{J}$ is not singular, *ii*) $\mathcal{J}$ is sufficiently regular and *iii*) $\mathcal{J}$ is a bijection between $\hat{\Omega}$ and Ω_e

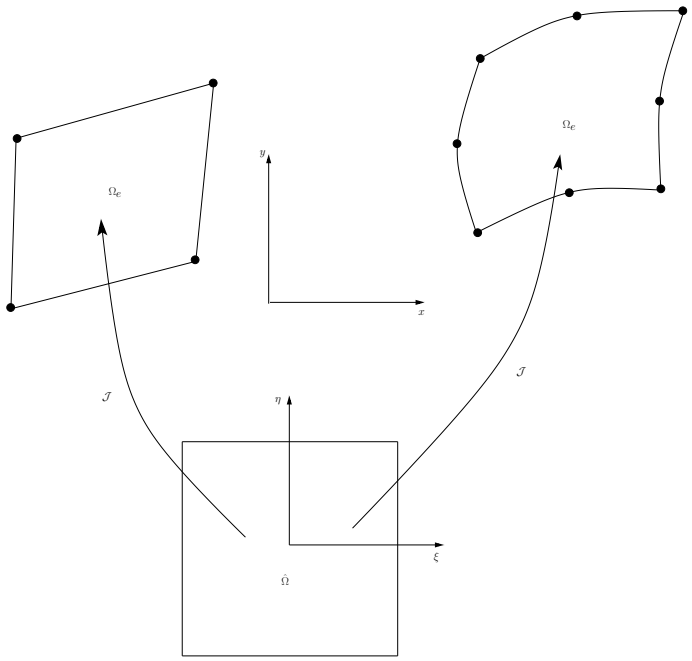


Figure 3.6: Example of mapping from the reference element to linear and quadratic elements.

Consider a generic three dimensional element, the mapping from the reference element in (ξ, η, ζ) to the physical element in (x, y, z), is given by

$$\begin{aligned} x(\xi,\eta,\zeta) &= \sum_{j=1}^{N_g} \hat{\psi}_j(\xi,\eta,\zeta)\, x_j, \\ y(\xi,\eta,\zeta) &= \sum_{j=1}^{N_g} \hat{\psi}_j(\xi,\eta,\zeta)\, y_j, \\ z(\xi,\eta,\zeta) &= \sum_{j=1}^{N_g} \hat{\psi}_j(\xi,\eta,\zeta)\, y_j, \end{aligned} \tag{3.12}$$

where N_g is the number of geometrical nodes of the element, $\hat{\psi}_j$ is the shape function for the node j and x_i, y_j, z_j are the node coordinates. The Jacobian of the

transformation is given by

$$J = \begin{pmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} & \frac{\partial z}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} & \frac{\partial z}{\partial \eta} \\ \frac{\partial x}{\partial \zeta} & \frac{\partial y}{\partial \zeta} & \frac{\partial z}{\partial \zeta} \end{pmatrix} = \begin{pmatrix} \sum_{j=1}^{N_g} \frac{\partial \hat{\psi}_j}{\partial \xi} x_j & \sum_{j=1}^{N_g} \frac{\partial \hat{\psi}_j}{\partial \xi} y_j & \sum_{j=1}^{N_g} \frac{\partial \hat{\psi}_j}{\partial \xi} z_j \\ \sum_{j=1}^{N_g} \frac{\partial \hat{\psi}_j}{\partial \eta} x_j & \sum_{j=1}^{N_g} \frac{\partial \hat{\psi}_j}{\partial \eta} y_j & \sum_{j=1}^{N_g} \frac{\partial \hat{\psi}_j}{\partial \eta} z_j \\ \sum_{j=1}^{N_g} \frac{\partial \hat{\psi}_j}{\partial \zeta} x_j & \sum_{j=1}^{N_g} \frac{\partial \hat{\psi}_j}{\partial \zeta} y_j & \sum_{j=1}^{N_g} \frac{\partial \hat{\psi}_j}{\partial \zeta} z_j \end{pmatrix}. \tag{3.13}$$

Derivation

Derivatives of the solution over the element e can be computed through the use of the derivatives of the shape functions of the element, namely

$$\begin{aligned} \frac{\partial u^h}{\partial x} &= \sum_{i \in \mathcal{N}_h^e} \frac{\partial \psi_i}{\partial x} u_i, \\ \frac{\partial u^h}{\partial y} &= \sum_{i \in \mathcal{N}_h^e} \frac{\partial \psi_i}{\partial y} u_i, \\ \frac{\partial u^h}{\partial z} &= \sum_{i \in \mathcal{N}_h^e} \frac{\partial \psi_i}{\partial z} u_i. \end{aligned} \tag{3.14}$$

The derivatives of the basis functions on the physical space can be easily computed from the derivatives of the basis functions on the reference element. By the usual rules of partial differentiation, the first component of the derivatives reads

$$\frac{\partial \psi_i}{\partial \xi} = \frac{\partial \psi_i}{\partial x}\frac{\partial x}{\partial \xi} + \frac{\partial \psi_i}{\partial y}\frac{\partial y}{\partial \xi} + \frac{\partial \psi_i}{\partial z}\frac{\partial z}{\partial \xi}. \tag{3.15}$$

Performing the same differentiation with respect to the other coordinates and using the matrix form, one obtains

$$\begin{pmatrix} \frac{\partial \psi_i}{\partial \xi} \\ \frac{\partial \psi_i}{\partial \eta} \\ \frac{\partial \psi_i}{\partial \zeta} \end{pmatrix} = \begin{pmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} & \frac{\partial z}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} & \frac{\partial z}{\partial \eta} \\ \frac{\partial x}{\partial \zeta} & \frac{\partial y}{\partial \zeta} & \frac{\partial z}{\partial \zeta} \end{pmatrix} \begin{pmatrix} \frac{\partial \psi_i}{\partial x} \\ \frac{\partial \psi_i}{\partial y} \\ \frac{\partial \psi_i}{\partial z} \end{pmatrix} = J \begin{pmatrix} \frac{\partial \psi_i}{\partial x} \\ \frac{\partial \psi_i}{\partial y} \\ \frac{\partial \psi_i}{\partial z} \end{pmatrix}, \tag{3.16}$$

thus the derivatives in the physical space can be obtained by inverting the previous

expression

$$\begin{pmatrix} \dfrac{\partial \psi_i}{\partial x} \\ \dfrac{\partial \psi_i}{\partial y} \\ \dfrac{\partial \psi_i}{\partial z} \end{pmatrix} = J^{-1} \begin{pmatrix} \dfrac{\partial \psi_i}{\partial \xi} \\ \dfrac{\partial \psi_i}{\partial \eta} \\ \dfrac{\partial \psi_i}{\partial \zeta} \end{pmatrix}. \tag{3.17}$$

The inverse of the Jacobian matrix usually exists, however the transformation can be singular in case of degenerate or highly distorted elements.

Volume Integration

As usually happens in numerical methods, integrals are evaluated numerically with quadrature rules. The quadrature rules are usually defined on reference elements, thus the integrals on the physical elements require the transformation of the integral from the physical to the reference space. For instance, the integral of a function $f(x, y, z)$ on the element e is written as

$$\int_{\Omega_e} f(x, y, z)\, \mathrm{d}\Omega = \int_{\widehat{\Omega}} f(\xi, \eta, \zeta)\, |J| \mathrm{d}\widehat{\Omega}, \tag{3.18}$$

where $\widehat{\Omega}$ is the reference element and $|J|$ is the determinant of the Jacobian of the transformation. Now, the integral can be evaluated numerically with a proper quadrature rule, as follows

$$\int_{\widehat{\Omega}} f(\xi, \eta, \zeta)\, |J| \mathrm{d}\widehat{\Omega} \approx \sum_{q=1}^{N_q} f(\xi_q, \eta_q, \zeta_q)\, |J|_q\, \omega_q = \sum_{q=1}^{N_q} f(\xi_q, \eta_q, \zeta_q)\, \tilde{\omega}_q, \tag{3.19}$$

where N_q is the number of the quadrature points, ξ_q, η_q, ζ_q are the coordinates of the quadratures points on the reference element and ω_q are the quadratures weights. The term $|J|_q$ indicates that the determinant of the Jacobian of the transformation is evaluated at the coordinates of the quadrature point. Note that the modified quadrature weights $\tilde{\omega}_q$ have been introduced, this notation is useful for implementation, since in numerical codes the terms $|J|_q$ are computed once for all, multiplied by the wights of the quadrature rules and then only $\tilde{\omega}_q$ are stored.

Surface Integration

Usually, numerical schemes require the calculation of the following type of surface integral for the vector function $\boldsymbol{f}(x, y, z)$

$$\int_{\partial\Omega} \boldsymbol{f}(x, y, z) \cdot \hat{\boldsymbol{n}}\, \mathrm{d}\partial\Omega, \tag{3.20}$$

where $\hat{\boldsymbol{n}}$ is the outward normal versor to the surface. The most convenient way to deal with the above expression is to consider the infinitesimal surface $\mathrm{d}\partial\Omega$ as

an oriented vector in the direction normal to the surface. For three dimensional problems the oriented vector is given by the following vector product

$$\hat{\boldsymbol{n}}\,\mathrm{d}\partial\Omega = \mathrm{d}\partial\boldsymbol{\Omega} = \begin{pmatrix} \dfrac{\partial x}{\partial \xi} \\ \dfrac{\partial y}{\partial \xi} \\ \dfrac{\partial z}{\partial \xi} \end{pmatrix} \times \begin{pmatrix} \dfrac{\partial x}{\partial \eta} \\ \dfrac{\partial y}{\partial \eta} \\ \dfrac{\partial z}{\partial \eta} \end{pmatrix} \mathrm{d}\xi\,\mathrm{d}\eta. \tag{3.21}$$

In two spatial dimensions, the previous relations reduces to

$$\hat{\boldsymbol{n}}\,\mathrm{d}\partial\Omega = \mathrm{d}\partial\boldsymbol{\Omega} = \begin{pmatrix} \dfrac{\partial x}{\partial \xi} \\ \dfrac{\partial y}{\partial \xi} \\ 0 \end{pmatrix} \times \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \mathrm{d}\xi. \tag{3.22}$$

By re-writing the oriented surface as $\mathrm{d}\partial\boldsymbol{\Omega} = \hat{\boldsymbol{n}}\,\|\mathrm{d}\partial\boldsymbol{\Omega}\|$, with $\hat{\boldsymbol{n}}$ the unit vector, the numerical integral reads

$$\begin{aligned} \int_{\widehat{\partial\Omega}} \boldsymbol{f}(\xi,\eta,\zeta)\cdot\hat{\boldsymbol{n}}|J|\mathrm{d}\widehat{\Omega} &\approx \sum_{q=1}^{N_q} \boldsymbol{f}(\xi_q,\eta_q,\zeta_q)\cdot\hat{\boldsymbol{n}}_q\,\|\mathrm{d}\partial\boldsymbol{\Omega}\|_q\,\omega_q \\ &= \sum_{q=1}^{N_q} \boldsymbol{f}(\xi_q,\eta_q,\zeta_q)\cdot\hat{\boldsymbol{n}}_q\,\tilde{\omega}_q \end{aligned} \tag{3.23}$$

3.1.3 Calculation of the Total Residual

As already introduced in Chapter 2, the first step in the construction of a RD scheme is the computation of the total residual on each element of the grid. For advection problems the total residual on the generic element e reads

$$\Phi^e(u_h) = \int_{\Omega_e} \boldsymbol{\nabla}\cdot\boldsymbol{f}(u_h)\,\mathrm{d}\Omega = \oint_{\partial\Omega_e} \boldsymbol{f}(u_h)\cdot\hat{\boldsymbol{n}}\,\mathrm{d}\partial\Omega, \tag{3.24}$$

where the divergence theorem has been used to transform the domain integral in the surface one, since the flux function is continuous on the face of the element. In such a way the direct calculation of the divergence is avoided and conservation is automatically satisfied.

In the construction of multidimensional upwind schemes, the total residual is usually computed by resorting the quasi-linear form of the governing equation, namely

$$\Phi^e(u_h) = \int_{\Omega_e} \boldsymbol{\nabla}\cdot\boldsymbol{f}(u_h)\,\mathrm{d}\Omega = \bar{\boldsymbol{a}}\cdot\int_{\Omega_e} \boldsymbol{\nabla}u_h\,\mathrm{d}\Omega. \tag{3.25}$$

A key aspect of this formulation is the mean-value linearization of the Jacobian flux. It is well know that if $\bar{\boldsymbol{a}}$ is not computed properly, the numerical scheme is not conservative and therefore it converges to the wrong solution in case of non-linear problems, when discontinuities are present. A practical way to construct a conservative scheme is to compute a proper average state $\bar{u}_h$, with $\bar{\boldsymbol{a}} = \boldsymbol{a}(\bar{u}_h)$, by the means of a conservative linearization, such that Eq. (3.25) and Eq. (3.24) are identical [118]. The main difficulty of this approach is that it is not possible to define the state $\bar{u}_h$ for general nonlinear conservation laws, such that the following relation is true

$$\bar{\boldsymbol{a}} = \boldsymbol{a}(\bar{u}_h) = \frac{1}{|\Omega_e|} \int_{\Omega_e} \boldsymbol{a}(u_h) \, \mathrm{d}\Omega. \tag{3.26}$$

An alternative approach was used by Abgrall and Barth [4], who proposed a family of non-conservative RD schemes in terms of entropy variables. The non-conservative integrals appearing in the quasi-linear form are approximated by an appropriate adaptive quadrature procedure such that the weak solutions are still obtained in the limit of mesh refinement, whenever sufficient order numerical quadrature is used. The method, however, is computational demanding due to the process of the adaptive quadrature and heavily relies on the symmetric formulation of the system.

In this work the formulation (3.24) is preferred to the others since it is more flexible and easier to apply to general non-linear conservation laws and to any kind of element. From a numerical point of view, the total residual on the generic element e is computed by the means of quadrature rules, as follows

$$\Phi^e(u_h) = \oint_{\partial\Omega_e} \boldsymbol{f}(u_h) \cdot \hat{\boldsymbol{n}} \, \mathrm{d}\partial\Omega \approx \sum_{f \in \mathcal{F}^e} \left(\sum_{q=1}^{N_q^f} \boldsymbol{f}(u_q) \cdot \hat{\boldsymbol{n}}_q \, \tilde{\omega}_q \right), \tag{3.27}$$

where $\mathcal{F}^e$ is the set of the faces of the element e, N_q^f is the number of quadrature points on the face f and the numerical solution at the quadrature point q is given by

$$u_q = \sum_{i \in \mathcal{N}^e} \psi_i(\xi_q, \eta_q, \zeta_q) \, u_i. \tag{3.28}$$

Consider now inhomogeneous problems written in the following from

$$\boldsymbol{\nabla} \cdot \boldsymbol{f}(u_h) = S(u_h), \tag{3.29}$$

where $S(u_h)$ is a source term, which might depend on u_h in the general case. The total residual for this problem reads

$$\begin{aligned} \Phi^e(u_h) &= \int_{\Omega_e} \Big(\boldsymbol{\nabla} \cdot \boldsymbol{f}(u_h) - S(u_h) \Big) \, \mathrm{d}\Omega \\ &= \oint_{\partial\Omega_e} \boldsymbol{f}(u_h) \cdot \hat{\boldsymbol{n}} \, \mathrm{d}\partial\Omega - \int_{\Omega_e} S(u_h) \, \mathrm{d}\Omega. \end{aligned} \tag{3.30}$$

The boundary integral has the same form already seen for homogeneous problems, and the domain integral is approximated with quadrature rules as follows

$$\int_{\Omega_e} S(u_h)\,\mathrm{d}\Omega \approx \sum_{q=1}^{N_q} S(u_q)\,\tilde{\omega}_q. \tag{3.31}$$

3.2 Multidimensional Upwind Schemes

After the calculation of the total residual, the next step in the construction of RD schemes is the distribution of the total residual to each degree of freedom of the element. The way the distribution process is performed is crucial and determines the behavior and the accuracy of the numerical scheme.

The first line of research within RD methods aimed to construct truly multidimensional upwind schemes in order to have more accurate schemes than FV [102]. Consider, for example, a standard node-centered FV method for multidimensional problems. For each node i of the grid, the method reduces to several one dimensional upwind schemes along the directions of the node-pairs composed by the node i and the neighboring nodes j. Hence the method is very dissipative and the overall accuracy of the FV scheme for multidimensional problems is lower than that guaranteed for one dimensional problems.

To see how multidimensional upwind schemes work, consider a two dimensional, homogeneous, scalar advection problem discretized on a grid of triangular elements; the description is limited to $\mathbb{P}1$ elements only. The total residual on the element e reads

$$\Phi^e = \int_{\Omega_e} \boldsymbol{\nabla}\cdot \boldsymbol{f}(u_h)\,\mathrm{d}\Omega = \bar{\boldsymbol{a}}\cdot \int_{\Omega_e} \boldsymbol{\nabla} u_h\,\mathrm{d}\Omega, \tag{3.32}$$

where the quasi-linear formulation has been resorted. For linear elements it is easy to put the previous expression in terms of the nodal values of the solution. Indeed, consider a node j of the element e, with $\boldsymbol{n}_j^e$ denoting the inward pointing vector normal to the edge of e opposite to the node j, scaled by the length of the edge (Figure 3.7), the gradient of the shape function is constant over the element and is given by

$$\boldsymbol{\nabla}\psi_j = \frac{\boldsymbol{n}_j^e}{2\,|\Omega_e|}, \tag{3.33}$$

thus the gradient of the numerical solution on the element e reads

$$\boldsymbol{\nabla} u_h = \sum_{j\in\mathcal{N}_h^e} \frac{\boldsymbol{n}_j^e}{2\,|\Omega_e|} u_j. \tag{3.34}$$

For u_h varying linearly on the element, the gradient of the numerical solution is

constant and Eq. (3.32) can be written as follows

$$\Phi^e = \bar{\boldsymbol{a}} \cdot \sum_{j \in \mathcal{N}_h^e} \frac{\boldsymbol{n}_j^e}{2\,|\Omega_e|} u_j = \sum_{j \in \mathcal{N}_h^e} k_j^e\, u_j, \tag{3.35}$$

where the following upwind parameter has been introduced

$$k_j^e = \frac{\bar{\boldsymbol{a}} \cdot \boldsymbol{n}_j^e}{2\,|\Omega_e|}. \tag{3.36}$$

It is easy to understand the reason of name upwind parameter: $k_j^e > 0$ if $\bar{\boldsymbol{a}}$ is oriented as $\boldsymbol{n}_j^e$, hence the node j is down-stream, conversely if $k_j^e < 0$ the node j is up-stream.

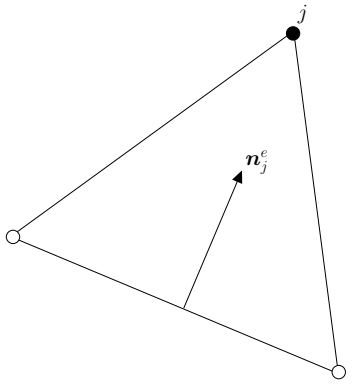


Figure 3.7: Definition of the scaled inward normal vector.

Note that, since the elements have a close boundary, the normal vectors $\boldsymbol{n}_j^e$ sum up to zero for each element, thus the following relation holds

$$\sum_{j \in \mathcal{N}_h^e} k_j^e = 0. \tag{3.37}$$

Introducing the following definitions

$$k_j^{e^+} = \max(k_j^e, 0) \qquad \text{and} \qquad k_j^{e^-} = \min(k_j^e, 0), \tag{3.38}$$

it is possible to re-write Eq. (3.32) as follows

$$\Phi^e = \sum_{j \in \mathcal{N}_h^e} k_j^{e^+} u_j + \sum_{j \in \mathcal{N}_h^e} k_j^{e^-} u_j = \sum_{j \in \mathcal{N}_h^e} k_j^{e^+} \left(\sum_{j \in \mathcal{N}_h^e} N^e k_j^{e^+} u_j + \sum_{j \in \mathcal{N}_h^e} N^e k_j^{e^-} u_j \right) \tag{3.39}$$

with

$$N^e = \left(\sum_{j \in \mathcal{N}_h^e} k_j^{e^+} \right)^{-1} = -\left(\sum_{j \in \mathcal{N}_h^e} k_j^{e^-} \right)^{-1} = \frac{1}{2} \left(\sum_{j \in \mathcal{N}_h^e} |k_j^e| \right)^{-1} > 0. \tag{3.40}$$

Defining the inflow

$$u^e_{\text{in}} = \frac{\sum_{j\in\mathcal{N}^e_h} k^{e^-}_j u_j}{\sum_{j\in\mathcal{N}^e_h} k^{e^-}_j} = -\sum_{j\in\mathcal{N}^e_h} N^e k^{e^-}_j u_j \tag{3.41}$$

and the outflow state on the element e

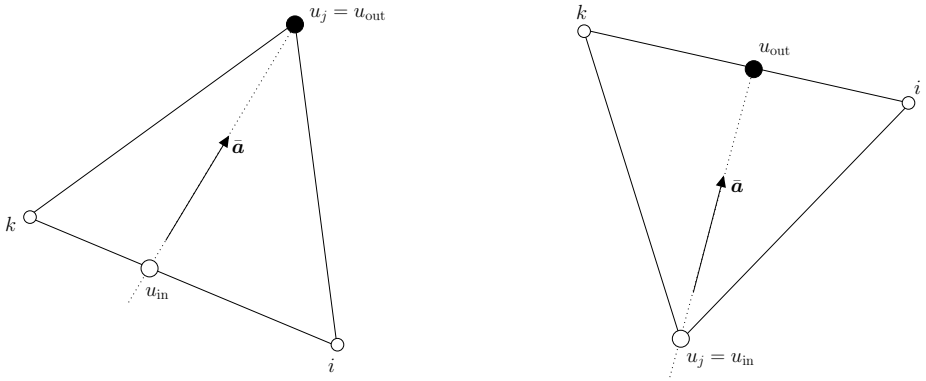
$$u^e_{\text{out}} = \frac{\sum_{j\in\mathcal{N}^e_h} k^{e^+}_j u_j}{\sum_{j\in\mathcal{N}^e_h} k^{e^+}_j} = \sum_{j\in\mathcal{N}^e_h} N^e k^{e^+}_j u_j \tag{3.42}$$

one finally obtains

$$\Phi^e = \sum_{j\in\mathcal{N}^e_h} k^{e^+}_j (u_{\text{out}} - u_{\text{in}}). \tag{3.43}$$

A geometrical interpretation of the previous expression can be found by noting that the inflow and outflow states represent the values of the numerical solution, respectively, in the most up-stream and down-stream points of e along the streamline direction $\bar{\boldsymbol{a}}$. As a consequence, Eq. (3.43) represents a one dimensional balance along the direction of $\bar{\boldsymbol{a}}$.

It is possible to distinguish two situations on the element e. If $\bar{\boldsymbol{a}}$ points in the direction of a single node, Figure 3.8-(a), then this node coincides with the outflow point; in this situation the element is said to be a one-target element. Conversely, if $\bar{\boldsymbol{a}}$ points in the direction of one edge, Figure 3.8-(b), then one node coincides with the up-stream point and the other two nodes are located down-stream; in this situation the element is said to be a two-target element.



(a) (b)

Figure 3.8: Definition of the inflow and outflow states for the one-target (a) and two-target (b) element.

The framework just described gives the possibility to construct multidimensional upwind schemes. These schemes must satisfy the following properties

i) For the one-target element e, if $k_j^e > 0$ and $k_i^e, k_k^e < 0$, then $\Phi_j^e = \Phi^e$ and $\Phi_i^e = \Phi_k^e = 0$

ii) For the two-target element e, if $k_j^e < 0$ and $k_i^e, k_k^e > 0$, then $\Phi_j^e = 0$

The previous properties define a one dimensional upwind scheme along the stream line direction, in particular in the case of a one-target element the whole residual is sent down-stream to the single outflow node. For a two-target element, the total residual has to be split between the two down-stream nodes; the way the residual is distributed between these two nodes in not unique. Here are described two ways to perform this task, which bring to two famous schemes in the RD framework.

3.2.1 LDA Scheme

The LDA (Low Diffusion A) scheme is a linearity preserving, multidimensional upwind scheme. It is defined by the following distribution coefficients

$$\beta_i^{e,\mathrm{LDA}} = k_i^{e^+} \left(\sum_{j \in \mathcal{N}_h^e} k_j^{e^+} \right)^{-1} = k_i^{e^+} N^e, \tag{3.44}$$

which give the following nodal residuals

$$\Phi_i^{e,\mathrm{LDA}} = \beta_i^{e,\mathrm{LDA}} \Phi^e = k_i^{e^+} (u_{\mathrm{out}} - u_{\mathrm{in}}). \tag{3.45}$$

Since the distribution coefficients remain bounded independently on Φ^e, the LDA scheme is formally second order accurate, but it is not Local Extrema Diminishing [89], thus the scheme is not monotone near discontinuities.

3.2.2 N Scheme

The N (Narrow) scheme, firstly proposed by Roe [102], is a first order, multidimensional upwind scheme. It is defined by the following nodal residuals

$$\Phi_i^{e,\mathrm{N}} = k_i^{e^+} (u_i - u_{\mathrm{in}}). \tag{3.46}$$

It differs from the LDA scheme only for the two-target element, in this case the N scheme reduces to a first order upwind scheme along the edges of the element originating from the inflow point.

Since the distribution coefficients are defined implicitly as

$$\beta_i^{e,\mathrm{N}} = \frac{\Phi_i^{e,\mathrm{N}}}{\Phi^e}, \tag{3.47}$$

they can be unbounded as $\Phi^e \to 0$, hence the scheme is not Linearity Preserving. On the other hand, it is easy to see that Positivity is guaranteed, which implies that the scheme is monotone.

While in the LDA scheme any arbitrary state can be used to determine the advection speed $\bar{\boldsymbol{a}}$, in the case of the N scheme a proper linearization must be performed otherwise the scheme is no longer consistent in case of non-linear problems. An alternative approach [42], which avoids the use of a consistent linearization, is based on the observation that the conditions

$$\begin{cases} \Phi_i^{e,\mathrm{N}} &= k_i^{e^+}(u_i - u_{\mathrm{in}}) \\ \Phi^e &= \sum_{i \in \mathcal{N}_h^e} \Phi_i^{e,\mathrm{N}} \end{cases} \tag{3.48}$$

uniquely define the state u_{in}. Thus, if the inflow state is computed as

$$u_{\mathrm{in}} = \left(\sum_{j \in \mathcal{N}_h^e} k_j^{e^+} \right)^{-1} \left(\sum_{j \in \mathcal{N}_h^e} k_j^{e^+} u_j - \Phi^e \right), \tag{3.49}$$

it is guaranteed that the scheme is formally identical to the N scheme and it does not violate the conservative definition of the residual, Eq. (3.24), independently on how the mean state $\bar{\boldsymbol{a}}$ is computed.

3.2.3 The Limit of the Multidimensional Upwind Schemes

Although multidimensional upwind RD schemes have shown great capabilities in the discretization of advection problems on simplexes, they suffer of a lack of flexibility. One limit is due to the fact that the extension of the multidimensional upwind mechanism to elements different from simplexes is not clear. For instance in [91,128], multidimensional linearity preserving schemes have been extended to quadrilateral elements for the discretization of two-dimensional scalar problems. However, it was found that an additional streamline stabilization term must be added because the schemes formulated on quadrilateral elements are marginally stable with respect to their counterpart on triangles.

Another important drawback of multidimensional upwind schemes is the difficulty to extend the approach to higher orders of approximation of the solution; most of the numerical results are limited to the case of linear interpolation. A possible way out to this limit consists in using sub-triangulation [130]. With this approach, the solution on each element is approximated with a k-th order polynomial, at the same time the degrees of freedom of the element are used to construct sub-cells within the main element. The total residual is computed on each sub-cell with the global polynomial expansion of the solution on the main element (*i.e.*, using all the degrees of freedom), then the residual is distributed to the degrees of freedom of each sub-cell by using a multidimensional upwind scheme, as done for linear elements (Figure 3.9)

It is clear that the sub-triangulation adds a large computational effort with respect to the linear interpolation of the solution and might make higher order schemes less efficient than low order ones in term of computational time. An alternative approach has been proposed in [26] to construct a third order RD scheme on triangles

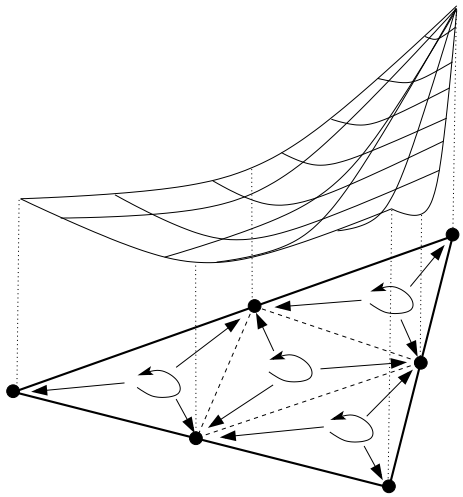


Figure 3.9: Graphical representation of the sub-triangulation approach.

and tetrahedra. In this approach, the order of accuracy is increased by considering a higher order reconstruction of the solution; the Hermite cubic interpolation is used to compute the midpoint value of the solution on the edge, this value is used together with the vertex values to construct a quadratic approximation of the solution on the edge. The total residual on the element, computed using a better approximation of the solution, is then distributed to the vertices with a multidimensional upwind RD scheme. Although this approach is less expensive that the use of the sub-triangulation, it is limited to formally third order approximation on triangles and tetrahedra. Furthermore, the Hermite interpolation requires the reconstruction of the gradient of the solution at the vertices of the element and this introduce two additional drawbacks: the stencil of the numerical scheme becomes larger and the accuracy of the numerical scheme depends strongly on the regularity of the mesh.

3.3 Central Schemes

To overcome the limits of multidimensional upwind schemes, a different family of scheme is now considered. The idea of truly multidimensional upwinding is abandoned in favor of a central distribution of the residual in order to construct a more flexible method.

In central schemes, differently from truly upwind ones, each degree of freedom of the element always receives a certain amount of the total residual. Obviously, a pure isotropic distribution of the residual would result in an unstable scheme for hyperbolic problems, hence a bias term must be added to the pure central part in order to stabilize the numerical scheme.

3.3.1 Petrov-Galerkin Scheme

Consider the Petrov-Galerkin scheme used in the FE framework for the discretization of the scalar advection problem (2.1). As well know, the scheme consists of the Galerkin discretization of the governing equation plus a streamline dissipation term, [20, 59, 60]

$$\int_\Omega \psi_i \, \boldsymbol{a}\cdot\boldsymbol{\nabla} u_h \, \mathrm{d}\Omega + \sum_{e\in\mathcal{E}_h} \int_{\Omega_e} \boldsymbol{a}\cdot\boldsymbol{\nabla}\psi_i \, \tau_h \, \boldsymbol{a}\cdot\boldsymbol{\nabla} u_h \, \mathrm{d}\Omega = 0, \qquad \forall i \in \mathcal{N}_h, \tag{3.50}$$

where τ_h is a positive parameter with the dimension of a time and it is defined as follows

$$\tau_h = \frac{1}{2}\frac{h_e}{\|\boldsymbol{a}\|}, \tag{3.51}$$

with h_e a characteristic length scale of the element.

Exploiting the compactness of the basis functions, in the case of linear triangular elements and constant advection speed $\boldsymbol{a}$, the first integral of Eq. (3.50) reduces to

$$\sum_{e\in\mathcal{E}_h^i} \int_{\Omega_e} \psi_i \, \boldsymbol{a}\cdot\boldsymbol{\nabla} u_h \, \mathrm{d}\Omega = \sum_{e\in\mathcal{E}_h^i} \frac{\Phi^e}{3|\Omega_e|}, \tag{3.52}$$

while using the relation (3.34) and the definition (3.36), the second integral of Eq. (3.50) reads

$$\sum_{e\in\mathcal{E}_h^i} \int_{\Omega_e} \boldsymbol{a}\cdot\boldsymbol{\nabla}\psi_i \, \tau_h \, \boldsymbol{a}\cdot\boldsymbol{\nabla} u_h \, \mathrm{d}\Omega = \sum_{e\in\mathcal{E}_h^i} \tau_h \frac{k_i}{2|\Omega_e|}\Phi^e. \tag{3.53}$$

Hence, the Petrov-Galerkin scheme (3.50) can be recasted in the following RD form

$$\Phi_i^{e,\mathrm{PG}} = \beta_i^{e,\mathrm{PG}}\Phi^e, \qquad \text{with} \qquad \beta_i^{e,\mathrm{PG}} = \frac{1}{3} + \tau_h\frac{k_i}{2|\Omega_e|}. \tag{3.54}$$

Strictly speaking, the previous analogy between RD and Petrov-Galerkin schemes is valid only for $\mathbb{P}1$ elements and constant advection speed. In the more general case Eq. (3.50) no longer reduces to Eq. (3.54).

Thanks to the property (3.37) the scheme is consistent. Furthermore, it is easy to see from the definition of the distribution coefficients (3.54) that the scheme is linearity preserving, thus formal second order accuracy is guaranteed for a linear interpolation of the solution.

3.3.2 Rusanov Scheme

The Rusanov scheme is a central scheme obtained from a central distribution of the total residual plus a stabilization term

$$\Phi_i^{e,\mathrm{Rv}} = \frac{1}{N_{\mathrm{dof}}^e}\Phi^e + \frac{1}{3}\alpha^e \sum_{\substack{j\in\mathcal{N}_h^e \\ j\neq i}} (u_i - u_j), \qquad \alpha^e \geq \max_{j\in\mathcal{N}_h^e} |k_j| > 0. \tag{3.55}$$

Since the distribution coefficients are defined implicitly as

$$\beta_i^{e,\mathrm{Rv}} = \frac{\Phi_i^{e,\mathrm{Rv}}}{\Phi^e}, \tag{3.56}$$

they can be unbounded as $\Phi^e \to 0$, hence the scheme is only first order accurate.

The streamline dissipation term of the Petrov-Galerkin scheme, Eq. (3.54), has residual properties, *i.e.* it is proportional to the residual of the governing equation through bounded coefficients, thus it vanishes as the residual tends to zero. On the other hand, the dissipation term of the Rusanov scheme, Eq. (3.55), has not a residual properties and, in addition, it has a purely isotropic character. Although very cheap, the Rusanov scheme is expected to be extremely dissipative.

3.3.3 Lax-Wendroff Scheme

Consider a discretization method consisting of a central scheme plus a streamline stabilization term, which gives the following nodal residual

$$\Phi_i^{e,\mathrm{LW}} = \frac{\Phi^e}{N_{\mathrm{dof}}^e} + \int_{\Omega_e} \boldsymbol{a}\cdot\boldsymbol{\nabla}\psi_i\,\tau\,\boldsymbol{a}\cdot\boldsymbol{\nabla}u_h\,\mathrm{d}\Omega, \tag{3.57}$$

with the positive parameter τ given by

$$\tau = \frac{1}{2}\frac{|\Omega_e|}{\max\limits_{j\in\mathcal{N}_e^h}(k_j^+)}. \tag{3.58}$$

The reason why this scheme is called here Lax-Wendroff comes from the fact that Eq. (3.57), in the case of $\mathbb{P}1$ element with constant advection speed, coincides with the classical Ni's version of the Lax-Wendroff scheme [81], for which the distribution coefficients can be defined as

$$\beta_i^{e,\mathrm{LW}} = \frac{1}{3} + \frac{\Delta t}{2|\Omega_e|}k_i. \tag{3.59}$$

Note that the previous expression of the distribution coefficients is very similar to that of the Petrov-Galerkin scheme, although the derivation of the two scheme is completely different.

3.4 Non-Linear Schemes

As already reported in Chapter 2, a linear scheme cannot be monotone and high-order at the same time, thus non-linear schemes have to be considered to overcome this limit. Two approaches are commonly used in the RD framework to construct non-linear schemes: the local non-linear blending of a linearity preserving scheme

with a positive one, and the non-linear limiting of a positive scheme into a linearity preserving one.

The first approach simply consists in blending together two linear schemes in order to combine the non-oscillatory properties of a Positive scheme with the capacity to get high-order of a linearity preserving scheme, namely

$$\Phi_i^{e,\mathrm{B}} = (1 - \Theta^e(u_h))\Phi_i^{e,\mathrm{LP}} + \Theta^e(u_h)\Phi_i^{e,\mathrm{P}}, \tag{3.60}$$

where $\Phi_i^{e,\mathrm{LP}}$ and $\Phi_i^{e,\mathrm{P}}$ are, respectively, the residuals computed with the linearity preserving scheme and the positive one. The blending parameter $\Theta^e(u_h)$, which depends on the numerical solution, must ensure that the condition (2.40) is fulfilled in smooth regions, while the monotonicity of the first order scheme must prevail across discontinuities. A very common choice [41,55,111] consists in taking $\Phi_i^{e,\mathrm{LP}} = \Phi_i^{e,\mathrm{LDA}}$ and $\Phi_i^{e,\mathrm{P}} = \Phi_i^{e,\mathrm{N}}$ with the blending parameter defined as

$$\Theta^e(u^h) = \frac{|\Phi^e|}{\sum_{i \in \mathcal{N}^e} \left|\Phi_i^{e,\mathrm{N}}\right|}, \tag{3.61}$$

even though more sophisticated expressions can be constructed [2]. Note that the previous definition of $\Theta^e(u_h)$ only assures that condition (2.40) is satisfied, but there is no guarantee that the scheme is really monotone across shocks.

Although the non-linear blending provides an easy way to construct effective high-order schemes, in practical applications it might be difficult to define a proper blending parameter that does not spoil accuracy and robustness of the numerical scheme. Furthermore, these schemes are not well suited for unsteady simulations [10].

Another approach for combining high-order discretization with monotonicity, consists in mapping a positive scheme into a linearity preserving one. This operation is generally called non-linear limiting and is described in the next section.

3.4.1 Limited Non-Linear Schemes

Consider a first order accurate monotone scheme, such a scheme has unbounded distribution coefficients $\beta_i^e = \Phi_i^e/\Phi^e$. The construction of a non-linear limited scheme consists in limiting the distribution coefficients of the first order scheme in order to have bounded values. The limiting procedure is obtained by applying a continuous mapping $\beta_i^e \mapsto \hat{\beta}_i^e$, from the space of the distribution coefficients of the low-order scheme to the one of the higher order scheme. The high-order, monotone distributed residuals are then computed as

$$\hat{\Phi}_i^e = \hat{\beta}_i^e(u_h)\Phi^e(u_h). \tag{3.62}$$

In order to have a well posed map, the following conditions must be satisfied

i) $|\hat{\beta}_i^e| < \infty,\ \forall i \in \mathcal{N}_h^e,$

ii) $\hat{\beta}_i^e \beta_i^e \geq 0$ and $\Phi_i^e = 0 \Rightarrow \hat{\beta}_i^e = 0,\ \forall i \in \mathcal{N}_h^e,$

iii) $\sum_{i \in \mathcal{N}_h^e} \hat{\beta}_i^e = 1.$

The first property assures that the scheme is linearity preserving because all the distribution coefficients remain bounded. The second property corresponds to require that the scheme is monotone; indeed

$$\hat{\Phi}_i^e = \frac{\hat{\Phi}_i^e}{\Phi^e}\frac{\Phi^e}{\Phi_i^e}\Phi_i^e = \frac{\hat{\beta}_i^e}{\beta_i^e} \sum_{\substack{j \in \mathcal{N}_h^e \\ j \neq i}} c_{ij}^e (u_i - u_j) = \sum_{\substack{j \in \mathcal{N}_h^e \\ j \neq i}} \hat{c}_{ij}^e (u_i - u_j), \tag{3.63}$$

where

$$\hat{c}_{ij}^e = c_{ij}^e \frac{\hat{\beta}_i^e}{\beta_i^e}. \tag{3.64}$$

Since, by definition, for a first order monotone scheme $c_{ij}^e \geq 0$, the positivity of $\hat{c}_{ij}^e$ requires $\hat{\beta}_i^e \beta_i^e \geq 0$. The last property assures the conservation of the numerical scheme.

One popular RD scheme based on this approach, the so-called PSI scheme [10,11, 118], uses the N scheme as first order starting method with the following definition of the map

$$\hat{\beta}_i^{e,\mathrm{PSI}} = \frac{\max\left(\beta_i^{e,\mathrm{N}}, 0\right)}{\sum_{j \in \mathcal{N}_h^e} \max\left(\beta_j^{e,\mathrm{N}}, 0\right)}, \tag{3.65}$$

in which negative values of $\beta_i^{e,\mathrm{N}}$ are set to zero and the maximum value cannot exceed the unity. Note that the new distribution coefficients sum up to the unity and all the properties which assure the well-posedness of the map are fulfilled.

The great drawback of the PSI scheme is the poor iterative convergence in the case of system of equations, furthermore the PSI scheme cannot be extended easily to orders of accuracy more than two. In [11] higher order accuracy was obtained with the PSI scheme through the use of sub-triangulation of the elements and linear interpolation of the solution on each sub-element, however it is easy to show that in order to ensure that the consistency condition is satisfied, the monotonicity one must be relaxed.

In [3] a limited non-linear scheme based on the Rusavon scheme is proposed with the use of the same limiting map of the PSI scheme (3.65). This last approach is more flexible than the PSI scheme, since the central scheme can be easily formulated on every type of element, for scalar problem and system of equations, as well. However, the use of a central scheme, like the Rusanov scheme, in combination with the limiting technique produces undamped spurious modes in smooth regions of the solution, although the scheme performs well near discontinuities. Furthermore, the

numerical scheme has a very poor iterative convergence to the steady state. The problem is due to the fact that in the limiting only the sign of the distribution coefficients is considered and no upwind mechanism is taken into account, hence the scheme can be locally downwind.

The problem is analyzed with more details in [3], where it is shown that an additional filtering term must be added to the limited residuals in order to obtain a convergent and accurate numerical scheme. The non-linear Rusanov scheme then reads

$$\hat{\Phi}_i^{e,\mathrm{Rv}} = \hat{\beta}_i^{e,\mathrm{Rv}}(u_h)\Phi^e(u_h) + \theta_h^e(u_h)\int_{\Omega_e} \boldsymbol{a}\cdot\boldsymbol{\nabla}\psi_i \; \boldsymbol{a}\cdot\boldsymbol{\nabla}u_h \,\mathrm{d}\Omega, \tag{3.66}$$

where the parameter $\theta_h^e(u_h)$ provides the correct scaling of the filtering term. In the general case θ_h^e depends on the solution itself, this means that the monotonicity of scheme is not formally guaranteed, however numerical experiments confirm that the scheme remains essentially non oscillatory near discontinuities. The following definition of the parameter θ_h^e is used here

$$\theta_h^e(u_h) = \varepsilon(u_h)\left(\frac{\displaystyle\sum_{j\in\mathcal{N}_h^e} |\bar{\boldsymbol{a}}\cdot\boldsymbol{n}_j|}{2|\Omega_e|}\right)^{-1}, \quad \text{with} \quad \boldsymbol{n}_j = \int_{\Omega_e} \boldsymbol{\nabla}\psi_j \,\mathrm{d}\Omega, \tag{3.67}$$

where $\bar{\boldsymbol{a}}$ is the arithmetic average of the advection speed on the element and $\varepsilon(u_h)$ is a smoothness sensor such that $\varepsilon(u_h) = 1$ in smooth regions and $\varepsilon(u_h) = 0$ across discontinuities [8].

Chapter 4

Extension to Multidimensional Second-Order PDEs on Unstructured Grids

The RD approach, previously detailed for the discretization of advection problems, is now extended to include the discretization of diffusion terms. Although RD methods have been successfully used to discretize advection problems for long time, their extension to advection-diffusion problems has not reached a complete level of understanding yet. The simple coupling of a RD method for the advective terms with a different approach for the diffusive ones generally results in a not optimal global scheme, although the two schemes separately perform well.

In this work advection and diffusion are discretized within the same framework in order to construct an effective and flexible RD method which maintains an optimal accuracy for different cases, ranging from advection dominated to diffusion dominated problems. The key point of this approach is the reconstruction of an unique value of the gradient of the numerical solution at the degrees of freedom of the grid, thus a continuous approximation of both the solution and its gradient can be used to compute the total residual on each element.

4.1 Discretization of Diffusion Terms with RD Methods

When in the governing equation (2.1) diffusive phenomena are considered together with the advective terms, the following advection-diffusion equation is obtained

$$\boldsymbol{\nabla}\cdot\boldsymbol{f}(u) = \boldsymbol{\nabla}\cdot\big(\nu\boldsymbol{\nabla}u\big), \tag{4.1}$$

where $\nu > 0$ is the viscosity, generally function of u. The relative importance of the advection and the diffusion is described by the non-dimensional parameter, Peclet number, $\mathrm{Pe} = \|\boldsymbol{a}\|\, h/\nu$. In advection and diffusion limits $\mathrm{Pe} \to \infty$, $\mathrm{Pe} \to 0$, respectively, while $\mathrm{Pe} \sim 1$ when advection and diffusion are equally important.

For the extension of RD methods to advection-diffusion problems, different strategies have been considered to compute and to distribute the residual associated with the diffusive terms. On a first attempt, based on the physical intuition that the diffusion has an isotropic behavior in space, RD schemes (for advection problems) were coupled with the Galerkin discretization of diffusive terms [90,124], but a truncation error analysis revealed that this simple approach results in a first order accurate scheme when advection and diffusion have the same order of magnitude [85]. A different approach, developed for two-dimensional problems on triangular grids, considered the hybridization of RD methods with a Petrov-Galerkin scheme by the means of a scaling parameter, function of the Peclet number [98].

A key aspect that emerges from the work of Nishikawa and Roe [85] is that a RD scheme with an uniform order of accuracy in the whole range of Peclet numbers should not consider two different distribution schemes for advective and diffusive terms, but only one distribution process has to be performed for the residual of the whole equation, namely

$$\Phi^e = \int_{\Omega_e} \left(\boldsymbol{\nabla} \cdot \boldsymbol{f}\left(u_h\right) - \boldsymbol{\nabla} \cdot \left(\nu \boldsymbol{\nabla} u_h \right) \right) \mathrm{d}\Omega. \tag{4.2}$$

To put the previous expression in term of a boundary integral, one has to cope with the fact that the normal component of the gradient of the numerical solution, $\boldsymbol{\nabla} u^h \cdot \hat{\boldsymbol{n}}$, is in general discontinuous across the face of two adjacent elements and this violates the continuous approximation hypothesis of the numerical scheme. Suppose, now, that an unique value of the gradient is available at each degree of freedom, the gradient can be interpolated with the same shape functions used for the solution. The total residual on the element can be written as follows

$$\Phi^e = \oint_{\partial\Omega_e} \left(\boldsymbol{f}\left(u_h\right) - \nu \widetilde{\boldsymbol{\nabla} u_h} \right) \cdot \hat{\boldsymbol{n}} \, \mathrm{d}\partial\Omega, \tag{4.3}$$

where $\widetilde{\boldsymbol{\nabla} u_h}$ is the interpolated gradient of the numerical solution, which is now continuous on the faces of the elements.

Once the total residual is evaluated, it can be distributed to the degrees of freedom of the elements. This strategy has been adopted in [88] to construct a second order RD scheme for advection-diffusion problems on triangular grids an has been extended to the third order in [25]. In both works, the distribution process is performed by means of purely advective distribution coefficients, which is not appropriate in the diffusion limit. A more general scheme consists in using distribution coefficients which are function of the local Peclet number in order to recover an isotropic scheme in the diffusion limit and an upwind one in the advection limit [31, 85].

The key idea of the Eq. (4.3) is the reconstruction of the gradient of the numerical solution at each degree of freedom of the grid and is one of the issue analyzed in this chapter. Numerical experiments show that in order to obtain high-order accurate solutions, the gradients must be reconstructed with the same order of the solution.

An alternative approach, proposed by Nishikawa for scalar diffusion problems [82] and scalar advection-diffusion problems [83], consists in reinterpreting the advection-diffusion scalar equation as an equivalent hyperbolic first order system, in this way the gradient reconstruction is no longer necessary, but the price to pay is the increment of the unknowns of the problem due to the fact that a system of equations must be solved instead of a single scalar equation.

4.2 Hyperbolic First Order System Formulation

The hyperbolic First Order System (FOS) formulation is here recalled for later convenience. The basic idea consists in re-writing the advection-diffusion scalar problem (4.1) as an equivalent first order system in which the second order derivatives of the original problem are replaced by the first order derivatives of auxiliary variables. At the steady state the two formulations will coincide and the values of the auxiliary variables will equal the values of the derivatives of the unknown in the original problem.

Consider the scalar advection-diffusion problem in $\Omega \in \mathbb{R}^2$, for simplicity,

$$\frac{\partial u}{\partial t} + \boldsymbol{a} \cdot \boldsymbol{\nabla} u - \boldsymbol{\nabla} \cdot \big(\nu \boldsymbol{\nabla} u\big) = 0, \tag{4.4}$$

with the advection speed $\boldsymbol{a} = (a_x, a_y)^{\mathrm{T}}$. The previous equation can be recasted as first order system (FOS) by writing

$$\begin{aligned} \frac{\partial u}{\partial t} + \boldsymbol{a} \cdot \boldsymbol{\nabla} u &= \nu \left(\frac{\partial p}{\partial x} + \frac{\partial q}{\partial y} \right) \\ \frac{\partial p}{\partial t} &= \frac{1}{T_r} \left(\frac{\partial u}{\partial x} - p \right) \\ \frac{\partial q}{\partial t} &= \frac{1}{T_r} \left(\frac{\partial u}{\partial y} - q \right) \end{aligned} \tag{4.5}$$

where p and q are the gradient variables and T_r is a relaxation time. At the steady state, the system (4.5) is equivalent to the original equation (4.4), independently on the parameter T_r, and p,q become equivalent to $\frac{\partial u}{\partial x}$ and $\frac{\partial u}{\partial y}$, respectively.

The system (4.5), in vector form reads

$$\frac{\partial \mathsf{u}}{\partial t} + \boldsymbol{A} \cdot \boldsymbol{\nabla} \mathsf{u} = \mathsf{S}, \tag{4.6}$$

with

$$\mathsf{u} = \begin{pmatrix} u \\ p \\ q \end{pmatrix}, \quad A_x \begin{pmatrix} a_x & -\nu & 0 \\ \frac{1}{T_r} & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad A_y \begin{pmatrix} a_y & 0 & -\nu \\ 0 & 0 & 0 \\ \frac{1}{T_r} & 0 & 0 \end{pmatrix}, \quad \boldsymbol{S} = \begin{pmatrix} 0 \\ -\frac{p}{T_r} \\ -\frac{q}{T_r} \end{pmatrix}, \tag{4.7}$$

and since, for an arbitrary vector $\boldsymbol{n} = (n_x, n_y)^{\mathrm{T}}$, it is possible to write the Jacobian matrix as $\boldsymbol{A} \cdot \boldsymbol{n} = A_x n_x + A_y n_y$, thus

$$A_n = A_x n_x + A_y n_y = \begin{pmatrix} a_n & -\nu n_x & -\nu n_y \\ -\dfrac{n_x}{T_r} & 0 & 0 \\ -\dfrac{n_y}{T_r} & 0 & 0 \end{pmatrix}, \tag{4.8}$$

with $a_n = \boldsymbol{a} \cdot \boldsymbol{n}$. The Jacobian matrix (4.8) has a set of real eigenvalues

$$\lambda_{1,n} = \frac{1}{2}\left(a_n - \sqrt{a_n^2 + \frac{4\nu}{Tr}}\right), \quad \lambda_{2,n} = \frac{1}{2}\left(a_n + \sqrt{a_n^2 + \frac{4\nu}{Tr}}\right), \quad \lambda_{3,n} = 0, \tag{4.9}$$

while the matrix of the right-eigenvectors reads

$$R_n = \begin{pmatrix} \lambda_{1,n} T_r & \lambda_{2,n} T_r & 0 \\ n_x & n_y & -n_y \\ n_y & n_y & n_x \end{pmatrix}, \tag{4.10}$$

and, as usual, the Jacobian matrix can be written as $A_n = R_n \Lambda_n L_n$, where Λ_n is the diagonal matrix of the eigenvalues and $L_n = R_n^{-1}$ is the matrix of the left eigenvectors.

The system (4.5) has real eigenvalues and linearly independent right-eigenvectors, thus the system is hyperbolic. The parameter T_r can be defined as the ratio of a length scale L_r to the characteristic wave speed of the system. When the maximum eigenvalue is considered, T_r can be written as

$$T_r = \frac{L_r}{\lambda_{2,n}}, \tag{4.11}$$

substituting in the previous relation the definition of $\lambda_{2,n}$ and solving for T_r one obtains

$$T_r = \frac{L_r}{|a_n| + \nu / L_r}, \tag{4.12}$$

where a_n has been replaces by $|a_n|$ to keep T_r positive. By substituting the definition (4.12) back into the definition of the eigenvalues one obtains

$$\lambda_{1,n} = a_n^- \left(1 - \frac{1}{Re_{L_r}^-}\right), \quad \lambda_{2,n} = a_n^+ \left(1 + \frac{1}{Re_{L_r}^+}\right), \quad \lambda_{3,n} = 0, \tag{4.13}$$

with

$$a_n^- = \min(0,\, a_n), \qquad a_n^+ = \max(0,\, a_n)$$
$$Re_{L_r}^- = \frac{a_n^- L_r}{\nu} \qquad Re_{L_r}^+ = \frac{a_n^+ L_r}{\nu}.$$

Note that $a_n^+ + a_n^- = a_n$ and $a_n^+ - a_n^- = |a_n|$. The matrix of the right-eigenvectors can be now written as follows

$$R_n = \begin{pmatrix} \dfrac{L_r}{Re_{L_r}^+ + 1} & \dfrac{L_r}{Re_{L_r}^- - 1} & 0 \\ n_x & n_y & -n_y \\ n_y & n_y & n_x \end{pmatrix}, \tag{4.14}$$

with the corresponding left-eigenvectors given by

$$L_n = \frac{1}{|Re_{L_r}| + 2} \begin{pmatrix} \dfrac{|Re_{L_r}| + 1}{L_r} & (1 + Re_{L_r}^+)n_x & (1 + Re_{L_r}^+)n_y \\ -\dfrac{|Re_{L_r}| + 1}{L_r} & (1 - Re_{L_r}^-)n_x & (1 - Re_{L_r}^-)n_y \\ 0 & -n_y & n_x \end{pmatrix}, \tag{4.15}$$

where $|Re_{L_r}| = \frac{|a_n|L_r}{\nu}$. The Jacobian matrix A_n can be decomposed as follows

$$A_n = \lambda_{1.n}\Pi_{1,n} + \lambda_{2,n}\Pi_{2,n}, \tag{4.16}$$

with the projection matrices $\Pi_{1,n}, \Pi_{2,n}$ given by

$$\Pi_{1,n} = \frac{1}{|Re_{L_r}| + 2} \begin{pmatrix} 1 - Re_{L_r}^- & L_r n_x & L_r n_y \\ \dfrac{|Re_{L_r}| + 1}{L_r} n_x & (1 + Re_{L_r}^+)n_x^2 & (1 + Re_{L_r}^+)n_x n_y \\ \dfrac{|Re_{L_r}| + 1}{L_r} n_y & (1 + Re_{L_r}^+)n_x n_y & (1 + Re_{L_r}^+)n_y^2 \end{pmatrix}, \tag{4.17}$$

$$\Pi_{2,n} = \frac{1}{|Re_{L_r}| + 2} \begin{pmatrix} 1 + Re_{L_r}^+ & -L_r n_x & -L_r n_y \\ -\dfrac{|Re_{L_r}| + 1}{L_r} n_x & (1 - Re_{L_r}^-)n_x^2 & (1 - Re_{L_r}^-)n_x n_y \\ -\dfrac{|Re_{L_r}| + 1}{L_r} n_y & (1 - Re_{L_r}^-)n_x n_y & (1 - Re_{L_r}^-)n_y^2 \end{pmatrix}. \tag{4.18}$$

It is worth noting that all the previous considerations are done on the continuous system, regardless the numerical method used.

4.2.1 Determination of the Length Scale

In the previous section, the parameter T_r has been defined as the ratio of a length scale L_r to the maximum wave speed of the system. The length scale L_r must be still determined. One can think to choose L_r to optimize some property of the numerical scheme or to ameliorate, in some way, the formulation of the continuous system. For example, one can choose L_r such that the stiffness of the continuous system is minimized, thereby reaching the steady state as fast as possible.

Following Nishikawa [83], consider a Fourier mode of phase angle $\boldsymbol{\beta} = (\beta_x, \beta_y)^{\mathrm{T}}$, with $\beta_x, \beta_y \in [0, \pi]$

$$\mathsf{u}^{\boldsymbol{\beta}} = e^{(\beta_x x/h + \beta_y y/h)} \mathsf{u}_0, \tag{4.19}$$

inserting this mode in the equation (4.6), one obtains

$$\frac{d\mathsf{u}^{\boldsymbol{\beta}}}{dt} = M^{\beta} \mathsf{u}^{\boldsymbol{\beta}}, \tag{4.20}$$

where

$$M^{\beta} = \begin{pmatrix} i\dfrac{\boldsymbol{a}\cdot\boldsymbol{\beta}}{h} & \nu\dfrac{i\beta_x}{h} & \nu\dfrac{i\beta_y}{h} \\ \dfrac{i\beta_x}{hT_r} & -\dfrac{1}{T_r} & 0 \\ \dfrac{i\beta_y}{hT_r} & 0 & -\dfrac{1}{T_r} \end{pmatrix}, \tag{4.21}$$

The eigenvalues of M^{β} are given by

$$\lambda^{\beta}_{1,2} = -\frac{1}{2}\left(\left(\frac{1}{T_r} + \frac{i\beta a_\beta}{h} \right) \pm \sqrt{ \left(\frac{1}{T_r} - \frac{i\beta a_\beta}{h} \right)^2 - \frac{4\nu\beta^2}{h^2 T_r} } \right), \quad \lambda^{\beta}_3 = -\frac{1}{T_r}, \tag{4.22}$$

with

$$a_\beta = \frac{\boldsymbol{a}\cdot\boldsymbol{\beta}}{\beta}, \quad \text{and} \quad \beta = \sqrt{\beta_x^2 + \beta_y^2}. \tag{4.23}$$

Since the third eigenvalue is zero, the attention is focused only on $\lambda^{\beta}_{1,2}$. These eigenvalues are complex, the imaginary part gives the propagation velocity and the real part gives the damping factor. If $\lambda^{\beta}_{1,2}$ are complex conjugate, the system will be perfectly conditioned because at the two eigenvalues will be associated the same propagation speed (magnitude of the imaginary part) and the same damping factor. Only in the diffusion limit the two eigenvalues will be complex conjugate, while in the advection limit the eigenvalues will be pure imaginary and pure real

$$\lambda^{\beta}_1 = -\frac{i\alpha a_\beta}{h} \quad \text{and} \quad \lambda_2 = -\frac{1}{T_r}. \tag{4.24}$$

One can consider to equalize the magnitude of the eigenvalues, $|\lambda^{\beta}_1|/|\lambda^{\beta}_2| = 1$, for the smoothest error mode, to obtain the following optimal length [83]

$$L_r = \frac{1}{2\pi}\left(\frac{Re_\pi}{\sqrt{1 + {Re_\pi}^2} + 1} + \sqrt{1 + \frac{2}{\left(\sqrt{1 + {Re_\pi}^2} + 1\right)}} \right), \tag{4.25}$$

where

$$Re_\pi = \frac{\|\boldsymbol{a}\|(1/\pi)}{\nu}. \tag{4.26}$$

It is easy to see that in the diffusion limit ($\|a\| \to 0$) $L_r = 1/(\sqrt{2}\pi)$, in the advection limit ($\nu \to 0$) $L_r = 1/\pi$, and the value of L_r remains confined between these two values, see Figure 4.1. With this choice of L_r the condition number $K = |\lambda_1^\beta|/|\lambda_2^\beta|$ is one on all the range of Re_π.

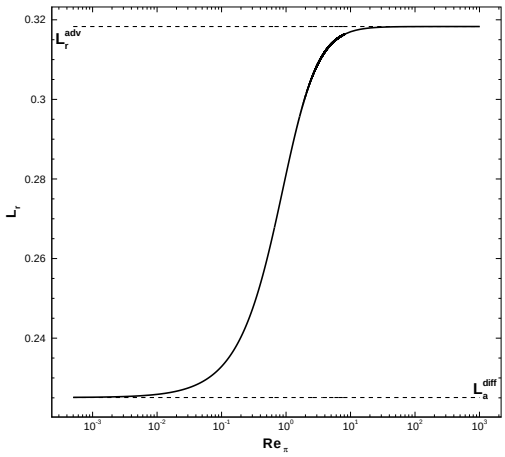


Figure 4.1: Variation of the characteristic length L_r as defined in Eq. (4.25).

4.2.2 Spatial Discretization

Since the system of equations (4.6) is hyperbolic, it can be discretized with any scheme already available for hyperbolic problems. If a RD scheme is used, the total residual of a generic element e is defined as follows

$$\Phi^e(\mathsf{u}^h) = \begin{pmatrix} \Phi_u^e \\ \Phi_p^e \\ \Phi_q^e \end{pmatrix} = \int_{\Omega_e} \Big(\boldsymbol{A} \cdot \boldsymbol{\nabla} \mathsf{u}_h - \mathsf{S}(\mathsf{u}^h) \Big) \, \mathrm{d}\Omega. \tag{4.27}$$

The system can be written in conservative form by introducing the flux function $\mathbf{f}(\mathsf{u}) = \Big(\mathsf{f}_x(\mathsf{u}), \mathsf{f}_y(\mathsf{u}) \Big)^{\mathrm{T}}$, such that $\boldsymbol{A} = \boldsymbol{\nabla}_{\mathsf{u}} \mathbf{f}(\mathsf{u})$, with

$$\mathsf{f}_x(\mathsf{u}) = \begin{pmatrix} a_x u - \nu p \\ -\dfrac{u}{T_r^\star} \\ 0 \end{pmatrix} \quad \text{and} \quad \mathsf{f}_y(\mathsf{u}) = \begin{pmatrix} a_y u - \nu q \\ 0 \\ -\dfrac{u}{T_r^\star} \end{pmatrix}, \tag{4.28}$$

where the parameter $T_r^\star$ is used instead of the parameter T_r, with

$$T_r^\star = \frac{L_r}{\|\boldsymbol{a}\| + \nu / L_r}, \tag{4.29}$$

so that $T_r^\star$ is constant on the element; this ensure that at the steady state the relations $\frac{\partial u}{\partial x} = p$ and $\frac{\partial u}{\partial y} = q$ will be satisfied in the integral sense. The total residual can be now written as

$$\begin{aligned}\Phi^e &= \int_{\Omega_e} \Big(\boldsymbol{\nabla} \cdot \mathbf{f}(\mathsf{u}^h) - \mathsf{S}(\mathsf{u}^h) \Big) \, \mathrm{d}\Omega \\ &= \oint_{\partial\Omega_e} \mathbf{f}(\mathsf{u}) \cdot n \, \mathrm{d}\partial\Omega - \int_{\Omega_e} \mathsf{S}(\mathsf{u}^h) \, \mathrm{d}\Omega,\end{aligned} \tag{4.30}$$

and RD schemes developed for hyperbolic systems of equations can be used to distribute the residual to each degree of freedom of the element, see Chapter 5.

4.3 RD Discretization of Advection-Diffusion Problems

The RD numerical schemes introduced in Chapter 3 for advection problems are now extended to the advection-diffusion case. Having in mind the goal to discretize the compressible Navier-Stokes equations on hybrid grids with second and third order accuracy, the attention is focused here only on the linear scheme (3.57) and the non-linear one (3.66) due to their flexibility compared to multidimensional upwind schemes. Furthermore, purely upwind schemes are expected to be not suited when diffusive effects prevail on the advective ones. In fact, some authors [31, 85] used a modified version of the distribution coefficients of the LDA scheme in order to recover an isotropic scheme in the diffusion limit.

4.3.1 Central Linear and Non-Linear RD Schemes

Given the total residual computed as in Eq. (4.3), the linear scheme for advection-diffusion problems reads

$$\Phi_i^{e,\mathrm{LW}} = \frac{\Phi^e}{N_{\mathrm{dof}}^e} + \int_{\Omega_e} \boldsymbol{a} \cdot \boldsymbol{\nabla} \psi_i \, \tau \left(\boldsymbol{a} \cdot \boldsymbol{\nabla} u_h - \boldsymbol{\nabla} \cdot \left(\nu \widetilde{\boldsymbol{\nabla} u^h} \right) \right) \mathrm{d}\Omega \tag{4.31}$$

where the scaling parameter τ is defined as follows

$$\tau = \frac{1}{2} \frac{|\Omega_e|}{\sum_{j \in \Sigma_h^e} \max(k_j, 0) + \nu}, \quad \text{with} \quad k_j = \frac{1}{2} \bar{\boldsymbol{a}} \cdot \boldsymbol{n}_j, \tag{4.32}$$

The scheme (4.31) is linearity preserving but not positive. An RD scheme which is high order preserving and monotone is given by the following non-linear scheme

$$\begin{aligned}\hat{\Phi}_i^{e,\mathrm{Rv}} = &\, \hat{\beta}_i^{e,\mathrm{Rv}}(u_h) \Phi^e(u_h) \\ &+ \theta_h^e(u_h) \int_{\Omega_e} \Big(\boldsymbol{a} \cdot \boldsymbol{\nabla} \psi_i - \boldsymbol{\nabla} \cdot \left(\nu \boldsymbol{\nabla} \psi_i \right) \Big) \left(\boldsymbol{a} \cdot \boldsymbol{\nabla} u_h - \boldsymbol{\nabla} \cdot \left(\nu \widetilde{\boldsymbol{\nabla} u_h} \right) \right) \mathrm{d}\Omega,\end{aligned} \tag{4.33}$$

The parameter $\theta_h^e(u^h)$ is taken as follows

$$\theta_h^e(u^h) = \varepsilon(u^h) \left(\frac{\sum_{j \in \Sigma_h^e} |\bar{\boldsymbol{a}} \cdot \boldsymbol{n}_j| + \nu}{2|\Omega_e|} \right)^{-1}, \tag{4.34}$$

with $\varepsilon(u^h)$ a smoothness sensor. The non-linear distribution coefficients, $\hat{\beta}_i^{e,\mathrm{Rv}}$, are obtained from the Rusanov's scheme, by applying the non-linear limiting technique described in Section 3.4.1, and a symmetric form is used in the filtering term.

Note that although the integrals at the second member of the linear and non-linear schemes are strictly computed on the element interior, for sake of robustness and consistency with the calculation of the total residual, the reconstructed gradient is used in place of the internal gradient for the viscous terms.

4.3.2 Improved Discretization of the Diffusion Terms

Numerical experiments reveal that the schemes (4.31) and (4.33) applied to the discretization of advection-diffusion problems are unsatisfactory from the point of view the robustness. In order to obtain a better discretization of the diffusive terms, the advection-diffusion equation (4.1) is written in the form of a first order system as follows

$$\begin{cases} \boldsymbol{\nabla} \cdot \boldsymbol{f}(u) - \boldsymbol{\nabla} \cdot (\nu \boldsymbol{q}) = 0 \\ \boldsymbol{q} - \boldsymbol{\nabla} u = 0 \end{cases} \tag{4.35}$$

Consider now a numerical scheme for the previous system obtained by writing the weak form of the system plus a streamline stabilization term. With an abuse of notation it is possible to write

$$\begin{aligned} &\int_{\Omega_e} \psi_i \begin{pmatrix} \boldsymbol{\nabla} \cdot \boldsymbol{f}(u_h) - \boldsymbol{\nabla} \cdot (\nu \boldsymbol{q}_h) \\ \boldsymbol{q}_h - \boldsymbol{\nabla} u_h \end{pmatrix} \mathrm{d}\Omega \\ &\qquad + \int_{\Omega_e} \boldsymbol{A} \cdot \boldsymbol{\nabla} \psi_i \, \boldsymbol{\tau} \begin{pmatrix} \boldsymbol{\nabla} \cdot \boldsymbol{f}(u_h) - \boldsymbol{\nabla} \cdot (\nu \boldsymbol{q}_h) \\ \boldsymbol{q}_h - \boldsymbol{\nabla} u_h \end{pmatrix} \mathrm{d}\Omega = 0, \end{aligned} \tag{4.36}$$

where $\boldsymbol{A} = (A_x, A_y)$ with

$$A_x = \begin{pmatrix} a_x & -\nu & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \quad \text{and} \quad A_y = \begin{pmatrix} a_y & 0 & -\nu \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{pmatrix}, \tag{4.37}$$

so that

$$\boldsymbol{A} \cdot \boldsymbol{\nabla} \psi_i = \begin{pmatrix} \boldsymbol{a} \cdot \boldsymbol{\nabla} \psi_i & -\nu \dfrac{\partial \psi_i}{\partial x} & -\nu \dfrac{\partial \psi_i}{\partial y} \\ -\dfrac{\partial \psi_i}{\partial x} & 0 & 0 \\ -\dfrac{\partial \psi_i}{\partial y} & 0 & 0 \end{pmatrix}. \tag{4.38}$$

The term $\boldsymbol{\tau}$ is assumed to be of the following form

$$\boldsymbol{\tau} = \begin{pmatrix} \tau_a & 0 & 0 \\ 0 & \tau_d & 0 \\ 0 & 0 & \tau_d \end{pmatrix}, \tag{4.39}$$

where τ_a and τ_d are strictly positive coefficients.

Supposing, now, that the gradient of the numerical solution has been reconstructed at each degree of freedom, one can replace the second equation of the system (4.35) with the approximation $\boldsymbol{\nabla} u^u \simeq \widetilde{\boldsymbol{\nabla} u_h}$ and consider only the first equation, which now reads

$$\begin{aligned} &\int_{\Omega_e} \psi_i \left(\boldsymbol{\nabla} \cdot \boldsymbol{f}(u^h) - \boldsymbol{\nabla} \cdot (\nu \widetilde{\boldsymbol{\nabla} u^h}) \right) \mathrm{d}\Omega \\ &+ \int_{\Omega_e} \boldsymbol{a} \cdot \boldsymbol{\nabla} \psi_i \, \tau_c \left(\boldsymbol{a} \cdot \boldsymbol{\nabla} u^h - \boldsymbol{\nabla} \cdot \left(\nu \widetilde{\boldsymbol{\nabla} u^h} \right) \right) \mathrm{d}\Omega \\ &+ \int_{\Omega_e} \nu \, \boldsymbol{\nabla} \psi_i \cdot \left(\tau_d \left(\boldsymbol{\nabla} u^h - \widetilde{\boldsymbol{\nabla} u^h} \right) \right) \mathrm{d}\Omega = 0. \end{aligned} \tag{4.40}$$

The first two integrals of the previous equation represent a discretization method for the scalar advection-diffusion equation with a central scheme plus a streamline stabilization term, in the same way as shown in the Eq. (4.31). The last integral represents an additional stabilization term, for the diffusive part only, which vanishes in the advective limit and the parameter τ_d is dimensionless. It is interesting to note that the additional term penalizes the difference between the discontinuous and the interpolated gradient, on each element.

With a slightly different procedure, a similar stabilization term for the diffusive part has been obtain by Nishikawa [84] for the RD discretization of diffusion problems.

With the Eq. (4.40) in mind, it is proposed here a modification of the schemes (4.31) and (4.33), previously introduced, in order to include the extra stabilization term for the diffusive part of the equation. In practice, the linear scheme reads

$$\begin{aligned} \Phi_i^{e,\mathrm{LW}} = \frac{\Phi^e}{N_{\mathrm{dof}}^e} &+ \Upsilon(\mathrm{Pe}^{\mathrm{e}}) \int_{\Omega_e} \boldsymbol{a} \cdot \boldsymbol{\nabla} \psi_i \, \tau \left(\boldsymbol{a} \cdot \boldsymbol{\nabla} u_h - \boldsymbol{\nabla} \cdot \left(\nu \boldsymbol{\nabla} u_h \right) \right) \mathrm{d}\Omega \\ &+ \left(1 - \Upsilon(\mathrm{Pe}^{\mathrm{e}}) \right) \int_{\Omega_e} \nu \, \boldsymbol{\nabla} \psi_i \cdot \left(\boldsymbol{\nabla} u^h - \widetilde{\boldsymbol{\nabla} u^h} \right) \mathrm{d}\Omega, \end{aligned} \tag{4.41}$$

while the non-linear scheme becomes

$$\begin{aligned} \hat{\Phi}_i^{e,\mathrm{Rv}} = &\, \hat{\beta}_i^{e,\mathrm{Rv}}(u_h) \Phi^e(u_h) \\ &+ \Upsilon(\mathrm{Pe}^{\mathrm{e}}) \, \theta_h^e(u_h) \int_{\Omega_e} \left(\boldsymbol{a} \cdot \boldsymbol{\nabla} \psi_i - \boldsymbol{\nabla} \cdot \left(\nu \boldsymbol{\nabla} \psi_i \right) \right) \left(\boldsymbol{a} \cdot \boldsymbol{\nabla} u_h - \boldsymbol{\nabla} \cdot \left(\nu \boldsymbol{\nabla} u_h \right) \right) \mathrm{d}\Omega \\ &+ \left(1 - \Upsilon(\mathrm{Pe}^{\mathrm{e}}) \right) \int_{\Omega_e} \nu \, \boldsymbol{\nabla} \psi_i \cdot \left(\boldsymbol{\nabla} u_h - \widetilde{\boldsymbol{\nabla} u^h} \right) \mathrm{d}\Omega, \end{aligned} \tag{4.42}$$

where it has been introduced the local Peclet number, define as $\mathrm{Pe}^{\mathrm{e}} = \|\boldsymbol{a}\| h^e/\nu$, with h^e the characteristic length of the element e, the function $\Upsilon(\mathrm{Pe}^{\mathrm{e}})$ is defined such that $\Upsilon(\mathrm{Pe}^{\mathrm{e}}) \to 0$ in the diffusive limit and $\Upsilon(\mathrm{Pe}^{\mathrm{e}}) \to 1$ in the advective limit. In the numerical simulations the following definition is used

$$\Upsilon(\mathrm{Pe}^{\mathrm{e}}) = \max\left(0, 1 - \frac{1}{\mathrm{Pe}^{\mathrm{e}}}\right). \tag{4.43}$$

Note that in the schemes (4.41) or (4.42), the use of the blending function $\Upsilon(\mathrm{Pe}^{\mathrm{e}})$, makes possible to recover, in the case of the pure advection, the same scheme used for the discretization of pure advective problems, while in the case of pure diffusion problems only the stabilization term for the diffusive terms is taken into account.

4.3.3 Discretization of the Hyperbolic FOS

The schemes introduced for the scalar advection-diffusion problem can be easily extended to case of system of equations, see Chapter 5, meaning that the discretization of the hyperbolic FOS is straightforward.

With respect to the original work of Nishikawa, where only strong boundary conditions are considered, here the boundary conditions are imposed in a weak sense as typical done for advection problems. The weak boundary conditions have been found to be more effective than the strong boundary conditions in terms of iterative convergence.

In numerical experiments, it has been observed that the high-order discretization of the hyperbolic FOS converges very slowly to the steady state, making the use of a explicit scheme almost impossible. For this reasons, when quadratic elements are used, an implicit Euler scheme is employed in combination with the pseudo-transient continuation strategy for which the local time step is defined as follows

$$\Delta t_i^n = \frac{\mathrm{CFL}^n}{\displaystyle\sum_{e \in \mathcal{E}_h^i} \left(\max_{j \in \mathcal{N}_h^e} |k_j^n| + \nu \right)}, \tag{4.44}$$

with the CFL law taken as

$$\mathrm{CFL}^n = \mathrm{CFL}^{n-1} \frac{\|R^{n-2}\|_{L^2}}{\|R^{n-1}\|_{L^2}}, \quad \text{with } \mathrm{CFL}^0 < 1, \tag{4.45}$$

where $\|R^{n-1}\|_{L^2}$ and $\|R^{n-2}\|_{L^2}$ are the L^2 norms of the residual at the time steps $n-1$ and $n-2$, respectively. In the simulation CFL^0 is takes as 0.9 and the maximum value of CFL is limited to 10^6. The implicit problem is solved by the means of the inexact Newton-Krylov method and the GMRES algorithm with the ILU0 preconditioner is used to solve the resulting linear systems.

4.4 Gradient Reconstruction Strategies

As previously explained, in order to compute the total residual for the whole advection-diffusion equation, one has to assume that a continuous value of the gradient of the numerical solution is available on the faces of the elements. The strategy adopted in this work to obtain a continuous approximation of gradient consists in reconstructing the gradient of the numerical solution at each degree of freedom of the grid and then the nodal values of gradients are interpolated with Lagrangian functions on each element.

The key point is the reconstruction of the gradient at the degrees of freedom, for this reason here are recalled some of the most used techniques in the field of the gradient reconstruction. Attention is focused on the possibility to obtain an high-order gradient reconstruction, *e.g.*, the gradient is recovered with the same order of accuracy of the solution. For simplicity, the description is always limited to the two-dimensional case.

4.4.1 Theory

Area-Weighted Method

One of the easiest way to reconstruct the gradient at the grid nodes is the area-weight average of the gradients in each element surrounding a node, namely

$$\widetilde{\boldsymbol{\nabla} u_i} = \frac{\displaystyle\sum_{e \in \mathcal{E}_h^i} \boldsymbol{\nabla} u_h(\boldsymbol{x}_i)\, |\Omega_e|}{\displaystyle\sum_{e \in \mathcal{E}_h^i} |\Omega_e|}, \quad \forall i \in \mathcal{N}_h, \tag{4.46}$$

The previous relation, in the case of linear elements is the so-called Green-Gauss formula.

L2-Projection

In the L2-Projection, the reconstructed gradients are obtained by solving the following equivalence $\boldsymbol{\nabla} u_h = \widetilde{\boldsymbol{\nabla} u_h}$ in a weak sense

$$\int_\Omega \psi \widetilde{\boldsymbol{\nabla} u_h}\, \mathrm{d}\Omega = \int_\Omega \psi \boldsymbol{\nabla} u_h\, \mathrm{d}\Omega, \qquad \forall \psi \in V_h. \tag{4.47}$$

From a numerical point of view, the weight function ψ is taken in the finite dimensional space of the Lagrangian functions. The gradient is expressed as follows

$$\widetilde{\boldsymbol{\nabla} u_h}|_e \simeq \sum_{j \in \mathcal{N}_h^e} \psi_j \widetilde{\boldsymbol{\nabla} u_j}, \tag{4.48}$$

where $\widetilde{\boldsymbol{\nabla} u_j}$ is the reconstructed gradient at the generic degree of freedom j. If, for the spatial components of the gradient, the following vectors of unknowns are

defined

$$\boldsymbol{x}_\partial = \left[\left. \widetilde{\frac{\partial u}{\partial x}} \right|_j \right]_{j=1,N_{\text{dof}}} \quad \text{and} \quad \boldsymbol{y}_\partial = \left[\left. \widetilde{\frac{\partial u}{\partial y}} \right|_j \right]_{j=1,N_{\text{dof}}}, \tag{4.49}$$

than the discrete solution of the problem (4.47) can be obtained by solving the following linear systems

$$M\boldsymbol{x}_\partial = \boldsymbol{b}_x \quad \text{and} \quad M\boldsymbol{y}_\partial = \boldsymbol{b}_y, \tag{4.50}$$

with

$$M_{ij} = \int_{\Omega_{ij}} \psi_i \psi_j \, \mathrm{d}\Omega, \quad b_{x_i} = \int_{\Omega_i} \psi_i \frac{\partial u^h}{\partial x} \, \mathrm{d}\Omega, \quad \text{and} \quad b_{y_i} = \int_{\Omega_i} \psi_i \frac{\partial u^h}{\partial y} \, \mathrm{d}\Omega, \tag{4.51}$$

where Ω_i is the support of the shape function ψ_i and $\Omega_{ij} = \Omega_i \cap \Omega_j$. The gradient $\boldsymbol{\nabla} u^h$ is computed by resorting the gradient of the shape functions, as standard practice in the FE framework.

This technique requires the solution of a global linear system that can be quite expensive for a high number of unknowns. Obliviously, since the matrix M depends only on the geometry of the grid, it can be inverted only once and can be used for several calculations on the same grid.

Least-Square Method

Another approach to reconstruct the gradient at each degree of freedom of the grid is the least-square method. The technique is unrelated to the mesh topology and it involves only the information associated to the neighboring nodes. Although the stencil is arbitrary, the natural choice involves only the nearest neighboring nodes.

The starting point consists in expanding the solution in a Taylor series around the node i for each node j belonging to the stencil of i, see Figure 4.2,

$$\begin{aligned} u_j = u_i &+ \left. \frac{\partial u}{\partial x} \right|_i (x_j - x_i) + \left. \frac{\partial u}{\partial y} \right|_i (y_j - y_i) \\ &+ \left. \frac{\partial^2 u}{\partial x^2} \right|_i (x_j - x_i)^2 + \left. \frac{\partial^2 u}{\partial y^2} \right|_i (y_j - y_i)^2 + \left. \frac{\partial^2 u}{\partial x \partial y} \right|_i (x_j - x_i)(y_j - y_i) + \dots, \end{aligned} \tag{4.52}$$

where $u_i = u(\boldsymbol{x}_i)$ and $u_j = u(\boldsymbol{x}_j)$. The gradient reconstruction is obtained by solving for the values of the gradient that minimize the following function

$$\sum_{j=1}^{N_i} \omega_{ij}^2 E_{ij}^2, \qquad \forall \; i \in \mathcal{N}_h \tag{4.53}$$

with

$$\begin{aligned} E_{ij}^2 = \Bigg(&-\Delta u_{ij} + \left. \frac{\partial u}{\partial x} \right|_i \Delta x_{ij} + \left. \frac{\partial u}{\partial y} \right|_i \Delta y_{ij} \\ &+ \left. \frac{\partial^2 u}{\partial x^2} \right|_i \Delta x_{ij}^2 + \left. \frac{\partial^2 u}{\partial y^2} \right|_i \Delta y_{ij}^2 + \left. \frac{\partial^2 u}{\partial x \partial x} \right|_i \Delta x_{ik} \Delta y_{ij} + \dots \Bigg)^2, \end{aligned} \tag{4.54}$$

where $\Delta u_{ij} = u_j - u_i$, $\Delta x_{ij} = x_j - x_i$, $\Delta y_{ij} = y_j - y_i$, while ω_{ij} is a weight factor. In the case of linear elements, the solution is expanded only up to the first derivatives in the Taylor series and the components of the gradient are obtained by solving the following minimization problems for the first derivatives

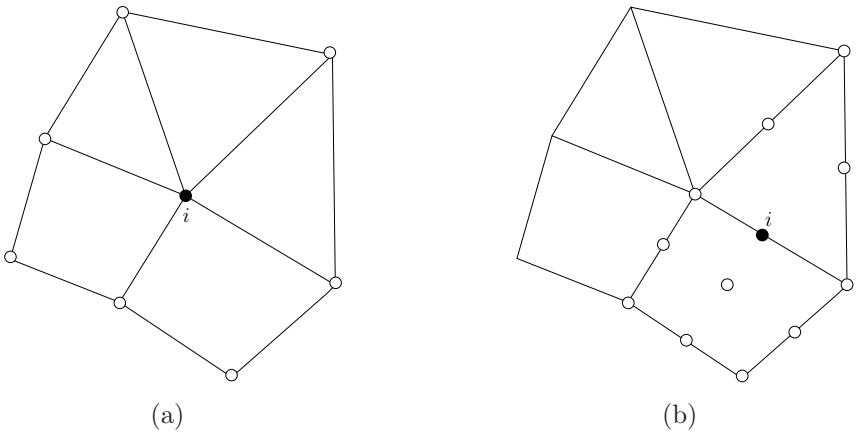
$$\frac{\partial \left(\sum_{j=1}^{N} \omega_{ij}^2 E_{ij}^2 \right)}{\partial \left(\left. \frac{\partial u}{\partial x} \right|_i \right)} = 0 \qquad \text{and} \qquad \frac{\partial \left(\sum_{j=1}^{N} \omega_{ij}^2 E_{ij}^2 \right)}{\partial \left(\left. \frac{\partial u}{\partial y} \right|_i \right)} = 0. \tag{4.55}$$

By simple algebra, it is easy to see that the previous minimization problems correspond to the solution of the following linear system

$$\begin{pmatrix} \sum_{j=1}^{N} \omega_{ij}^2 \Delta x_{ij}^2 & \sum_{j=1}^{N} \omega_{ij}^2 \Delta x_i \Delta y_{ij} \\ \sum_{j=1}^{N} \omega_{ij}^2 \Delta x_{ij} \Delta y_{ij} & \sum_{j=1}^{N} \omega_{ij}^2 \Delta y_{ij}^2 \end{pmatrix} \begin{pmatrix} \left. \frac{\partial u}{\partial x} \right|_i \\ \\ \left. \frac{\partial u}{\partial y} \right|_i \end{pmatrix} = \begin{pmatrix} \sum_{j=1}^{N} \omega_{ij}^2 \Delta x_{ij} \Delta u_{ij} \\ \sum_{j=1}^{N} \omega_{ij}^2 \Delta y_{ij} \Delta u_{ij} \end{pmatrix}. \tag{4.56}$$

The weight factor ω_{ij} is generally taken as the inverse of the distance between the nodes i and j.

The extension to the case of quadratic elements is straightforward, it consists in taking also the second derivatives in the Taylor expansion and the minimization is done with respect to first and second derivatives.



(a) (b)

Figure 4.2: Illustration of the stencil for least square gradient reconstruction at the node i. The symbol (•) indicates the node around which the Taylor series expansion is done while the symbols (○) indicate the node used to construct the least square problem. Left: stencil with linear elements. Right: stencil with quadratic elements for a midpoint node.

Super-Convergent Patch Recovery

In the field of the FE method applied to mechanical structures, it is know that the stresses (gradients of the displacements) sampled at certain points in the element possess a super-convergent property, thus the stresses can be computed with the same order of accuracy of the displacements [142]. It can be shown that in the case of a segment element, such particular points correspond to the Gauss-Legendre points [56], obviously by tensor product such points can be defined also for quadrangles and hexahedra. For triangles or tetrahedra such a property cannot be rigorously shown, but numerical experiments confirm that the stresses sampled at certain points have higher order of accuracy.

Accepting the fact that gradients are sampled with higher order accuracy in certain points of the element, it is possible to compute gradients with good accuracy within all the element. Indeed, if at sampling points the values of the gradient are accurate to the order $k+1$, by using a polynomial of degree k (the same order used to interpolate the solution) it is possible to obtain an approximation which has high order accuracy everywhere within the element, if this polynomial is made to fit the values of the sampled gradients in a least square manner. Such a technique, called super-convergent patch recovery, has been introduced by Zienkiewicz and Zhu (SPR-ZZ) [140, 141].

Consider, for simplicity, a two dimensional scalar problem and assume that u_h is the piecewise continuous polynomial interpolation of the solution, with order k. The aim is to obtain a value for the reconstructed gradient, $\widetilde{\boldsymbol{\nabla} u_h}$ at all the degrees of freedom, with the same order of accuracy of the solution. The components of the recovered gradient, at the generic node i, are written in a polynomial form as follows

$$\left.\widetilde{\frac{\partial u_h}{\partial x}}\right|_i = \boldsymbol{p}^{\mathrm{T}}\boldsymbol{a}_x \quad \text{and} \quad \left.\widetilde{\frac{\partial u_h}{\partial y}}\right|_i = \boldsymbol{p}^{\mathrm{T}}\boldsymbol{a}_y, \tag{4.57}$$

with

$$\boldsymbol{p}^{\mathrm{T}}(\boldsymbol{x}) = (1, x, y, x^2, \ldots, x^k, x^{k-1}y, \ldots, y^k), \tag{4.58}$$

$$\boldsymbol{a}_x = (a_{x_1}, a_{x_2}, \ldots, a_{x_m}) \quad \text{and} \quad \boldsymbol{a}_y = (a_{y_1}, a_{y_2}, \ldots, a_{y_m}). \tag{4.59}$$

Assuming that N_s sampling points, (x_j, y_j), $j = 1 \ldots N_s$, are available for each grid vertex i, the objective is to minimize the following functions

$$F_x = \sum_{k=1}^{N_s} \left(\frac{\partial u_h}{\partial x}(\boldsymbol{x}_k) - \boldsymbol{p}_k^{\mathrm{T}}\boldsymbol{a}_x \right) \quad \text{and} \quad F_y = \sum_{k=1}^{N_s} \left(\frac{\partial u_h}{\partial y}(\boldsymbol{x}_k) - \boldsymbol{p}_k^{\mathrm{T}}\boldsymbol{a}_y \right), \tag{4.60}$$

with $\boldsymbol{p}_k = \boldsymbol{p}(\boldsymbol{x}_k)$. The vectors of the coefficients $\boldsymbol{a}_x$ and $\boldsymbol{a}_y$ are obtained by solving the following minimization problems

$$\frac{\partial F_x}{\partial \boldsymbol{a}_x} = 0 \quad \text{and} \quad \frac{\partial F_y}{\partial \boldsymbol{a}_y} = 0. \tag{4.61}$$

It is easy to verify that the minimization problems correspond to the solution of the following linear systems [1]

$$A^{\mathrm{T}}A\boldsymbol{a}_x = A^{\mathrm{T}}\boldsymbol{b}_x^h, \quad \text{and} \quad A^{\mathrm{T}}A\boldsymbol{a}_y = A^{\mathrm{T}}\boldsymbol{b}_y^h, \tag{4.62}$$

where

$$\boldsymbol{b}_x^h = \begin{pmatrix} \dfrac{\partial u^h}{\partial x}(\boldsymbol{x}_1) \\ \dfrac{\partial u^h}{\partial x}(\boldsymbol{x}_2) \\ \vdots \\ \dfrac{\partial u^h}{\partial x}(\boldsymbol{x}_{N_s}) \end{pmatrix}, \quad \boldsymbol{b}_y^h = \begin{pmatrix} \dfrac{\partial u^h}{\partial y}(\boldsymbol{x}_1) \\ \dfrac{\partial u^h}{\partial y}(\boldsymbol{x}_2) \\ \vdots \\ \dfrac{\partial u^h}{\partial y}(\boldsymbol{x}_{N_s}) \end{pmatrix} \quad \text{and} \quad A = \begin{pmatrix} 1 & x_1 & y_1 & \dots & y_1^k \\ 1 & x_2 & y_2 & \dots & y_2^k \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_{N_s} & y_{N_s} & \dots & y_{N_s}^k \end{pmatrix}. \tag{4.63}$$

To compute the coefficients $\boldsymbol{a}_x$ and $\boldsymbol{a}_y$ a small linear system must be solved for each vertex of the grid. The dimension of the matrix A are determined by the number of the sampling points (N_s) and by the degree of the polynomial used to express the reconstructed gradient, that is $A \in \mathbb{R}^{N_s \times m}$, where m is the number of the coefficients in the vectors $\boldsymbol{a}_x$ and $\boldsymbol{a}_y$. The problems in the Eq. (4.62) admit an unique solution if $\mathrm{Rank}\, A = m$, that is always satisfied in the case in which $N_s \geq m$. It is worth also noticing that, since the matrix A depends only on the geometry, for a given grid, the matrix $(A^{\mathrm{T}}A)^{-1}A^{\mathrm{T}}$ needs to be computed only once.

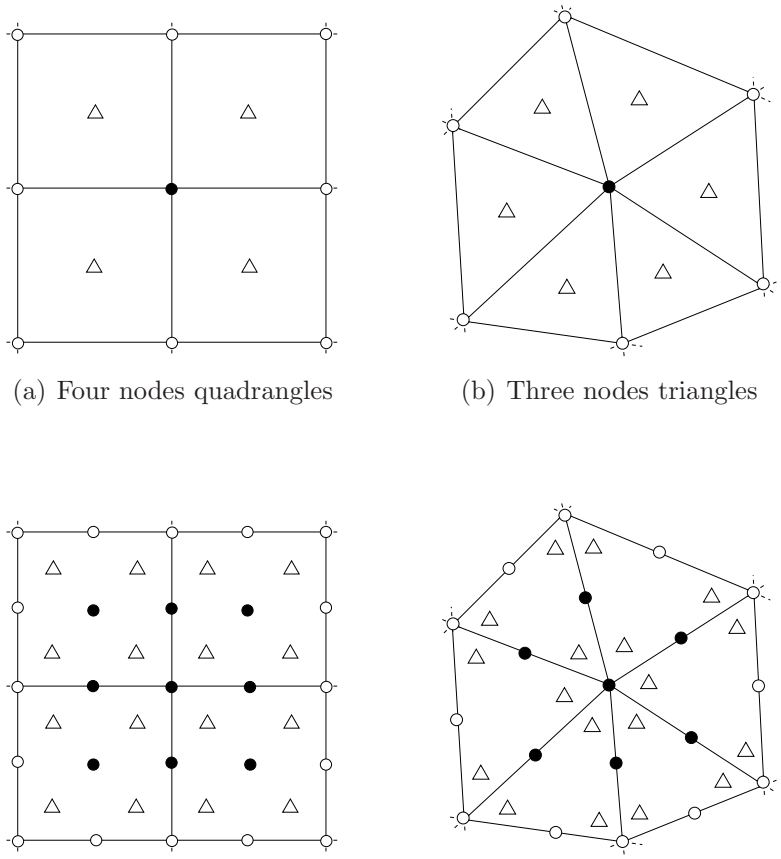
The least-square problem is solved for each grid vertex, but not for the extra nodes introduced by a higher approximation of the solution. For these nodes, the gradient is reconstructed by simply evaluating at the coordinates of the nodes the polynomial function constructed for the nearest grid vertex. For each node associated with the higher order degrees of freedom, there may be two or more equidistant grid vertices, each of them is equally valid to evaluate the gradient. To resolve this ambiguity, an arithmetic average of the values of gradients, evaluated using the polynomial expansion of all the equidistant vertices, is used to uniquely define the reconstructed gradient at the high-order nodes.

Generally, the number of elements that share the same node is such that the condition $N_s \geq m$ is always satisfied, this means that the gradient reconstruction is compact because it involves only the elements contained in the support of a grid node. For the nodes belonging to the boundary of the domain, the condition $N_s \geq m$ might not be satisfied without enlarging the stencil, otherwise the problem will be ill conditioned. In this case, to avoid the use of larger stencils for the boundary nodes, it is possible to obtain the value of the reconstructed gradient with the same polynomial expansion used for the nearest domain node.

In Figure 4.3 are shown examples of patches used to reconstruct the gradient for a domain node in the case of quadrangular and triangular elements. Note that, although in this section the description of the SPR-ZZ technique is given only for

[1] The least-square problem could be solved also with a QR factorization of the matrix A.

two dimensional elements, the extension to three dimensional elements is straightforward.



(a) Four nodes quadrangles (b) Three nodes triangles

(c) Nine nodes quadrangles (d) Six nodes triangles

Figure 4.3: Interior super-convergent patches for quadrilateral and triangular elements: top linear elements, bottom quadratic elements. The symbols (∘) indicate the patch assembly points, the symbols (•) indicate the points where the gradient is reconstructed and the symbols (△) indicate the super-convergent sampling points.

For a quadrangle, the sampling points are defined uniquely. Considering a reference segment defined as $\xi = [-1, 1]$, the sampling point is the point $\xi = 0$ in the case of a linear element, while in the case of a quadratic element the sampling points have coordinate $\xi = \pm 1/\sqrt{3}$. The sampling points on the reference quadrangle are simply obtained by tensor product of the points defined on the reference segment.

For a linear triangle, the sampling point is the point with barycentric coordinates $\boldsymbol{\lambda} = (1/3, 1/3, 1/3)$, while in the case of a quadratic triangle the sampling points are not unique; different choices are available. In Figure 4.4 are shown three examples of sampling points used. In the first option (Figure 4.4-(a)) are used three points

with barycentric coordinates

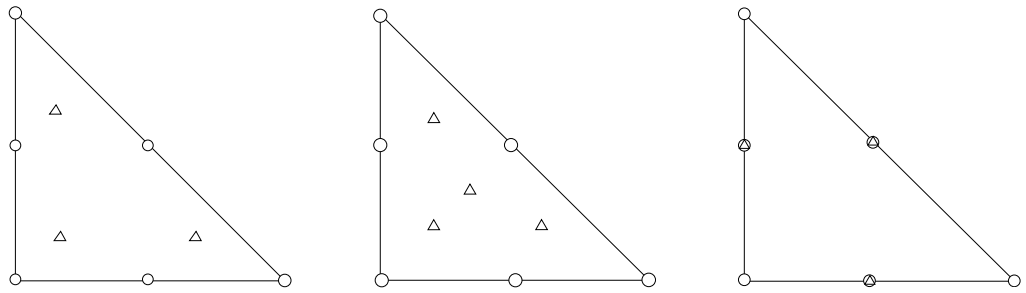
$$\boldsymbol{\lambda}_1 = \left(\frac{2}{3}, \frac{1}{3}, \frac{1}{3}\right), \quad \boldsymbol{\lambda}_2 = \left(\frac{1}{3}, \frac{2}{3}, \frac{1}{3}\right), \quad \text{and} \quad \boldsymbol{\lambda}_3 = \left(\frac{1}{3}, \frac{1}{3}, \frac{2}{3}\right). \tag{4.64}$$

in the second option (Figure 4.4-(b)) are used four points with barycentric coordinates

$$\begin{aligned} \boldsymbol{\lambda}_1 &= \left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right), \qquad \boldsymbol{\lambda}_2 = (0.6, 0.2, 0.2), \\ \boldsymbol{\lambda}_3 &= (0.2, 0.6, 0.2), \quad \boldsymbol{\lambda}_4 = (0.2, 0.2, 0.6). \end{aligned} \tag{4.65}$$

Another option (Figure 4.4-(c)) consists in taking as sampling points the three points with barycentric coordinates

$$\boldsymbol{\lambda}_1 = \left(\frac{1}{2}, \frac{1}{2}, 0\right), \quad \boldsymbol{\lambda}_2 = \left(0, \frac{1}{2}, \frac{1}{2}\right), \quad \boldsymbol{\lambda}_3 = \left(\frac{1}{2}, \frac{1}{2}, 0\right). \tag{4.66}$$



(a) (b) (c)

Figure 4.4: Three different examples of sampling points for quadratic triangles.

4.4.2 Results and Discussion

To study the accuracy of the presented gradient reconstruction strategies, the following function is used

$$u = -\cos(2\pi\eta)\exp\left(\frac{\xi\left(1-\sqrt{1+16\pi^2\nu^2}\right)}{2\nu}\right), \tag{4.67}$$

with $\eta = a_y x - a_x y$ and $\xi = a_x x + a_y$. Here $a_x = 0.5$, $a_y = \sqrt{3}/2$ and $\nu = 0.01$. The solution, shown in Figure 4.5, is infinitely differentiable with continuous gradient. The computations of the reconstructed gradient are performed on four different type of grids, shown in Figure 4.6, namely unstructured grids of triangles, quadrangles and hybrid elements and highly distorted unstructured meshes of triangles, obtained randomly perturbing regular grids.

The SPR-ZZ technique is compared against the weighted-area (hereafter called also Green-Gauss), L^2-projection and least-square procedures. Note that in the least-square method, for each node, only the direct neighboring nodes are considered, to avoid large computational stencils.

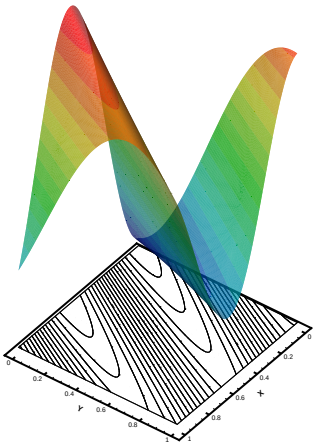


Figure 4.5: Graphical representation of the Eq. (4.67) used to test the gradient reconstruction methods.

The error of the reconstruction procedure is computed as the L^2 norm of the difference between the computed gradient, $\widetilde{\boldsymbol{\nabla} u^h}$, and the exact gradient, $\boldsymbol{\nabla} u_{\text{ex}}$, for each spatial component

$$\epsilon_{L^2} = \sqrt{\frac{\int_\Omega \left(\widetilde{\boldsymbol{\nabla} u^h} - \boldsymbol{\nabla} u_{\text{ex}}\right)^2 \mathrm{d}\Omega}{\int_\Omega \left(\boldsymbol{\nabla} u_{\text{ex}}\right)^2 \mathrm{d}\Omega}}. \tag{4.68}$$

In Figure 4.7 are shown the L^2 errors of different gradient reconstruction methods with triangular grids. In the case of linear elements, the differences between the reconstruction procedures are small and all the schemes reach almost the second order accuracy. The L^2-Projection and the SPR-ZZ methods have the smallest level of error, but the former scheme is more expensive because it requires the solution of a global (large) linear system. In the case of quadratic elements, the methods have an order of accuracy no more than two, except for the SPR-ZZ method which is third order accurate. It is also worth noticing that the errors obtained with the SPR-ZZ method are one order of magnitude smaller than those obtained with other methods. In Figure 4.8 are reported the errors obtained with the SPR-ZZ procedure on quadratic triangular elements, for the three different sampling strategies shown in Figure 4.4. It is evident that the first strategy guaranties the smallest level of error, while the four-points strategy does not introduce any improvement. The first strategy has been used to produce the results of Figure 4.7, for the SPR-ZZ method.

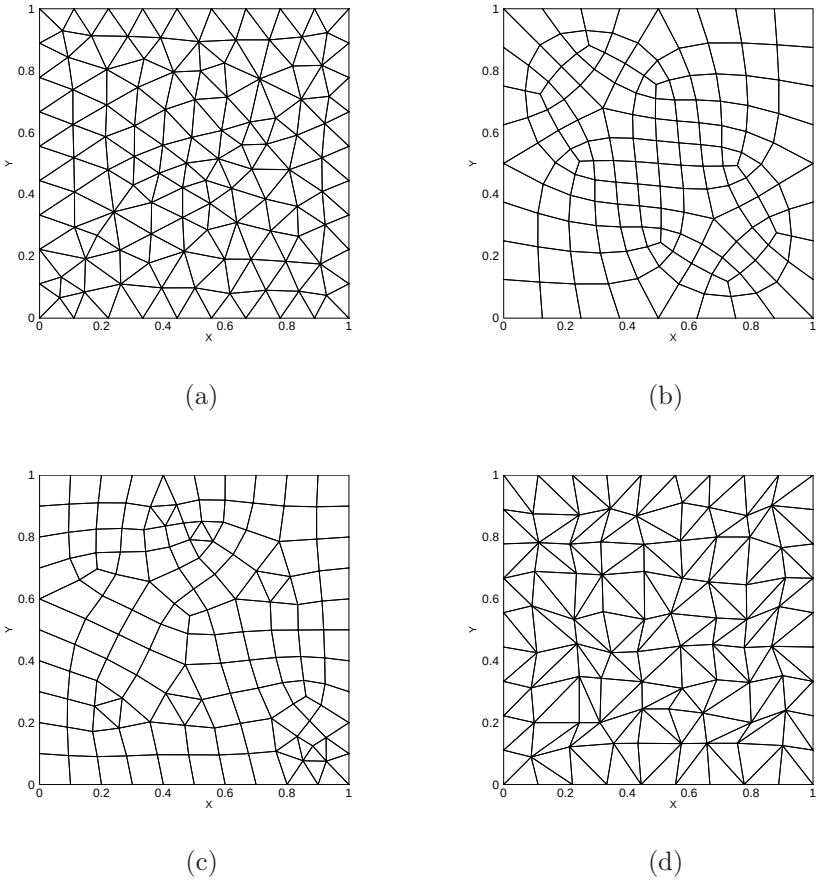


Figure 4.6: Example of different type of grid used to test the accuracy of the gradient reconstruction procedures.

The errors of the reconstruction methods on unstructured grids of quadrangles and of hybrid elements are reported in Figure 4.9 and Figure 4.10, respectively. The behavior of the reconstruction methods is the same observed in the case of triangular grids. Figure 4.11 shows the errors computed on a sequence of highly distorted triangular grids, the performance of the reconstruction methods is not optimal anymore due to very poor quality of the meshes, nevertheless the errors obtained with the SPR-ZZ methods are always much smaller than those obtained with other procedures.

4.5 Numerical Experiments

This section presents an extensive evaluation of the numerical schemes proposed. The objective is to show that

i) high-order RD schemes previously proposed can be successfully used in the

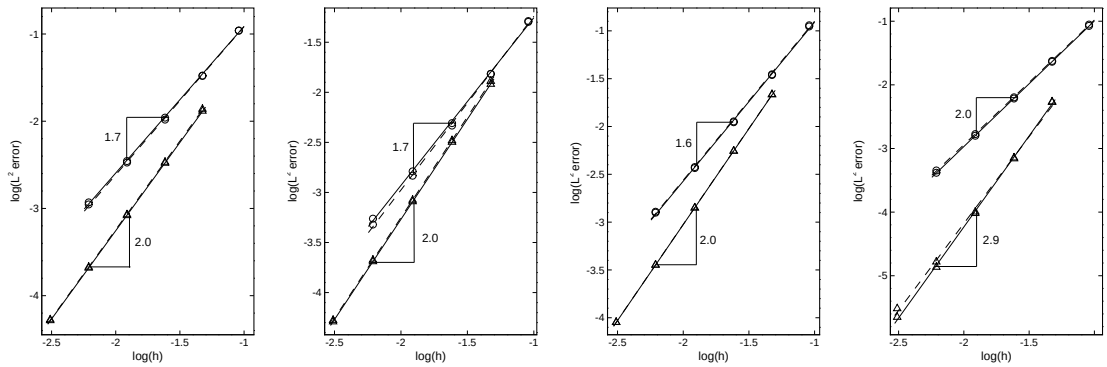


Figure 4.7: L^2 errors of different gradient reconstruction methods with linear ($\circ$) and quadratic ($\triangle$) elements on unstructured grids of triangles. Solid and dashed lines represent the error of the x and y components of the gradient, respectively. From left to right: Weighted area, L^2-Projection, Least Square and SPR-ZZ methods. The mean slopes of curves are also reported and $h = 1/\sqrt{N_{\text{dof}}}$

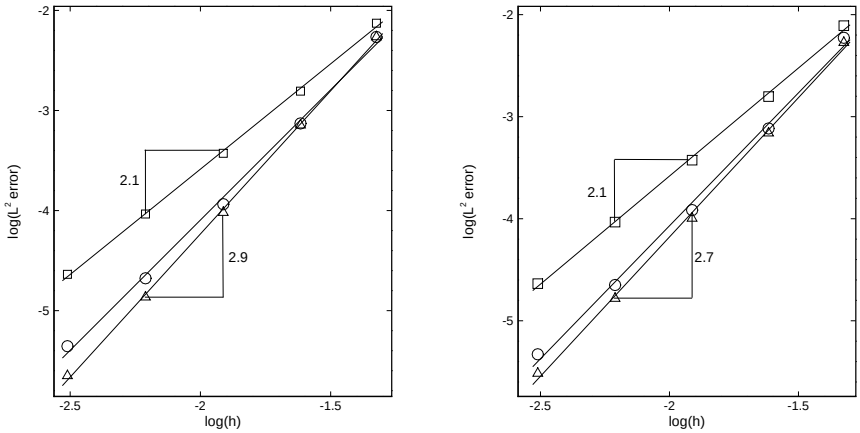


Figure 4.8: L^2 errors of the SPR-ZZ method for quadratic triangular elements with different sampling strategies: ($\triangle$) strategy Figure 4.4-(a), ($\circ$) strategy Figure 4.4-(b) and ($\square$) strategy Figure 4.4-(c). On the left errors for the x component of the gradient, on the right errors for the y component of the gradient and $h = 1/\sqrt{N_{\text{dof}}}$.

.

discretization of advection-diffusion problems,

ii) high-order accuracy is preserved in the whole range of the Peclet number.

The last requirement could not by satisfied by the method proposed in [5] for which the theoretical accuracy was spoiled in the region $Pe \approx 1$

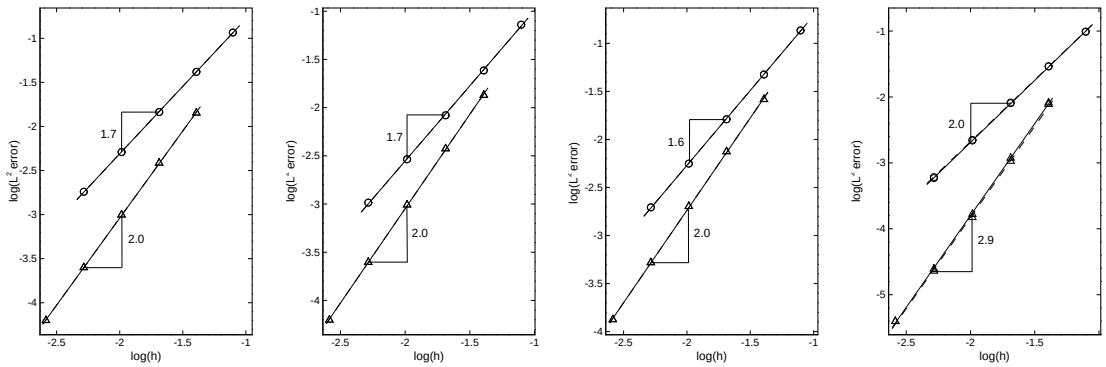


Figure 4.9: L^2 errors of different gradient reconstruction methods with linear ($\circ$) and quadratic ($\triangle$) elements on unstructured grids of quadrangles. Solid and dashed lines represent the error of the x and y components of the gradient, respectively. From left to right: Weighted area, L^2-Projection, Least Square and SPR-ZZ methods. The mean slopes of curves are also reported and $h = 1/\sqrt{N_{\text{dof}}}$

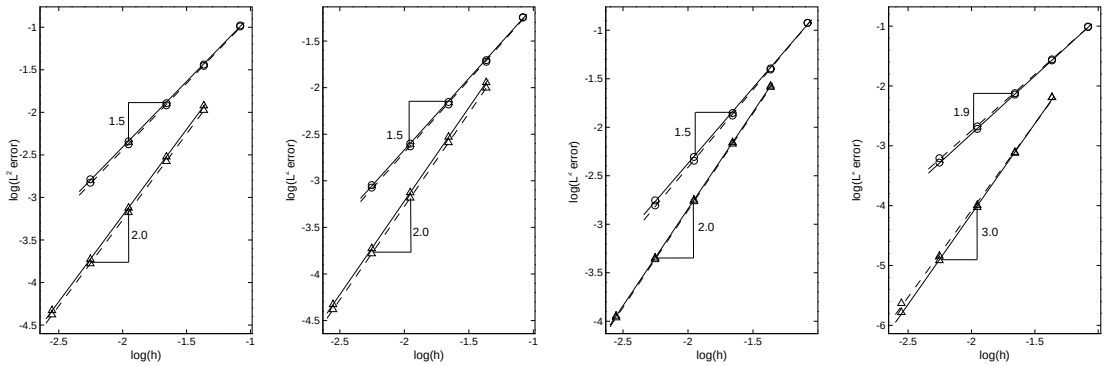


Figure 4.10: L^2 errors of different gradient reconstruction methods with linear ($\circ$) and quadratic ($\triangle$) elements on hybrids grids. Solid and dashed lines represent the error of the x and y components of the gradient, respectively. From left to right: Weighted area, L^2-Projection, Least square and SPR-ZZ methods. The mean slopes of curves are also reported also and $h = 1/\sqrt{N_{\text{dof}}}$

Here, the steady state is considered to be reached when the L^2 norm of the initial residual is reduced by ten orders of magnitude. If the residual of the scheme stagnates at a higher level, it is marked that the simulation did not converged. The CFL number is taken as 0.9 and 0.6, respectively for the second and third order schemes. The same type of grids shown in Figure 4.6 are considered in the numerical simulations. In all the simulations the following definition of the L^2 norm of the

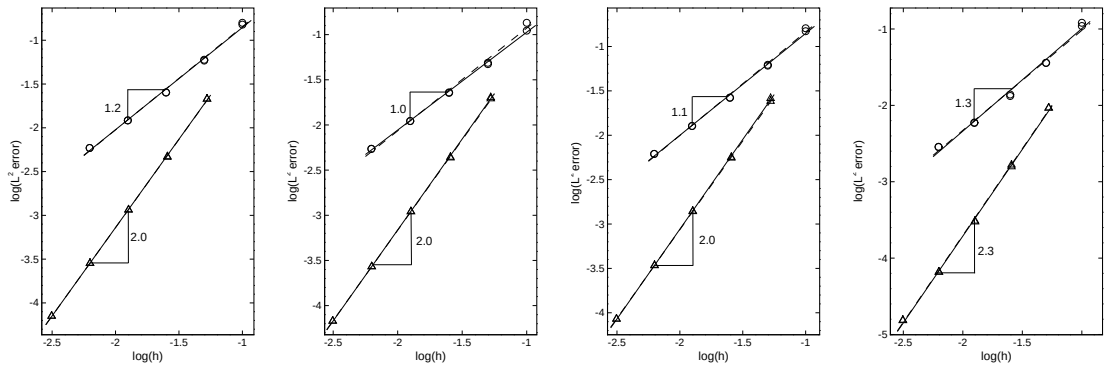


Figure 4.11: L^2 errors of different gradient reconstruction methods with linear ($\circ$) and quadratic ($\triangle$) elements on an unstructured grids of randomly distorted triangles. Solid and dashed lines represent the error of the x and y components of the gradient, respectively. From left to right: Weighted area, L^2-Projection, Least Square and SPR-ZZ methods. The mean slopes of curves are also reported and $h = 1/\sqrt{N_{\text{dof}}}$

error is used

$$\epsilon_{L^2} = \sqrt{\frac{\int_\Omega \left(u^h - u_{\text{ex}}\right)^2 \mathrm{d}\Omega}{\int_\Omega {u_{\text{ex}}}^2 \mathrm{d}\Omega}}. \tag{4.69}$$

4.5.1 Linear Advection-Diffusion Equation

To check the order of accuracy of the linear and non-linear schemes, as well as to study the influence of different gradient reconstruction methods on the accuracy of the numerical solution, the linear advection-diffusion problem with constant viscosity is considered here; namely $\boldsymbol{a}\cdot\boldsymbol{\nabla} u = \nu \boldsymbol{\Delta} u$ on $\Omega = [0,1]^2$, with boundary conditions such that the exact solution of the problem is given by

$$u = -\cos(2\pi\eta)\exp\left(\frac{\xi\left(1-\sqrt{1+16\pi^2\nu^2}\right)}{2\nu}\right), \tag{4.70}$$

with $\eta = a_y x - a_x y$ and $\xi = a_x x + a_y y$. Here $\boldsymbol{a} = (0,1)^{\mathrm{T}}$ and $\nu = 0.01$, which is the most critical case because the advection and the diffusion have similar orders of magnitude and traditional high-order RD schemes generally loose an order of accuracy in this regime. On the left, right and bottom boundaries of the domain the exact solution is imposed as Dirichlet boundary condition while on the top boundary nothing is done. The solution is initialized with a zero value everywhere in the domain except on the inflow boundaries where the exact solution is imposed.

In Figure 4.12 and Figure 4.13 are reported the L^2 norms of the errors of the numerical solution obtained on a sequence of triangular grids with the linear scheme (4.41) and the non-linear one (4.42), for different gradient reconstruction strategies; linear and quadratic elements are considered. For sake of completeness, the errors of the solution are shown together with the errors of the gradients of the numerical solution. In the case of linear elements, the accuracy of the schemes with different gradient reconstruction methods is almost identical, for both linear and non-linear schemes. This is in accordance with the accuracy results observed for the different gradient reconstruction techniques and it underlines also the fact that the high cost of the L^2-projection method is not justified, since less expensive methods produce results with the same level of accuracy.

The situation is very different in the case of quadratic elements; the weighted area and the L^2-Projection gradient reconstruction methods produce sub-optimal schemes with second order only accurate solutions for both linear and non-linear schemes. The use of the SPR-ZZ method allows the construction of an optimal third order accurate scheme and it is worth noticing that also the x-component of the gradient of the numerical solutions is third order accurate, meaning that solution and gradients are computed with the same order of accuracy. On the y-component of the gradient this optimal behavior is lost, and this due to the combined effects of the gradient reconstruction with the solution error on the outflow boundary, where no boundary condition is imposed. An optimal accuracy on both the components of the gradient has been observed in numerical simulations of the linear advection-diffusion problems with Dirichlet boundary conditions imposed on all the boundaries of the domain.

It is worth noticing that the combination of the non-linear scheme with the least square gradient reconstruction technique produces an almost optimal scheme although the least square reconstruction does not allow a high-order gradient reconstruction by itself.

In Figure 4.14 are reported the errors of the solution and of gradient components obtained for the discretization of the linear advection-diffusion problem on a sequence of unstructured grids of quadrangles with the linear and the non-linear schemes. For simplicity, only the weighted area and the SPR-ZZ reconstruction strategies are used. As previously observed, with linear elements there is no significant difference in the level of accuracy between different gradient reconstruction techniques, however with quadratic elements only the SPR-ZZ gradient reconstruction guarantees third order accurate solutions. Of course, the same considerations done for triangular and quadrangular grids are still valid with grids with hybrid elements, as it is evident from Figure 4.15.

In Figure 4.16 are reported the errors obtained on a sequence of highly distorted triangular grids. The behavior of the schemes is similar to that observed with more regular meshes, in particular it is important to note that the poor quality of the grids has only a limited influence on the overall accuracy of the scheme.

In order to highlight the effectiveness of higher order schemes over the second

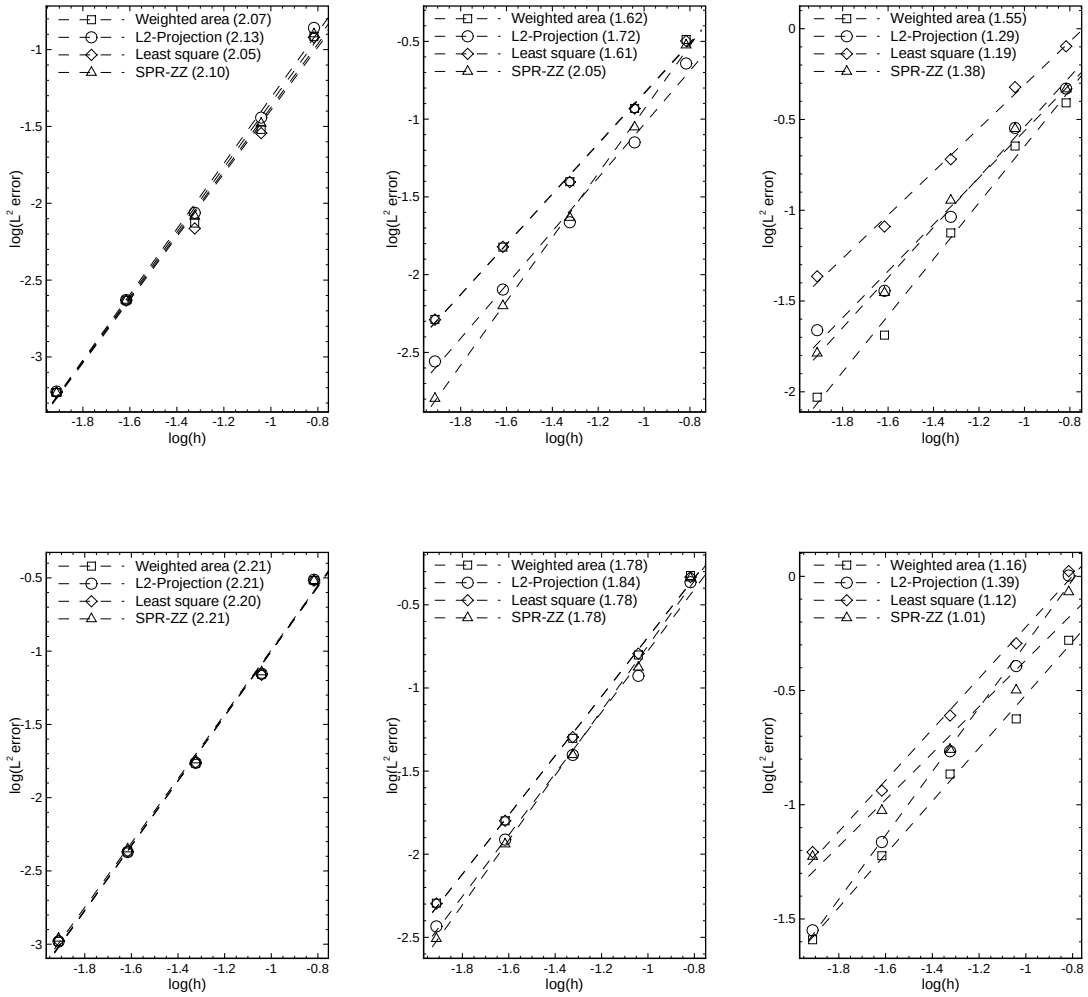


Figure 4.12: L^2 errors for the solution of the linear advection-diffusion problem on triangular girds with linear elements. Errors of the solution (first column), errors of the x-component of the gradient (second column) errors of the y-component of the gradient (third column). Linear scheme (upper), non-linear scheme (lower). In the legends are reported also the mean slopes of curves and $h = 1/\sqrt{N_{\text{dof}}}$.

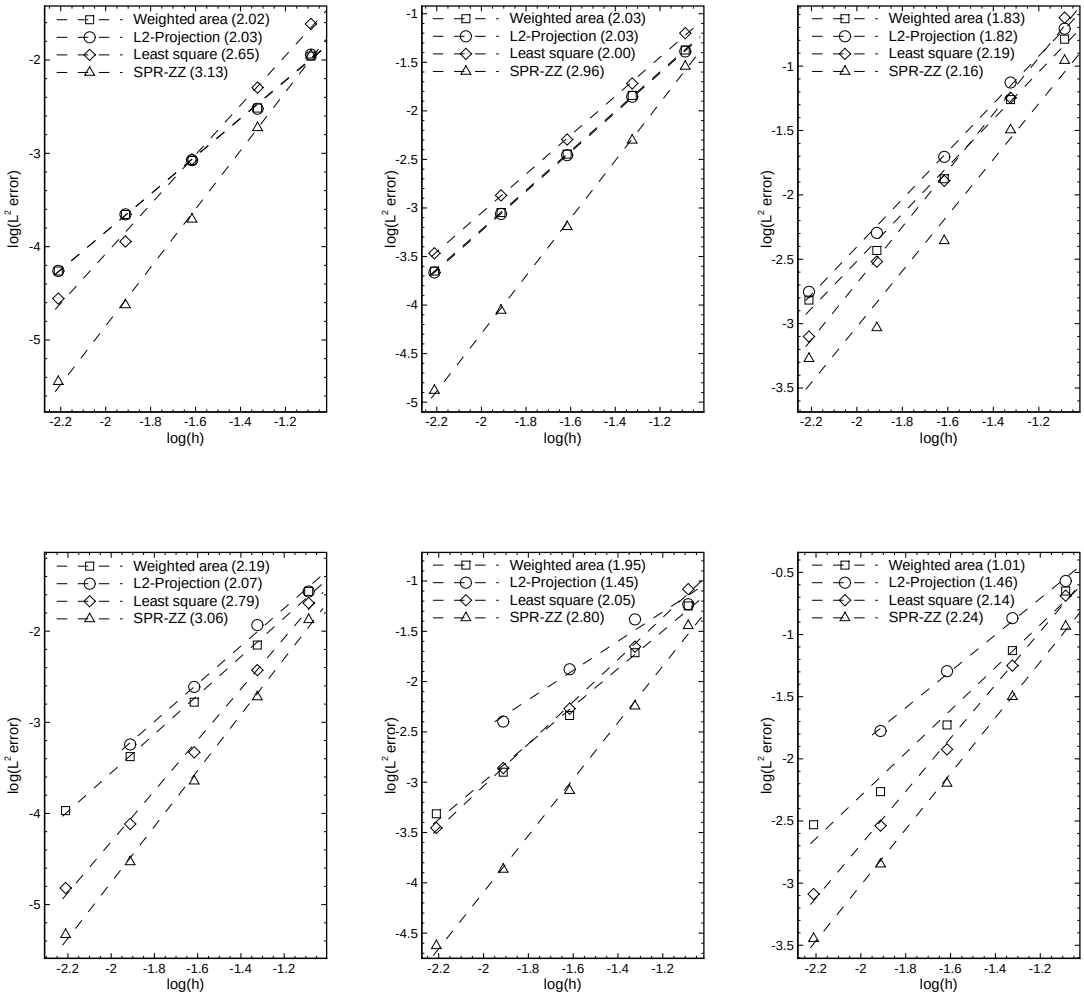


Figure 4.13: L^2 errors for the solution of the linear advection-diffusion problem on triangular girds with quadratic elements. Errors of the solution (first column), errors of the x-component of the gradient (second column) errors of the y-component of the gradient (third column). Linear scheme (upper), non-linear scheme (lower). In the legends are reported also the mean slopes of curves and $h = 1/\sqrt{N_{\text{dof}}}$.

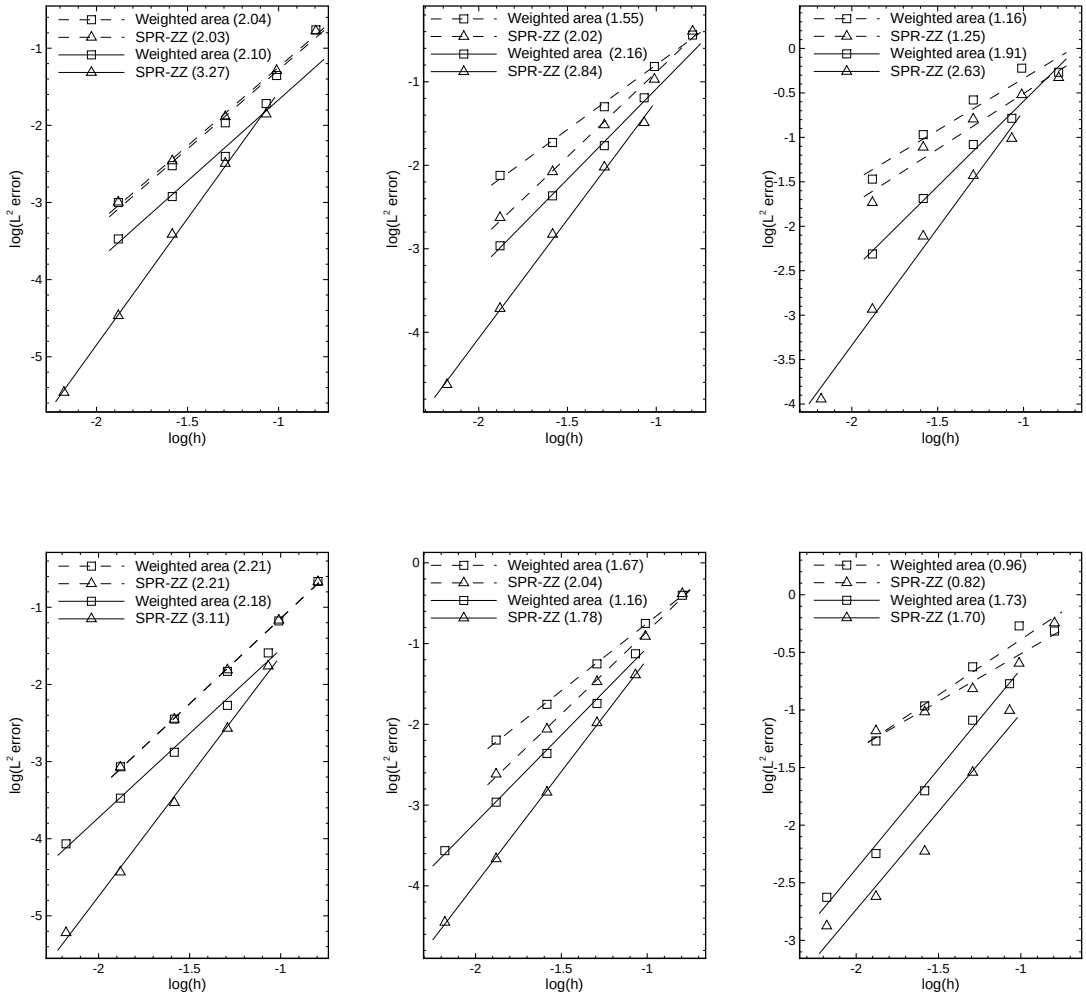


Figure 4.14: L^2 errors for the solution of the linear advection-diffusion problem on quadrangular girds with linear (dashed lines) and quadratic (solid lines) elements. Errors of the solution (first column), errors of the x-component of the gradient (second column) errors of the y-component of the gradient (third column). Linear scheme (upper), non-linear scheme (lower). In the legends are reported also the mean slopes of curves and $h = 1/\sqrt{N_{\text{dof}}}$.

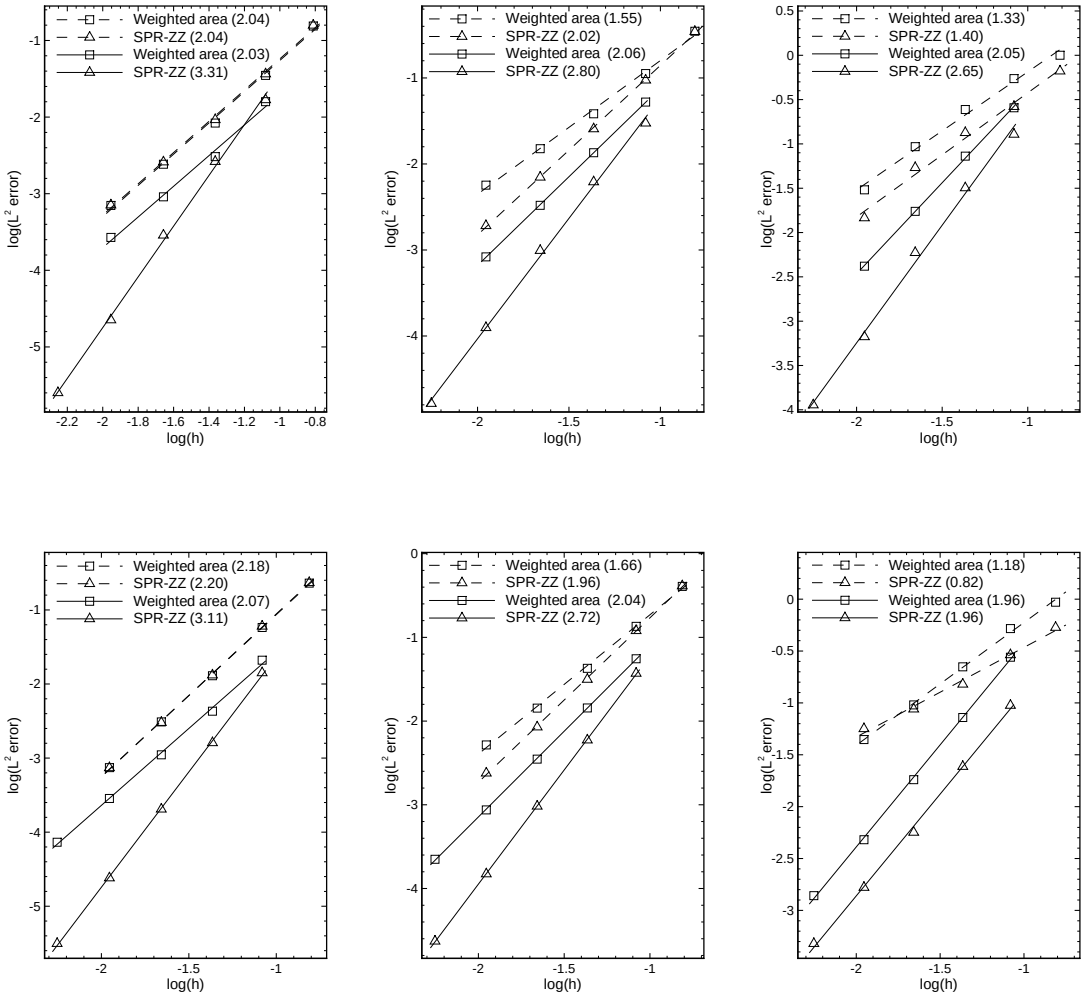


Figure 4.15: L^2 errors for the solution of the linear advection-diffusion problem on hybrid girds with linear (dashed lines) and quadratic (solid lines) elements. Errors of the solution (first column), errors of the x-component of the gradient (second column) errors of the y-component of the gradient (third column). Linear scheme (upper), non-linear scheme (lower). In the legends are reported also the mean slopes of curves and $h = 1/\sqrt{N_{\text{dof}}}$.

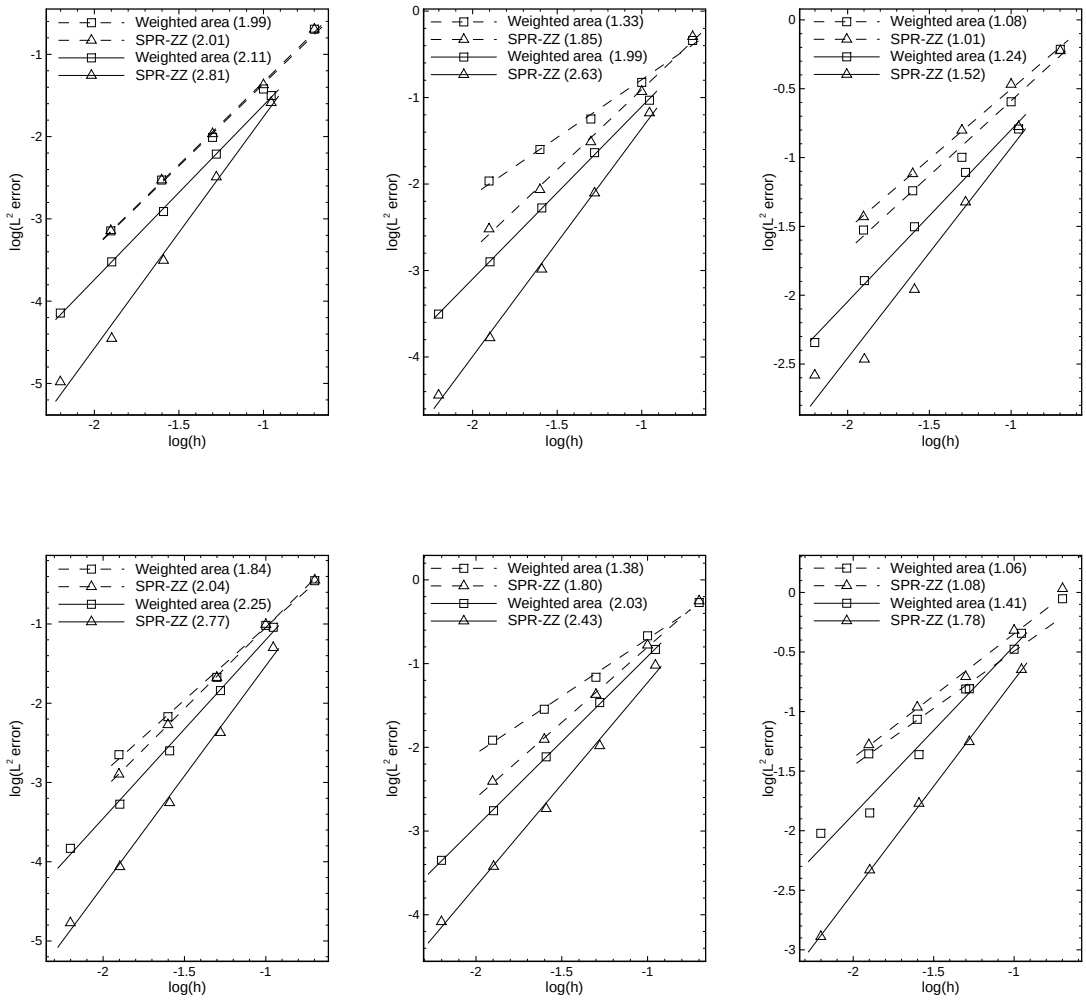


Figure 4.16: L^2 errors for the solution of the linear advection-diffusion problem on highly distorted triangular girds with linear (dashed lines) and quadratic (solid lines) elements. Errors of the solution (first column), errors of the x-component of the gradient (second column) errors of the y-component of the gradient (third column). Linear scheme (upper), non-linear scheme (lower). In the legends are reported also the mean slopes of curves and $h = 1/\sqrt{N_{\text{dof}}}$.

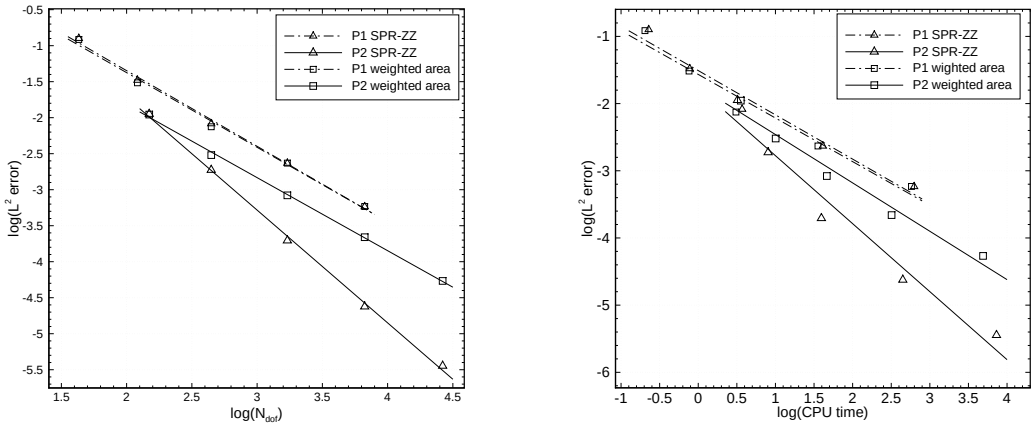


Figure 4.17: L^2 errors for the solution of the linear advection-diffusion problem, on triangular grids, versus the number of degrees of freedom and the CPU time in seconds.

order ones, in Figure 4.17 the discretization errors of the solution are reported as function of the number of degrees of freedom and the CPU time; for brevity only results for the linear scheme on triangular grids are shown. One can see that to get a fixed level of error, 10^{-5} for example, an actual third order scheme requires about 12 000 degrees of freedom and 25 minutes to perform the computation. A second order scheme, on the other hand, requires about 31 000 degrees of freedom and 5 hours to get the same level of error.

The effect of the filtering term for the viscous part only is now investigate, i.e. the linear scheme (4.41) is compared against the scheme (4.31) and the non-linear scheme (4.42) is compared against the scheme (4.33). The comparison is done in term of solution accuracy and number of iterations necessary to reach the steady state; results are reported in Table 4.1 for the linear and non-linear schemes, with the SPR-ZZ reconstruction strategy. It can be observed that in the case of linear elements there is no appreciable difference in term of error between the schemes with and without the stabilization term for the viscous part, however the use of the extra stabilization term makes the linear scheme converge much faster to the steady state. The effect becomes even more important for the non-linear scheme, for which it is observed that the absence of the extra dumping term prevents the scheme to converge in several cases. For the quadratic elements, the presence of the extra dumping term has two effects, it improves the convergence of the numerical methods and it introduces a crucial improvement in the level of accuracy, for both linear and non-linear schemes.

For sake of completeness, the linear advection-diffusion problem is also solved with a very small viscosity coefficient, $\nu = 10^{-6}$, in order to verify that the numerical schemes are able to preserve the theoretical accuracy in the advection limit. The

Linear elements					Quadratic elements				
N_{dof}	$\epsilon_{L^2}(u^h)$ improved	Ite	$\epsilon_{L^2}(u^h)$ standard	Ite	N_{dof}	$\epsilon_{L^2}(u^h)$ improved	Ite	$\epsilon_{L^2}(u^h)$ standard	Ite
Linear scheme									
43	$1.2639\ 10^{-1}$	454	$1.1945\ 10^{-1}$	465	149	$1.1247\ 10^{-2}$	3148	$1.4256\ 10^{-2}$	4443
121	$3.3130\ 10^{-2}$	538	$3.1174\ 10^{-2}$	726	445	$1.8777\ 10^{-3}$	2507	$3.1427\ 10^{-3}$	7065
445	$8.2461\ 10^{-3}$	532	$8.8271\ 10^{-3}$	1456	1705	$1.9648\ 10^{-4}$	2566	$9.3957\ 10^{-3}$	7125
1705	$2.3337\ 10^{-3}$	1319	$2.5600\ 10^{-3}$	2877	6673	$2.3797\ 10^{-5}$	7029	not converged	–
6673	$5.8201\ 10^{-4}$	4428	$8.1456\ 10^{-4}$	6310	26401	$3.5754\ 10^{-6}$	23431	not converged	–
Non-linear scheme									
43	$3.0257\ 10^{-1}$	474	$2.9418\ 10^{-1}$	529	149	$1.3349\ 10^{-2}$	4747	$1.4830\ 10^{-2}$	5566
121	$7.2141\ 10^{-2}$	686	$6.7668\ 10^{-2}$	992	445	$1.8975\ 10^{-3}$	4544	$6.5685\ 10^{-3}$	11414
445	$1.8068\ 10^{-2}$	835	$1.7301\ 10^{-2}$	2003	1705	$2.2616\ 10^{-4}$	3991	$8.8542\ 10^{-4}$	9935
1705	$4.4622\ 10^{-3}$	1791	not converged	–	6673	$2.9410\ 10^{-5}$	5075	$1.0561\ 10^{-3}$	31636
6673	$1.0974\ 10^{-3}$	4897	not converged	–	26401	$4.6791\ 10^{-6}$	42370	not converged	–

Table 4.1: L^2 errors and orders of accuracy for the solution of the linear advection-diffusion problem on triangular girds with the linear and non-linear schemes, for different gradient reconstruction strategies, and with linear and quadratic elements. The standard and improved identification stands for the basic and the modified scheme with the additional filtering term for the viscous part, respectively.

errors, obtained on a sequence of triangular grids, are reported in Figure 4.18 for the linear and the non-linear schemes. As expected, the theoretical order of accuracy is reached independently on the gradient reconstruction method used, because the diffusive effects are negligible in this regime. Nevertheless, the level of accuracy of the gradients obtained with the SPR-ZZ technique is hgher.

Discretization with the Hyperbolic FOS

The linear advection-diffusion problem is now discretized by the means of the hyperbolic Fist Order System scheme described in Section 4.3.3. The objective is to compare the accuracy and the performance of this formulation with the scalar discretization method. The linear scheme is used to discretize the hyperbolic FOS and the scalar equation. In the latter case the SPR-ZZ gradient reconstruction strategy is used. A sequence of triangular grids is considered, with linear and quadratic elements, and the viscosity coefficient ν is takes as 0.01. In Figure 4.19 are shown the errors on the solution and the x-component of the gradient, together with the CPU time (in seconds), needed to reach the steady state.

With linear elements, there is only a small difference in the errors discretization between the scalar and the FOS formulation, however in terms of CPU time, it is evident that the scalar scheme is much more effective than the FOS formulation.

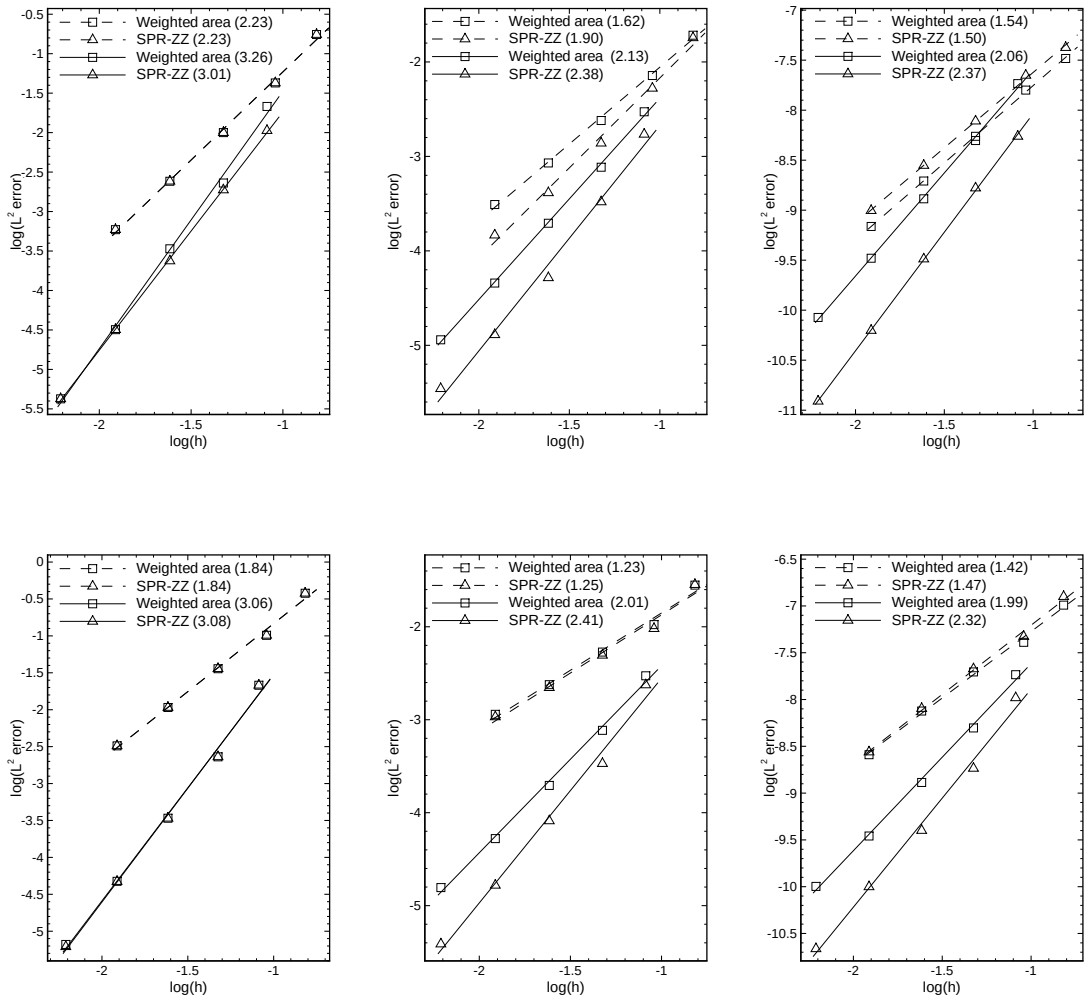


Figure 4.18: L^2 errors for the solution of the linear advection-diffusion problem, with $\nu = 10^{-6}$, on triangular girds with linear (dashed lines) and quadratic (solid lines) elements. Errors of the solution (first column), errors of the x-component of the gradient (second column) errors of the y-component of the gradient (third column). Linear scheme (upper), non-linear scheme (lower). In the legends are reported also the mean slopes of curves and $h = 1/\sqrt{N_{\text{dof}}}$. Note that the error of the y-component of the gradient is not normalized in this case, due to the very small value of the exact expression.

The slope of the curve CPU time–error for the scalar scheme is in accordance with the analysis reported in [83] for which, in two spatial dimensions, $\epsilon = \mathcal{O}(\text{CPU}^{-p/4})$, where ϵ is some norm of the solution error, CPU is the total CPU time and p is the order of convergence of the numerical scheme. For the FOS scheme, using an explicit method, the theoretical CPU time–error curve is $\epsilon = \mathcal{O}(\text{CPU}^{-p/3})$, in 2D. The correct slope $-2/3$ can be observed in Figure 4.19 for the FOS scheme

with linear elements, while, since an implicit method has been used with quadratic elements, the previous estimations cannot be applied. In particular, a slope -0.85 has been obtained for the implicit third-order simulation with the FOS scheme.

With quadratic elements, the accuracy of the solution obtained using the FOS scheme is slightly better than that obtained with the scalar scheme, but the situation is completely different if one looks at the accuracy of the gradient. As already pointed out in [83], the discretization of the hyperbolic FOS with RD schemes does not allow to recover the gradients with the same accuracy of the solution, unless the mesh is regular. It can be noticed how the use of the SPR-ZZ strategy allows to obtain a third order accurate gradient while the FOS scheme gives only second order accuracy. The last remark concerns the CPU time. An implicit Euler method has been used for the FOS scheme and an explicit Euler method has been used for the scalar scheme; the scalar scheme is still much more effective than the FOS scheme.

Note that in the advection limit the smallest eigenvalue of the hyperbolic FOS vanishes, this means that two of the three eigenvalues are zero and the problem becomes ill conditioned. It has been observed here that the FOS scheme is not able to converge for the linear advection-diffusion problem with the viscous coefficient ν taken as 10^{-6}.

4.5.2 Viscous Burger Equation

The viscous Burger equation is now considered in order to test the accuracy of the numerical schemes with a non-linear problem. The governing equation reads

$$\frac{\partial}{\partial x}\left(\frac{u^2}{2}\right) + \frac{\partial u}{\partial y} = \nu \frac{\partial^2 u}{\partial x^2}, \qquad \text{on } \Omega = [0,1]^2, \tag{4.71}$$

the problem admits the following exact solution

$$u = \frac{2\nu\pi \exp(-\nu y \pi^2)\sin(\pi x)}{a + \exp(-\nu y \pi^2)\cos(\pi x)}, \quad \text{with } a > 1. \tag{4.72}$$

Note that the exact solution of the steady two-dimensional problem is obtained from the unsteady one-dimensional problem, in which the time coordinate is substituted by the y coordinate. In the simulations, the parameter a is taken as 1.5 and the viscosity coefficient ν is taken as 0.05. On the bottom, left and right boundaries the exact solution is imposed as Dirichlet boundary condition. The solution is initialized with a zero value everywhere, except on the inflows boundaries where the exact solution is imposed.

A sequence of unstructured triangular grids is considered, the weighted area and the SPR-ZZ gradient reconstruction methods are used. The errors for the solution are reported in Figure 4.20 together with the errors for the gradients components. With linear elements, the level of accuracy of the schemes is almost identical, while with quadratic elements the situation is very different. The use of the weighted area

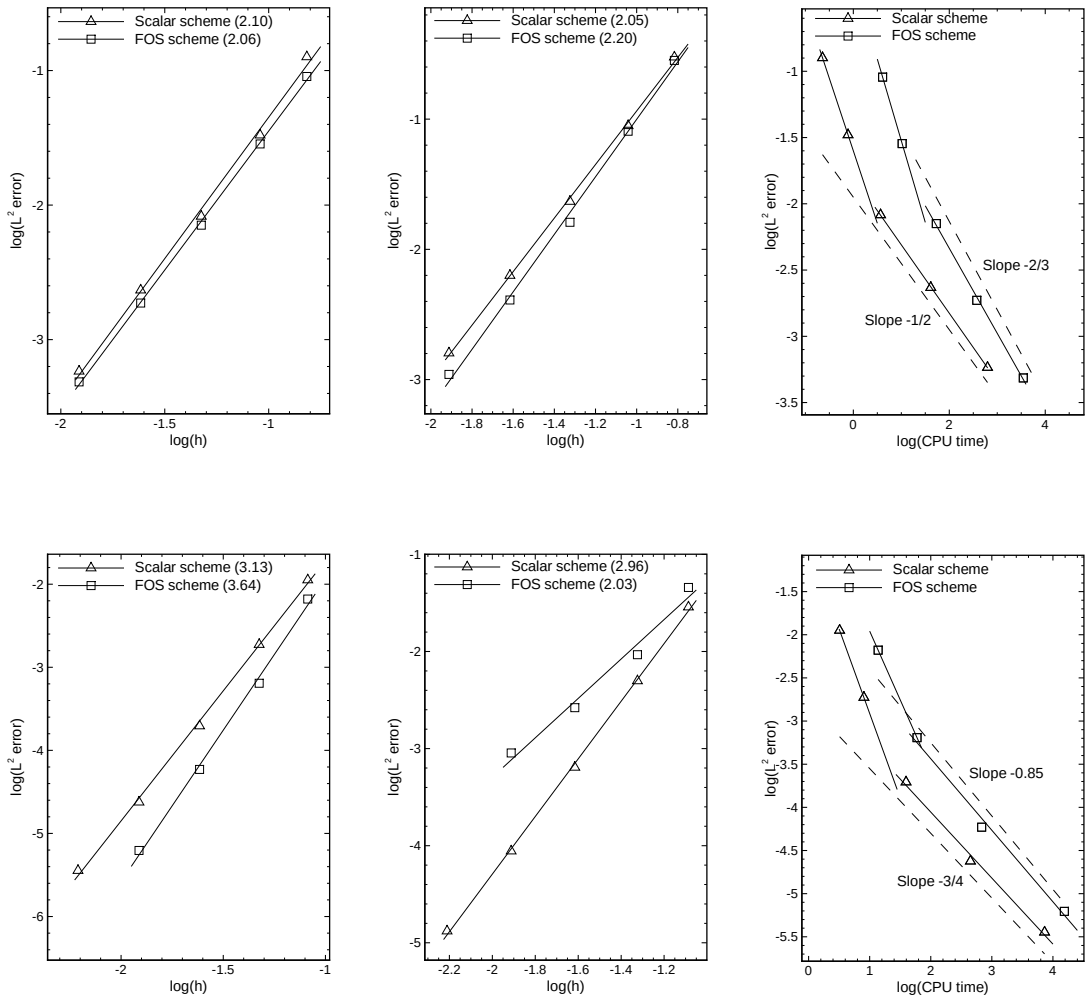


Figure 4.19: L^2 errors for the solution of the linear advection problem with $\nu = 0.01$ on triangular girds, with the scalar and FOS solver. Errors of the solution (first column), and of the x-component of the gradient (second column) as function of the number of DofS. Error of the solution as function of the CPU time in seconds (third column). Linear elements (upper), quadratic elements (lower). In the legends are reported also the mean slopes of the curves and $h = 1/\sqrt{N_{\text{dof}}}$.

gradient reconstruction has a disastrous effect on the accuracy of the solution, indeed the theoretical third order scheme has the same level of accuracy of the second order scheme. On the other had, the use of the SPR-ZZ reconstruction method allows to construct an optimal third order scheme and also the accuracy of the gradient is noticeably improved.

4.5.3 Anisotropic Diffusion Problem

A two-dimensional diffusion problem is now addressed, the viscosity is not considered to be a scalar anymore but an anisotropic tensor. The aim of this test case is to study the accuracy of the proposed RD schemes with a pure diffusion problem, and the anisotropy of the viscous tensor is introduced to test also the robustness of the numerical scheme.

The diffusion problem is formulated as follows

$$-\boldsymbol{\nabla}\cdot\left(\mathbb{K}\boldsymbol{\nabla}u\right)=0,\quad \text{on}\quad \Omega=[0,1]^2, \tag{4.73}$$

with

$$\mathbb{K}=\begin{pmatrix}1 & 0\\ 0 & \delta\end{pmatrix}, \tag{4.74}$$

the problem has the following exact solution $u=\sin(2\pi x)\,e^{-2\pi y\sqrt{1/\delta}}$; in the numerical simulations δ is takes as 10^3.

A sequence on unstructured triangular grids is considered, the scalar schemes with the weighted area and SPR-ZZ gradient reconstruction strategies are considered. In Figure 4.21 are reported the errors of the solution and of the gradients for linear and quadratic elements. As usual, the second order schemes have the same level of accuracy independently on the gradient reconstruction method used, but with quadratic elements only the use of the SPR-ZZ method allows to get a third order accuracy on the solution as well as on the gradient. It is interesting to note that the accuracy of the non-linear scheme is severely spoiled by the use of the simple weighted area method with quadratic elements.

In the end, the anisotropic viscous problem is solved on a uniform, structured mesh of quadrangles, results are shown in Figure 4.22 and indicated that the theoretical accuracy of the schemes is achieved independently on the gradient reconstruction technique used, although the accuracy of the gradient is always better with the SPR-ZZ method. This remark is important because for advection-diffusion problems, even with the use of uniform structured grids, the formal accuracy of the scheme is not preserved, unless the gradients are recovered with high order accuracy.

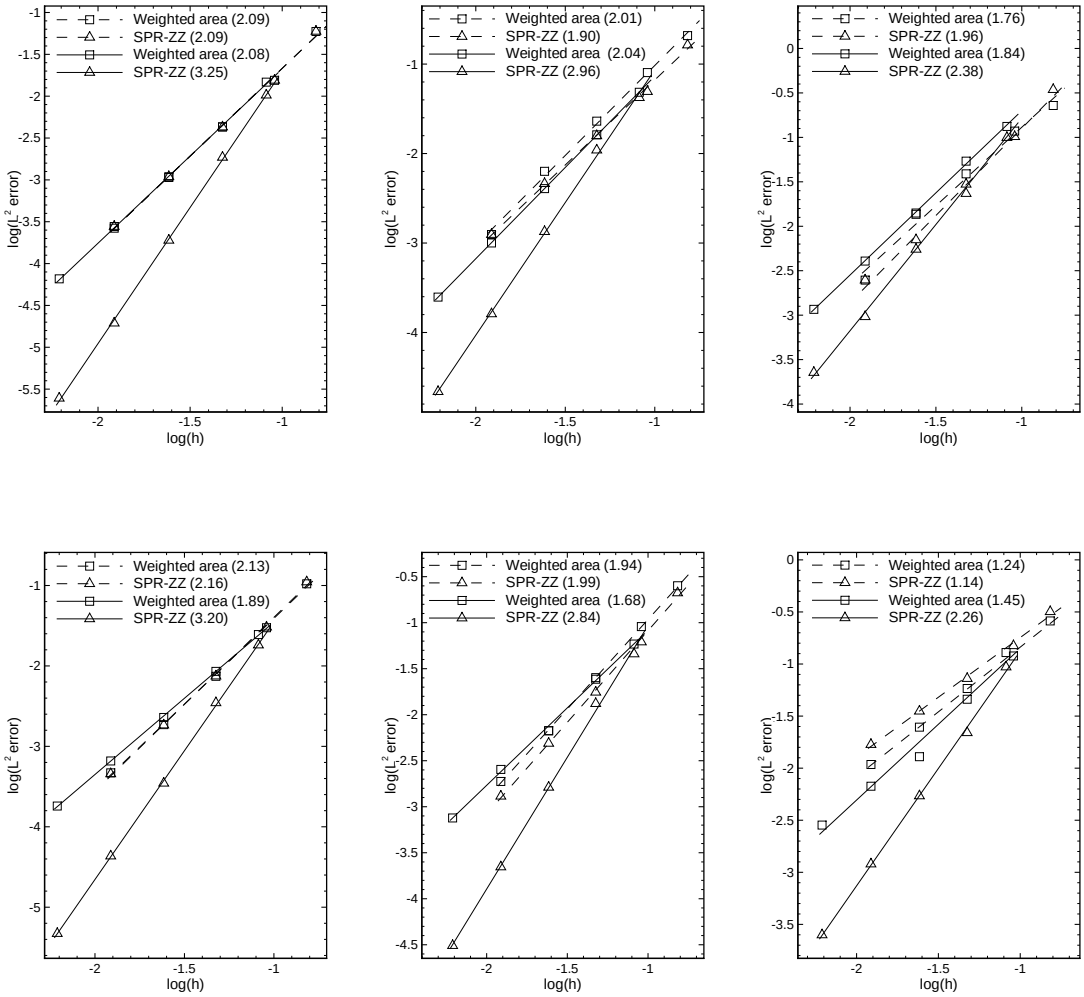


Figure 4.20: L^2 errors for the solution of the viscous Burger problem on triangular girds with linear (dashed lines) and quadratic (solid lines) elements. Errors of the solution (first column), errors of the x-component of the gradient (second column) errors of the y-component of the gradient (third column). Linear scheme (upper), non-linear scheme (lower). In the legends are reported also the mean slopes of the curves and $h = 1/\sqrt{N_{\text{dof}}}$.

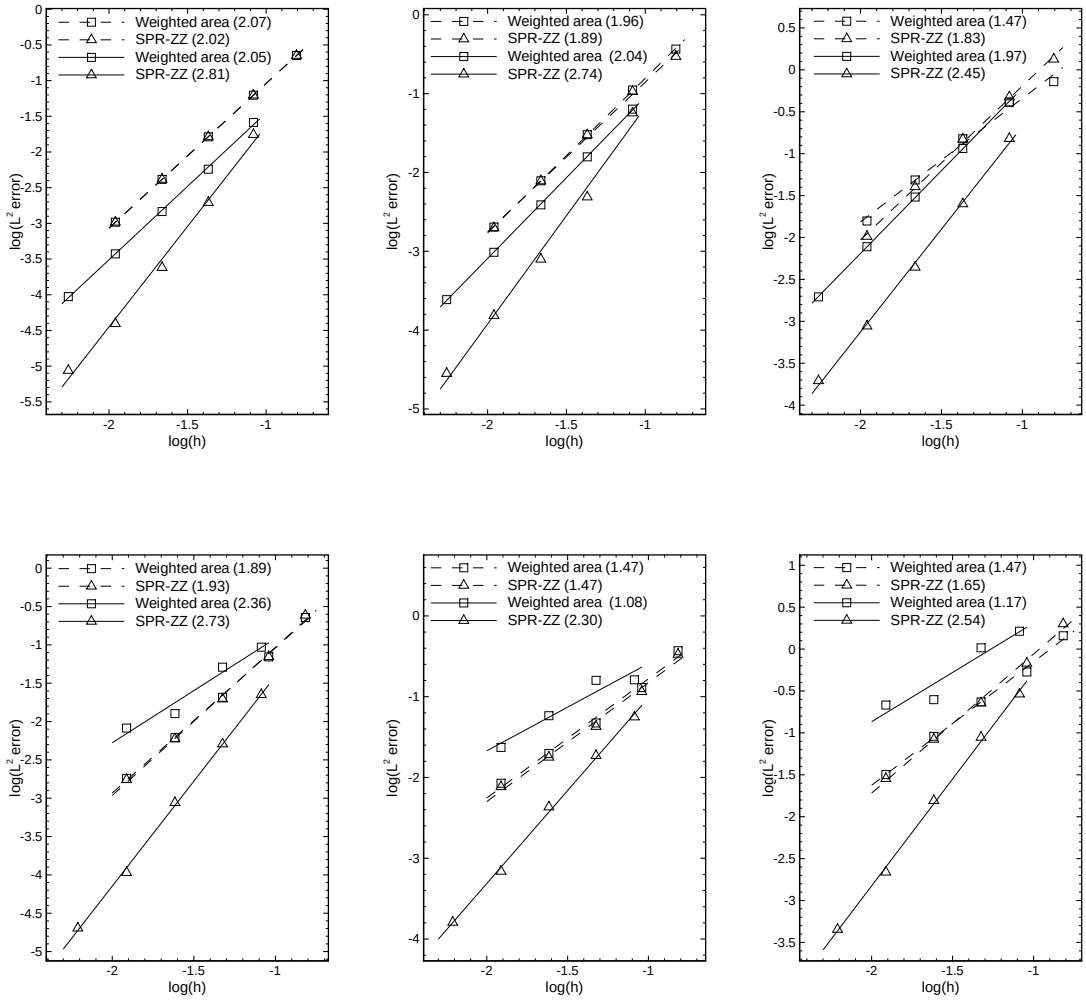


Figure 4.21: L^2 error in the solution of the anisotropic diffusion problem on triangular girds with linear (dashed lines) and quadratic (solid lines) elements. Error of the solution (first column), error of the x-component of the gradient (second column) error of the y-component of the gradient (third column). Linear scheme (upper), non-linear scheme (lower). In the legends are reported also the mean slopes of the curves and $h = 1/\sqrt{N_{\text{dof}}}$.

4.5.4 Linear Advection-Diffusion Equation with Discontinuous Solution

To check the capacity of the numerical scheme to get oscillations-free discontinuous solutions, the following advection-diffusion problem is considered

$$\boldsymbol{a}\cdot\boldsymbol{\nabla}u = \nu\boldsymbol{\Delta}u, \quad \text{on} \quad \Omega = [0,1]^2 \tag{4.75}$$

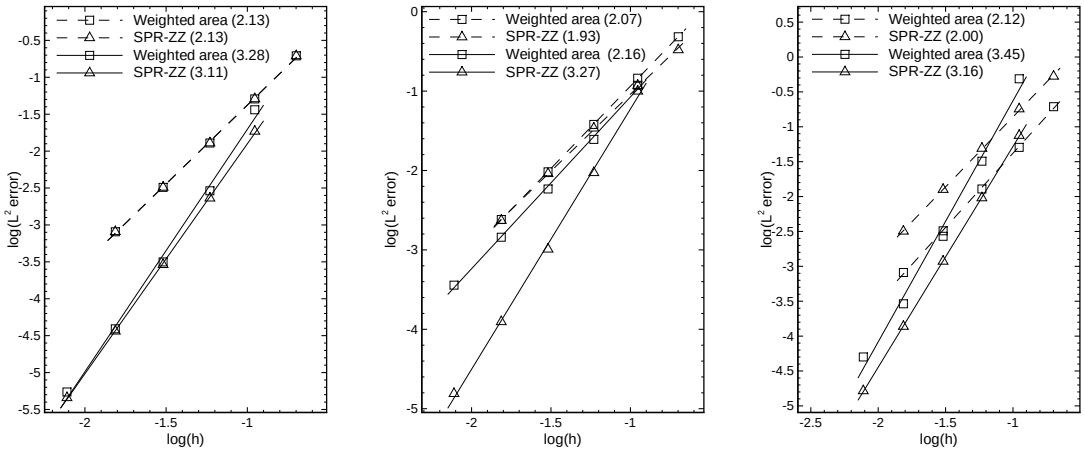


Figure 4.22: L^2 error in the solution of the anisotropic diffusion problem, with the linear scheme, on uniform structured grids of quadrangles, with linear (dashed lines) and quadratic (solid lines) elements. Error of the solution (first column), error of the x-component of the gradient (second column) error of the y-component of the gradient (third column). In the legends are reported also the mean slopes of the curves and $h = 1/\sqrt{N_{\text{dof}}}$.

with $\boldsymbol{a} = (1/2, \sqrt{3}/2)^{\text{T}}$, $\nu = 10^{-3}$ and with the following boundary conditions on the inflow boundaries

$$u = \begin{cases} 1, & \text{on } \{y = 0,\, 0 \leq x \leq 1\}, \\ 1, & \text{on } \{x = 0,\, y \leq 1/5\}, \\ 0, & \text{elsewhere.} \end{cases} \tag{4.76}$$

In Figure 4.23 are reported the approximated solutions obtained on an unstructured grid of 3 264 triangles, with the SPR-ZZ method for the gradient reconstruction. Note that the linear scheme produces spurious oscillations near the discontinuity, which are reduced using quadratic elements, but never disappear. The non-linear scheme, on the other hand, gives smooth solutions for both linear and quadratic elements. This can be better observed in Figure 4.24, where the profiles of the solutions along $x = 0.05$ are reported.

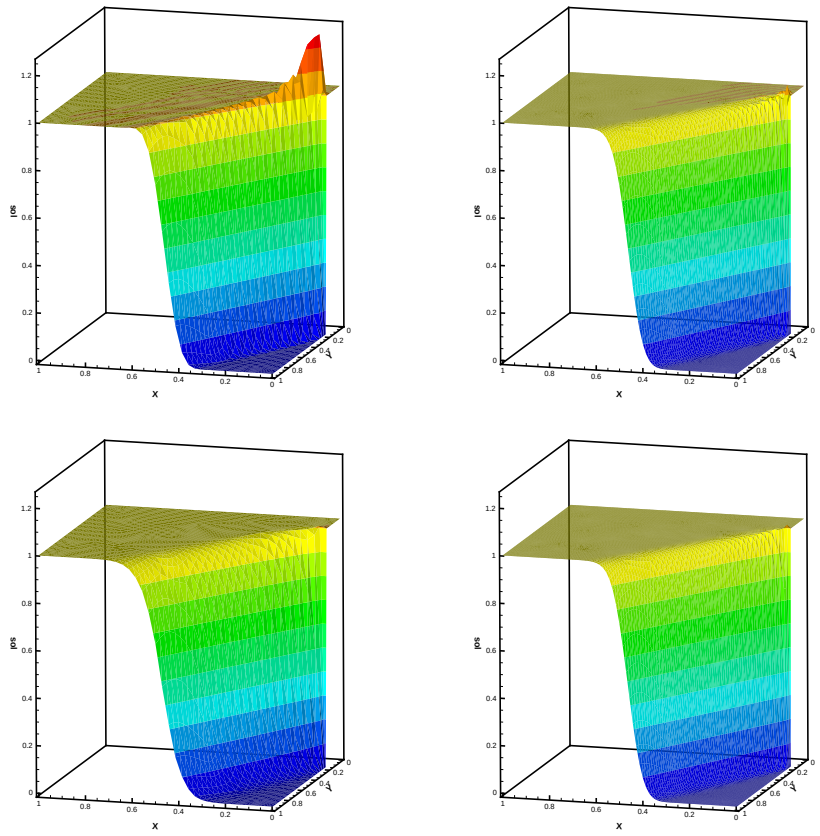


Figure 4.23: Solutions of the advection-diffusion problem with discontinuous boundary conditions. Top row: solutions with the liner scheme, bottom row: solution with the non-linear scheme. Left column: linear elements, right column: quadratic elements.

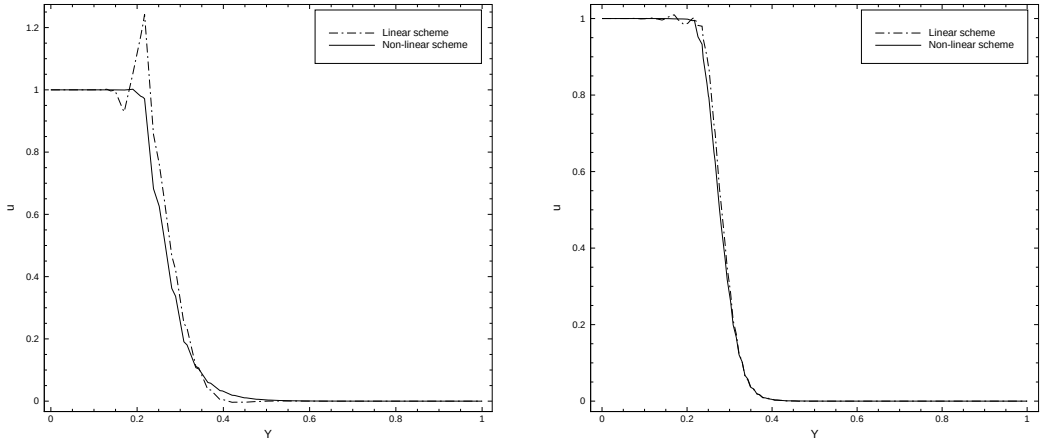


Figure 4.24: Profiles at $x = 0.05$ of the solutions of the advection-diffusion problem with discontinuous boundary conditions with the linear and the non-linear scheme. Left: linear elements, right: quadratic elements.

Part II

Systems of Conservation Laws

Chapter 5

RD Discretization of Systems of Equations

In this chapter the RD method developed for scalar advection-diffusion problems is now extended to the case of systems of conservation laws, with particular focus on the Navier-Stokes equations, although some problems with the Euler equations are considered as well, for sake of completeness.

After the introduction of the governing equations, the numerical method for their discretization is presented. Given the numerical method described in Chapter 4 for scalar equations, the extension to systems of equations is almost straightforward and does not introduce any new aspect for the spatial discretization. However, in addition to the scalar formulation, here is described with more details the way used to impose the boundary conditions and also how to treat high order representation of the boundaries. Furthermore, an implicit solver based on the matrix-free approach is adopted in order to accelerate the convergence of the solution to the steady state.

5.1 Governing Equations

The mathematical model considered in this work describe the dynamics of the compressible flows with the proper initial and boundary conditions. In the particular case in which the viscous and thermal effects are neglected, the governing equations are denoted as Euler equations, while in the more general case the governing equations are denoted as Navier-Stokes equations.

5.1.1 Euler Equations

In the general case of three spatial dimensions, the dynamic of a compressible flow, for which the viscous and thermal effect are neglected, is governed by the following

set of non-linear equations written in conservative form

$$\begin{cases} \dfrac{\partial \rho}{\partial t} + \boldsymbol{\nabla} \cdot \boldsymbol{m} = 0 \\ \dfrac{\partial \boldsymbol{m}}{\partial t} + \boldsymbol{\nabla} \cdot \left(\dfrac{\boldsymbol{m} \otimes \boldsymbol{m}}{\rho} + P\mathbb{I} \right) = \mathbf{0} \\ \dfrac{\partial E^t}{\partial t} + \boldsymbol{\nabla} \cdot \left(\left(E^t + P \right) \dfrac{\boldsymbol{m}}{\rho} \right) = 0 \end{cases} \tag{5.1}$$

where ρ is the density, P is the pressure, $\mathbb{I} \in \mathbb{R}^3$ is the identity matrix, the momentum vector $\boldsymbol{m} = \rho \boldsymbol{v}$, with $\boldsymbol{v}$ the velocity vector, and E^t is the total energy per unit volume defined as

$$E^t = \rho e + \frac{\|\boldsymbol{m}\|^2}{2} = \rho e^t, \tag{5.2}$$

with e the specific (*i.e,* per unit mass) internal energy and e^t the specific total energy.

The system (5.1) must be completed with thermodynamic relations for the variables P and e, In the case of the polytropic ideal gas, the following equations of state can be written

$$e(T) = \frac{\mathcal{R}T}{\gamma - 1} \qquad \text{and} \qquad P(T, \rho) = \mathcal{R}T\rho \tag{5.3}$$

where T is the temperature, $\gamma = c_p/c_v$ is the specific heat ratio, and $\mathcal{R}$ is the gas constant. For typical aerodynamic applications $\gamma = 1.4$. By combining the relations in (5.3), one obtains the pressure as function of the specific variables e and ρ

$$P(e, \rho) = (\gamma - 1)\rho e. \tag{5.4}$$

Since in the Euler equations the actual unknown is the total energy per unit volume (E^t) and not the specif energy (e), it is convenient to rewrite the previous relation as

$$P = P(E^t, \rho) = (\gamma - 1)\left(E^t - \frac{\|\boldsymbol{m}\|^2}{2} \right), \tag{5.5}$$

where with a slight abuse of notation the same symbol has been used to indicate the pressure as function of the ρ and e and as function of the conservative variables.

It is common practice to introduce the vector u of the conservative variables and the advective flux function $\mathbf{f}^a(\mathsf{u})$

$$\mathsf{u} = \begin{pmatrix} \rho \\ \boldsymbol{m} \\ E^t \end{pmatrix} \qquad \text{and} \qquad \mathbf{f}^a(\mathsf{u}) = \begin{pmatrix} \boldsymbol{m} \\ \dfrac{\boldsymbol{m} \otimes \boldsymbol{m}}{\rho} + P\mathbb{I} \\ \left(E^t + P \right) \dfrac{\boldsymbol{m}}{\rho} \end{pmatrix}, \tag{5.6}$$

such that the system (5.1) can be recasted in the following vector form

$$\frac{\partial \mathsf{u}}{\partial t} + \boldsymbol{\nabla} \cdot \mathbf{f}^a(\mathsf{u}) = \mathbf{0}. \tag{5.7}$$

5.1.2 Navier-Stokes Equations

The complete set of equations includes the description of viscous effects, as well as heat conduction phenomena. The system of equations in conservative form, in the general case of three spatial dimensions reads

$$\begin{cases} \dfrac{\partial \rho}{\partial t} + \boldsymbol{\nabla \cdot} \boldsymbol{m} = 0 \\ \dfrac{\partial \boldsymbol{m}}{\partial t} + \boldsymbol{\nabla \cdot} \left(\dfrac{\boldsymbol{m} \otimes \boldsymbol{m}}{\rho} + P\mathbb{I} \right) = \boldsymbol{\nabla \cdot} \mathbb{S} \\ \dfrac{\partial E^t}{\partial t} + \boldsymbol{\nabla \cdot} \left(\left(E^t + P \right) \dfrac{\boldsymbol{m}}{\rho} \right) = \boldsymbol{\nabla \cdot} \left(\mathbb{S} \cdot \dfrac{\boldsymbol{m}}{\rho} - \boldsymbol{q} \right) \end{cases} \tag{5.8}$$

where $\mathbb{S}$ is the viscous stress tensor and $\boldsymbol{q}$ is the heat flux vector.

The viscous stress tensor for a Newtonian fluid is: $\mathbb{S} = 2\mu\mathbb{D} + \lambda(\boldsymbol{\nabla \cdot} \boldsymbol{v})\mathbb{I}$, with $\mathbb{D} = \frac{1}{2}(\boldsymbol{\nabla}^{\mathrm{T}}\boldsymbol{v} + \boldsymbol{\nabla}\boldsymbol{v})$ the symmetric tensor deformation, and μ the viscosity coefficient. Using the Stokes hypothesis, the coefficient λ can be related to μ by the relation $\lambda + \dfrac{2}{3}\mu = 0$ and the stress tensor reads

$$\mathbb{S} = -\frac{2}{3}\mu(\boldsymbol{\nabla \cdot} \boldsymbol{v})\mathbb{I} + \mu(\boldsymbol{\nabla}^{\mathrm{T}}\boldsymbol{v} + \boldsymbol{\nabla}\boldsymbol{v}). \tag{5.9}$$

The heat flux, using the Fourier law, can be written as $\boldsymbol{q} = -\kappa\boldsymbol{\nabla}T$, where the thermal conductivity (κ) is evaluated using the constant Prandtl number hypothesis

$$\kappa = c_p \frac{\mu}{Pr}, \tag{5.10}$$

with c_p the specific heat at constant pressure and Pr the Prandtl number. Typically, $Pr = 0.72$.

In the simplest case, the viscosity coefficient (μ) is considered constant, however, for a more realistic modeling, the viscosity coefficients depends on the thermodynamics conditions as well as the type of fluid. For fluids at moderate temperature $\big(100\ [\mathrm{K}] \leq T \leq 2000\ [\mathrm{K}]\big)$, the viscosity coefficient can be assumed to be function of the temperature only, and the following equation can be used

$$\mu = \mu_0\, \tilde{\mu}(T, T_0) = \mu_0 \left(\frac{T}{T_0} \right)^{\frac{2}{3}} \frac{T_0 + C_s}{T + C_s}, \tag{5.11}$$

know as Sutherland's law, in which the temperature is in Kelvin degrees as well as the Sutherland's constant $C_s = 110.4\,[\mathrm{K}]$. The values of μ_0 and T_0 can be arbitrarily chosen provided that

$$\mu \frac{T + C_s}{T^{\frac{2}{3}}} = \mu_0 \frac{T_0 + C_s}{T_0^{\frac{2}{3}}} = 1.458 \times 10^{-6}. \tag{5.12}$$

By introducing the viscous flux function $\mathbf{f}^v(\mathsf{u}, \boldsymbol{\nabla}\mathsf{u})$

$$\mathbf{f}^v(\mathsf{u}, \boldsymbol{\nabla}\mathsf{u}) = \begin{pmatrix} 0 \\ \mathbb{S} \\ \mathbb{S} \cdot \dfrac{\boldsymbol{m}}{\rho} + \kappa \boldsymbol{\nabla} T \end{pmatrix}, \tag{5.13}$$

the Navier-Stokes system of equations can be recasted in the following vector form

$$\frac{\partial \mathsf{u}}{\partial t} + \boldsymbol{\nabla} \cdot \mathbf{f}^a(\mathsf{u}) = \boldsymbol{\nabla} \cdot \mathbf{f}^v(\mathsf{u}, \boldsymbol{\nabla}\mathsf{u}) \tag{5.14}$$

It is well know that the viscous flux function is homogeneous with the respect to the gradient of the conservative variables, namely

$$\mathbf{f}^v(\mathsf{u}, \boldsymbol{\nabla}\mathsf{u}) = \mathbb{K}(\mathsf{u}) \boldsymbol{\nabla}\mathsf{u}, \tag{5.15}$$

where the tensor $\mathbb{K}(\mathsf{u}) \in \mathbb{R}^{m \times d \times m \times d}$, with m the number of the equations and d the number of the spatial dimensions, is defined as

$$K_{ijrs}(\mathsf{u}) = \frac{\partial f_{ij}^v(\mathsf{u}, \boldsymbol{\nabla}\mathsf{u})}{\partial \left(\dfrac{\partial u_r}{\partial x_s} \right)} \tag{5.16}$$

with $i, r = 1, \ldots, m$ and $j, s = 1, \ldots, d$. The expression of the tensor $\mathbb{K}$ is reported in Appendix A.

5.1.3 Dimensionless Form of the Equations

From a numerical point of view it is always preferable to work with dimensionless equations. The non-denationalization process simply consists in applying a change of variable so that a generic dimensional quantity w is made non-dimensional by defining

$$w = \tilde{w} w_r, \tag{5.17}$$

where w_r is a dimensional reference quantity. If one introduces the reference length l_r and the reference velocity v_r, it is possible to define the dimensionless time and the dimensionless spatial coordinates as follows

$$\tilde{t} = \frac{t}{l_r / v_r}, \quad \tilde{x} = \frac{x}{l_r}, \quad \tilde{y} = \frac{y}{l_r}, \quad \tilde{z} = \frac{z}{l_r}, \tag{5.18}$$

the dimensionless temporal and spatial differential operators are related to the respective dimensional operators by the following relations

$$\frac{\partial}{\partial t} = \frac{v_r}{l_r} \frac{\partial}{\partial \tilde{t}}, \quad \boldsymbol{\nabla} = \frac{1}{l_r} \tilde{\boldsymbol{\nabla}}. \tag{5.19}$$

If the dimensionless version of the following variables is introduced

$$\begin{aligned} \tilde{\rho} &= \frac{\rho}{\rho_r}, \quad \tilde{\boldsymbol{v}} = \frac{\boldsymbol{v}}{v_r}, \quad \tilde{e^t} = \frac{e^t}{e^t_r}, \quad \tilde{P} = \frac{P}{P_r}, \\ \tilde{T} &= \frac{T}{T_r}, \quad \tilde{\kappa} = \frac{\kappa}{\kappa_r}, \quad \tilde{c_p} = \frac{c_p}{c_{p_r}}, \quad \tilde{\mu} = \frac{\mu}{\mu_r}, \end{aligned} \tag{5.20}$$

after easy simplifications, the non-dimensional Navier-Stokes equations read

$$\begin{cases} \dfrac{\partial \tilde{\rho}}{\partial \tilde{t}} + \tilde{\boldsymbol{\nabla}} \cdot \tilde{\boldsymbol{m}} = 0 \\ \dfrac{\partial \tilde{\boldsymbol{m}}}{\partial \tilde{t}} + \tilde{\boldsymbol{\nabla}} \cdot \left(\dfrac{\tilde{\boldsymbol{m}} \otimes \tilde{\boldsymbol{m}}}{\tilde{\rho}} + \left[\dfrac{P_r}{\rho_r v_r^2} \right] \tilde{P} \mathbb{I} \right) = \dfrac{1}{Re_r} \tilde{\boldsymbol{\nabla}} \cdot \tilde{\mathbb{S}} \\ \dfrac{\partial \tilde{E}^t}{\partial \tilde{t}} + \tilde{\boldsymbol{\nabla}} \cdot \left(\left(\tilde{E}^t + \left[\dfrac{P_r}{\rho_r e^t_r} \right] \tilde{P} \right) \dfrac{\tilde{\boldsymbol{m}}}{\tilde{\rho}} \right) = \dfrac{1}{Re_r} \tilde{\boldsymbol{\nabla}} \cdot \left(\left[\dfrac{v_r^2}{e^t_r} \right] \tilde{\mathbb{S}} \cdot \dfrac{\tilde{\boldsymbol{m}}}{\tilde{\rho}} + \dfrac{1}{Pr_r} \left[\dfrac{c_{p_r} T_r}{e^t_r} \right] \tilde{\kappa} \tilde{\boldsymbol{\nabla}} \tilde{T} \right) \end{cases} \tag{5.21}$$

where

$$Re_r = \frac{\rho_r v_r l_r}{\mu_r} \qquad \text{and} \qquad Pr_r = \frac{\mu_r c_{p_r}}{\kappa_r}, \tag{5.22}$$

are the reference Reynolds number and the reference Prandtl number, respectively.

The guiding principle of non-dimensionalization imposes that the dimensionless system of equations recovers exactly the same formal structure as its dimensional counterpart. For this to happen, in the system (5.21), all terms in the square brackets must amount to unity, as well as the reference Reynolds number and the reference Prandtl number, so that the reference values must satisfy the following conditions

$$\frac{P_r}{\rho_r v_r^2} = 1, \quad \frac{v_r^2}{e^t_r} = 1, \quad \frac{\rho_r v_r l_r}{\mu_r} = 1 \quad \text{and} \quad \frac{\mu_r c_{p_r}}{\kappa_r} = 1. \tag{5.23}$$

The non-dimensional thermodynamic relations are obtained in the same way showed for the Navier-Stokes equations. Namely, the relations (5.3) can be put in dimensionless form as follows

$$\begin{aligned} P_r \tilde{P} = \rho_r \tilde{\rho}\, \mathcal{R}_r \tilde{\mathcal{R}}\, T_r \tilde{T} \quad &\Rightarrow \quad \left[\frac{P_r}{\mathcal{R}_r T_r \rho_r} \right] \tilde{P} = \tilde{\mathcal{R}} \tilde{T} \tilde{\rho}, \\ e_r \tilde{e} = \frac{\mathcal{R}_r \tilde{\mathcal{R}}\, T_r \tilde{T}}{\gamma - 1} \quad &\Rightarrow \quad \left[\frac{e_r}{\mathcal{R}_r T_r} \right] \tilde{e} = \frac{\tilde{\mathcal{R}} \tilde{T}}{\gamma - 1}. \end{aligned}$$

By imposing that the terms in the square brackets are equal to unity, the dimensionless thermodynamic relations are equal to their dimensional counterpart if the following conditions are satisfied

$$\mathcal{R}_r = \frac{P_r}{T_r \rho_r} \quad \text{and} \quad e_r = \frac{P_r}{\rho_r}. \tag{5.24}$$

Furthermore, since the following thermodynamic relation holds

$$\mathcal{R} = (\gamma - 1)c_v = \frac{\gamma - 1}{\gamma} c_p, \tag{5.25}$$

where c_v is the specific heat at constant volume, it is easy to see that the following condition must be also satisfied

$$c_{p_r} = c_{v_r} = \mathcal{R}_r. \tag{5.26}$$

In order to define a complete non-dimensionalization of the equations, it is necessary to choose four independent reference quantities and the remaining quantities are obtained as function of the fundamental ones by using the conditions (5.23 – 5.26). Any set of four independent reference quantities can be used to make the equations dimensionless, but obviously some sets of variables are more practical than others. For external aerodynamic problems the following set of variables is generally used

$$(l_r, \rho_r, v_r, T_r), \tag{5.27}$$

and by using the conditions (5.23 – 5.26) the remaining reference quantities read

$$P_r = \rho_r v_r^2, \quad e_r^t = v_r^2, \quad \mu_r = \rho_r v_r l_r, \quad \mathcal{R}_r = \frac{v_r^2}{T_r}, \quad \kappa_r = \frac{\rho_r v_r^3 l_r}{T_r}. \tag{5.28}$$

The choice of the values of the fundamental reference quantities is mainly matter of convenience and depends on the applications considered. A typical choice for external aerodynamic problems consists in taking as reference quantities the variables at the infinity

$$\begin{cases} l_r = l & \text{some length: chord, mean aerodynamic chord, } \ldots \\ \rho_r = \rho_\infty & \text{upstream infinity static density,} \\ v_r = c_\infty & \text{upstream infinity speed of sound,} \\ T_r = T_\infty & \text{upstream infinity static temperature.} \end{cases} \tag{5.29}$$

According to the previous choice of the reference quantities, the dimensionless thermodynamic quantities now write as follows

$$\tilde{\mathcal{R}} = \frac{\mathcal{R}}{\mathcal{R}_r} = \mathcal{R}\frac{T_\infty}{c_\infty^2} = \mathcal{R}\frac{T_\infty}{\gamma \mathcal{R} T_\infty} = \frac{1}{\gamma}, \tag{5.30}$$

$$\tilde{c_v} = \frac{c_v}{c_{v_r}} = \frac{\mathcal{R}}{(\gamma - 1)\mathcal{R}_r}, = \frac{1}{\gamma(\gamma - 1)}, \tag{5.31}$$

$$\tilde{c_p} = \frac{c_p}{c_{p_r}} = \frac{\gamma \mathcal{R}}{(\gamma - 1)\mathcal{R}_r} = \frac{1}{\gamma - 1}, \tag{5.32}$$

and the dimensionless variables at infinity now read

$$\begin{aligned}
\tilde{P}_\infty &= \frac{P_\infty}{P_r} = \frac{P_\infty}{\rho_\infty c_\infty^2} = \frac{1}{\gamma}, \qquad \tilde{\rho}_\infty = \frac{\rho_\infty}{\rho_r} = 1, \\
\tilde{e}_\infty &= \frac{\tilde{P}_\infty}{\tilde{\rho}_\infty(\gamma-1)} = \frac{1}{\gamma(\gamma-1)}, \quad \tilde{c}_\infty = \frac{c_\infty}{c_r} = 1, \\
\tilde{\boldsymbol{v}}_\infty &= \frac{\boldsymbol{v}_\infty}{v_r} = \frac{\boldsymbol{v}_\infty}{c_\infty} = M_\infty \hat{\boldsymbol{v}}_\infty, \qquad \tilde{T}_\infty = \frac{T_\infty}{T_r} = 1,
\end{aligned} \tag{5.33}$$

where $\hat{\boldsymbol{v}}_\infty$ is the unit length vector directed along the velocity vector.

The dimensionless dynamic viscosity can be written as

$$\tilde{\mu} = \frac{\mu}{\mu_r} = \frac{1}{l\rho_\infty c_\infty}\mu_0 \left(\frac{T}{T_0}\right)^{\frac{2}{3}} \frac{T_0 + C_s}{T + C_s}, \tag{5.34}$$

taking $\mu_0 = \mu_\infty$ and therefore $T_0 = T_\infty$ in the Sutherland law, is possible to rewrite the previous expression as

$$\begin{aligned}
\tilde{\mu} &= \frac{\mu_\infty}{l\rho_\infty c_\infty} \left(\frac{T}{T_\infty}\right)^{\frac{2}{3}} \frac{T_\infty + C_s}{T + C_s} \\
&= \frac{\mu_\infty}{l\rho_\infty v_\infty} \frac{v_\infty}{c_\infty} \frac{1 + \tilde{C}_s}{\tilde{T} + C_s} \tilde{T}^{\frac{2}{3}} \\
&= \frac{M_\infty}{Re_\infty} \frac{1 + \tilde{C}_s}{\tilde{T} + \tilde{C}_s} \tilde{T}^{\frac{2}{3}},
\end{aligned} \tag{5.35}$$

where M_∞ and Re_∞ are, respectively, the Mach number and the Reynolds number computed with quantities at the infinity, while $\tilde{C}_s = C_s/T_\infty$.

The non-dimensional thermal conductivity can be expressed as

$$\begin{aligned}
\tilde{\kappa} &= \frac{\kappa}{\kappa_r} = \kappa \frac{T_\infty}{l\rho_\infty c_\infty^3} = \frac{\kappa}{l\rho_\infty c_\infty \mathcal{R}_r} \\
&= \frac{\kappa}{l\rho_\infty v_\infty \mathcal{R}_r} M_\infty = \frac{\mu}{l\rho_\infty v_\infty} M_\infty \frac{\kappa c_p}{\mu(\gamma-1)} \\
&= \frac{\tilde{\mu}}{(\gamma-1)Pr}
\end{aligned} \tag{5.36}$$

where the Prandtl number is defined as $Pr = \dfrac{\mu c_p}{\kappa}$.

Obviously the dimensionalization procedure for the Euler equations is the same showed for the Navier-Stokes equation without considering the viscous and the thermal terms.

5.2 RD Spatial Discretization of System of Equations

Given the RD space discretization for the scalar equation, as seen in Chapters 3 and 4, the extension to the system of equations is almost straightforward; the key

elements for the RD discretization remain the same. In this section the linear and non-linear central schemes are extended to the discretization of the Euler and the Navier-Stokes equations.

As seen in the scalar case, the starting point for the RD space discretization is the approximation of the continuous solution, in particular each component of the vector of the conservative variables is expanded with Lagrangian functions. In vector form one gets

$$\mathsf{u}_h(\boldsymbol{x}) = \sum_{i \in \mathcal{N}_h} \psi_i(\boldsymbol{x})\, \mathsf{u}_i, \tag{5.37}$$

where u_h is the approximation of the vector of the conservative variables, $\mathcal{N}_h$ is as usual the set of all the degrees of freedom of the grid, $\psi_i(\boldsymbol{x})$ are the same basis functions used for the scalar case and u_i is the vector of the conservative variables at the degree of freedom i.

5.2.1 Discretization of the Euler Equations

In case of purely advective problems, *i.e.*, Euler equations, the total residual is computed as follows

$$\Phi^e\big(\mathsf{u}^h\big) = \int_{\Omega_e} \boldsymbol{\nabla}\cdot \mathbf{f}^a(\mathsf{u}_h)\, \mathrm{d}\Omega = \oint_{\partial\Omega_e} \mathbf{f}^a(\mathsf{u}_h)\cdot \hat{\boldsymbol{n}}\, \mathrm{d}\partial\Omega, \tag{5.38}$$

where Φ^e is now a vector whose components are the total residual of each component of the system. Once computed the total residual, the next step consists in distributing the residual to each degree of freedom of the element.

Multidimensional upwind RD schemes are based on the matrix formulation, for example the N and the LDA schemes.In these schemes the scalar formulation is extended to the system by introducing the matrix counterpart of the parameter (3.36) defined for scalar problems, namely

$$K_j^e = \frac{\bar{\boldsymbol{A}}\cdot \boldsymbol{n}_{\boldsymbol{j}}^e}{2|\Omega_e|}, \tag{5.39}$$

where $\boldsymbol{A} = \boldsymbol{\nabla}_{\mathsf{u}}\mathbf{f}^a$ is the Jacobian matrix of the Euler flux function. Another approach makes use of central schemes, which are the ones used in this work.

Linear Scheme

The vector formulation of the linear scheme introduced for the scalar advection scalar equation reads

$$\Phi_i^{e,\mathrm{LW}} = \frac{\Phi^e}{N_{\mathrm{dof}}^e} + \int_{\Omega_e} \boldsymbol{A}\cdot\boldsymbol{\nabla}\psi_i\, \Xi\, \boldsymbol{A}\cdot\boldsymbol{\nabla}\mathsf{u}_h\, \mathrm{d}\Omega, \tag{5.40}$$

where Ξ is a scaling matrix which is taken as

$$\Xi = \frac{1}{2}|\Omega_e| \left(\sum_{i\in\mathcal{N}_i^e} \mathsf{R}_{n_i}(\bar{\mathsf{u}})\, \Lambda_{n_i}^{+}(\bar{\mathsf{u}})\, \mathsf{L}_{n_i}(\bar{\mathsf{u}}) \right)^{-1} \tag{5.41}$$

where $\bar{\mathsf{u}}$ is the arithmetic average of the conservative variables on the element, R_n, L_n are respectively the matrices of the right and left eigenvectors along the direction of the generic vector $\boldsymbol{n}$, and $\Lambda_n = \mathrm{diag}(\lambda_n)$ is the corresponding diagonal matrix of the eigenvalues. The operator $(\cdot)^+$ selects only the positive values and sets the negative ones to zero. The vector $\boldsymbol{n}_i$ is taken as follows

$$\boldsymbol{n}_i = \frac{1}{N_{\mathrm{dim}}} \int_{\Omega_e} \boldsymbol{\nabla} \psi_i \, \mathrm{d}\Omega, \tag{5.42}$$

such that it has the dimensions of a length (surface) in two (three) spatial dimensions.

Non-Linear Scheme

A central non-linear scheme for a system of equations is constructed in the same way shown for the scalar equation. The starting point is the first order accurate Rusanov scheme, which is re-written here for the system

$$\Phi_i^{e,\mathrm{Rv}} = \frac{1}{N_{\mathrm{dof}}^e} \Phi^e + \frac{1}{N_{\mathrm{dof}}^e} \alpha^e \sum_{\substack{j \in \mathcal{N}_h^e \\ j \neq i}} (\mathsf{u}_i - \mathsf{u}_j), \tag{5.43}$$

with α^e taken as

$$\alpha^e = \max_{j \in \mathcal{N}_h^e} |\lambda_{n_j}| > 0. \tag{5.44}$$

In the next step, the distribution coefficients of the low-order scheme are mapped into non-linear bounded distribution coefficients by the means of a non-linear map. As described in [3], the mapping for a system of equations is constructed in the characteristic space, consequently the distributed and the total residuals are first rewritten as

$$\Phi_i^{e,\star} = \mathsf{L}_n \Phi_i^{e,\mathrm{Rv}} \qquad \text{and} \qquad \Phi^{e,\star} = \sum_{i \in \mathcal{N}_h^e} \Phi_i^{e,\star}, \tag{5.45}$$

where the mean fluid velocity vector on the element is used as direction vector for the computation of the eigenvectors. The distributed high-order residuals are obtained by applying a non-linear mapping to the original unbounded distribution coefficients, $\beta_i^{e,\star} = \Phi_i^{e,\star} / \Phi^{e,\star}$. The map is constructed as follows

$$\hat{\beta}_i^{e,\star} = \frac{\left(\dfrac{\Phi_i^{e,\star}}{\Phi^{e,\star}} \right)^+}{\displaystyle\sum_{j \in \mathcal{N}_h^e} \left(\frac{\Phi_j^{e,\star}}{\Phi^{e,\star}} \right)^+}, \tag{5.46}$$

which corresponds to the scalar PSI limiter. The limited residual in the characteristic space is firstly computed as follows

$$\hat{\Phi}_i^{e,\star} = \hat{\beta}_i^{e,\star} \, \Phi^{e,\star}, \tag{5.47}$$

and finally, the high order distributed residuals are projected back onto the physical space: $\hat{\Phi}_i^e = \mathsf{R}_n \hat{\Phi}_i^{e,\star}$.

As already mentioned in the case of scalar problem, the use of a central scheme in combination with the limiting technique, produces undamped spurious modes and a poor iterative convergence to the steady state solution, due to the fact that no upwind mechanism is included. The cure to this problem consists in adding a filtering term by means of a streamline dissipation term, hence the final form of the non-linear scheme reads

$$\hat{\Phi}_i^{e,\mathrm{Rv}} = \hat{\Phi}_i^e + \varepsilon_h^e(\mathsf{u}_h) \int_{\Omega_e} \boldsymbol{A} \cdot \boldsymbol{\nabla} \psi_i \, \Xi \, \boldsymbol{A} \cdot \boldsymbol{\nabla} \mathsf{u}_h \, \mathrm{d}\Omega, \tag{5.48}$$

where

$$\Xi = \frac{1}{2} |\Omega_e| \left(\sum_{i \in \mathcal{N}_h^e} \mathsf{R}_{n_i}(\bar{\mathsf{u}}) \, \Lambda_{n_i}^+(\bar{\mathsf{u}}) \, \mathsf{L}_{n_i}(\bar{\mathsf{u}}) \right)^{-1}, \tag{5.49}$$

and $\varepsilon_h^e(\mathsf{u}_h)$ is a smoothness sensor which assures that the filtering term is added only in the smooth regions of the solution, namely $\varepsilon_h^e(\mathsf{u}_h) \sim 1$ in smooth regions and $\varepsilon_h^e(\mathsf{u}_h) \sim 0$ near discontinuities [8].

The limited procedure can be applied also to the N scheme, resulting in the popular PSI scheme, widely used for the numerical discretization of the Euler equation [118], also in a blended form with the LDA scheme [42], however it is well know that such a scheme has a very poor iterative convergence in the case of systems, making it less attractive than the central non-linear scheme.

5.2.2 Discretization of the Navier-Stokes Equations

As seen in Chapter 4, in the case of advection-diffusion scalar problems, the discretization of diffusive terms in the proposed RD framework requires the reconstruction of an unique value of the gradient of the numerical solution at the degrees of freedom. The continuous approximation of the gradient allows to express the total residual on each element as follows

$$\Phi^e = \oint_{\partial\Omega_e} \left(\mathbf{f}^a(\mathsf{u}_h) - \mathbb{K}(\mathsf{u}_h) \widetilde{\boldsymbol{\nabla} \mathsf{u}_h} \right) \cdot \hat{\boldsymbol{n}} \, \mathrm{d}\partial\Omega, \tag{5.50}$$

where $\widetilde{\boldsymbol{\nabla} \mathsf{u}_h}$ is an unique value of the gradient of the numerical solution reconstructed with one of the techniques described in Section 4.4.

From a practical point of view, the gradient of each component of the vector of the conservative variables is reconstructed, although one can consider to reconstruct the gradient of the primitive variables which is then used to compute the gradient of the conservative variables. In numerical experiments it has been observed that the reconstruction of the gradient of the primitive variables produces only a marginal improvement in the accuracy of the reconstructed gradient, hence in numerical simulation the gradient of the conservative variables is reconstructed directly.

Linear Scheme

The linear scheme described for the Euler equations is now extended to include viscous effects. The scheme is the vector counterpart of the scheme (4.31) proposed for the scalar advection-diffusion equation, hence it reads

$$\Phi_i^{e,\mathrm{LW}} = \frac{\Phi^e}{N_{\mathrm{dof}}^e} + \int_{\Omega_e} \boldsymbol{A}\cdot\boldsymbol{\nabla}\psi_i\,\Xi\left(\boldsymbol{A}\cdot\boldsymbol{\nabla}\mathsf{u}_h - \boldsymbol{\nabla}\cdot\left(\mathbb{K}\widetilde{\boldsymbol{\nabla}\mathsf{u}}\right)\right)\mathrm{d}\Omega, \tag{5.51}$$

with

$$\Xi = \frac{1}{2}|\Omega_e|\left(\sum_{i\in\mathcal{N}_h^e}\mathsf{R}_{n_i}(\bar{\mathsf{u}})\,\Lambda_{n_i}^{+}(\bar{\mathsf{u}})\,\mathsf{L}_{n_i}(\bar{\mathsf{u}}) + \sum_{j=1}^{N_{\mathrm{dim}}}\mathsf{K}_{jj}(\bar{\mathsf{u}})\right)^{-1}. \tag{5.52}$$

As described in Section 4.3.2, an improved version of the previous scheme takes into account of the difference between the reconstructed and the internal gradient on the element, namely

$$\begin{aligned}\Phi_i^{e,\mathrm{LW}} = \frac{\Phi^e}{N_{\mathrm{dof}}^e} &+ \int_{\Omega_e}\boldsymbol{A}\cdot\boldsymbol{\nabla}\psi_i\,\Xi\left(\boldsymbol{A}\cdot\boldsymbol{\nabla}\mathsf{u}_h - \boldsymbol{\nabla}\cdot\left(\mathbb{K}\widetilde{\boldsymbol{\nabla}\mathsf{u}}\right)\right)\mathrm{d}\Omega \\ &+ \int_{\Omega_e}\mathbb{K}\boldsymbol{\nabla}\psi_i\cdot\left(\boldsymbol{\nabla}\mathsf{u}_h - \widetilde{\boldsymbol{\nabla}\mathsf{u}_h}\right)\mathrm{d}\Omega\end{aligned} \tag{5.53}$$

Non-Linear Scheme

With the Eq. (5.40) and the Eq. (4.33) in mind, the non-linear central RD scheme for the discretization of the Navier-Stokes equation reads

$$\hat{\Phi}_i^{e,\mathrm{Rv}} = \hat{\Phi}_i^e + \varepsilon_h^e(\mathsf{u}_h)\int_{\Omega_e}\left(\boldsymbol{A}\cdot\boldsymbol{\nabla}\psi_i - \mathbb{K}\boldsymbol{\nabla}\psi_i\right)\Xi\left(\boldsymbol{A}\cdot\boldsymbol{\nabla}\mathsf{u}_h - \boldsymbol{\nabla}\cdot\left(\mathbb{K}\widetilde{\boldsymbol{\nabla}\mathsf{u}_h}\right)\right)\mathrm{d}\Omega, \tag{5.54}$$

with $\hat{\Phi}_i^e$ the limited version of the residual distributed with the Rusanov scheme and

$$\Xi = \frac{1}{2}|\Omega_e|\left(\sum_{i\in\mathcal{N}_h^e}\mathsf{R}_{n_i}(\bar{\mathsf{u}})\,\Lambda_{n_i}^{+}(\bar{\mathsf{u}})\,\mathsf{L}_{n_i}(\bar{\mathsf{u}}) + \sum_{j=1}^{N_{\mathrm{dim}}}\mathsf{K}_{jj}(\bar{\mathsf{u}})\right)^{-1}. \tag{5.55}$$

Also the non-linear scheme is modified with the addition of the extra stabilization term for the viscous part

$$\begin{aligned}\hat{\Phi}_i^{e,\mathrm{Rv}} = \hat{\Phi}_i^e + \varepsilon_h^e(\mathsf{u}_h)&\int_{\Omega_e}\left(\boldsymbol{A}\cdot\boldsymbol{\nabla}\psi_i - \mathbb{K}\boldsymbol{\nabla}\psi_i\right)\Xi\left(\boldsymbol{A}\cdot\boldsymbol{\nabla}\mathsf{u}_h - \boldsymbol{\nabla}\cdot\left(\mathbb{K}\widetilde{\boldsymbol{\nabla}\mathsf{u}_h}\right)\right)\mathrm{d}\Omega \\ &+ \int_{\Omega_e}\mathbb{K}\boldsymbol{\nabla}\psi_i\cdot\left(\boldsymbol{\nabla}\mathsf{u}_h - \widetilde{\boldsymbol{\nabla}\mathsf{u}_h}\right)\mathrm{d}\Omega\end{aligned} \tag{5.56}$$

Note that for the system, in the non-linear scheme, as well as the linear one, the streamline stabilization term is not blended with the stabilization term for the

viscous part via a parameter function of the local Reynolds number, as it has been done instead in the scalar case; this choice gives more robustness to the numerical solver, since it was observed that the blending might cause a breakdown of the numerical simulations.

5.3 Boundary Conditions

The imposition of the boundary conditions in the RD framework has not reached yet a mature level of understanding, for this reason it is shown here how boundary conditions are imposed in this work.

Consider the weak form of a generic steady conservation law

$$\int_{\Omega} \psi \boldsymbol{\nabla} \cdot \mathbf{f}(\mathsf{u}_h) \, \mathrm{d}\Omega = 0, \tag{5.57}$$

an integration by parts gives

$$\int_{\partial\Omega} \mathbf{f}(\mathsf{u}^{\partial}) \cdot \boldsymbol{n} \, \mathrm{d}\partial\Omega - \int_{\Omega} \boldsymbol{\nabla}\psi \cdot \mathbf{f}(\mathsf{u}_h) \, \mathrm{d}\Omega = 0, \tag{5.58}$$

where u^{∂} represents the solution state that takes into account the boundary conditions. Replacing the basis function ψ with the standard Lagrangian basis functions ψ_i, $\forall\, i \in \mathcal{N}_h$, and applying again the integration by parts to the second member on the left-hand side of the previous equation, one obtains

$$\int_{\Omega_i} \psi_i \boldsymbol{\nabla} \cdot \mathbf{f}(\mathsf{u}_h) \, \mathrm{d}\Omega + \int_{\partial\Omega_i \cap \partial\Omega} \psi_i \big(\mathbf{f}(\mathsf{u}^{\partial}) - \mathbf{f}(\mathsf{u}_h)\big) \cdot \boldsymbol{n} \, \mathrm{d}\partial\Omega = 0, \tag{5.59}$$

where Ω_i is the union of the elements that have the degree of freedom i in common. The boundary integral on the previous equation can be regarded as the contribution that must be added to the residuals computed on the domain elements without considering the boundary conditions. Note that the correction flux, $\mathbf{f}(\mathsf{u}^{\partial}) - \mathbf{f}(\mathsf{u}_h)$, becomes null as the boundary conditions are correctly enforced into the numerical scheme. With the Eq. (2.12) in mind, the complete RD space discretization, which includes the imposition of the weak boundary conditions, can be written as follows

$$\sum_{e \in \mathcal{E}_{h,i}} \Phi_i^e + \sum_{f \in \mathcal{F}_{h,i}} \Phi_i^{e,\partial} = 0, \quad \forall i \in \mathcal{N}_h, \tag{5.60}$$

where the boundary residuals reads

$$\Phi_i^{e,\partial} = \int_{\partial\Omega_i \cap \partial\Omega} \psi_i \big(\mathbf{f}(\mathsf{u}^{\partial}) - \mathbf{f}(\mathsf{u}_h)\big) \cdot \boldsymbol{n} \, \mathrm{d}\partial\Omega \tag{5.61}$$

and $\mathcal{F}_{h,i}$ is the set of the boundary faces which share the degree of freedom i; the set is empty if i is not on the boundary.

Slip Wall

For an inviscid wall, the boundary condition requires that the normal component of the velocity to the plane is null: $\boldsymbol{v} \cdot \hat{\boldsymbol{n}} = 0$, with $\hat{\boldsymbol{n}}$ the outward normal versor to the boundary face. The flux function and the correction flux read

$$\mathbf{f}^a(\mathsf{u}^\partial_{\text{wall}}) \cdot \hat{\boldsymbol{n}} = \begin{pmatrix} 0 \\ P\hat{\boldsymbol{n}} \\ 0 \end{pmatrix}, \qquad \Big(\mathbf{f}^a(\mathsf{u}^\partial_{\text{wall}}) - \mathbf{f}^a(\mathsf{u}_h)\Big) \cdot \hat{\boldsymbol{n}} = -v_n \begin{pmatrix} \rho \\ \rho\boldsymbol{v} \\ E^t + P \end{pmatrix}, \tag{5.62}$$

respectively, with $v_n = \boldsymbol{v} \cdot \hat{\boldsymbol{n}}$.

Subsonic Inflow/Outflow

The imposition of inflow/outflow boundary condition has to rely on a characteristics-based procedure in order to impose the correct number of conditions. The number of characteristics pointing into or out of the domain depends on the fact that the boundary is an inflow or outflow boundary.

In the case of a subsonic inflow boundary, one of the characteristics points out of the domain and the others point into the domain, this means that for a two (three) dimensional flows, three (four) conditions must be specified. For a subsonic outflow only one characteristic point into the domain, thus only one condition must be specified.

The procedure here described takes into account automatically the number of conditions which need to be imposed, provided the proper inflow/outflow state $\mathsf{u}^\partial_{\text{in/out}}$. As standard practice, the advective flux function is linearized as

$$\mathbf{f}^a(\mathsf{u}^\partial_{\text{in/out}}) \cdot \hat{\boldsymbol{n}} \simeq \boldsymbol{A}^+_n(\mathsf{u}_h)\mathsf{u}_h + \boldsymbol{A}^-_n(\mathsf{u}_h)\mathsf{u}^\partial_{\text{in/out}}, \tag{5.63}$$

where $\mathsf{A}^\pm_n(\mathsf{u}) = \mathsf{R}_n(\mathsf{u})\,\Lambda^\pm_n(\mathsf{u})\,\mathsf{L}_n(\mathsf{u})$; the operator $+$ selects the components of the solution state that must be taken from the interior of the domain, while the operator $-$ selects those components that must be imposed from the exterior. By linearizing the flux function at the interior state as follows

$$\mathbf{f}^a(\mathsf{u}_h) \cdot \hat{\boldsymbol{n}} = \boldsymbol{A}_n(\mathsf{u}_h)\mathsf{u}_h = \Big(\boldsymbol{A}^-_n(\mathsf{u}^h) + \boldsymbol{A}^+_n(\mathsf{u}^h)\Big)\mathsf{u}_h, \tag{5.64}$$

the correction flux for the imposition of the inflow/outflow boundary conditions reads

$$\Big(\mathbf{f}^a(\mathsf{u}^\partial_{\text{in/out}}) - \mathbf{f}^a(\mathsf{u}^h)\Big) \cdot \hat{\boldsymbol{n}} = \mathsf{A}^-_n(\mathsf{u}^h)(\mathsf{u}^\partial_{\text{in/out}} - \mathsf{u}^h). \tag{5.65}$$

Usually, at inflow boundary the total pressure, the total temperature (or density) and the flow direction are specified, while the velocity magnitude is taken from the interior; with these quantities it is possible to define the state $\mathsf{u}^\partial_{\text{in}}$. At the outflow boundary the static pressure is generally prescribed, while density and velocity are taken from the interior, giving the state $\mathsf{u}^\partial_{\text{out}}$.

Far-Field

In the imposition of the far-field boundary conditions, the state u^{∂} is taken as the far-field condition u_{∞}, and the procedure previously illustrated to impose subsonic inflow/outflow boundary conditions is applied. Although the number of conditions imposed might be over-specified, the procedure ignores the superfluous conditions.

Adiabatic Slip Wall

For viscous fluids at the solid surface, the adiabatic and no-slip boundary condition is applied; in this case the velocity and the normal heat flux on the wall must be zero: $\boldsymbol{v}|_{\text{wall}} = 0$, $\boldsymbol{q} \cdot \hat{\boldsymbol{n}}|_{\text{wall}} = 0$. In the RD scheme, the adiabatic no-slip boundary conditions are generally applied by initializing the velocity field such that the velocity components are zero on the wall and the residual of the momentum equation is enforced to be zero on the boundary nodes at each iteration; this corresponds to impose strongly the boundary condition for the momentum equation. In some works, nothing is explicitly done for the residual associated to the energy equation, this is partially justified by the fact that on the wall, since the velocity is zero, one has

$$\int_{\partial\Omega_{e_i}\cap\partial\Omega} (\boldsymbol{v} \cdot \mathbb{S} - \boldsymbol{q}) \cdot \hat{\boldsymbol{n}} = -\int_{\partial\Omega_{e_i}\cap\partial\Omega} \boldsymbol{q} \cdot \hat{\boldsymbol{n}}, \tag{5.66}$$

thus not computing the previous integral is equivalent to assume that $\boldsymbol{q} \cdot \hat{\boldsymbol{n}}$ is zero on the wall. However, this consideration is not true in practice and imposing the boundary conditions in such a way does not guarantee that the normal heat flux is zero on the wall.

An effective way to impose the boundary condition for the energy equation makes use of the correction flux technique showed before. Since the velocity is zero on the wall, the viscous flux function in the normal direction to the wall becomes

$$\mathbf{f}^{v}(\mathsf{u}) \cdot \hat{\boldsymbol{n}} = \begin{pmatrix} 0 \\ \mathbb{S} \cdot \hat{\boldsymbol{n}} \\ \kappa \boldsymbol{\nabla} T \cdot \hat{\boldsymbol{n}} \end{pmatrix}, \tag{5.67}$$

consequently the correction flux on the wall is

$$\mathbf{f}^{v}(\mathsf{u}^{\partial}_{\text{wall}}) = \begin{pmatrix} 0 \\ 0 \\ 0 \\ -\kappa \boldsymbol{\nabla} T \cdot \boldsymbol{n} \end{pmatrix}. \tag{5.68}$$

The gradient of the temperature for a perfect gas can be easily expressed as function of the conservative variables and their gradient, for example, considering explicitly

the x component for simplicity

$$\begin{aligned}\frac{\partial T}{\partial x} &= \frac{\gamma-1}{\mathcal{R}}\frac{\partial}{\partial x}\left(\frac{E^t}{\rho}-\frac{1}{2}\|\boldsymbol{v}\|^2\right)\\ &= \frac{\gamma-1}{\mathcal{R}}\left[\frac{\partial}{\partial x}\left(\frac{E^t}{\rho}\right)-\frac{1}{2}\left(\frac{\partial}{\partial x}\left(\frac{m_x^2}{\rho^2}\right)+\frac{\partial}{\partial x}\left(\frac{m_y^2}{\rho^2}\right)\right)\right]\\ &= \frac{\gamma-1}{\mathcal{R}}\left[\frac{1}{\rho}\frac{\partial E^t}{\partial x}-\frac{E^t}{\rho^2}\frac{\partial \rho}{\partial x}-\underbrace{\left(\frac{m_x}{\rho^2}\frac{\partial m_x}{\partial x}-\frac{m_x^2}{\rho^3}\frac{\partial \rho}{\partial x}+\frac{m_y}{\rho^2}\frac{\partial m_y}{\partial x}-\frac{m_y^2}{\rho^3}\frac{\partial \rho}{\partial x}\right)}_{=0,\text{wall}}\right],\end{aligned} \tag{5.69}$$

and therefore

$$\boldsymbol{\nabla} T = \frac{\gamma-1}{\mathcal{R}}\left(\boldsymbol{\nabla} E^t - \frac{E^t}{\rho}\boldsymbol{\nabla}\rho\right)\frac{1}{\rho}. \tag{5.70}$$

The reconstructed gradient of the conservatives variables are used in the previous equation, in this way an unique value of the gradient of the temperature at each node is obtained. As usual, the residual of the momentum equation is explicitly imposed to be zero in order to apply the strong boundary conditions for the velocity on the wall.

5.3.1 High Order Boundary Representation

Here the problem of the boundary representation is briefly recalled, While in standard FV methods a linear representation of the geometry is generally used, it is well know that when high-order methods are used, if the spatial discretization is accurate enough, the error due to the approximation of the geometry may dominate and the formal accuracy of the scheme is spoiled [18, 71].

Another consideration must be also done. The main advantage of high-order methods over the low-order ones is the capability of the high-order methods to achieve an higher level of accuracy given the same computational resources. Obviously, since a high-order method introduces more degrees of freedom than a lower one, a fair comparison between high and low order methods must be done with the same number of degrees of freedom. In other words the computational grids used by high-order methods should be coarser than those used by low-order methods, hence high-order approximation of the boundaries have to be considered to preserve the correct representation of the geometry.

As previously explained, boundary conditions are prescribed (entirely or partially) in a weak manner by specifying the flux function which takes into account the boundary state. The flux function depends on the normal vector to the boundary face, thus in order to take into account the real geometry of the boundary, the normal vector to the boundary face must be computed using the high-order element approximation.

In this work a simple approach is used for the boundary approximation: the solution and the geometry are represented with the same order of accuracy. The geometry of high-order elements is defined by the following isoparametric transformation from the reference element, $\hat{\Omega}_e$, to the physical one, Ω_e, (Figure 5.1)

$$\boldsymbol{x}(\boldsymbol{\xi}) = \sum_{i=1}^{N_g} \boldsymbol{x}_i \psi_i(\boldsymbol{\xi}), \tag{5.71}$$

where $\boldsymbol{\xi}$ is the independent variable in the reference space, $\boldsymbol{x}_i$ are the coordinates of the nodes defined in the physical space and ψ_i are the same Lagrangian basis functions used for the discretization of the solution. Note that the nodes on the faces of the element belonging to the boundary $\partial\Omega$ are places on the real geometry of the boundary. A curved boundary is therefore approximated by piecewise polynomial elements. The use of Lagrangian functions guarantees the geometric continuity between neighboring elements.

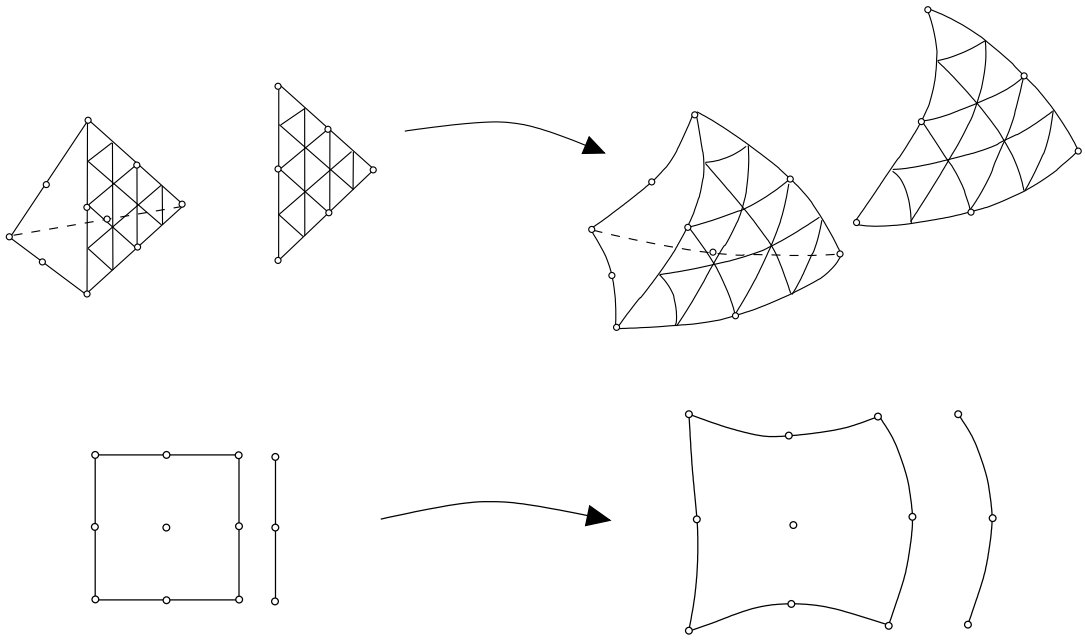
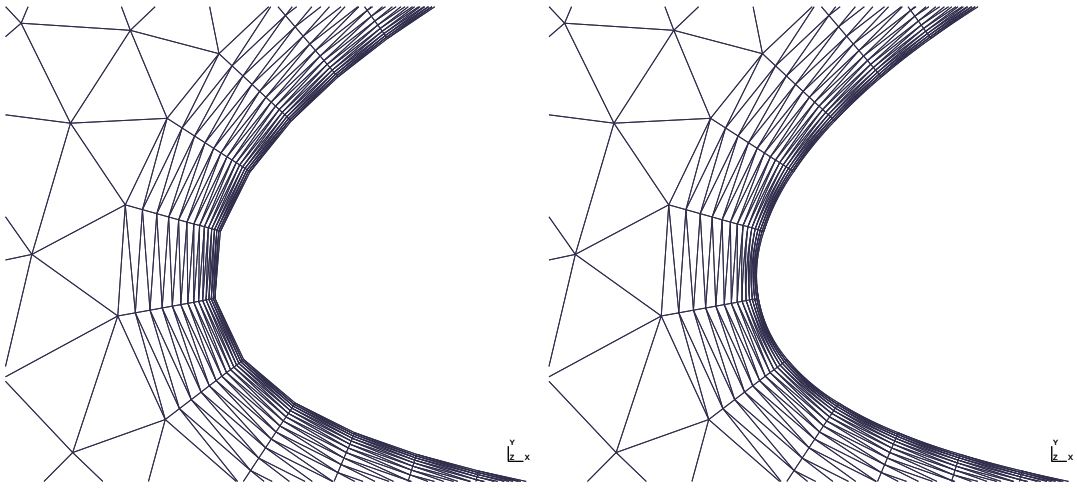


Figure 5.1: Three and two dimensional mapping of standard elements to arbitrary curved ones in the physical space.

It must be pointed out that, since the inverse of the mapping Jacobian is a rational function for curved elements, the accuracy of the quadrature formula used to computed domain and boundary integrals must be higher than that used with linear elements (with the same order of interpolation of the solution).

The generation of high-order grids is still a very active field of research [7,67,96], due to the necessity to create robust mesh generators which are able to generate high quality elements. The use of high-order elements makes the problem more stiff and if elements are very distorted high-order numerical solvers could fail. In Figure 5.2



(a) (b)

Figure 5.2: Example of linear (a) and quadratic (b) mesh around the leading edge of an airfoil.

is reported an example of linear and quadratic meshes around the leading edged of a NACA-23012 airfoil, generated with the Gmsh software [50]. Note that due to the clustering of the elements in the boundary layer, some curved faces must be generated also at the interior of the domain, otherwise the elements overlap each other.

5.4 Implicit Numerical Solver

The numerical solution of the RD method is obtained by solving the non-linear system of equations (5.60), which here is rewritten in the following form

$$\mathsf{R}(\mathsf{u}_h) = 0. \tag{5.72}$$

Among all kinds of methods, the Newton's method is one of the most popular for the solution of non-linear equations, and has a local quadratic convergence. The general form of the Newton's method for solving Eq. (5.72) is

$$\mathsf{u}_h^{k+1} = \mathsf{u}_h^k - \mathsf{J}^{-1}\mathsf{R}(\mathsf{u}_h^k), \quad k = 0, 1, 2, \ldots \tag{5.73}$$

where u_h^0 is an initial guess of the solution and $\mathsf{J} = \partial\mathsf{R}(\mathsf{u})/\partial\mathsf{u}$ the Jacobian of R, is non singular at each iteration. In practice, the Newton iteration (5.73) is implemented in the following form

$$\mathsf{J}(\mathsf{u}_h^k)\Delta\mathsf{u}_h^k = -\mathsf{R}(\mathsf{u}_h^k), \quad \text{and} \quad \mathsf{u}_h^{k+1} = \mathsf{u}_h^k + \Delta\mathsf{u}_h^k. \tag{5.74}$$

Usually the original problem (5.72) is relaxed as a pseudo-transient problem

$$\mathsf{R}(\mathsf{u}_h) = 0 \qquad \longrightarrow \qquad \frac{\mathrm{d}\mathsf{u}_h}{\mathrm{d}t} = -\mathsf{R}(\mathsf{u}_h), \tag{5.75}$$

where t is a scaled, pseudo-time variable. The steady state solution is the limit for t that tends to infinity. The presence of the time derivative enables a better convergence of the Newton's method, overcoming the harsh start-up phase when the solution is far from an optimal initial guess. Furthermore, the Jacobian of the modified problem is better conditioned than the Jacobian of the original problem during the start-up phase.

If the backward Euler formula is used for the discretization of the time derivative, the fully discrete form of the problem reads

$$\frac{\mathsf{u}_h^{n+1} - \mathsf{u}_h^n}{\Delta t^n} = -\mathsf{R}(\mathsf{u}_h^{n+1}), \qquad n = 0, 1, 2, \ldots \tag{5.76}$$

where n is the number of the time steps and u^0 is the initial value of the solution. For each time step n, a non linear problem should be solved with the Newton's method, namely

$$\left[\frac{\mathbb{I}}{\Delta t^n} + J(\mathsf{u}_{h_k}^n)\right] \Delta \mathsf{u}_{h_k}^n = -\mathsf{R}(\mathsf{u}_{h_k}^n) \tag{5.77}$$

$$\mathsf{u}_{h_{k+1}}^n = \mathsf{u}_{h_k}^n + \Delta \mathsf{u}_{h_k}^n, \qquad k = 0, 1, 2, \ldots, \tag{5.78}$$

in practice at each time step only one Newton iteration is performed.

The parameter Δt^n is the discrete time step, with $\Delta t^n \to \infty$ as $n \to \infty$. Note that for $\Delta t^n \to \infty$ the iteration of the original Newton's method (5.73) is retrieved. The evolution of the time step is controlled by the CFL number, that is chosen according to the following law [123]

$$\mathrm{CFL}^n = \mathrm{CFL}^{n-1} \frac{\|\mathsf{R}(\mathsf{u}^{n-2})\|_\infty}{\|\mathsf{R}(\mathsf{u}^{n-1})\|_\infty}, \tag{5.79}$$

starting from a low CFL number. The iterative process is stopped when the residual of the equations becomes small enough with respect to the initial residual.

5.4.1 Jacobian-Free Newton/GMRES Method

At each non-linear iteration, the linear system (5.77) must be solved; the linear system is re-written for convenience as follows

$$A(\mathsf{u}_h^n)\Delta \mathsf{u}_h^n = -\mathsf{R}(\mathsf{u}_h^n), \qquad \text{with} \qquad A(\mathsf{u}_h^n) = \left(\frac{\mathbb{I}}{\Delta t^n} + J(\mathsf{u}_h^n)\right). \tag{5.80}$$

Complete solving of the linear system is unnecessary for the convergence of the scheme, usually inexact Newton's method is used to reduce the computational effort and avoid over-solving of the system [47]. The linear system is solved until

$$\|\mathsf{R}(\mathsf{u}_h^n) + A(\mathsf{u}_h^n)\Delta \mathsf{u}_h^n\| \leq \eta_h^n \|\mathsf{R}(\mathsf{u}_h^n)\| \tag{5.81}$$

with a tolerance $\eta_h^n < 1$. In this work, a constant value for η_h^n is used during the simulation; typical values for η are 10^{-1} or 10^{-2}.

Furthermore, the matrix of the linear problem is non symmetric and very sparse, with dimension $N = N_{\text{dof}} \times N_{\text{eq}}$, thus the number of the non-zero elements can be very high. Krylov methods can be used to solve this class of linear systems, in particular the GMRES method [109] is widely used. This method has the property of minimizing the L^2-norm of the residual over all the vectors in the Krylov subspace. The GMRES method computes a new search vector every iteration. The vector is added to the Krylov subspace to progressively improve the solution. However, more search directions incur higher memory and computational costs. For large problems, this limits the maximum number of iterations that can be used. The restarted version of the algorithm can be used, where the algorithm is restarted from the most recent solution.

To accelerate the convergence of the iterative linear solver, preconditioning is used, this consists in solving a modified linear system

$$AP^{-1}P\Delta\mathsf{u}_h = -\mathsf{R}, \tag{5.82}$$

with P a preconditioning matrix for the matrix A of the linear system. When the right preconditioning is used, one first solves

$$AP^{-1}\mathsf{w} = -\mathsf{R}, \tag{5.83}$$

for w, and the solves

$$\Delta\mathsf{u}_h = P^{-1}\mathsf{w}, \tag{5.84}$$

for $\Delta\mathsf{u}_h$. Only P^{-1} is required. For sake of clarity the right-preconditioned GMRES pseudo-code is reported in Algorithm 1.

The construction of the matrix A requires to compute the Jacobian of R. In order to obtain quadratic convergence for the Newton's method, the linearization of the residual must be exact. Unfortunately, explicit calculation of the Jacobian matrix resulting from the linearization of the high-order, non-linear residuals is extremely expensive, if not impossible. The Jacobian of the low-order residual is generally used in the construction of the matrix A, but the quadratic convergence of the Newton's method is lost due to the inconsistency between the right-hand side of the linear system, constructed with the high-order residual, and the matrix on the left-hand side, constructed with the low order scheme.

In Algorithm 1 it is clear that the GMRES solver requires the matrix A only in a matrix-vector product, thus remembering that the matrix contains the Jacobian of the numerical scheme, one can approximate the product of the matrix A with a generic vector w as follows [21, 69, 121]

$$\begin{aligned} A\mathsf{w} &= \left(\frac{\mathbb{I}}{\Delta t^n} + J(\mathsf{u}_h^n)\right)\mathsf{w} \\ &\simeq \frac{\mathbb{I}}{\Delta t^n}\mathsf{w} + \frac{\mathsf{R}(\mathsf{u}_h^n + \epsilon\mathsf{w}) - \mathsf{R}(\mathsf{u}_h^n)}{\epsilon}, \end{aligned} \tag{5.85}$$

Algorithm 1 Right preconditioned GMRES. Solve $A\boldsymbol{x} = \boldsymbol{b}$

1: Choose $\boldsymbol{x}_0$
2: $\boldsymbol{r}_0 = \boldsymbol{b} - A\boldsymbol{x}_0$
3: $\beta = \|\boldsymbol{r}_0\|_2$
4: $\boldsymbol{v}_1 = \boldsymbol{r}_0/\beta$
5: Define $H_m = \{h_{i,j}\}_{1\leq i\leq m+1, 1\leq j\leq m}$
6: $H_m = 0$
7: **for** $j = 1$ to m **do**
8: $\quad \boldsymbol{w}_1 = AP^{-1}\boldsymbol{v}_j$
9: $\quad$ **for** $i = 1$ to j **do**
10: $\quad\quad h_{i,j} = (\boldsymbol{w}_j, \boldsymbol{v}_i)$
11: $\quad\quad \boldsymbol{w}_j = \boldsymbol{w}_j - h_{ij}\boldsymbol{v}_i$
12: $\quad\quad h_{j+1,j} = \|\boldsymbol{w_j}\|_2$
13: $\quad\quad$ **if** $h_{j+1,j} = 0$ **then**
14: $\quad\quad\quad$ m = j
15: $\quad\quad\quad$ goto 20
16: $\quad\quad$ **end if**
17: $\quad\quad \boldsymbol{v}_{j+1} = \boldsymbol{w}_j/h_{j+1,j}$
18: $\quad$ **end for**
19: **end for**
20: $\boldsymbol{y}_m = \underset{\boldsymbol{y}}{\text{argmin}}\ \|\beta e_1 - H_m\boldsymbol{y}\|_2$
21: $V_m = [\boldsymbol{v}_1, \ldots, \boldsymbol{v}_m]$
22: $\boldsymbol{x}_m = \boldsymbol{x}_0 + P^{-1}V_m\boldsymbol{y_m}$
23: **if** $\|\beta e_1 - H_m y\|_2 < \eta^n\|\boldsymbol{b}\|_2$ **then**
24: $\quad$ EXIT
25: **else**
26: $\quad \boldsymbol{x}_0 = \boldsymbol{x}_m$
27: $\quad$ goto 2
28: **end if**

with ϵ a small step size chosen as [94]

$$\epsilon = \frac{\sqrt{1 + \|\mathsf{u}\|_{L_2}}}{\|\mathsf{w}\|_{L_2}}\,\epsilon_{\text{rel}}, \qquad \epsilon_{\text{rel}} = 10^{-8}. \tag{5.86}$$

Since there is no need to compute explicitly the Jacobian, this approach is called Jacobian-free. However, a rough approximation of the Jacobian is always computed at each non-linear iteration, this matrix is used as a preconditioner in the GMRES algorithm.

Among all the types of preconditioning developed for CFD applications, in this work the LU-SGS [64] preconditioning is considered. This method represents a good compromise between robustness and memory requirement, and guarantees a good scalability in parallel algorithms. Consider the linear system (5.80) and given D, U,

and L, the diagonal block, the lower and the upper part of the matrix, respectively, it is easy to verify that the following factorization is true

$$(D+L)D^{-1}(D+U)\Delta \mathsf{u}_h = -\mathsf{R} + (LD^{-1}U)\Delta \mathsf{u}_h. \tag{5.87}$$

In the LU-SGS method the system is approximately factorized by neglecting the last term on the right-hand side of the previous equation, so the preconditioning matrix reads

$$P = (D+L)D^{-1}(D+U). \tag{5.88}$$

The solution of the Eq. (5.84) is obtained in two steps, each of them requires the inversion only of the diagonal block matrices:

Forward sweep

$$(D+L)\Delta \mathsf{u}_h^\star = w \qquad \Rightarrow \quad \Delta \mathsf{u}_i^\star = D_i^{-1}\left(\mathsf{w}_i - \sum_{\substack{j\in\mathcal{N}_h^i \\ j<i}} A_{ij}\Delta \mathsf{u}_j^\star\right), \quad i = 1,2,\ldots,N_{\text{dof}}$$

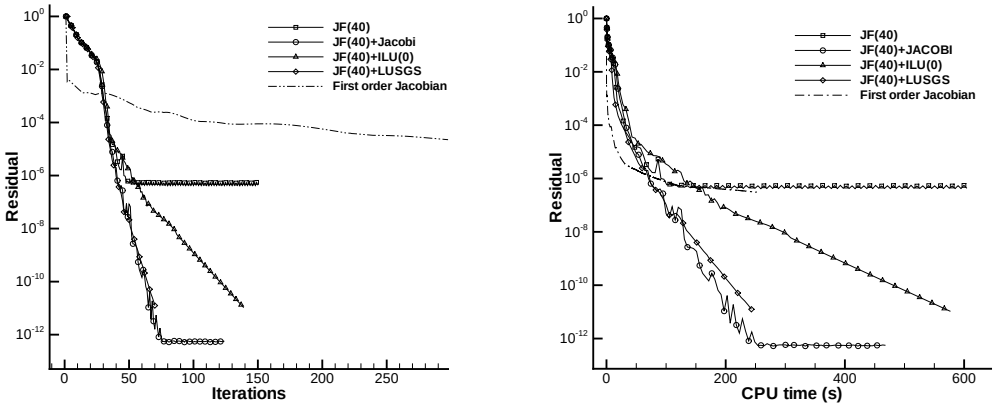
Backward sweep

$$(D+U)\Delta \mathsf{u}_h = D\Delta \mathsf{u}_h^\star \quad \Rightarrow \quad \Delta \mathsf{u}_i = \Delta \mathsf{u}_i^\star - D_i^{-1}\sum_{\substack{j\in\mathcal{N}_h^i \\ j>i}} A_{ij}\Delta \mathsf{u}_j, \quad i = N_{\text{dof}},\ldots,2,1$$

The LU-SGS algorithm here is preferred over other preconditioners since it is not computational expensive and has good convergence properties. For instance, the LU-SGS method requires only the inversion of the diagonal block matrix but it is much more effective that the Jacobi preconditioner, and at the same time it does not require to store the entire factorized matrix, like for example in ILU based preconditioners, but it is still very effective.

The Jacobian-free approach allows quadratic convergence of Newton's method because the matrix of the linear system is a complete linearization of the residual vector. The price to pay for using this technique is an increment of the computational effort, because at each time step it is necessary to compute several times the residual $\mathsf{R}(\mathsf{u})$ on the whole domain. This is largely compensated by a drastic reduction of the iterations number.

As an illustrative example, consider the simple subsonic, inviscid flow over a circular bump discretized with a third order RD non-linear scheme on an unstructured grid of triangles, see [6] for details. In Figure 5.3 the convergence curves are reported in terms of number of iterations and CPU time (in seconds) for the Jacobian-free technique with different type of preconditioners. The residual is computed as the L^2 norm of the continuity equation and is normalized with respect to the residual at the first iteration. In all the computations, 40 search directions has been used in the GMRES algorithm. It can be seen that the GMRES algorithm without preconditioning does not converge even for a simple problem, as that considered



(a) (b)

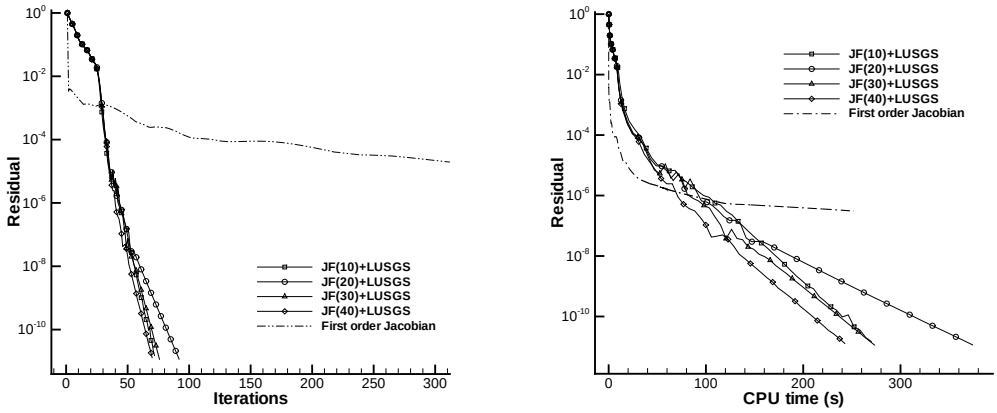
Figure 5.3: Iterative convergences as function of the number of iterations (a) and the CPU time (b) for an inviscid problem. Different preconditioners has been used for the Jacobian-free technique, the convergence history for the matrix-based implicit scheme, using the Jacobian of the first order scheme, is also reported.

here. The ILU(0) preconditioner does not perform as well as the LU-SGS preconditioners, probably because no reordering technique has been adopted, and might be for the same reason that the Jacobi and the LU-SGS preconditioner have similar performances for this particular test case.

In the Jacobian-free technique, the preconditioner matrix is computed using the Jacobian of the first order Rusanov scheme. The same matrix used directly as Jacobian in the implicit scheme does not allow the implicit scheme to converge to the zero machine, whatever the preconditioning strategy is. In Figure 5.4 is evaluated the influence of the number of the search directions in the GMRES algorithm in combination with the LU-SGS preconditioner. As expected the matrix-based implicit method, with the Jacobian of the first order scheme, does not make the scheme converge, whatever the number of search directions is.

5.5 Numerical Results for Inviscid Problems

In this section, some numerical results for inviscid problems are reported as illustrative examples of the discretization of the Euler equations with the RD approach. A smooth flow over a bump and a transonic flow over an airfoil are considered to verify the practical order of accuracy of the method and the non oscillatory properties of the numerical scheme.



(a) (b)

Figure 5.4: Iterative convergences as function of the number of iterations (a) and the CPU time (b) for an inviscid problem. Different preconditioners has been used for the Jacobian-free technique, the convergence history for the matrix-based implicit scheme, using the Jacobian of the first order scheme, is also reported.

5.5.1 Subsonic Flow over a Smooth Bump

A subsonic flow over a smooth bump is here considered, the inflow Mach number is M = 0.5, with 0° angle of incidence. The computational domain is bounded between $x = -1.5$ and $x = 1.5$, and between the bump and $y = 0.8$. The form of the bump is given by $y = 0.0625e^{-25x^2}$. An example of coarse grid with quadratic elements is reported in Figure 5.5, finer grids are obtained with an uniform refinement of the coarse grid. Slip wall boundary conditions are used for the top and bottom boundaries, while inflow and outflow conditions are applied on the left and right boundary, respectively. An example of the numerical solution is shown in Figure 5.6, for the third order non-linear scheme on a fine grid.

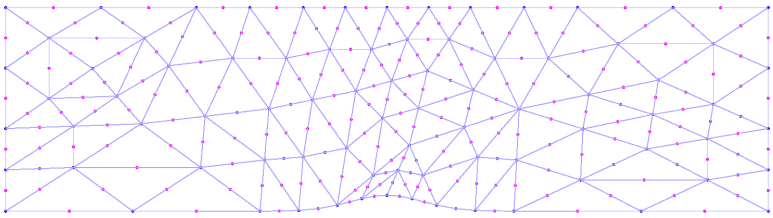


Figure 5.5: Coarse mesh, with quadratic elements, used for the simulation of the subsonic flow over the smooth bump.

Since the entropy for this case should be constant in the flow field, the following

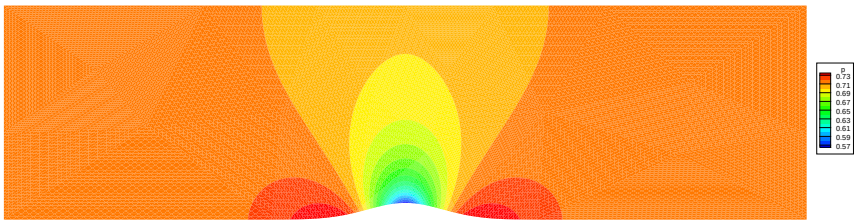


Figure 5.6: Pressure contours for the subsonic flow over the smooth bump, obtained with a third order approximation of the solution on a fine grid.

error definition is used to study the accuracy of the numerical method

$$\epsilon_{L^2} = \sqrt{\frac{\int_\Omega \left(\frac{p/\rho^\gamma - p_\infty/\rho_\infty^\gamma}{p_\infty/\rho_\infty^\gamma} \right)^2 \mathrm{d}\Omega}{\int_\Omega \mathrm{d}\Omega}}. \tag{5.89}$$

In Figure 5.7, the error is reported as function of the number of the degrees of freedom and the normalized CPU time [1], with the non-linear scheme. Note that the correct order of accuracy of the scheme is obtained for the h-refinement, with the linear and quadratic approximation of the solution. Furthermore, it is clear that the high-order discretization performs better than the low-order one also in term of computational time with respect to the second order method.

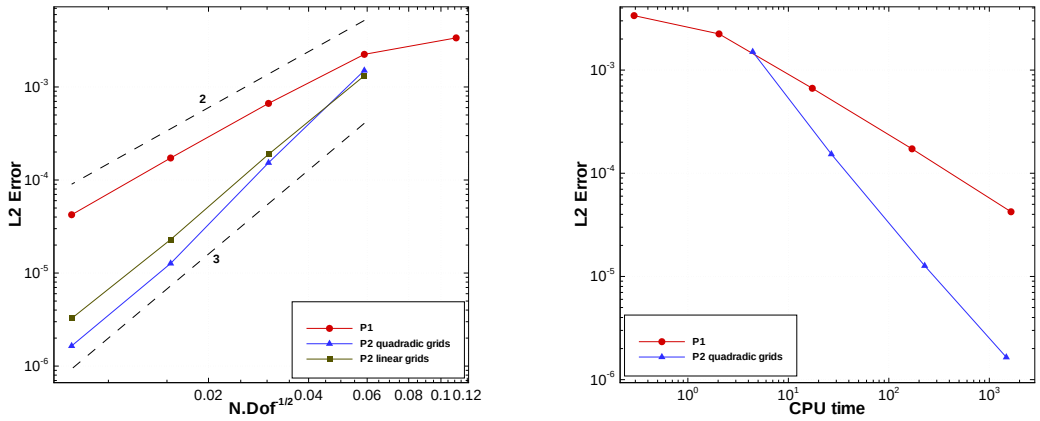
In addition, Figure 5.7-(a) reports also the variation of the entropy error with the mesh refinement for quadratic approximation of the solution and linear approximation of the geometry. It is evident that the numerical error is bigger compared to that obtained with the quadratic approximation of both geometry and solution, although for this particular test case there is not a severe loss of accuracy, which may occur for others test cases.

5.5.2 Transonic Flow over a NACA-0012 Airfoil

The transonic flow over over a NACA-0012 airfoil at 5° of incidence and free-stream Mach number M = 0.7 is now considered. The flow field is characterized by the presence of the shock located about the 45% of the chord an by a slip line detaching from the trailing edge. The aim of this case is to test the non-oscillatory property of the non-linear scheme, for both second and third order simulations.

The mesh used for the simulations is made of 12 108 elements and is shown in Figure 5.8, together with the Mach number contours for the third order simulation. The far-field boundary is situated about 25 chords away from the airfoil. All the

[1]The CPU time has been normalized with the time needed to execute the TAU benchmark code, *i.e.*, 8.22 seconds. All the simulations have been performed on a single processor.



(a) (b)

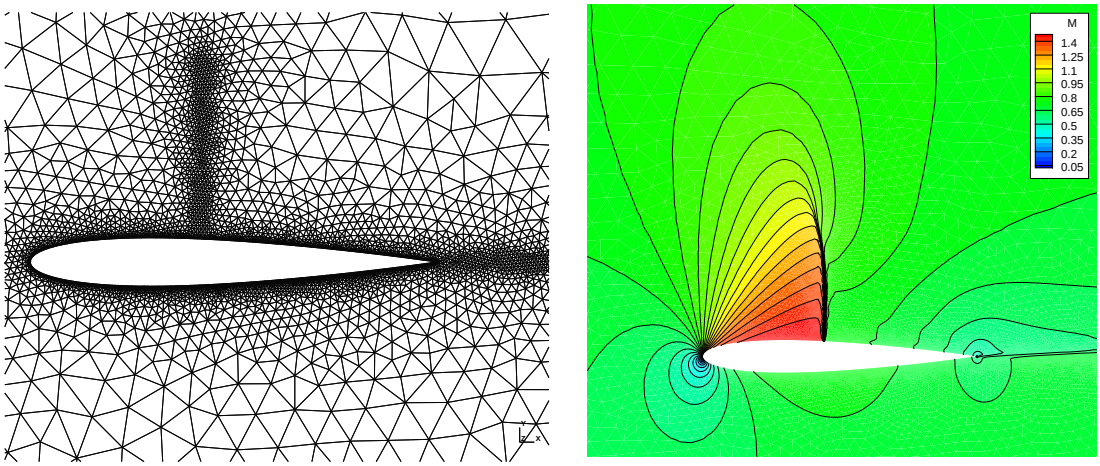
Figure 5.7: L^2 norm of the entropy error for the subsonic flow over a smooth bump as function of the number of degrees of freedom (a) and the normalized CPU time (b).

	N_{dof}	C_L	C_D
2^{nd} order	6391	0.856262	0.0531105
2^{nd} order	24890	0.871845	0.0471140
3^{rd} order	24890	0.878573	0.0455687

Table 5.1: Lift and drag coefficients of the transonic NACA-0012 airfoil as function of the number of degrees of freedom, for the second and third order simulations.

simulations are considered to be at the steady state when the L^2 norm of the density residual is reduced by ten orders of magnitude.

In Figure 5.9 is reported the value of the pressure coefficients over the airfoil for the second and third order simulations and for a second order simulation performed on a finer mesh, obtained by splitting each element of the coarser grid with four elements, such that the number of degrees of freedom is the same of the third order simulation. The numerical method naturally handles smooth and discontinuous solutions within the same framework without the necessity of using tuning parameters. Note that the solution has no oscillations near the discontinuity, furthermore the third order solution is more regular than that obtain with the second order scheme, even with the same number of degrees of freedom. For completeness, in Table 5.1 are reported the values of lift and drag coefficients for the second and third order simulations.



(a) (b)

Figure 5.8: Zoom near the airfoil showing (a) the grid used for the simulations of the transonic flow over the NACA-0012 airfoil, and (b) the Mach number contours for the third order simulation.

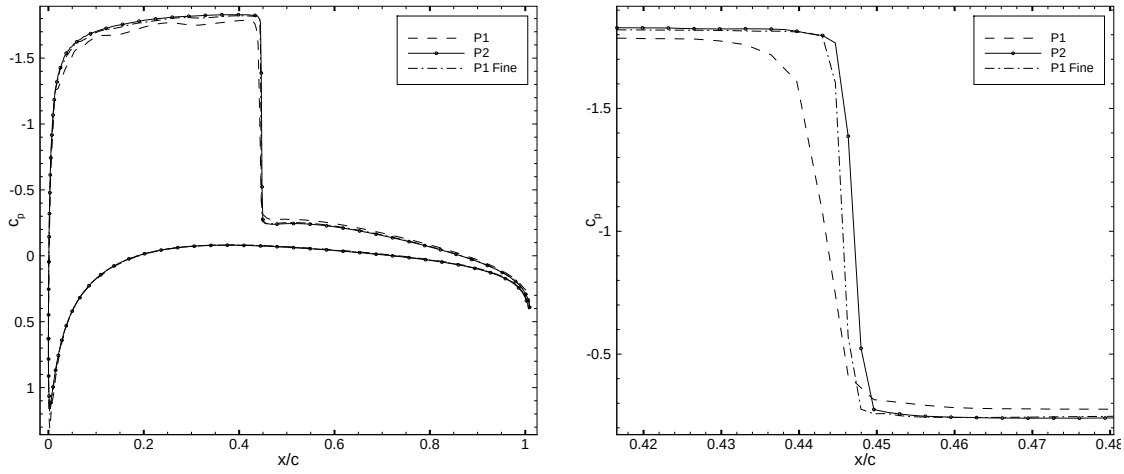


Figure 5.9: Distribution of the pressure coefficient over the airfoil for the simulation of the transonic flow over the NACA-0012 airfoil and zoom near the discontinuity.

5.6 Numerical Results for Viscous Laminar Problems

In this section the numerical discretization of laminar flows at the second and third order of accuracy is reported; two and three dimensional test cases are considered to show the flexibility of the solver. The main objective is to show that a third order accurate scheme can be obtained if a proper gradient reconstruction technique is adopted and the third order discretization gives better results that the second order one for the same number of degrees of freedom.

The method of the manufactured solution is first used to study the accuracy

of the linear and non-linear schemes, then simple test cases, as the laminar flow over a flat plate and a NACA-0012 airfoil, are considered to study the effect of the boundary conditions and of the gradient reconstruction on the accuracy of the solution. As a more realist test case, the three dimensional flow over a delta wing is considered and finally the interaction of a shock wave with a laminar boundary layer is simulated.

5.6.1 Manufactured Solutions

The method of the manufactured solutions gives a flexible way to create analytical solutions for problems of interest, in this way it is possible to verify rigorously the accuracy of the numerical scheme. With this approach, the form of the solution is chosen a priori, and the differential operator of the governing equation is applied to the chosen solution. The reminder of this operation is then used as source term for the original governing equation.

Although the form of the solution is arbitrary, it should be smooth and infinitely differentiable. Here, the form of the primitive variables is taken as a function of sines and cosines [106]

$$\begin{aligned}
\rho(x,y) &= \rho_0 + \rho_x \sin\left(\frac{a_x \pi x}{L}\right) + \rho_y \cos\left(\frac{a_y \pi y}{L}\right) + \rho_{xy} \cos\left(\frac{a_{xy} \pi xy}{L^2}\right) \\
u(x,y) &= u_0 + u_x \sin\left(\frac{a_x \pi x}{L}\right) + u_y \cos\left(\frac{a_y \pi y}{L}\right) + u_{xy} \cos\left(\frac{a_{xy} \pi xy}{L^2}\right) \\
v(x,y) &= v_0 + v_x \cos\left(\frac{a_x \pi x}{L}\right) + v_y \sin\left(\frac{a_y \pi y}{L}\right) + v_{xy} \cos\left(\frac{a_{xy} \pi xy}{L^2}\right) \\
P(x,y) &= P_0 + P_x \cos\left(\frac{a_x \pi x}{L}\right) + P_y \sin\left(\frac{a_y \pi y}{L}\right) + P_{xy} \sin\left(\frac{a_{xy} \pi xy}{L^2}\right)
\end{aligned} \tag{5.90}$$

for the density, the velocity components and the pressure, and where L is the length of the edge of the domain, assumed to be a square. In Table 5.2 are reported the constants used in the previous expressions. The conservative variables are first

Variable ϕ	ϕ_0	ϕ_x	ϕ_y	ϕ_{xy}	a_{ϕ_x}	a_{ϕ_y}	$a_{\phi_{xy}}$
$\rho\,[Kg/m^3]$	1	0.1	-0.1	0.08	0.75	1.0	1.25
$u\,[m/s]$	70	4	-12	7	5/3	1.5	0.6
$v\,[m/s]$	90	-20	4	-11	51.5	1.0	0.9
$P\,[N/m^2]$	1×10^5	-0.3×10^5	0.2×10^5	-0.25×10^5	51.5	1.25	0.75

Table 5.2: Values of the constants used in the manufactured solutions method.

computed from the previous definition of the primitive variables and then are made dimensionless using as reference quantities the values at the point of coordinates

$(0, 0)$, while the spatial coordinates are made dimensionless with L. In figure 5.10 are reported the contours of the conservative variables used as manufactured solutions.

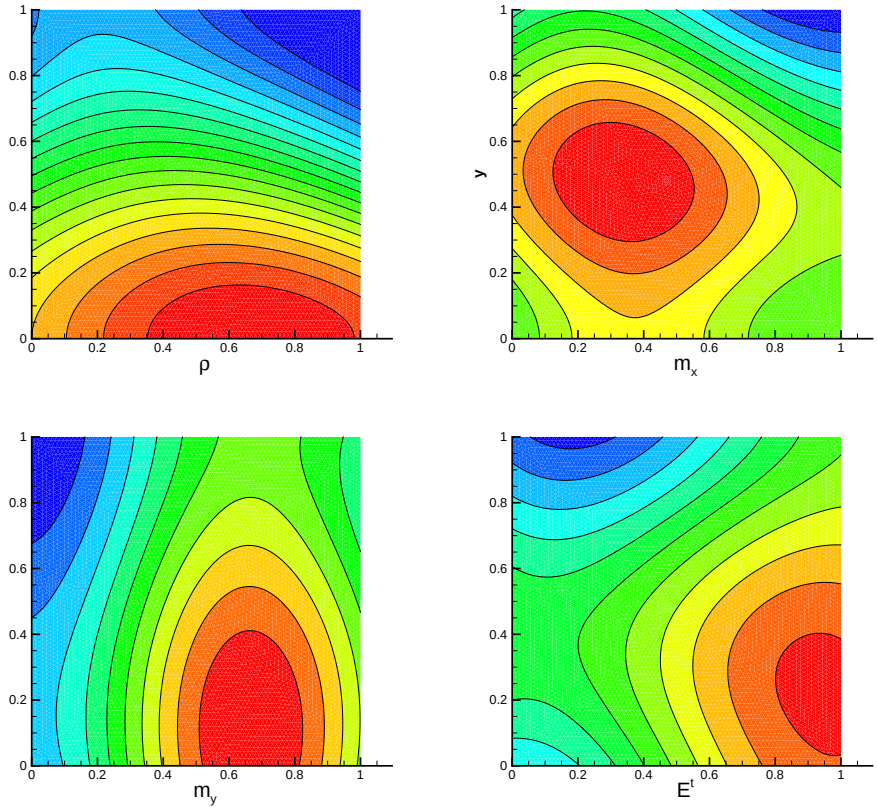


Figure 5.10: Form of the conservative variables used for the manufactured solution method

Dirichlet boundary conditions are applied on the boundaries of the domain, and starting from an uniform initialization, the solution is made evolve until the L^2 norm of the residual of the conservative variables is dropped by ten orders of magnitude. The discretization error is taken as the normalized L^2 norm of the difference between the numerical solution and the exact one, namely

$$\epsilon_{L^2} = \sqrt{\frac{\int_\Omega \|\mathsf{u}_h - \mathsf{u}_{\text{ex}}\|^2 \, \mathrm{d}\Omega}{\int_\Omega \|\mathsf{u}_{\text{ex}}\|^2 \, \mathrm{d}\Omega}}, \tag{5.91}$$

and the accuracy study is performed on a sequence of four unstructured grids, of the same type shown in Figure 4.6-(a), made of 228, 898, 3 588 and 14 412 triangles. The aim of the present study, is not only to verify the formal order of accuracy of the numerical scheme, but also to investigate if the observed order is kept constant when the relative importance of advection and diffusion changes. To this purpose, the observed order of accuracy is computed for different Reynolds numbers.

Two different type of gradient reconstruction methods are considered, the SPR-ZZ and the Green-Gauss methods, because it has been already observed in the study

performed for a scalar function, that there is only a limited difference between the Green-Gauss method and other more sophisticated approaches.

In Figure 5.11, are reported, for different Reynolds numbers, the observed orders of accuracy, for the numerical solution and the reconstructed gradient, with the linear and non-linear schemes and the two type of gradient reconstruction. The observed order of accuracy is taken as the mean slope, computed with a linear least square interpolation, of the variables $\log(1/\sqrt{N_{\text{dof}}})$ and $\log(\epsilon_{L^2})$. Note that only the results for the x component of the gradients are reported, results for the y component are very similar.

Consider first the linear scheme, at the second order there is almost no difference between the results obtained with the two types of gradient reconstructions in the observed order of accuracy of the solution: both methods guarantee formal second order accuracy for the solution. However only the SPR-ZZ method gives also second order of accuracy for the gradient. With quadratic elements, the situation changes drastically, only the SPR-ZZ method gives third order of accuracy on the whole range of Reynolds numbers, for both solution and gradient. It is interesting to observe that in the diffusion limit, the use of a higher order reconstruction method for the gradients gives an increment of accuracy for the solution with respect to the theoretical third order; eventually the theoretical order is reached in the advection dominated regime. With the Green-Gauss method, the formal accuracy of the solution is lost in the diffusion dominated regime, and is recovered only when advective terms dominate the diffusive ones. It is worth noting also that the Green-Gauss method gives at most second order accuracy for the gradient, to be compared with the third order accuracy guaranteed by the SPR-ZZ method.

The same considerations done above for the linear scheme are also valid for the non-linear scheme, however in this case the loss of accuracy in the diffusion regime is more severe if gradients are not reconstructed properly, furthermore the loss of accuracy affects also the solution computed with linear elements.

5.6.2 Laminar Flow over a Flat Plate

In this test case a laminar flow over a flat plate is simulated, the main objective here is to verify the correct imposition of the wall boundary conditions. The linear scheme with the SPR-ZZ gradient reconstruction technique is considered.

A free stream Mach number M = 0.3 is considered and the Reynolds number based on the free-stream conditions and the flat plate length is Re = 5 000. The length of the plate is L = 1. The range of the computational domain in the x-direction is $[-1, 1]$ with the leading edge of the flat plate at $x = 0$. The size of the computational domain in the y-direction is 1, which is 10 times the boundary layer thickness at $x = 1$. At the inlet, the inflow boundary condition is imposed, at the top and the exit the static pressure is imposed. Along the plane $y = 0$, the symmetry boundary condition is imposed for $-1 \leq x \leq 0$ and the adiabatic wall

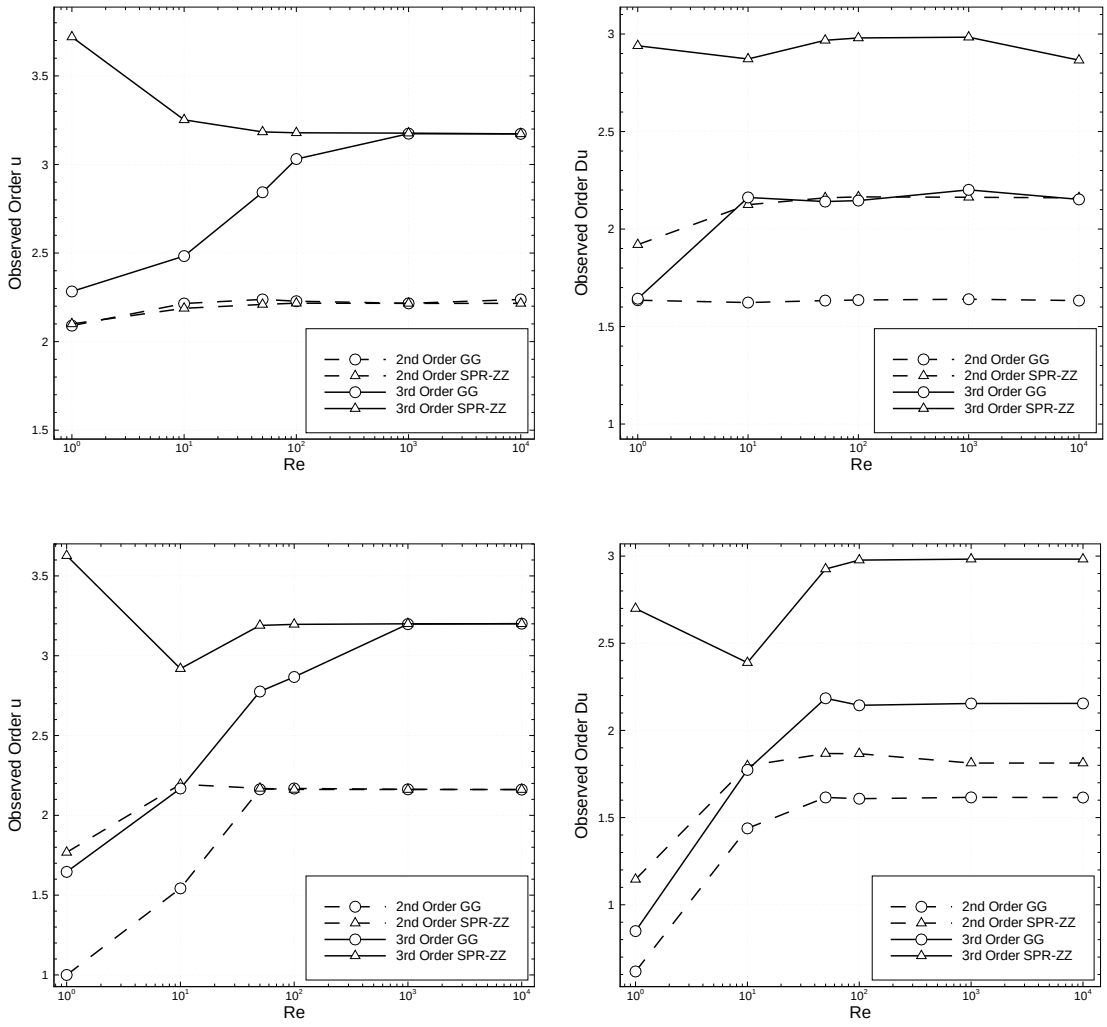


Figure 5.11: Observed orders of accuracy for the solution (left) and the gradient (right) in the manufactured solutions test, at different Reynolds numbers and with two type of gradient reconstruction. On the top row are reported the results for the linear scheme, on the bottom row are reported the results for the non-linear scheme.

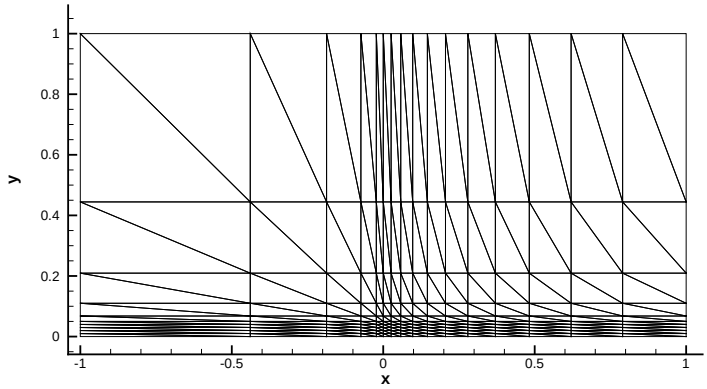
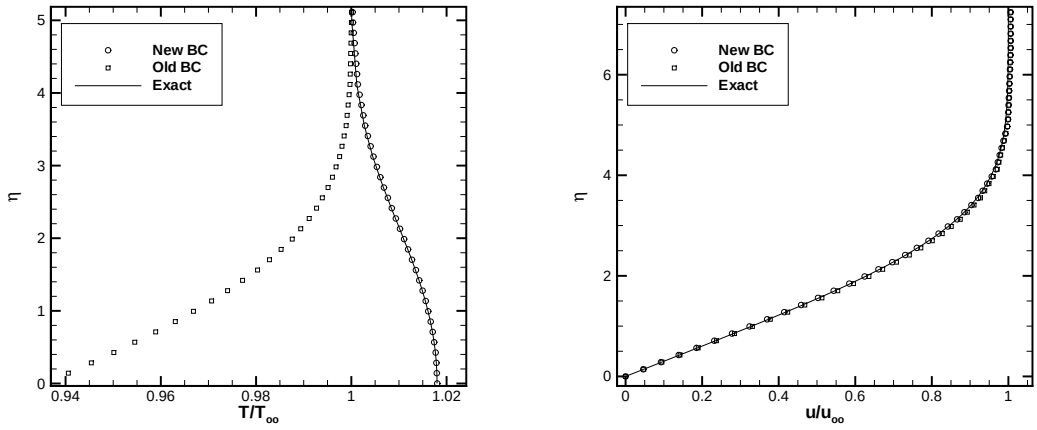


Figure 5.12: The coarsest mesh used for the flat plate boundary layer case.

no-slip boundary condition is imposed for $0 \leq x \leq 1$. The coarsest mesh used is shown in Figure 5.12, it consists of 187 degrees of freedom. The medium and the fine grids are obtained successively splitting each triangle in four triangles.

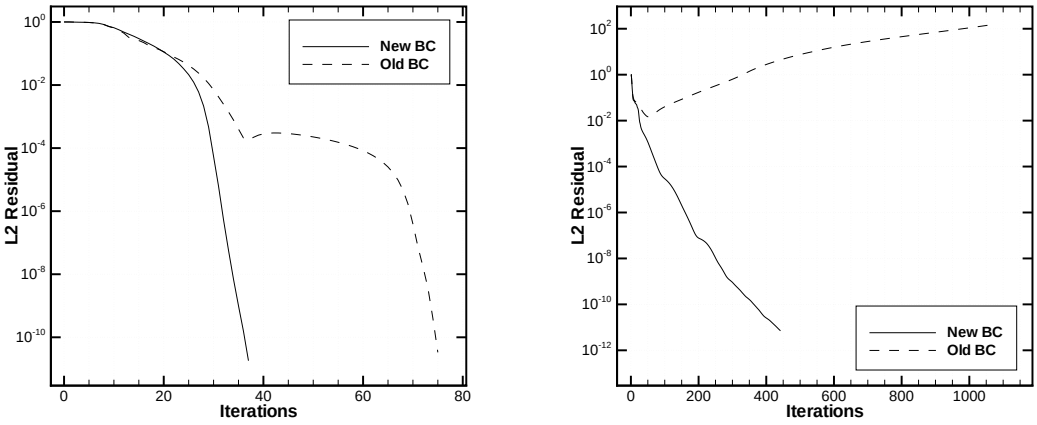
In Figure 5.13 are compared the temperature and the axial velocity profiles at $x = 1$ for the second order scheme on the medium grid, using the approach of imposing the weak boundary condition for the energy equation proposed in this work and when nothing is done. The results are compared against the exact solution of the thermal boundary layer [110]. Clearly the new approach is able to correctly impose the adiabatic condition, while the old approach completely fails to predict the temperature profile and also the normal heat flux is not zero on the wall. The velocity profile is instead in good agreement with the exact solution since the no-slip condition is strongly imposed in both procedures. From the Figure 5.14-(a) can be also deduced that the new way to handle the adiabatic condition assures a fast convergence to the steady state solution. The third order scheme is much more sensible to the imposition of the boundary conditions as can be seen in Figure 5.14-(b), only the new way to impose the boundary condition on the wall makes the scheme converge.

For sake of completeness, in Figure 5.15 is shown the axial velocity profile at $x = 1$ for the second and third order schemes for three levels of grid refinement. The numerical results are in good agreement with the analytical solution even on the coarsest grid. In Figure 5.16 are reported the skin friction coefficients along the flat plate for the second and third order schemes, note that for the same number of degrees of freedom, third order solutions are closer to the exact values with respect to the second order solutions.



(a) (b)

Figure 5.13: Temperature profile (a) and axial velocity profile (b) at $x = 1$, as function of the dimensionless wall distance for the old and the improved way to impose the adiabatic boundary conditions. The numerical solutions (second order scheme) are compared against the theoretical solution.

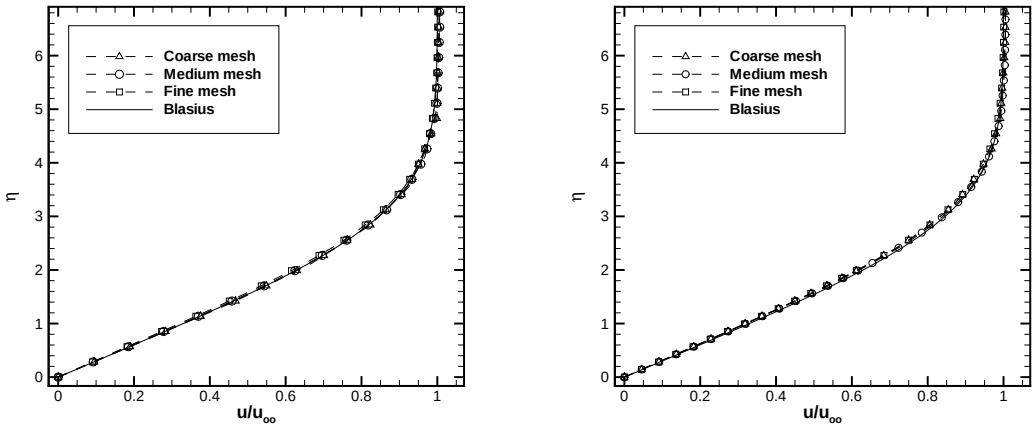


(a) (b)

Figure 5.14: L^2 iterative residual of the axial momentum equation for the old and the improved way to impose the adiabatic boundary conditions for the second (a) and third (b) order scheme.

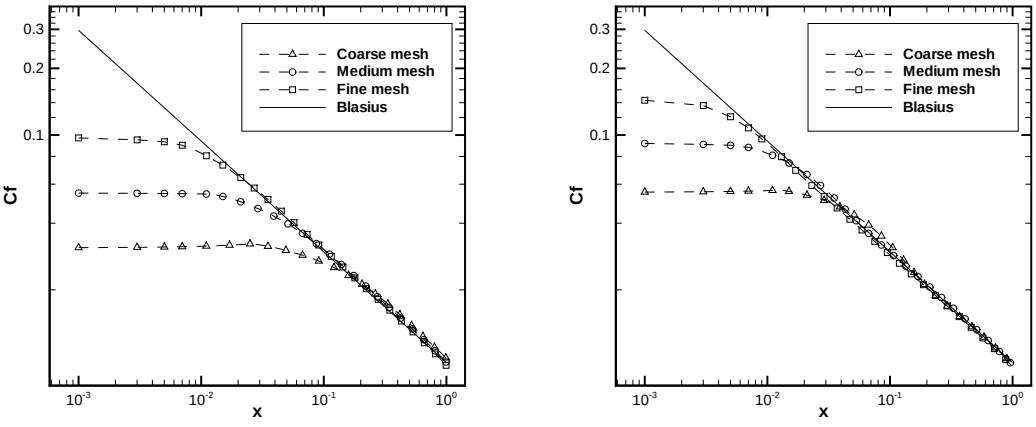
5.6.3 Laminar Flow over a NACA-0012 Airfoil

In this test, a subsonic viscous flow over a NACA-0012 airfoil at zero angle of attack is considered, the free stream mach number is M = 0.5 and Reynolds number is Re = 5 000, based on the chord and the free-stream conditions. This is a widely used test case for two dimensional laminar flows; a distinctive feature of this test case



(a) (b)

Figure 5.15: Axial velocity profile for the second (a) and third (b) order schemes at $x = 1$ as function of the dimensionless wall distance on three different grids. The numerical solutions are compared against the theoretical solution.



(a) (b)

Figure 5.16: Skin friction coefficient along the flat plate fort he second (a) and third (b) order schemes on three different grids. The numerical solutions are compared against the theoretical solution and semi-log scale has been used.

is a steady separation bubble near the trailing edge of the airfoil. An example of computational grid is displayed in Figure 5.17, and three levels of grid refinement are considered in the simulations. The grid extends about 50 chords away from the airfoil. The airfoil boundary is considered a no-slip adiabatic wall and is represented by piecewise linear and quadratic elements, for second and third order simulations, respectively. The far-field boundary condition is applied on the outer boundary of

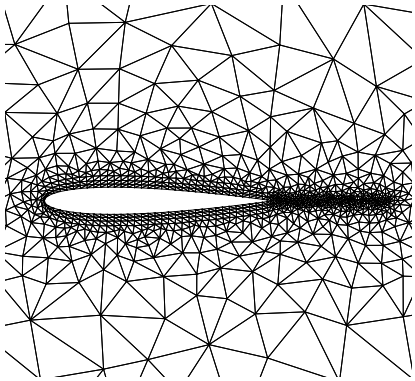


Figure 5.17: An example of computational grid used for the NACA-0012 test case.

the domain. In all the simulation the steady state is considered to be reached when the L^2 norm of the density residual is drop by ten orders of magnitudes.

In Figure 5.18 are depicted the solutions computed with the linear scheme and the SPR-ZZ gradient reconstruction, for P1 and P2 elements. The solution with the P1 elements has been computed on a grid obtained from that with P2 elements (4 216 elements) and splitting each P2 triangle with four P1 triangles, in such a way the number of the degrees of freedom for the second and third order simulations is exactly the same. Note, in Figure 5.18, that although there is not much difference in the Mach number contours between the second and the third order simulations, the streamlines near the trailing edge are very different, and only the third order scheme is able to reproduce the symmetric recirculation bubble. For the same simulations, in Figure 5.19 and Figure 5.20 are reported the pressure and skin friction coefficients profiles, respectively. Note the more regularity of the solution of the third order simulation with respect to the second order one, for the same number of degrees of freedom.

In order to compare the effect of the gradient reconstruction on the quality of the solution, in Figure 5.21 are reported the absolute values of the lift coefficient as function of the number of the degrees of freedom, for linear and quadratic elements, with the SPR-ZZ and the Green-Gauss gradient reconstruction. Since the airfoil is symmetric and the angle of attack is zero, the theoretical value of the lift coefficient is zero; note that the mesh has no particular symmetries which could cancel out numerical errors. It is evident that, for the same number of degrees of freedom, the RD scheme with quadratic elements gives a better approximation of the problem, in particular the use the SPR-ZZ guarantees a great improvement in the numerical discretization. If a proper gradient reconstruction is not used, the gain in using a higher order RD scheme remains marginal with respect to a formal second approximation of the solution. For sake of completeness, in Table 5.3 are also reported the values of pressure and viscous contributions to the drag coefficient.

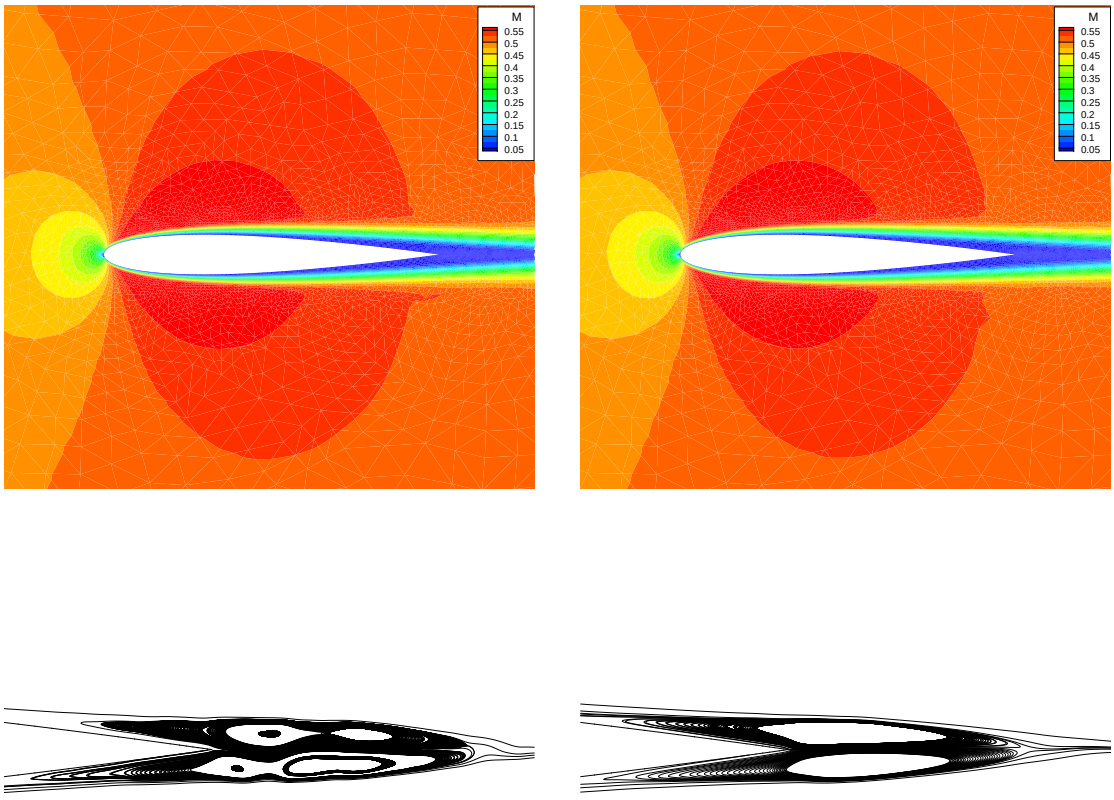


Figure 5.18: Mach number contours (top) and streamlines near the trailing edge (bottom) for the second (left) and third (right) order linear scheme.

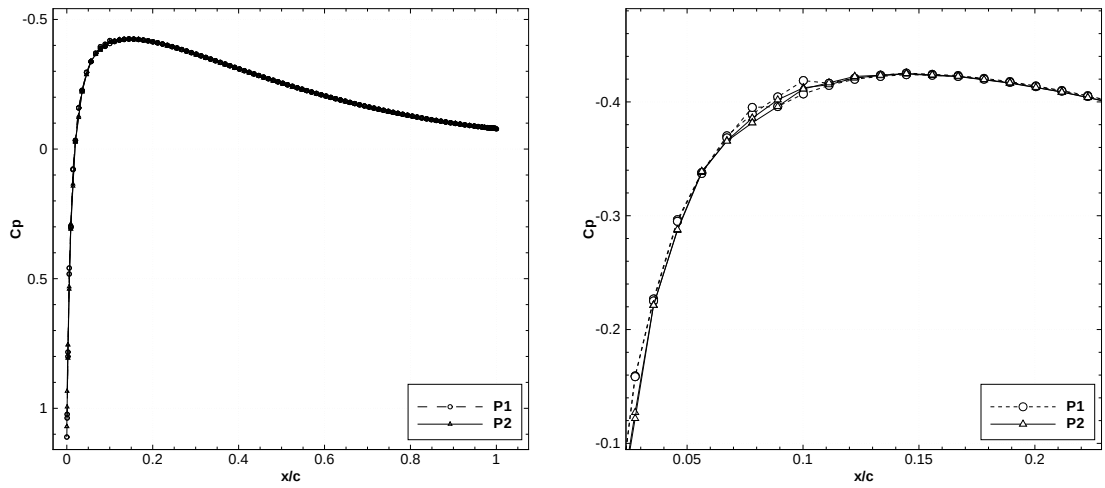


Figure 5.19: Pressure coefficient along the NACA-0012 airfoil for the second and third order simulations, with the same number of degrees of freedom.

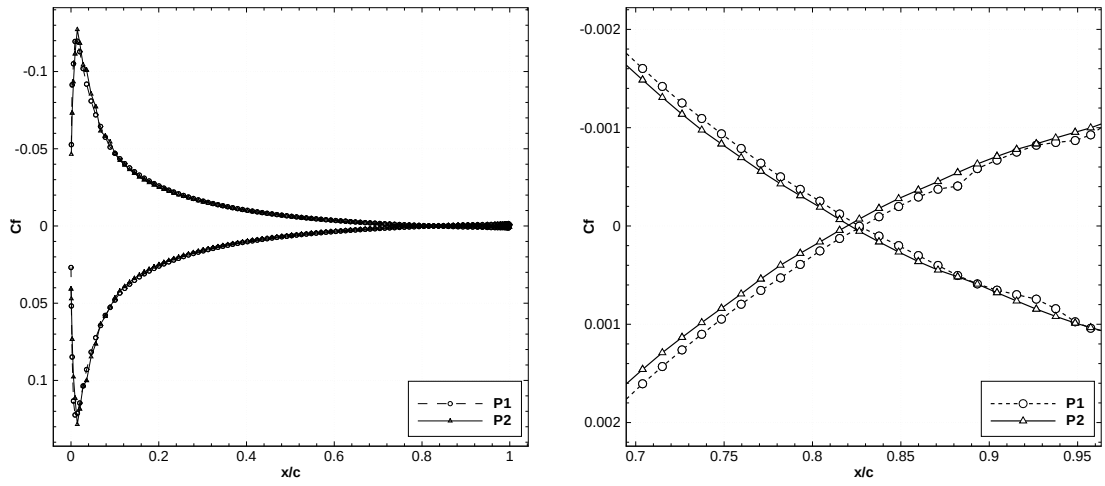


Figure 5.20: Skin friction coefficient along the NACA-0012 airfoil for the second and third order simulations, with the same number of degrees of freedom.

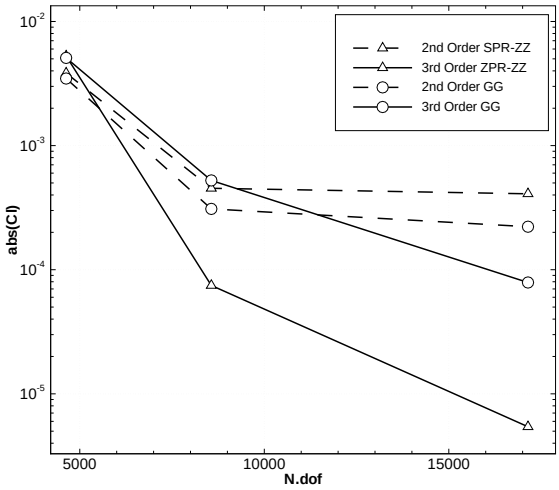


Figure 5.21: Absolute values of the computed lift coefficients on different grids, with linear and quadratic elements, and with the SPR-ZZ and the Green-Gauss gradient reconstruction methods.

	N_{dof}	C_{D_p}	C_{D_v}
SPR-ZZ P1	4630	0.0220	0.0345
	8564	0.0222	0.0327
	17146	0.0224	0.0325
SPR-ZZ P2	4630	0.0247	0.0369
	8564	0.0228	0.0324
	17146	0.0228	0.0324
GG P1	4630	0.0225	0.0313
	8564	0.0226	0.0311
	17146	0.0224	0.0310
GG P2	4630	0.0251	0.0366
	8564	0.0230	0.0326
	17146	0.0229	0.0320

Table 5.3: Pressure and viscous contributions to the drag coefficient of the NACA-0012 airfoil, for linear and quadratic elements with the SPR-ZZ and the Green-Gauss gradient reconstruction.

5.6.4 Laminar Flow around a Delta Wing

The laminar flow around a delta wing with sharp edges, at high angle of attack is now considered. As the flow passes the leading edge, it rolls up and creates a big vortex structure which is convected far behind the wing, at the same time, near the leading edge a smaller secondary vortex appears. A free stream Mach number $\mathrm{M} = 0.5$ is considered, the Reynolds number, based on the root chord of the wing

and the free-stream conditions, is $\mathrm{Re} = 4\,000$, the angle of attack is $\alpha = 12.5°$.

The geometry of the delta wing is depicted in Figure 5.22, together with an example of a coarse grid of tetrahedra used for the simulations. Note the presence of very stretched elements on the wing. Finer levels of grids are obtained by uniformly splitting each tetrahedron of the coarser level with eight tetrahedra. Three levels of refinement are considered and simulations have been run in parallel on 8, 16 and 32 processors, for the coarse, medium and fine grids, respectively. The wing surface is treated as no-slip adiabatic wall, the vertical plane intersecting the root of wing is treated as a symmetry plane, while far field boundary conditions are applied on the outer boundaries of the domain.

The solution is initialized with an uniform flow, the lower order solution is used

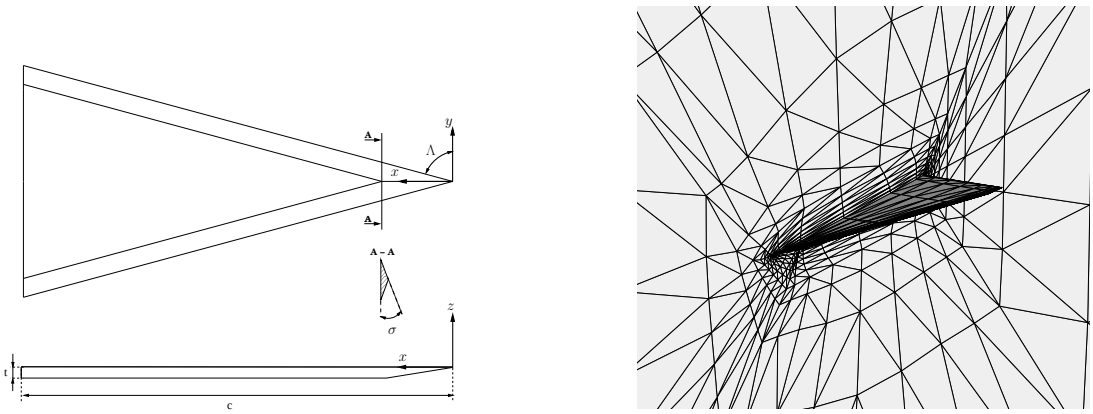


Figure 5.22: Left: Bottom and side views of the model of the delta wing: $\Lambda = 75°$, $\sigma = 60°$ and $t/c = 0.024$. Right: a coarse mesh of tetrahedra use for the simulations.

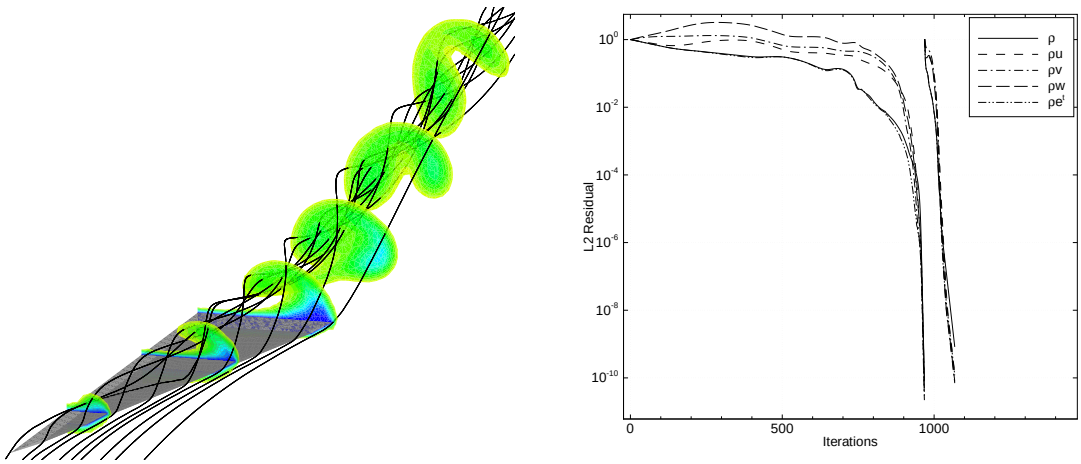


Figure 5.23: Left: Streamlines and slices of Mach number contours along and behind the delta wing, for a third order simulation on a fine grid. Right: Convergence history for the order sequencing (second and third order).

as initial solution for the third order computation. For this test case, the linear scheme is used with the SPR-ZZ gradient reconstruction method; in Figure 5.23 are reported the streamlines and Mach number contours, at different stations, of the third order solution on the finest grid. The convergence history is also reported for the second and third order simulations. Note the quadratic convergence in the final stage of the non-linear solver and also the fact that the thanks to the order sequencing strategy, very few iterations are required by the third order method to converge to the steady state.

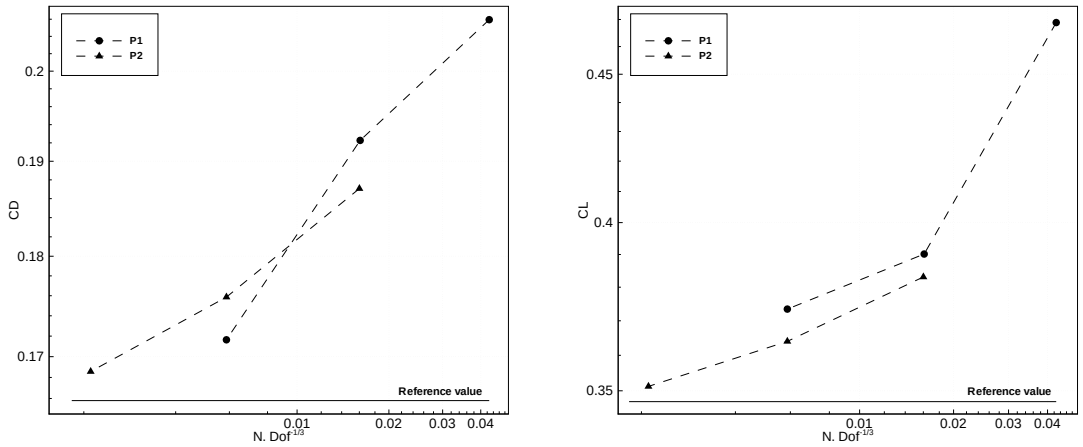
In Figure 5.24 are reported the drag and lift coefficients computed with linear and quadratic elements, on three uniformly refined grids. For comparison, are reported also the reference values computed in [73] by extrapolating the results obtained with a higher order DG method. Observing the convergence of the drag coefficient as function of the number of degrees of freedom, it can be noted that there is no significant gain in using a higher order approximation, with respect to the second order. This behavior can be caused by the singularity at the leading edge of the wing, which might mask the benefits of a higher order approximation with an uniform mesh refinement. Regarding the convergence of the lift coefficient, it could be observed a clear benefit of using a higher order approximation, because the big vortex structure over the wing, which generates the great part of the lift force, is better captured with higher order elements. For completeness, in Figure 5.25 are reported the errors of the lift and drag coefficients as function of the number of the degrees of freedom; the error is computed as the absolute value of the difference between the numerical and the reference values.

5.6.5 Shock-Wave/Laminar Boundary Layer Interaction

As last test case, the interaction of an oblique shock wave with a laminar boundary layer is considered. The aim of this test is to verify the non-oscillatory properties of the non-linear scheme in presence of discontinuities of the solution and at the same time, the capability to maintain the accuracy required for the discretization of the boundary layer.

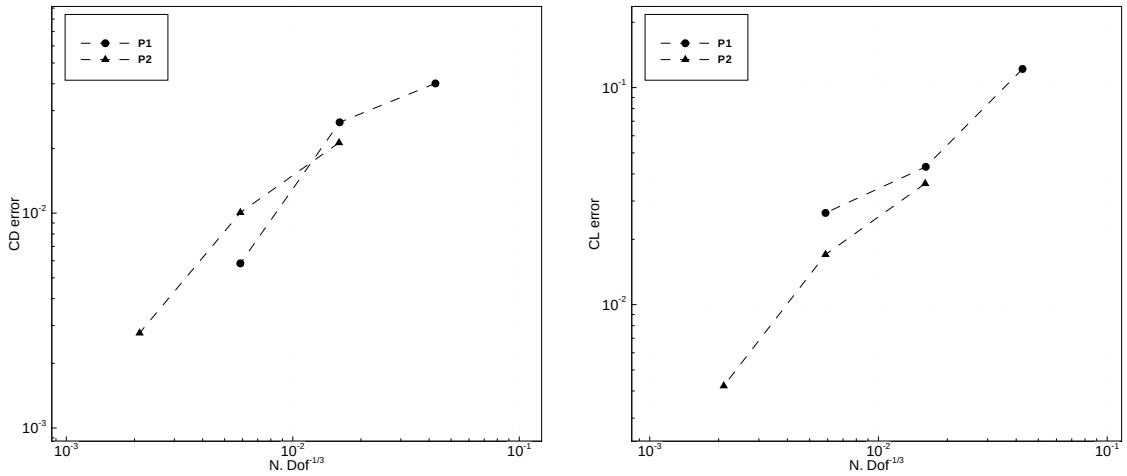
The test consists in a laminar boundary layer developing over a flat plate and an incident shock impinging the boundary layer. Since the flow is supersonic, a shock appears at the leading edge of the flat plate, that interacts with the oblique shock. Furthermore, at the impinging point, the incident shock produces a separation of the boundary layer, the shock is then reflected and an expansion fan appears, turning the flow toward the wall and causing a reattachment of the boundary layer, as it is depicted in Figure 5.26.

In the numerical simulations, the oblique shock is generated by imposing the incoming supersonic flow state on the lower part of left boundary, while another supersonic state is imposed on the upper part of the left boundary and on the top boundary; this state is computed using the relations of the oblique shocks,



(a) (b)

Figure 5.24: Drag (a) and lift (b) coefficients as function of the degrees of freedom for the delta wing simulation, with linear and quadratic elements.



(a) (b)

Figure 5.25: Errors, with respect to the reference values, of the drag (a) and lift (b) coefficients as function of the degrees of freedom, for the delta wing simulation, with linear and quadratic elements.

such that the incident shock has a certain angle of incidence θ_s. The height of the computational domain is 0.94, while the range of the computational domain in the x direction is $[-0.2, 2]$, the flat plate has length $L = 2$, with the leading edge at $x = 0$. Along the plate, the no-slip adiabatic wall boundary condition is applied, while the symmetry boundary condition is applied on the remaining part of the bottom boundary. On the right boundary, the outflow boundary condition is applied, see Figure 5.26. The inflow states are chosen such that the free-stream Mach number is $M = 2.15$ and the angle of the incident shock is $\theta_s = 30.8°$, in this case the impingement point would be at center of the plate for an inviscid fluid. The Reynolds number based on the free-stream values and the distance between the plate leading edge and the inviscid shock impingement point is $\mathrm{Re} = 1 \times 10^5$.

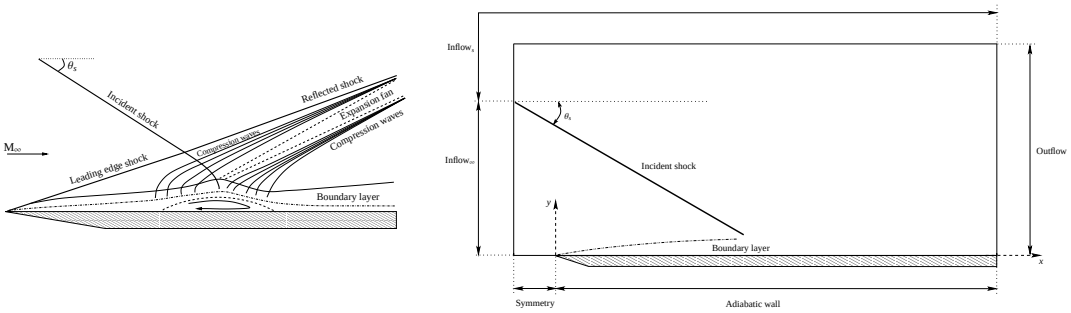
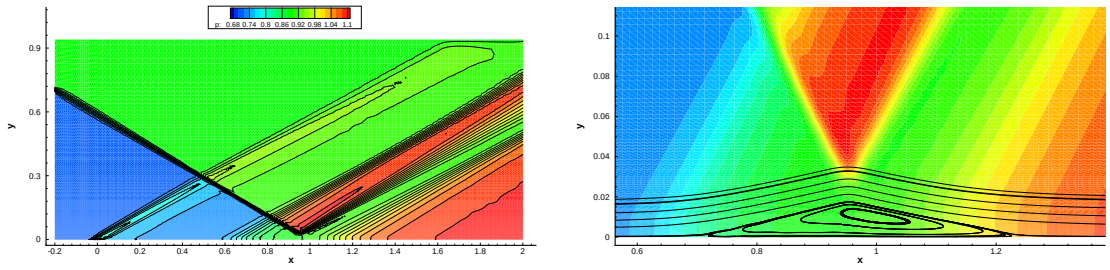


Figure 5.26: Schematic representation of the waves pattern (left) and of the computational domain with boundary conditions (right) for the shock-wave/boundary layer interaction problem

The non-linear scheme with the SPR-ZZ gradient recovery strategy is used to perform the numerical simulations at second and third order of accuracy. The computational domain is generated from the triangulation of a 90×85 structured grid; the first number refers to the number of elements on the horizontal boundaries, with



(a) (b)

Figure 5.27: Left: contours of the pressure obtained with the third order scheme for the shock/boundary layer interaction. Right: zoom of the solution near the impinging point of the shock with the boundary layer, streamlines are also reported to show the separation bubble.

80 elements along the plates, the second number refers to the number of elements on the vertical boundaries. The element distribution is uniform on the x direction, while along the y direction a non-uniform distribution of the elements is used, with a mesh spacing $\Delta y = 0.5 \times 10^{-3}$ near the bottom boundary. For comparison, a second order simulation is also performed on a finer grid with the same number of degrees of freedom of the third order simulation on the coarse grid. The simulation is initialized with an uniform solution and the second order solution is used as initial solution for the third order approximation. Except the case of the second order simulation on the coarse grid, for which the initial residual is reduces by ten orders of magnitude, the residual for the third order simulation and the second order one on the finer grid could not be reduced by more than eight orders of magnitude.

In Figure 5.27-(a) are shown the contours of the pressure for the third order simulation; all the features of this problem are well represented. In Figure 5.27-(b) is reported a zoom of the solution where the incident shock impinges the boundary layer. Two features are evident: the reflection of the incident shock and the recirculation bubble as a consequence of the separation and subsequent reattachment of the boundary layer produced by the incident shock and the expansion fan, respectively.

The profiles of density, pressure and Mach number along the lines at $y = 0.29$ and $y = 0.15$ are reported in Figure 5.28. Note that the third order scheme gives a very sharp and monotone representation of the discontinuities and also smooth portions of the solution are better represented compared to the second order solution. It is important to remember that smooth and discontinuous solutions are treated within the same non-linear scheme, without any special treatment or tuning parameter. For a fair comparison, it is also reported the solution obtained with the second order scheme on a finer mesh; it is worth noticing that, although a mesh refinement produces an improvement of the numerical solution, the level of accuracy obtained with the second order scheme is still lower than that obtained with the third order scheme, for the same number of degrees of freedom.

Finally, in Figure 5.29 are reported the values of the pressure and of the friction coefficient along the plate. The oscillations near the point $x = 0$ are due to the singularity of the solution at the leading edge of the flat plate, but they are limited only in small region around the leading edge. The third order scheme seems less sensitive to this singularity compared to the second order simulations. The separation bubble can easily detected by the negative values of friction coefficients, note also the pressure plateau in the detached zone.

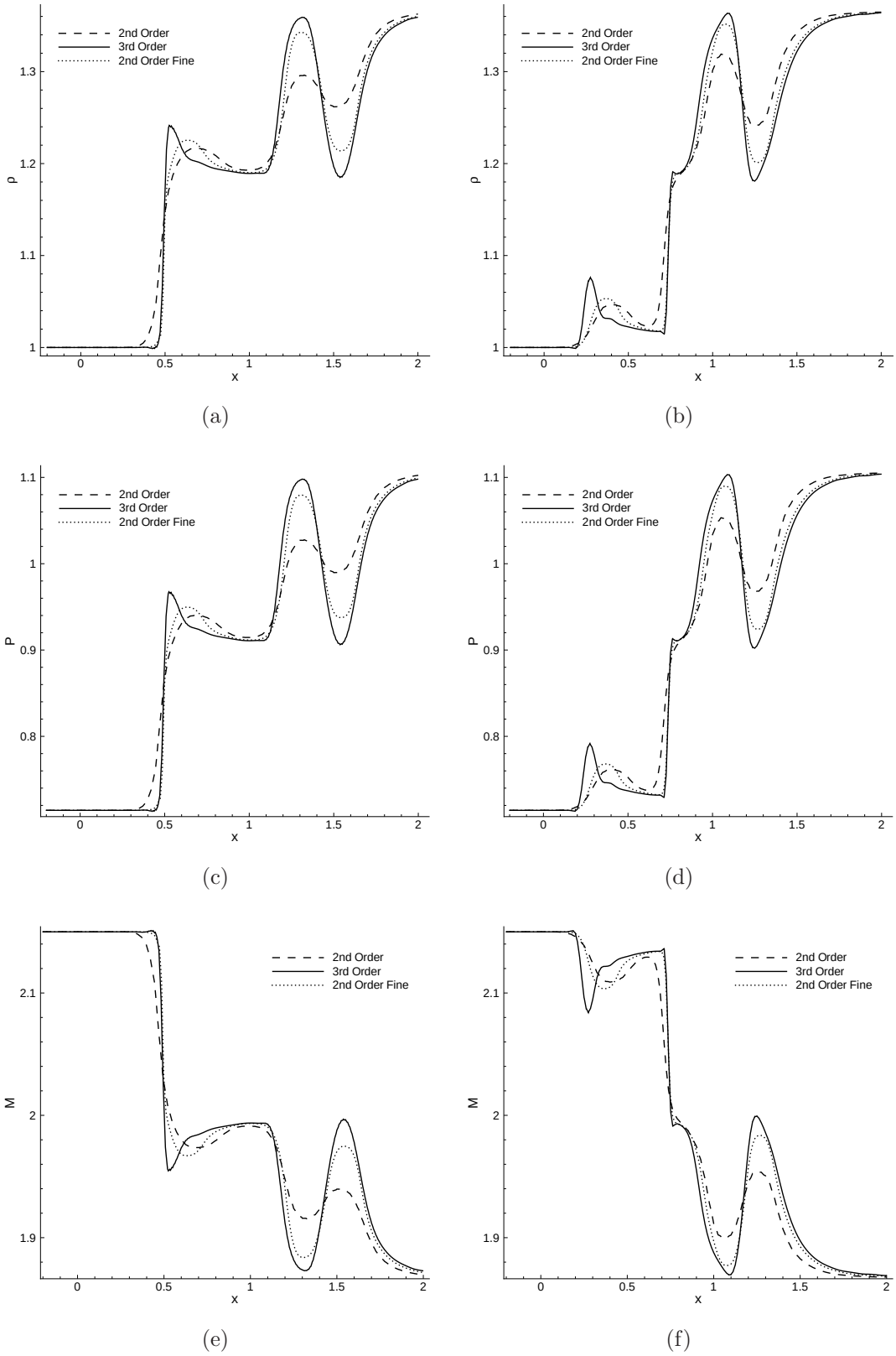
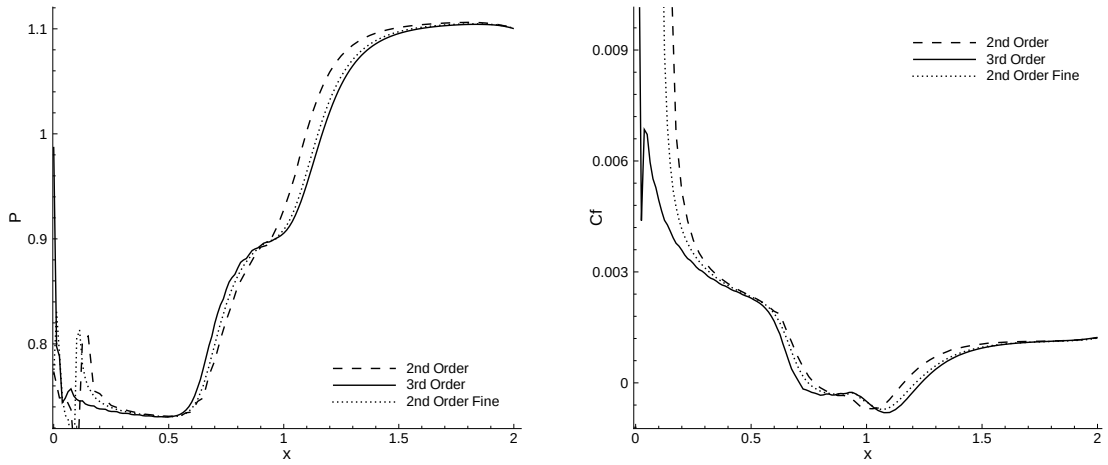


Figure 5.28: Density, pressure and Mach number profiles along the line $y = 0.29$ (a, c, e) and the line $y = 0.15$ (b, d, f) for the shock/boundary layer interaction problem.



(a) (b)

Figure 5.29: Pressure (a) and skin friction (b) profiles along the flat plate for the shock/boundary layer interaction problem.

Chapter 6

Simulations of Turbulent Flows

In this section a robust high-order RD solver for the RANS equations is developed. The Spalart-Allmaras turbulence model fully-coupled with the mean flow equations is adopted. The fully-coupled approach allows a straightforward extension of the RD method from laminar to turbulent problems, furthermore, the mean flow and turbulent equations are discretized with the same order of accuracy.

To make the method robust for practical applications, the original form of the Spalart-Allmaras equation has been modified and a non-linear LU-SGS method is used to make the scheme converge to the steady state. The numerical results show that the resulting solver is robust enough for the high-order simulation of transonic turbulent flows. Furthermore, despite the poor regularity of the turbulent flow field, there is still benefit to employing high-order RD methods for the mean flow and the turbulent equation.

6.1 Turbulence Modeling: the RANS Approach

One of the most popular method for dealing with turbulent flows in aeronautical applications is the Reynolds Averaged Navier-Stokes (RANS) approach, which provides information about the mean flow properties [137]. The main idea of the RANS is to express a generic variable $f(x,t)$ as the sum of a mean value and a fluctuating component, namely

$$f = \langle f \rangle + f', \tag{6.1}$$

where $\langle f \rangle$ is the mean value and f' is the fluctuation. For stationary turbulence, the time averaging operator is defined as

$$\langle f \rangle(x) = \lim_{\tau \to \infty} \frac{1}{\tau} \int_t^{t+\tau} f(x,t)\, \mathrm{d}t. \tag{6.2}$$

By applying the decomposition (6.1) to each variable of the Navier-Stokes equations and taking the time average of the equation, the governing equations for the mean

flow variables are obtained.

In the case of compressible flows the averaging of the Navier-Stokes equations would produce too many terms due to the correlation between the fluctuations, for this reason the density weighted average is preferred

$$\overline{f} = \frac{\langle \rho f \rangle}{\langle \rho \rangle}, \tag{6.3}$$

which yields to the definition of a new fluctuation f'', such that

$$f = \overline{f} + f'', \tag{6.4}$$

with $\langle \rho f'' \rangle = 0$. It is easy to see that

$$\langle \rho f \rangle = \langle \rho \overline{f} \rangle = \langle \rho \rangle \overline{f}. \tag{6.5}$$

6.1.1 Derivation of the Averaged Equations

The compressible RANS equations are obtained by first substituting the decomposition (6.4) for each variable of the Navier-Stokes equation and then one averages the entire equations in time.

Continuity equation

The averaged continuity equations takes the following form

$$\frac{\partial \langle \rho \rangle}{\partial t} + \boldsymbol{\nabla} \cdot \left(\langle \rho \rangle \overline{\boldsymbol{v}} \right) = 0, \tag{6.6}$$

which resembles to its non-averaged counterpart.

Momentum equation

The averaged momentum equations writes

$$\frac{\partial \left(\langle \rho \rangle \overline{\boldsymbol{v}} \right)}{\partial t} + \boldsymbol{\nabla} \cdot \left(\langle \rho \rangle \overline{\boldsymbol{v}} \otimes \overline{\boldsymbol{v}} + \langle P \rangle \mathbb{I} \right) = \boldsymbol{\nabla} \cdot \left(\langle \mathbb{S} \rangle - \langle \rho \boldsymbol{v}'' \otimes \boldsymbol{v}'' \rangle \right), \tag{6.7}$$

where the extra term $\mathbb{S}_t = -\langle \rho \boldsymbol{v}'' \otimes \boldsymbol{v}'' \rangle$ is know as the Reynolds stress tensor. Moreover, the averaged pressure for a perfect gas is defined as

$$\langle P \rangle = \langle \rho \mathcal{R} T \rangle = \langle \rho \rangle \mathcal{R} \overline{T}, \tag{6.8}$$

and the averaged stress tensor is defined by

$$\langle \mathbb{S} \rangle = -\frac{2}{3} \mu (\boldsymbol{\nabla} \cdot \overline{\boldsymbol{v}}) \mathbb{I} + \mu (\boldsymbol{\nabla}^{\mathrm{T}} \overline{\boldsymbol{v}} + \boldsymbol{\nabla} \overline{\boldsymbol{v}}), \tag{6.9}$$

with the mean dynamic viscosity coefficient expressed as $\mu = \mu(\overline{T})$.

Energy equation

Averaging the energy equation yields to the following terms

$$\begin{aligned}
\langle \rho e^t \rangle &= \langle \rho \rangle \left(\overline{e} + \frac{1}{2} \overline{\|\boldsymbol{v}''\|^2} \right), \\
\langle (\rho e + P)\boldsymbol{v} \rangle &= \langle \rho h \boldsymbol{v} \rangle = \langle \rho \rangle \overline{h}\, \overline{\boldsymbol{v}} + \langle \rho h'' \boldsymbol{v}'' \rangle, \\
\left\langle \rho \frac{1}{2} \|\boldsymbol{v}\|^2 \boldsymbol{v} \right\rangle &= \frac{1}{2} \langle \rho \rangle \, \|\overline{\boldsymbol{v}}\|^2 \, \overline{\boldsymbol{v}} + \frac{1}{2} \overline{\|\boldsymbol{v}''\|^2} \, \overline{\boldsymbol{v}} - \mathbb{S}_t \overline{\boldsymbol{v}} + \frac{1}{2} \overline{\|\boldsymbol{v}''\|^2 \boldsymbol{v}''}, \\
\langle \mathbb{S} \rangle \cdot \boldsymbol{v} &= \langle \mathbb{S} \rangle \cdot \overline{\boldsymbol{v}} + \langle \mathbb{S} \cdot \boldsymbol{v}'' \rangle.
\end{aligned} \tag{6.10}$$

The following definitions are introduced for convenience

$$k = \frac{1}{2} \langle \rho \rangle \overline{\|\boldsymbol{v}''\|^2}, \quad \langle \boldsymbol{q} \rangle = - \langle \kappa \boldsymbol{\nabla} T \rangle, \quad \text{and} \quad \boldsymbol{q}_t = - \langle \rho h'' \boldsymbol{v}'' \rangle, \tag{6.11}$$

which define the turbulent kinetic energy, the averaged heat diffusion flux and the turbulent enthalpy diffusion flux, respectively. By neglecting the terms $\overline{\|\boldsymbol{v}''\|^2 \boldsymbol{v}''}$ and $\langle \mathbb{S} \cdot \boldsymbol{v}'' \rangle$, the averaged energy equation can be written as

$$\frac{\partial}{\partial t} \left(\langle \rho \rangle (e^t + k) \right) + \boldsymbol{\nabla} \cdot \left(\langle \rho \rangle \left(\overline{h} + \frac{1}{2} \|\overline{\boldsymbol{v}}\|^2 + k \right) \right) = \boldsymbol{\nabla} \cdot \left((\mathbb{S} + \mathbb{S}_t) \cdot \overline{\boldsymbol{v}} - \langle \boldsymbol{q} \rangle - \boldsymbol{q}_t \right). \tag{6.12}$$

The averaging process introduces additional fluxes, but no new equations have been obtained to take into account for these new unknowns. Turbulence models provide closure relations for the average equations by modeling the additional turbulent quantities. The simplest (and the most used) modeling is bases on the Bousssinesq's eddy-viscosity concept, which assumes that, in analogy to the viscous stresses in laminar flows, the turbulent stresses are proportional to the mean velocity gradient. According to the eddy viscosity model, the Reynolds stress tensor is written as

$$\mathbb{S}_t = - \langle \rho \boldsymbol{v}'' \otimes \boldsymbol{v}'' \rangle = - \frac{2}{3} \left(\mu_t \boldsymbol{\nabla} \cdot \overline{\boldsymbol{v}} + \langle \rho \rangle k \right) \mathbb{I} + \mu_t (\boldsymbol{\nabla}^{\mathrm{T}} \overline{\boldsymbol{v}} + \boldsymbol{\nabla} \overline{\boldsymbol{v}}), \tag{6.13}$$

where the term $-\frac{2}{3} \langle \rho \rangle k$ ensures that the trace of the Reynolds stress tensor is $-2 \langle \rho \rangle k$. The part $-\frac{2}{3} \langle \rho \rangle k$ of the normal stresses acts like a pressure force and can be absorbed into the pressure definition by replacing the static pressure with the following modified pressure

$$\hat{P} = P + \frac{2}{3} \langle \rho \rangle k. \tag{6.14}$$

In analogy to the turbulent stresses model, the turbulent heat flux is assumed to be related to the gradient of the mean quantities. The turbulent heat flux can be expressed as

$$\boldsymbol{q}_t = \langle \rho h'' \boldsymbol{v}'' \rangle = c_p \langle \rho \rangle T'' \boldsymbol{v}'' = - c_p \langle \rho \rangle \frac{\mu_t}{Pr_t} \boldsymbol{\nabla} \overline{T}, \tag{6.15}$$

with Pr_t the turbulent Prandtl number.

In conclusion, leaving out the average symbols, the RANS equations read

$$\begin{cases} \dfrac{\partial \rho}{\partial t} + \boldsymbol{\nabla} \cdot \boldsymbol{m} = 0 \\ \dfrac{\partial \boldsymbol{m}}{\partial t} + \boldsymbol{\nabla} \cdot \left(\boldsymbol{m} \otimes \boldsymbol{m} + \hat{P} \mathbb{I} \right) = \boldsymbol{\nabla} \cdot \mathbb{S}_t \\ \dfrac{\partial \hat{E}^t}{\partial t} + \boldsymbol{\nabla} \cdot \left(\hat{E}^t + \hat{P} \right) \dfrac{\boldsymbol{m}}{\rho} = \boldsymbol{\nabla} \cdot \left(\mathbb{S}_t \cdot \dfrac{\boldsymbol{m}}{\rho} - \boldsymbol{q}_t \right) \end{cases} \tag{6.16}$$

where the definition of the total energy has been modified to include the turbulent kinetic energy

$$\hat{E}^t = E^t + \langle \rho \rangle k. \tag{6.17}$$

Making use of the definitions (6.13) and (6.15), the stress tensor and the heat flux are written as

$$\mathbb{S}_t = -\frac{2}{3}(\mu + \mu_t)(\boldsymbol{\nabla} \cdot \boldsymbol{v})\mathbb{I} + (\mu + \mu_t)(\boldsymbol{\nabla}^{\mathrm{T}} \boldsymbol{v} + \boldsymbol{\nabla} \boldsymbol{v}), \tag{6.18}$$

$$\boldsymbol{q}_t = c_p \left(\frac{\mu}{Pr} + \frac{\mu_t}{Pr_t} \right) \boldsymbol{\nabla} T. \tag{6.19}$$

The system of equations (6.16) can be further simplified since in most cases the turbulent kinetic energy is small compared to the kinetic energy of the mean flow, and thus

$$\hat{E}^t = E^t, \quad \text{and} \quad \hat{P} = P. \tag{6.20}$$

The system to be solved then reads

$$\begin{cases} \dfrac{\partial \rho}{\partial t} + \boldsymbol{\nabla} \cdot \boldsymbol{m} = 0 \\ \dfrac{\partial \boldsymbol{m}}{\partial t} + \boldsymbol{\nabla} \cdot \left(\boldsymbol{m} \otimes \boldsymbol{m} + P \mathbb{I} \right) = \boldsymbol{\nabla} \cdot \mathbb{S}_t \\ \dfrac{\partial E^t}{\partial t} + \boldsymbol{\nabla} \cdot \left(E^t + P \right) \dfrac{\boldsymbol{m}}{\rho} = \boldsymbol{\nabla} \cdot \left(\mathbb{S}_t \cdot \dfrac{\boldsymbol{m}}{\rho} - \boldsymbol{q}_t \right) \end{cases} \tag{6.21}$$

which closely resembles to the Navier-Stokes equations, except for the new definition of the stress tensor and the heat flux.

6.2 The Spalart-Allmaras Turbulence Model

The closure of the RANS equations is obtained by adding one or more turbulence modeling equations. For aerodynamic applications one of the most used turbulence

model is the Spalart–Allmaras model [116], which has showed to be quite accurate for attached and mildly separated flows, keeping reduced the complexity of the turbulence modeling.

The model consists of a single transport equation for the working variable $\mu_t^\star$ and for fully turbulent simulations it reads

$$\frac{\partial \mu_t^\star}{\partial t} + \boldsymbol{\nabla}\cdot\left(\mu_t^\star \frac{\boldsymbol{m}}{\rho} - \frac{\eta}{\sigma_{SA}}\boldsymbol{\nabla}\left(\frac{\mu_t^\star}{\rho}\right)\right) = S_{SA}, \tag{6.22}$$

where $\eta = \mu + \mu_t^\star$ and the source term S_{SA} is given by

$$S_{SA} = c_{b1}\hat{\omega}\mu_t^\star - \rho c_{w1} f_w \left(\frac{\mu_t^\star}{\rho\, d_{\min}}\right)^2 + \frac{1}{\sigma_{SA}}\rho c_{b2}\left\|\boldsymbol{\nabla}\left(\frac{\mu_t^\star}{\rho}\right)\right\|^2, \tag{6.23}$$

the three terms in the source term represent respectively the production, the destruction and the diffusion.

The turbulent eddy viscosity μ_t is calculated from the Spalart–Allmaras working variable as

$$\mu_t = \mu_t^\star f_{v1}, \tag{6.24}$$

where f_{v1} is a dimensionless function, which reads

$$f_{v1} = \frac{\chi^3}{\chi^3 + c_{v1}^3}, \quad \text{with} \quad \chi = \frac{\mu_t^\star}{\mu_t}. \tag{6.25}$$

In the production term, the modified vorticity function is expressed as

$$\hat{\omega} = \|\boldsymbol{\omega}\| + \bar{\omega} \tag{6.26}$$

where $\|\boldsymbol{\omega}\|$ is the magnitude of the vorticity vector, and

$$\bar{\omega} = \frac{\mu_t^\star f_{v2}}{\rho k^2 d_{\min}^2}, \quad \text{with} \quad f_{v2} = 1 - \frac{\chi}{1 + \chi f_{v1}}. \tag{6.27}$$

The remaining closure functions are

$$f_w = g\left(\frac{1 + c_{w3}^6}{g^6 + c_{w3}^6}\right)^{\frac{1}{6}}, \quad g = r + c_{w2}\big(r^6 - r\big), \quad r = \frac{\mu_t^\star}{\rho\hat{\omega}k^2 d_{\min}^2}, \tag{6.28}$$

where $d_{\min}$ is the distance to the nearest wall, $c_{b1} = 0.1355$, $\sigma_{SA} = 2/3$, $c_{b2} = 0.622$, $k = 0.41$, $c_{w1} = \dfrac{c_{b1}}{k^2} + \dfrac{(1 + c_{b2})}{\sigma_{SA}}$, $c_{w2} = 0.3$, $c_{w3} = 2$, $c_{v1} = 7.1$.

It is well know that in order to correctly discretize the turbulent boundary layer, the first grid point off the wall must lie within the viscous sublayer [137]. This requirement can be very strict for FV schemes which require quite regular elements, because the grid resolutions on the wall and along the direction orthogonal to the wall should be the same. This requirement might be circumvented with the use of wall functions, which model the effect of the boundary layer instead of completely

solving it, thus the grid resolution near the wall can be reduced. In this work no wall functions are used, thus the boundary layer is completely resolved. The grid resolution requirement is reduced with the use of high stretched element on the wall region, in this way the grid resolution along the normal direction to the wall is enough to discretized the boundary layer with good accuracy, but at the same time the number of the elements remains limited. Furthermore, the use of the higher order discretization of the solution reduces the mesh resolution requirement, making the grid resolution near the wall region less problematic.

6.2.1 Improvement of the Robustness for Turbulent Flow Simulations

Although the use of the Spalart-Allmaras turbulence model is quite common in the numerical simulations of aerodynamic flows, getting a robust solver is still a challenging task. A very important issue of the Spalart-Allmaras model is the behavior of the turbulent governing equation in zones with high gradients of the working variable. High values of the gradients are generally found near solid walls, where the turbulent variable rises rapidly from zero to a high value, and in the outer zone of the boundary layer where the turbulent variable rapidly decrease to the free-stream value.

While the space discretization is generally such fine in the near-wall zones that the turbulent variable is computed monotonously, the mesh resolution in the outer part of the boundary layer may be insufficient to resolve the turbulent field. In this case, the turbulent field shows numerical oscillations and negative values of the turbulent variable which can make the algorithm unstable.

One common modification with respect to the original model equation consists in limiting the value of the eddy viscosity to zero, so it never becomes negative

$$\mu_t = \begin{cases} \mu_t^\star f_{v1}, & \chi > 0, \\ 0, & \chi \leq 0. \end{cases} \tag{6.29}$$

However it can be observed that for negative values of $\mu_t^\star$ the Spalart–Allmaras model becomes unstable. One of the first modifications introduced to avoid numerical instabilities consists simply in limiting the values of $\hat{\omega}$ such that it always remains positive [116].

Considering the energy associated with the negative values of the working variables, it is possible to introduce more sophisticated modifications [86]. One first multiplies the Spalart–Allmaras equation (6.22) by the turbulent working variable to obtain the following governing equation for the energy of $\nu_t^\star = \mu_t^\star/\rho$

$$\frac{1}{2}\frac{\partial \rho \nu_t^{\star 2}}{\partial t} + \frac{1}{2}\boldsymbol{\nabla}\cdot\left(\nu_t^{\star 2}\boldsymbol{m}\right) - \nu_t^\star \boldsymbol{\nabla}\cdot\left(\frac{\eta}{\sigma_{SA}}\boldsymbol{\nabla}\nu_t^\star\right) = \nu_t^\star S_{SA}. \tag{6.30}$$

Defining Ω^+ and Ω^- the sub-domains of $\Omega = \Omega^+ \cup \Omega^-$, in which $\nu_t^\star$ is positive and negative, respectively, and assuming that $\partial\Omega \cap \partial\Omega^- = 0$ and that $\mu_t^\star$ is $C_0(\Omega)$, the

integration of the previous equation over Ω^- gives

$$\frac{1}{2}\int_{\Omega^-}\frac{\partial\rho\nu_t^{\star 2}}{\partial t}\,\mathrm{d}\Omega+\frac{1}{2}\int_{\partial\Omega^-}\nu_t^{\star 2}\boldsymbol{m}\cdot\boldsymbol{n}\,\mathrm{d}\partial\Omega-\int_{\partial\Omega^-}\nu_t^{\star}\frac{\eta}{\sigma_{SA}}\boldsymbol{\nabla}\nu_t^{\star}\cdot\boldsymbol{n}\,\mathrm{d}\partial\Omega + \int_{\Omega^-}\frac{\eta}{\sigma_{SA}}\|\boldsymbol{\nabla}\nu_t^{\star}\|^2\,\mathrm{d}\Omega=\int_{\Omega^-}\nu_t^{\star}S_{SA}\,\mathrm{d}\Omega. \tag{6.31}$$

The two boundary integrals on the right-hand side of the previous equation are zero due to the continuity of $\nu_t^{\star}$, *i.e.*, $\nu_t^{\star}|_{\partial\Omega^-}=0$. By using the definition of the source term (6.23), one obtains

$$\frac{1}{2}\int_{\Omega^-}\frac{\partial\rho\nu_t^{\star 2}}{\partial t}\,\mathrm{d}\Omega=\int_{\Omega^-}\frac{\rho c_{b2}\nu_t^{\star}-\eta}{\sigma_{SA}}\|\boldsymbol{\nabla}\nu_t^{\star}\|^2\,\mathrm{d}\Omega + \int_{\Omega^-}\rho c_{b1}\hat{\omega}\nu_t^{\star 2}\,\mathrm{d}\Omega-\int_{\Omega^-}\frac{\rho c_{w1}f_w}{d_{\min}^2}\nu_t^{\star 3}\,\mathrm{d}\Omega. \tag{6.32}$$

The previous equation represents the rate of change of the energy of negative values of $\nu_t^{\star}$ associated with a control volume moving together with the boundary of Ω^-. In order to have a stable scheme, the right hand side of the previous equation should be negative for negative values of $\nu_t^{\star}$.

The first integral on the right-hand side of Eq. (6.32) becomes positive for $\nu_t^{\star}<0$ if $(\rho c_{b2}\nu_t^{\star}-\eta)>0$ and thus the following alternative definition of η is adopted to assure that the term remains negative

$$\eta=\begin{cases}\mu(1+\chi), & \chi\geq 0,\\ \mu\Big(1+\chi+\dfrac{1}{2}\chi^2\Big), & \chi<0.\end{cases} \tag{6.33}$$

In order for the second integral on the right-hand side of Eq. (6.32) to be negative for $\nu_t^{\star}<0$, the production term is modified as

$$P_{\mu_t^{\star}}=\begin{cases}c_{b1}\hat{\omega}\mu_t^{\star}, & \chi\geq 0,\\ c_{b1}\omega\mu_t^{\star}g, & \chi<0.\end{cases} \tag{6.34}$$

$$g=1-\frac{1000\chi^2}{1+\chi^2}. \tag{6.35}$$

where the modified vorticity is re-defined as follows

$$\hat{\omega}=\begin{cases}\|\boldsymbol{\omega}\|+\bar{\omega}, & \bar{\omega}>-c_{v2}\|\boldsymbol{\omega}\|\\ \|\boldsymbol{\omega}\|+\dfrac{\|\boldsymbol{\omega}\|\big(c_{v2}^2\|\boldsymbol{\omega}\|+c_{v3}\bar{\omega}\big)}{(c_{v3}-2c_{v2})\|\boldsymbol{\omega}\|-\bar{\omega}}, & \bar{\omega}\leq -c_{v2}\|\boldsymbol{\omega}\|,\end{cases} \tag{6.36}$$

with $c_{v2}=0.7$, and $c_{v3}=0.9$.

Finally since the the last integral on the right-hand side of Eq. (6.32) should be positive for negative values of $\nu_t^\star$, the destruction term is modified as

$$D_{\mu_t^\star} = \begin{cases} \rho c_{w1} f_w \left(\dfrac{\mu_t^\star}{\rho\, d_{\min}} \right)^2, & \chi \geq 0, \\ -\rho c_{w1} \left(\dfrac{\mu_t^\star}{\rho\, d_{\min}} \right)^2, & \chi < 0. \end{cases} \tag{6.37}$$

The modifications introduced have the role to make the Spalart–Allmaras equation stable in the zone with negative $\mu_t^\star$ and make the equation continuous and differentiable at $\mu_t^\star = 0$. However, the modifications remove the negative contribution from the source term by making the production very strong in the negative χ region. Although the modification is mathematically stable, it has been observed in the numerical simulations that the strong effect induced by the modifications of the production and destruction terms can induce a sudden and strong transient during the iterative convergence which could make the solver to fail. In order to have a more robust solver, the elimination of negative values of $\mu_t^\star$ must be addressed directly.

In the actual implementation of the Spalart–Allmaras equation, two modificationshave been adopted which have shown to be very effective. The first modification consists in replacing the variable χ with the following function

$$\psi_{\mu_t^\star} = \begin{cases} 0.05 \log\left(1 + e^{20\chi}\right), & \chi \leq 10, \\ \chi, & \chi > 10, \end{cases} \tag{6.38}$$

the same modification has been proposed in [79], and it has the role to deactivate the production and the destruction terms of the turbulence model equation when $\mu_t^\star$ becomes negative. The function $\psi_{\mu_t^\star}$ is continuous and differentiable for all values of χ as can be observed in Figure 6.1. The form of the turbulence model is identical to the original form with the exception that the variable χ is replaced by $\psi_{\mu_t^\star}$ and the modified vorticity function (6.36) is retained. The second modification simply consists in getting rid of the diffusion contribution in the source term when $\mu_t^\star \leq 0$.

In conclusion, in the improved version of the Spalart–Allmaras turbulence model equation, the eddy viscosity is given by

$$\mu_t = \mu \psi_{\mu_t^\star} f_{v1}, \tag{6.39}$$

and the turbulent model equations is written as

$$\frac{\partial \mu_t^\star}{\partial t} + \boldsymbol{\nabla} \cdot \left(\mu_t^\star \frac{\boldsymbol{m}}{\rho} - \frac{\eta}{\sigma_{SA}} \boldsymbol{\nabla} \left(\frac{\mu_t^\star}{\rho} \right) \right) = S_{SA}, \tag{6.40}$$

with $\eta = \mu(1 + \psi_{\mu_t^\star})$ and the source term given by

$$S_{SA} = \begin{cases} c_{b1} \hat{\omega} \mu \psi_{\mu_t^\star} - \rho c_{w1} f_w \left(\dfrac{\mu \psi_{\mu_t^\star}}{\rho\, d_{\min}} \right)^2 + \dfrac{1}{\sigma_{SA}} \rho c_{b2} \left\| \boldsymbol{\nabla} \left(\dfrac{\mu_t^\star}{\rho} \right) \right\|^2, & \chi > 0, \\ c_{b1} \hat{\omega} \mu \psi_{\mu_t^\star} - \rho c_{w1} f_w \left(\dfrac{\mu \psi_{\mu_t^\star}}{\rho\, d_{\min}} \right)^2 & \chi \leq 0, \end{cases} \tag{6.41}$$

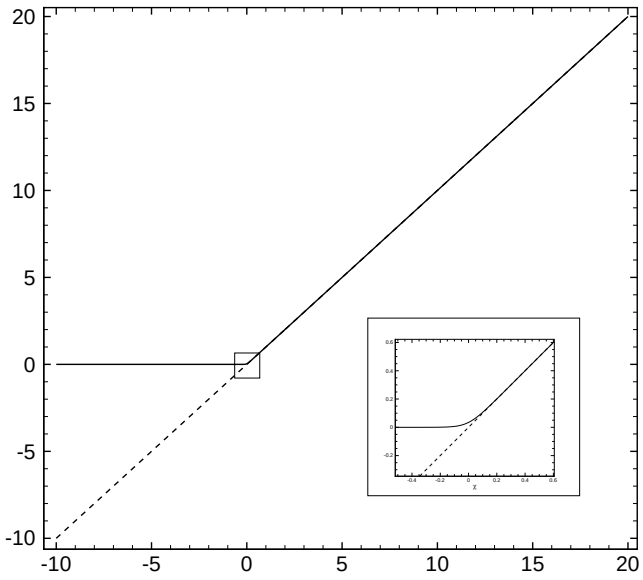


Figure 6.1: Comparison between $\psi_{\mu_t^\star}$ (solid line) and χ (dashed line) as function of $\mu_t^\star/\mu$.

and the different functions involved are defined as

$$\hat{\omega} = \begin{cases} \|\boldsymbol{\omega}\| + \bar{\omega}, & \bar{\omega} > -c_{v2}\|\boldsymbol{\omega}\|, \\ \|\boldsymbol{\omega}\| + \dfrac{\|\boldsymbol{\omega}\|\big(c_{v2}^2\|\boldsymbol{\omega}\| + c_{v3}\bar{\omega}\big)}{(c_{v3} - 2c_{v2})\|\boldsymbol{\omega}\| - \bar{\omega}}, & \bar{\omega} \leq -c_{v2}\|\boldsymbol{\omega}\|, \end{cases} \tag{6.42}$$

$$\begin{gathered} \bar{\omega} = \frac{\mu\psi_{\mu_t^\star} f_{v2}}{\rho\, k^2\, d_{\min}^2}, \qquad f_{v1} = \frac{\psi_{\mu_t^\star}^3}{\psi_{\mu_t^\star}^3 + c_{v1}^3}, \qquad f_{v2} = 1 - \frac{\psi_{\mu_t^\star}}{1 + \psi_{\mu_t^\star} f_{v1}}, \\ f_w = g\left(\frac{1 + c_{w3}^6}{g^6 + c_{w3}^6}\right)^{\frac{1}{6}}, \quad g = r + c_{w2}\big(r^6 - r\big), \quad r = \frac{\mu\psi_{\mu_t^\star}}{\rho\,\hat{\omega}\, k^2\, d_{\min}^2}. \end{gathered} \tag{6.43}$$

Note that with the definition (6.41), the source term tends to zero when $\mu_t^\star$ becomes negative. In the approach of [40], the source term is directly set to zero for negative values of $\mu_t^\star$.

6.2.2 Dimensionless Form of the Spalart–Allmaras Equation

The non-dimensionalization of the RANS equations is obtained with same approach shown in Section 5.1.3. In particular, considering the equation for the momentum in Eq. (6.16) and letting μ_{t_r} be the reference turbulent viscosity, one has

$$\frac{\partial\tilde{\boldsymbol{m}}}{\partial\tilde{t}} + \tilde{\boldsymbol{\nabla}}\boldsymbol{\cdot}\left(\tilde{\boldsymbol{m}} \otimes \tilde{\boldsymbol{m}} + \left[\frac{P_r}{\rho_r v_r^2}\right]\tilde{P}\mathbb{I}\right) = \left[\frac{\mu_r}{\rho_r v_r l_r} + \frac{\mu_{t_r}}{\rho_r v_r l_r}\right]\tilde{\boldsymbol{\nabla}}\boldsymbol{\cdot}\tilde{\mathbb{S}}. \tag{6.44}$$

Since the momentum equation must recovers the same form of the dimensional counterpart, in the previous equation the terms in the square brackets must amount to unity, namely

$$\frac{P_r}{\rho_r v_r^2} = 1, \quad \frac{\mu_r}{\rho_r v_r l_r} = 1 \quad \text{and} \quad \frac{\mu_{tr}}{\rho_r v_r l_r} = 1. \tag{6.45}$$

The first two conditions have been already seen in the case of the Navier-Stokes equations, while the last relation gives an additional condition for the reference eddy viscosity, namely

$$\mu_{tr} = \rho_r v_r l_r = \mu_r. \tag{6.46}$$

Consider now the Spalart–Allmaras Equation, with Eq. (6.39) in mind, the non-dimensional eddy viscosity is given by

$$\tilde{\mu_t} = \frac{\mu_t}{\mu_{tr}} = \frac{\mu \psi_{\mu_t^\star}}{\mu_{tr}} f_{v1} = \tilde{\mu} \psi_{\mu_t^\star} f_{v1}. \tag{6.47}$$

With this definition of the dimensionless eddy viscosity and with the dimensionless quantity defined for the Navier-Stokes equation, it is possible to write the dimensionless Spalart–Allmaras equation exactly as its dimensional counterpart [1].

6.3 Spatial Discretization of the RANS Equations

The complete set of the governing equations for RANS simulations is given by the equations for the mean quantities (6.21) and the equations for the turbulence modeling, in the case considered here the one equation Spalart–Allmaras model (6.22).

Generally, two approaches are followed in the discretization of the RANS equations. In the fist approach [16, 17, 30, 51] the equations for the turbulence modeling are fully coupled with the mean flow equations and the problem is solved for both the mean flow quantities and turbulence variables. In the second approach [15, 19] the governing equations for the mean flow are solved first, the mean flow quantities are then used as inputs for the solver of the turbulence equations and the process is generally reiterated for a certain number of iterations.

Despite the difference of the two approaches, there seems to be no firm analytical basis on how to couple the RANS equations and the turbulence model equations. For example, in a loosely coupled approach the turbulence equations might be discretized with a scheme different from that used for the mean flow equations; usually a more dissipative scheme is used for the turbulence equations to improve the robustness of the numerical solver. However, it has been observed that using a fully coupled

[1] Note that in Eq. (6.47) the term $\tilde{\mu}$ is generally very small, for this reason in the numerical code instead of solving directly for $\tilde{\mu_t^\star}$ it is better to solve for $\bar{\mu_t^\star} = (Re_\infty / M_\infty)\tilde{\mu_t^\star}$, *cf* Eq. (5.35), and then compute $\tilde{\mu_t^\star}$.

approach might improve the iterative convergence of the scheme [72], especially when the turbulent kinetic energy contributions are considered in the equations for the mean flow, and there would not be much gain by using the strongly coupled method when the turbulent kinetic energy contribution is ignored.

In this work the Spalart–Allmaras equation is fully coupled with the RANS equations and the whole set of the governing equations is discretized with the RD scheme described in Chapter 5 for the Navier-Stokes equations. This approach guaranties that both the mean flow variables and the Spalart–Allmaras working variable are discretized with the same level of accuracy, Furthermore, the numerical implementation of the turbulence model is straightforward in the Navier-Stokes solver.

With respect to the Navier-Stokes equations, the set of the unknowns includes the turbulence working variable and the advective and viscous flux functions are augmented to include the advective and viscous terms of the Spalart–Allmaras equation, furthermore the source has to be also considered. In vector form one can writes

$$\frac{\partial \mathsf{u}}{\partial t} + \boldsymbol{\nabla}\cdot \mathbf{f}^a(\mathsf{u}) - \boldsymbol{\nabla}\cdot \mathbf{f}^v(\mathsf{u}, \boldsymbol{\nabla}\mathsf{u}) = \mathsf{S}(\mathsf{u}, \boldsymbol{\nabla}\mathsf{u}) \tag{6.48}$$

with

$$\mathsf{u} = \begin{pmatrix} \rho \\ \boldsymbol{m} \\ E^t \\ \mu_t^\star \end{pmatrix}, \qquad \mathbf{f}^a(\mathsf{u}) = \begin{pmatrix} \boldsymbol{m} \\ \dfrac{\boldsymbol{m}\otimes\boldsymbol{m}}{\rho} + P\mathbb{I} \\ \big(E^t + P\big)\dfrac{\boldsymbol{m}}{\rho} \\ \mu_t^\star \dfrac{\boldsymbol{m}}{\rho} \end{pmatrix}, \qquad \mathbf{f}^v(\mathsf{u}, \boldsymbol{\nabla}\mathsf{u}) = \begin{pmatrix} 0 \\ \mathbb{S} \\ \mathbb{S}\cdot\dfrac{\boldsymbol{m}}{\rho} + \kappa\boldsymbol{\nabla}T \\ \dfrac{\eta}{\sigma_a}\boldsymbol{\nabla}\left(\dfrac{\mu_t^\star}{\rho}\right) \end{pmatrix}, \tag{6.49}$$

and $\mathsf{S}(\mathsf{u}, \boldsymbol{\nabla}\mathsf{u}) = (0, \mathbf{0}, 0, S_{SA})^{\mathrm{T}}$. The definitions of the eigen-structure and the homogeneity tensor associated with the new definition of the advective and viscous flux functions, respectively, are reported in Appendix B.

The total residual for a generic element e is computed as

$$\Phi^e = \oint_{\partial\Omega_e} \left(\mathbf{f}^a(\mathsf{u}_h) - \mathbb{K}(\mathsf{u}_h)\widetilde{\boldsymbol{\nabla}\mathsf{u}_h}\right)\cdot\hat{\boldsymbol{n}}\,\mathrm{d}\partial\Omega - \int_{\Omega_e} \mathsf{S}(\mathsf{u}_h, \widetilde{\boldsymbol{\nabla}\mathsf{u}_h}), \tag{6.50}$$

note that in order to improve the accuracy of the numerical discretization, the reconstructed gradient is used in the calculation of the source term, even if the integral is strictly defined on the interior of the element.

Analogously to the Navier-Stokes equation, the centred linear scheme reads

$$\begin{aligned} \Phi_i^{e,\mathrm{LW}} = \frac{\Phi^e}{N^e_{\mathrm{dof}}} &+ \int_{\Omega_e} \boldsymbol{A}\cdot\boldsymbol{\nabla}\psi_i\,\Xi\left(\boldsymbol{A}\cdot\boldsymbol{\nabla}\mathsf{u}_h - \boldsymbol{\nabla}\cdot\left(\mathbb{K}\widetilde{\boldsymbol{\nabla}\mathsf{u}}\right) - \mathsf{S}(\mathsf{u}_h, \widetilde{\boldsymbol{\nabla}\mathsf{u}_h})\right)\mathrm{d}\Omega \\ &+ \int_{\Omega_e} \mathbb{K}\boldsymbol{\nabla}\psi_i\cdot\left(\boldsymbol{\nabla}\mathsf{u}_h - \widetilde{\boldsymbol{\nabla}\mathsf{u}_h}\right)\mathrm{d}\Omega \end{aligned} \tag{6.51}$$

and the limited non-linear scheme reads

$$\begin{aligned}\hat{\Phi}_i^{e,\mathrm{Rv}} = \; & \hat{\Phi}_i^e \\ & + \varepsilon_h^e(\mathsf{u}_h)\int_{\Omega_e}\Big(\boldsymbol{A}\cdot\boldsymbol{\nabla}\psi_i - \mathbb{K}\boldsymbol{\nabla}\psi_i\Big)\,\Xi\left(\boldsymbol{A}\cdot\boldsymbol{\nabla}\mathsf{u}_h - \boldsymbol{\nabla}\cdot\big(\mathbb{K}\widetilde{\boldsymbol{\nabla}\mathsf{u}_h}\big) - \mathsf{S}(\mathsf{u}_h, \widetilde{\boldsymbol{\nabla}\mathsf{u}_h})\right)\mathrm{d}\Omega, \\ & + \int_{\Omega_e}\mathbb{K}\boldsymbol{\nabla}\psi_i\cdot\left(\boldsymbol{\nabla}\mathsf{u}_h - \widetilde{\boldsymbol{\nabla}\mathsf{u}_h}\right)\mathrm{d}\Omega\end{aligned} \tag{6.52}$$

with $\hat{\Phi}_i^e$ the limited version of the residual distributed with the Rusanov scheme.

The boundary conditions are imposed in the same way showed for the Navier-Stokes equations, with the addition of the condition $\mu_t^\star = 0$ on the wall, which is imposed in a strong way. At the far-field, the free-stream value of $\mu_t^\star$ is weakly imposed, and as suggested by Allmaras and coworkers [14], the free-stream value $\mu_t^\star/\mu = 3 - 5$ is imposed for fully turbulent simulations.

6.4 Non-Linear LU-SGS Implicit Solver

The equations for the turbulence model make the system of the governing equations stiffer compared to the Navier-Stokes equations, in addition, since in turbulent flows the variation of the solution in the direction normal to wall is much higher than that in the tangential direction, very stretched grids are generally used in numerical simulations of turbulent flows. These factors, makes the solution of turbulent problems much more difficult than laminar ones and the robustness of the numerical solver is drastically reduced.

Since high-order methods are generally less robust than standard low-order methods, the construction of a reliable numerical scheme becomes crucial. In fact, it has been observed in numerical simulations of turbulent flows at vary high Reynolds number, typical of aerodynamic applications, that the iterative scheme is very prone to fail with a sudden breakdown of the solver or it stagnates at high values of the residual. A possible explanation for this behavior is strictly related to the nature of the implicit solver described in Section 5.4.1. In the matrix-free approach, the Jacobian matrix is approximated with a finite difference method, this approach might produce ill conditioned Jacobian matrices for very stiff problems that results in a failure of the numerical solver. Furthermore, since the computed Jacobian used as preconditioning matrix is only an approximation of the true Jacobian of the numerical scheme, the use of more sophisticated preconditioning techniques will be probably ineffective.

For thees reasons, instead of using a matrix-free based Newton Krylov method, a new strategy has been adopted for the solution of the non-linear system obtained from the discretization of the RANS equations. The new approach is based on the

manipulation of the Gauss-Seidel method in order to avoid the calculation of the off-diagonal elements in the Jacobian matrix.

Consider the solution of the non linear system with the Newton's scheme (5.77), which is re-written here for convenience

$$\left(\frac{\mathbb{I}}{\Delta t^n}+\frac{\partial \mathsf{R}}{\partial \mathsf{u}}\right)\Delta \mathsf{u}_h^n=-\mathsf{R}(\mathsf{u}_h^n), \tag{6.53}$$

The previous equation can be recasted in the following form

$$\left(\frac{\mathbb{I}}{\Delta t^n}+\frac{\partial \mathsf{R}_i}{\partial \mathsf{u}_i}\right)\Delta \mathsf{u}_i^n=-\mathsf{R}_i(\mathsf{u}^n)-\sum_{\substack{j\in\mathcal{N}_h^i\\ j\neq i}}\frac{\partial \mathsf{R}_i}{\partial \mathsf{u}_j}\Delta \mathsf{u}_j^n,\quad \forall\, i\in\mathcal{N}_h. \tag{6.54}$$

When the Gauss-Seidel method is applied to solve (6.54), the forward and backward sweep steps read

$$\left(\frac{\mathbb{I}}{\Delta t^n}+\frac{\partial \mathsf{R}_i}{\partial \mathsf{u}_i}\right)\Delta \mathsf{u}_i^*=-\mathsf{R}_i(\mathsf{u}^n)-\sum_{\substack{j\in\mathcal{N}_h^i\\ j<i}}\frac{\partial \mathsf{R}_i}{\partial \mathsf{u}_j}\Delta \mathsf{u}_j^*,\quad i=1,2,\ldots,N_{\text{dof}}, \tag{6.55}$$

$$\left(\frac{\mathbb{I}}{\Delta t^n}+\frac{\partial \mathsf{R}_i}{\partial \mathsf{u}_i}\right)(\Delta \mathsf{u}_i^n-\Delta \mathsf{u}_i^*)=-\sum_{\substack{j\in\mathcal{N}_h^i\\ j>i}}\frac{\partial \mathsf{R}_i}{\partial \mathsf{u}_j}\Delta \mathsf{u}_j^n,\quad i=N_{\text{dof}},\ldots,2,1. \tag{6.56}$$

A better scheme can be obtained by applying a symmetric variation of the Gauss-Seidel method, in which both forward and backward steps are applied alternatively. The forward and backward sweep steps can be written in concise form as

$$\left(\frac{\mathbb{I}}{\Delta t^n}+\frac{\partial \mathsf{R}_i}{\partial \mathsf{u}_i}\right)\Delta \mathsf{u}_i^{(k+1)}=-\mathsf{R}_i(\mathsf{u}^n)-\sum_{\substack{j\in\mathcal{N}_h^i\\ j\neq i}}\frac{\partial \mathsf{R}_i}{\partial \mathsf{u}_j}\Delta \mathsf{u}_j^{(*)},\quad \forall\, i\in\mathcal{N}_h, \tag{6.57}$$

where k is an inner iteration index, the superscript $^{(*)}$ indicates the most recently updated solution, and $\Delta \mathsf{u}^{(0)}=0$. At the end, $\mathsf{u}^{n+1}=\mathsf{u}^n+\Delta \mathsf{u}^{(k+1)}$.

The right-hand side of the previous equation can be further manipulated as follows. Denoting the latest available solution as $\mathsf{u}^{(*)}=\mathsf{u}^n+\Delta \mathsf{u}^{(*)}$, it is possible to linearize $\mathsf{R}_i\big(\mathsf{u}^{(*)}\big)$ as

$$\begin{aligned}\mathsf{R}_i\big(\mathsf{u}^{(*)}\big)&\approx \mathsf{R}_i\big(\mathsf{u}^n\big)+\sum_{j\in\mathcal{N}_h^i}\frac{\partial \mathsf{R}_i}{\partial \mathsf{u}_j}\Delta \mathsf{u}_j^{(*)}\\ &=\mathsf{R}_i\big(\mathsf{u}^n\big)+\frac{\partial \mathsf{R}_i}{\partial \mathsf{u}_i}\Delta \mathsf{u}_i^{(*)}+\sum_{\substack{j\in\mathcal{N}_h^i\\ j\neq i}}\frac{\partial \mathsf{R}_i}{\partial \mathsf{u}_j}\Delta \mathsf{u}_j^{(*)},\end{aligned} \tag{6.58}$$

and the following relation is obtained

$$\mathsf{R}_i\big(\mathsf{u}^n\big)+\sum_{\substack{j\in\mathcal{N}_h^i\\ j\neq i}}\frac{\partial \mathsf{R}_i}{\partial \mathsf{u}_j}\Delta \mathsf{u}_j^{(*)}=\mathsf{R}_i\big(\mathsf{u}^{(*)}\big)-\frac{\partial \mathsf{R}_i}{\partial \mathsf{u}_i}\Delta \mathsf{u}_i^{(*)}. \tag{6.59}$$

Substituting the previous relation in Eq. (6.57), one obtains the following scheme

$$\left[\frac{\mathbb{I}}{\Delta t^n} + \frac{\partial \mathsf{R}_i}{\partial \mathsf{u}_i}\right] \Delta \mathsf{u}_i^{(k+1)} - \frac{\partial \mathsf{R}_i}{\partial \mathsf{u}_i} \Delta \mathsf{u}_i^{(*)} = -\mathsf{R}_i\big(\mathsf{u}^{(*)}\big), \tag{6.60}$$

which at last can be recasted as

$$\left[\frac{\mathbb{I}}{\Delta t^n} + \frac{\partial \mathsf{R}_i}{\partial \mathsf{u}_i}\right] \Big(\Delta \mathsf{u}_i^{(k+1)} - \Delta \mathsf{u}_i^{(*)}\Big) = -\mathsf{R}_i\big(\mathsf{u}^{(*)}\big) + \frac{\Delta \mathsf{u}_i^{(*)}}{\Delta t^n}. \tag{6.61}$$

The Eq. (6.61) is solved with the forward and backward sweeps, at the beginning of each step the small diagonal blocks of the Jacobian matrix in the left-hand side can be inverted using LU decomposition. Note that the right-hand side of the previous equation is nothing but the residual evaluated with the latest available solutions. The last term in the right hand side of the (6.61) is usually dropped for steady simulations, in order to accelerate the convergence rate. When the symmetric Gauss-Seidel converges, one is actually solving the original equation (5.76) instead of the linearized version (6.54), for this reason the algorithm is called non-linear LU-SGS [120].

The expression (6.61) still requires the diagonal blocks of the Jacobian matrix, in some works [92, 120] the diagonal blocks of the Jacobian are approximated with a finite difference method, however this strategy has been found unsatisfactory for RANS simulations, thus the diagonal of the approximated Jacobian is used.

6.5 Numerical Results

6.5.1 High Reynolds Number Flow over a Flat Plate

The two dimensional, turbulent flow over a flat plate is here considered. This test case is widely used for the verification and the validation of numerical schemes and turbulence models, furthermore, it is also useful to study the accuracy of numerical schemes in resolving boundary layer features.

The free-stream Mach number is $\mathrm{M} = 0.2$ and the Reynolds number, computed with the free-stream conditions and a unity length of the plate, is $\mathrm{Re} = 5 \times 10^6$, and the length of the plate is $\mathrm{L} = 2$. The range of the computational domain in the x-direction is $[-0.33, 2]$ with the leading edge of the flat plate at $x = 0$. The size of the computational domain in the y-direction is 1, which is about 66 times the boundary layer thickness at $x = 2$. At the inlet, the inflow boundary condition is imposed, while at the top and the exit, the static pressure is imposed. Along the plane $y = 0$, the symmetry boundary condition is imposed for $-0.33 \leq x \leq 0$ and the adiabatic wall no-slip boundary condition is imposed for $0 \leq x \leq 2$. The free-stream value of the turbulent working variable is such that $\mu_t^\star / \mu_\infty = 3$.

A sequence of uniformly refined grids is considered. The coarsest grid is obtained by the triangulation of a 35×25 structured grid; the first number refers to the number of nodes on the horizontal boundaries, with 29 nodes along the plate, the second number refers to the number of nodes on the vertical boundaries. The grid is stretched in the wall-normal direction, and is also clustered near the plate leading edge; the height of the elements along the wall is 8.32×10^{-6}. Finer grids are obtained by applying successive uniform refinements to the coarsest grid. The grids used are the same reported on the NASA Turbulence Modeling Resource repository [107].

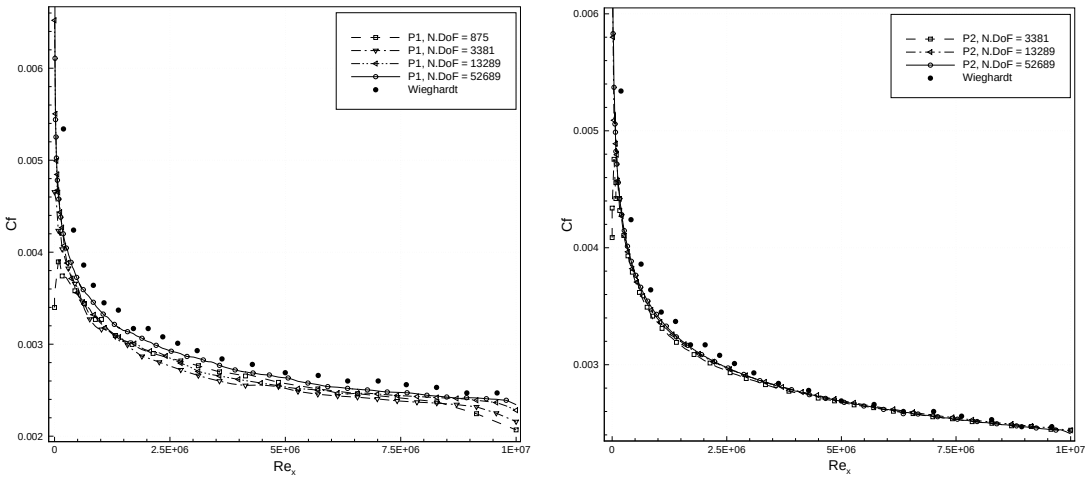
Except for the singularity of the solution at the leading edge of the plate, the flow is smooth and for this reason the linear scheme is used, with the SPR-ZZ method. Despite the high aspect ratio of the elements, this test case is not very problematic for the non-linear solver, thus the Jacobian-free method with the LU-SGS preconditioner is used, for this particular test case. For each grid, simulations with linear and quadratic approximation of the solution are performed, and the L^2 norm of the residuals of the mean flow and turbulence equations is reduced by ten orders of magnitude.

The skin friction distributions along the plate for the linear and quadratic approximation of the solution, for grids with different resolutions, are reported in Figure 6.2 as function of the quantity $\mathrm{Re}_x = x\|\boldsymbol{v}_\infty\|/\mu_\infty$; for reference, the experimental measures of Wieghardt [136] are also reported. It is worth noting that the experimental data were obtained in a essentially incompressible regime ($\mathrm{M} = 0.096$), however in the numerical simulations, although the Mach number is low ($\mathrm{M} = 0.2$), compressibility effects cannot be disregarded.

Note that the linear approximation of the solution gives a quite poor resolution of the friction coefficient along the plate; the distribution is not regular and the solution is still far from a grid independent value, even for the finest grid. The quadratic approximation of the solution, on the other hand, gives a very regular profile of the friction coefficient. The values agree well with the experimental measures and it is also clear that a reasonable accurate solution is already obtained on the coarsest mesh.

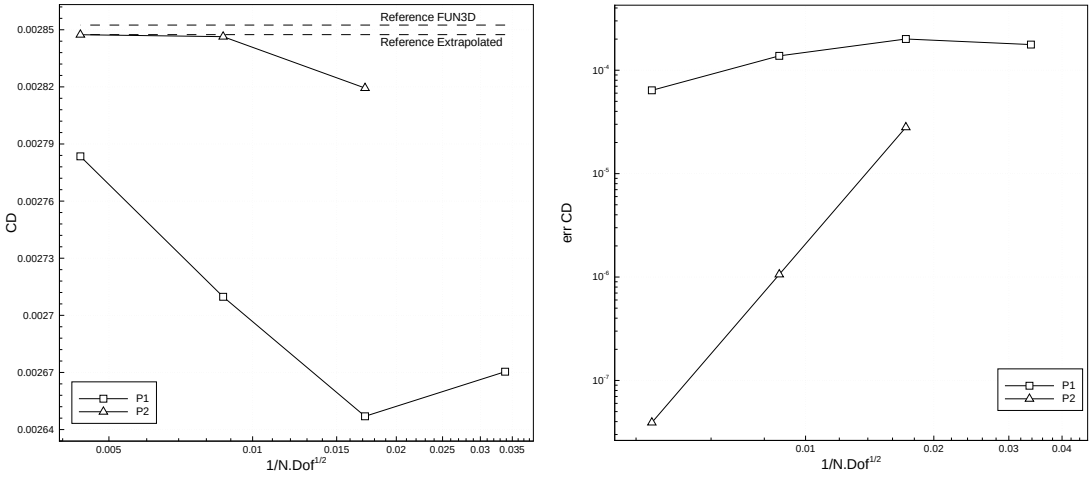
A quantitative comparison between the linear and quadratic approximation of the solution is reported in Figure 6.3-a, where the values of the drag coefficient are reported as function of the number of the degrees of freedom. Two reference values of the drag coefficient are also reported; the first one was computed with the NASA's unstructured FV code FUN3D [80] on a finer grid [108], the second reference value was obtained with a Richardson extrapolation [100] of the drag coefficients computed with quadratic elements. The difference between the two values is about 0.17%.

It is evident how, for the same number of degrees of freedom, the solution computed with the quadratic elements converges faster to the reference value, than the solution computed with the linear approximation. Thus, a second order scheme requires much more degrees of freedom than the third order scheme, to get a grid independent and accurate solution. This can be observed also in figure Figure 6.3-b, where the error for the drag coefficient is reported as function of the number of



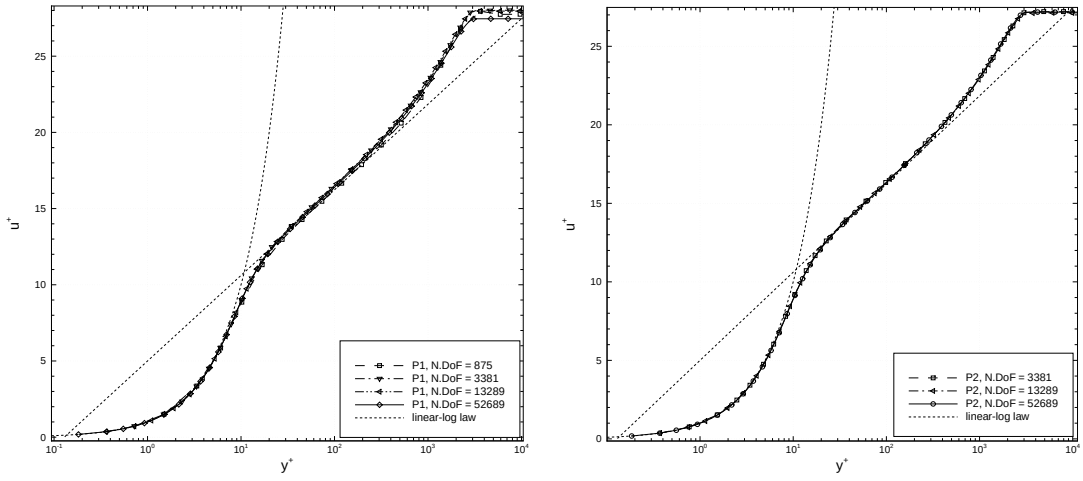
(a) (b)

Figure 6.2: Friction coefficient along the plate for the turbulent flow over a flat plate, with different grid resolutions and with linear (a) and quadratic (b) approximation of the solution.



(a) (b)

Figure 6.3: Effects of the grid and polynomial refinement on the solution accuracy for the turbulent flow over a flat plate: (a) drag coefficients (b) error on the computation of the drag coefficient.



(a) (b)

Figure 6.4: Dimensionless axial velocity profiles in the boundary layer at the point $x = 0.97$ for the turbulent flow over a flat plate, with different grid resolutions and with linear (a) and quadratic (b) approximation of the solution.

the degrees of freedom. The error is computed as the absolute values of the difference between the numerical value of the drag coefficient and the reference one, *i.e.*, $err_{\mathrm{CD}} = |c_D - c_{D_{\mathrm{ref}}}|$. The reference drag coefficient used for the calculation of the error is the value obtained with the Richardson extrapolation, because the FV reference value is reported with not enough digits, and an early stagnation of the error for the solution obtained with quadratic elements would be observed.

In Figure 6.4 are also reported the profiles of the axial velocity in the boundary layer at the station $x = 0.97$, in terms of dimensionless quantities. The dimensionless distance from the plated is defined as $y^+ = y\rho u_\tau/\mu$, with the friction velocity given by $u_\tau = \sqrt{\left(\mu \frac{\partial v_x}{\partial y}\right)\Big|_{\mathrm{wall}}}$, the dimensionless velocity is defined as $u^+ = v_x/u_\tau$. The figures report also the law-of-wall profiles computed according to the following equations [97]

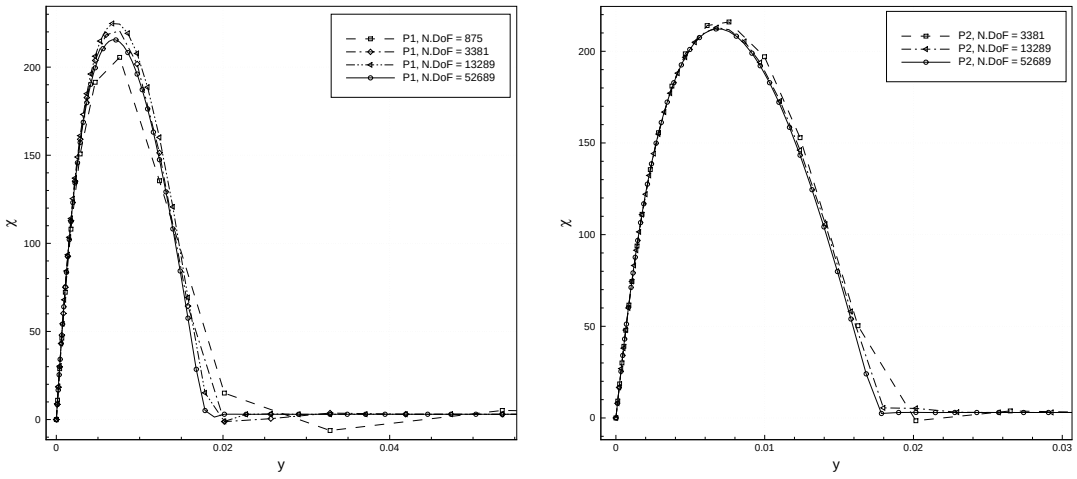
$$\begin{cases} u^+ = y^+, & \text{in the viscous sublayer} \\ u^+ = \dfrac{1}{\kappa} \log y^+ + B, & \text{in the loglayer} \end{cases} \tag{6.62}$$

where $\kappa = 0.41$ is the von Karman constant and $B = 5$. Since the friction velocity is based on the computed skin friction values at the wall, the dimensionless quantity depends on the accuracy of the solution. For the linear approximation of the solutions, a good agreement with the theoretical profiles in the viscous sub-layer and in the log-layer regions is obtained, also in the outer part of the boundary layer the velocity profiles dependent only a little on the grid resolution. Differences in the velocity profiles for quadratic elements can hardly observed from the figure.

Since no wall-functions have been used for the turbulent equation, it is important

	Grid 35×25	Grid 69×49	Grid 137×97	Grid 273×193
$y^+_{1_{\mathrm{P1}}}$	1.500	0.722	0.359	0.182
$y^+_{1_{\mathrm{P2}}}$	0.765	0.372	0.184	–

Table 6.1: Values of the dimensionless heights of the elements on the wall at $x = 0.97$ for the turbulent flow over a flat plate, for different grids and for linear and quadratic elements.

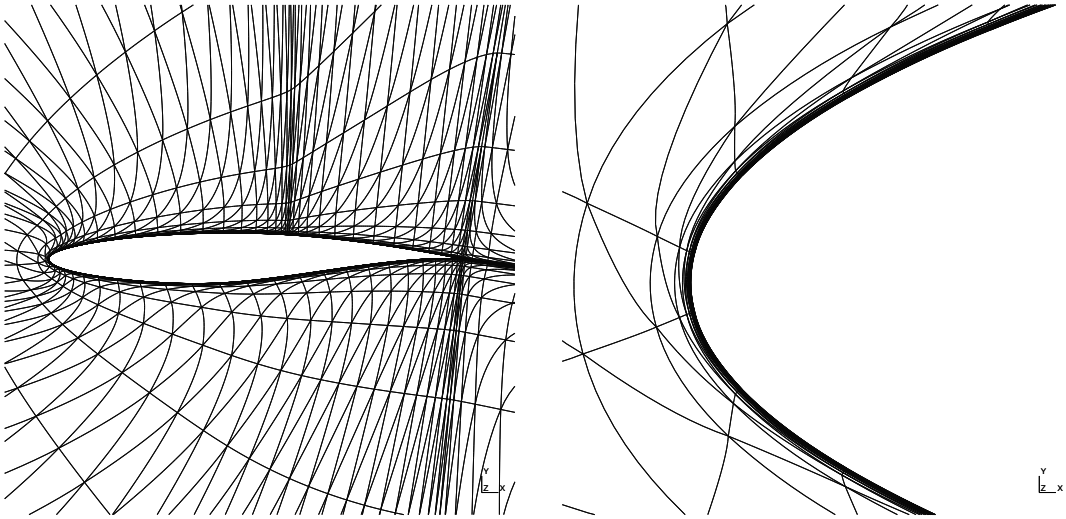


(a) (b)

Figure 6.5: Dimensionless turbulent working variable profiles in the boundary layer at the point $x = 0.97$ for the turbulent flow over a flat plate, with different grid resolutions and with linear (a) and quadratic (b) approximation of the solution.

to verify that the height of the elements on the wall is contained within the viscous sub-layer. In Table 6.1 are reported the values of the dimensionless heights of the element on the wall, computed at $x = 0.97$, for different grids, and for linear and quadratic elements. Note that in the case of quadratic elements, the height of the elements is computed as the distance from the plate of the middle node on the edge of the element, this means that the distance from the wall is half the height of the element.

In Figure 6.5 are reported the profiles of the quantity $\chi = \mu_t^\star/\mu$ at the station $x = 0.97$, to show the value of the turbulent working variable along the boundary layer. Note that when the spatial resolution is poor (generally at the outer part of the boundary layer) negative values of the turbulent working variable might be obtained, however this effect is canceled out with the mesh refinement and/or the order increment of the numerical solution.



(a) (b)

Figure 6.6: An example of coarse grid used for the numerical simulation of a turbulent flow over the RAE2822 airfoil and a zoom of the mesh near the leading edge of the airfoil to show the curvilinear elements.

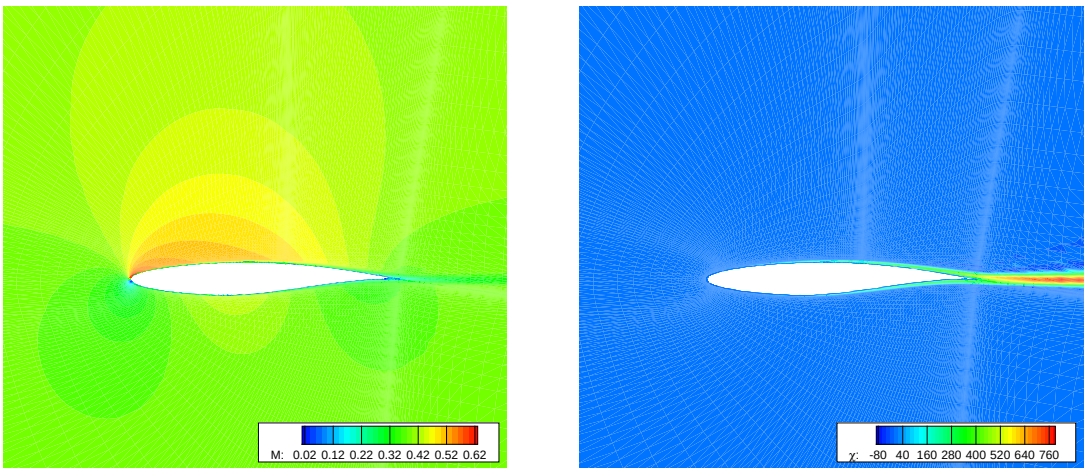
6.5.2 Subsonic Flow over a RAE2822 Airfoil

The second test case considered is the subsonic flow over a RAE2822 airfoil at angle of incidence $\alpha = 2.79°$, the Mach number based on the free-stream conditions is $\mathrm{M} = 0.4$ and the Reynolds number based on the free-stream conditions and the airfoil chord is $\mathrm{Re} = 6.5 \times 10^6$.

The problem is solved, with the linear scheme and the SPR-ZZ method, on a sequence of unstructured grids of triangles obtained with subsequent uniform refinements of the coarsest grid, shown in Figure 6.6, which is made of 4 048 elements. As practice in this work, an isoparametric representation is used for the solution and the geometry. Due to the strong stretching of the boundary layer elements, some of the quadratic elements have curved boundaries non only on the airfoil wall but also at the interior. The solution is converged using the non-linear LU-SGS method; solutions are considered to be at convergence when the L^2 norm of the residual of all the variables (mean flow and turbulence equation) is reduced by ten orders of magnitude.

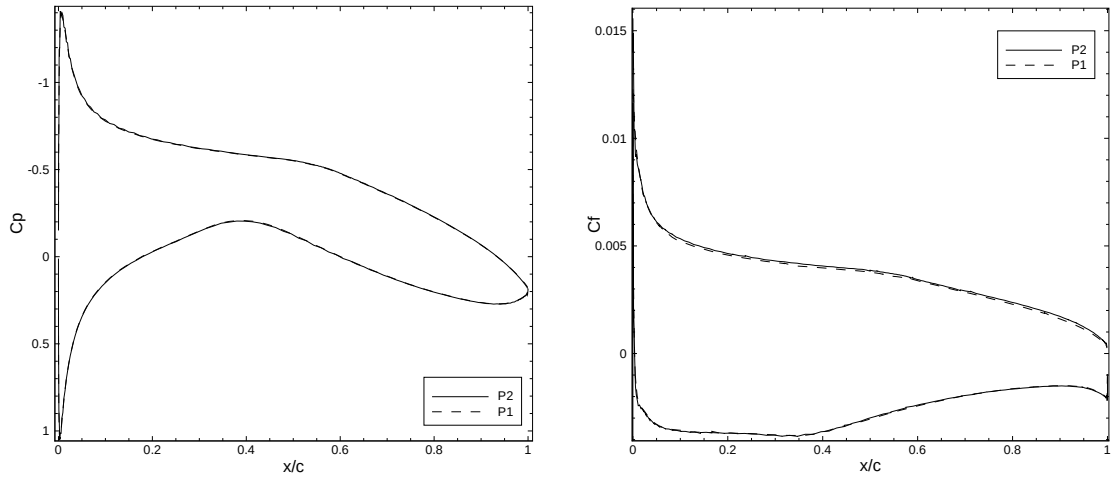
In Figure 6.7 are reported the contours of the Mach number and of the turbulence variable $\chi = \mu_t^\star/\mu$ for the third order simulation on a fine grid, and Figure 6.8 shows the pressure and friction coefficient along the airfoil computed with linear and quadratic elements on two different grids, such that the number of degrees of freedom of the two simulations is the same ($N_{\mathrm{dof}} = 32\,784$); the respective iterative convergence histories are reported in Figure 6.9.

For completeness, in Figure 6.10 are reported the values of the lift and drag coefficients computed on different grids with linear and quadratic elements. Note



(a) (b)

Figure 6.7: Mach number (a) and turbulent working variable (b) contours for the subsonic flow over a RAE2822 airfoil discretized with quadratic elements.



(a) (b)

Figure 6.8: Pressure (a) and friction (b) coefficient along the RAE2822 airfoil with linear and quadratic elements on two different grids, such that $N_{\text{dof}} = 32\,784$.

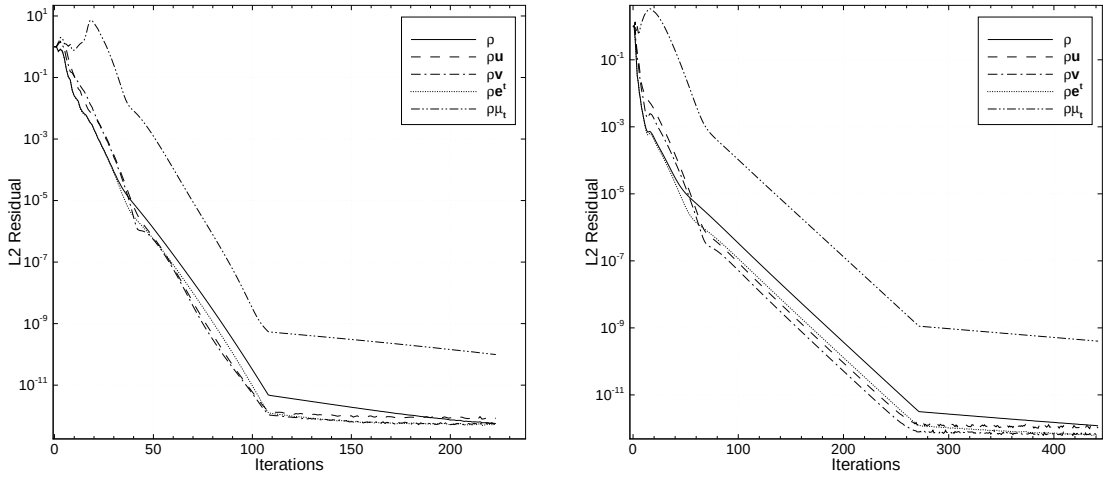


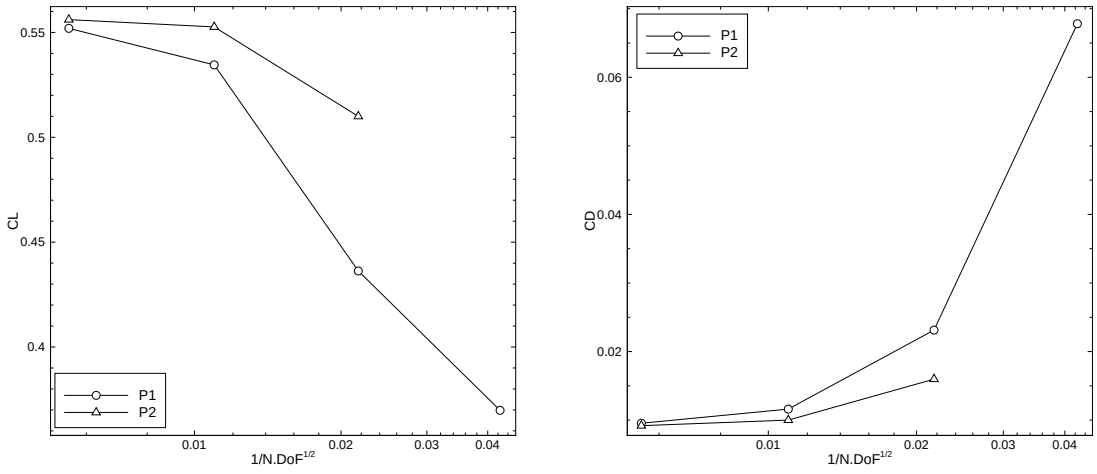
Figure 6.9: Iterative convergence for the simulation of the subsonic flow over the RAE2822 aifoil with linear (a) and quadratic (b) elements on two different grids, such that $N_{\text{dof}} = 32\,784$.

how the use of quadratic elements allows a faster convergence with the mesh refinement of the lift and drag coefficients to their respective asymptotic values. In addition, Figure 6.11 reports the errors for the force coefficients; the error for the lift coefficient is computed as $err_{\text{CL}} = |c_{L_h} - c_{L_{\text{ex}}}|$, where c_{L_h} is the value of the lift coefficient computed numerically and $c_{L_{\text{ex}}}$ is a reference value obtained with a Richardson extrapolation [100] of the lift coefficients computed with quadratic elements. The same procedure is also used to compute the error on the drag coefficient. For the lift coefficient, it is clear the advantage of using a high-order approximation over standard second order schemes even for the simulations of RANS equations. For the drag coefficient, the difference between second and third order schemes reduces but nevertheless the quadratic approximation of the solutions gives a smaller error, for the same number of degrees of freedom.

6.5.3 Transonic Flow over a RAE2822 Airfoil

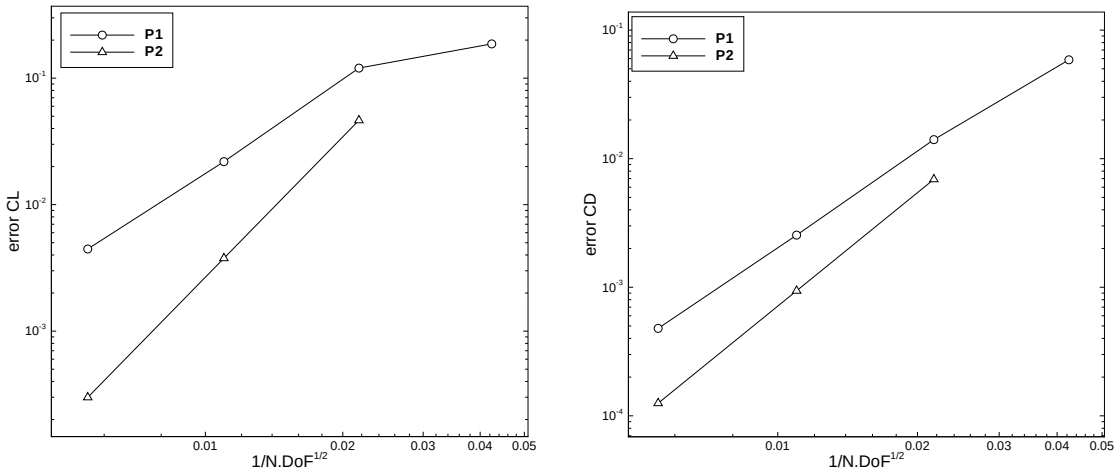
In this test case a transonic, turbulent flow over a RAE2822 airfoil is considered. The free-stream Mach number is $\text{M} = 0.734$, the angle of incidence and the Reynolds number are the same of the previous test case, namely $\alpha = 2.79°$ and $\text{Re} = 6.5 \times 10^6$. In these conditions, the upper surface of the airfoil is characterized by the presence of a shock wave which impinges the boundary layer, generating a small recirculation bubble behind the shock.

Due to the presence of the shock, the problem is discretized with the non-linear scheme, and the SPR-ZZ method is used for the gradient reconstruction. Furthermore, the solution is converged using the non-linear LU-SGS method; the L^2 norm



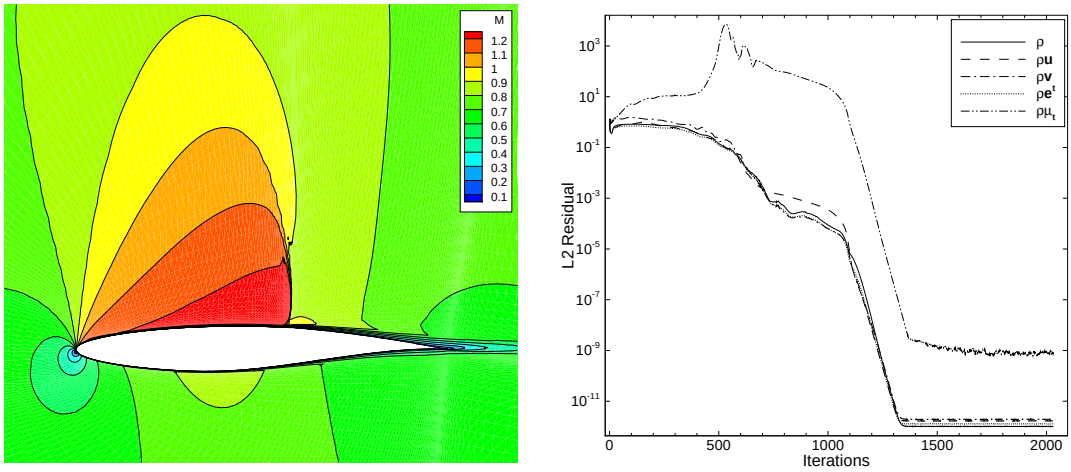
(a) (b)

Figure 6.10: Values of the lift (a) and drag (b) coefficients as function of the number of the degrees of freedom, for the subsonic turbulent flow over the RAE2822 airfoil, with linear and quadratic elements.



(a) (b)

Figure 6.11: Errors with linear and quadratic elements on the lift (a) and drag (b) coefficients for the subsonic turbulent flow over the RAE222 airfoil.



(a) (b)

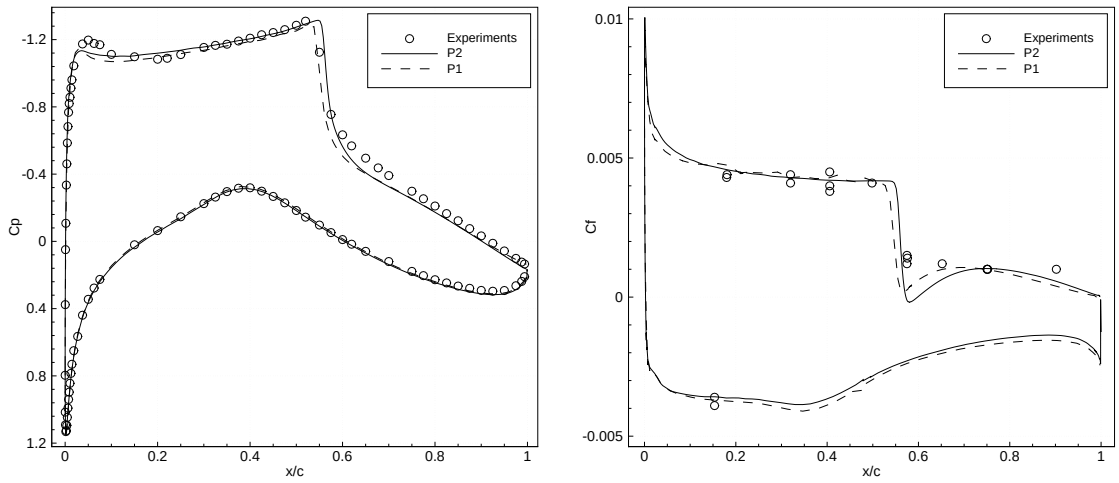
Figure 6.12: Mach number contours (a) and iterative convergence history (b) for the transonic turbulent flow over a RAE2822 airfoil computed with quadratic elements.

of the residual of the mean flow and of the turbulent equation is reduced at least by nine orders of magnitudes.

The same type of grids used in the previous test case is adopted and three levels of refinement are considered. In Figure 6.12 are reported the Mach number contours for the solution on the fine grid (64 768 elements) with quadratic elements, together with the iterative convergence history. Note that the shock is sharply captured, and the number of iterations to reach the steady state is much higher compared to the subsonic test case, due to the presence of the shock and also due to the use of the non-linear scheme.

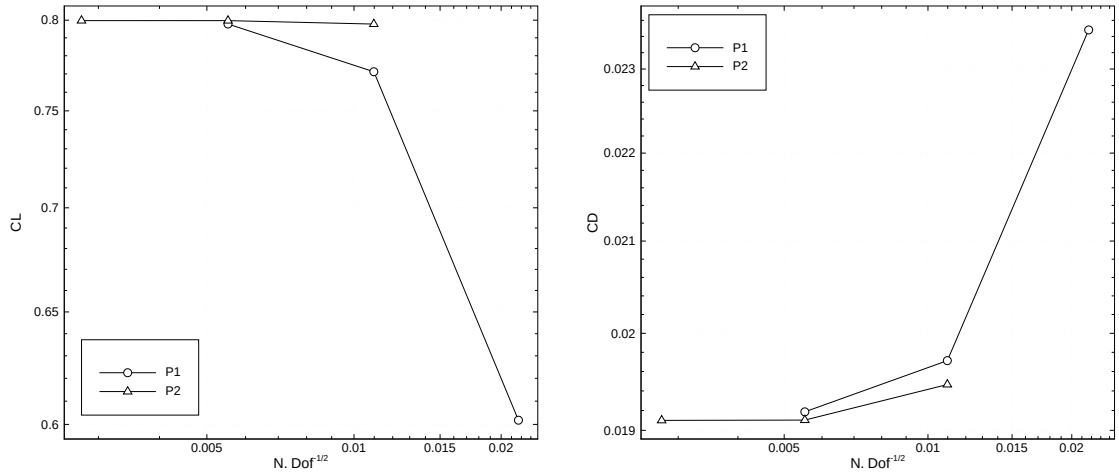
In Figure 6.13 are shown the pressure and the friction coefficients along the airfoil with linear elements on a finer grid and quadratic elements on a coarse grid, such that the number of degrees of freedom in the two cases is the same: 32 784. The values of the computed pressure coefficient agree very well with the experimental data and with the quadratic discretization of the solution, the agreement is even improved. The shock is sharply captured and its position correctly predicted. The leading edge suction peak of the pressure is slightly under-predicted by the numerical simulations, due to the fact that the simulations have been performed in a fully turbulent regime. Even the friction coefficient agrees well with the experimental data. On the upper part of the airfoil, note the negative values of the friction coefficient just behind the shock, obtained with the quadratic approximation of the solution, that identify the separation bubble induced by the shock.

Finally, in Figure 6.14 are reported the values of lift and drag coefficients for linear and quadratic approximation of the solution with three levels of grid refinement; the quadratic solution reaches much faster than the linear one the reference values, for the same number of degrees of freedom.



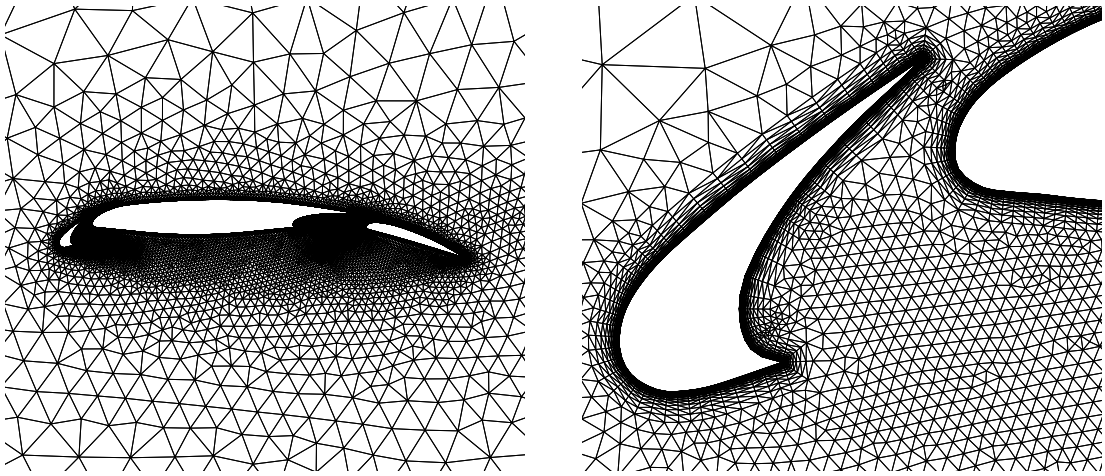
(a) (b)

Figure 6.13: Pressure (a) and friction (b) coefficients over the RAE2822 airfoil, for the transonic turbulent flow computed with linear and quadratic elements on two grids, such that $N_{\text{dof}} = 32\,784$.



(a) (b)

Figure 6.14: Lift (a) and drag (b) coefficients on different grids, for the transonic turbulent flow over the RAE2822 airfoil, with linear and quadratic elements.



(a) (b)

Figure 6.15: Unstructured grid used for the simulation of the turbulent flow over the L1T2 multi-element airfoil, and zoom of the grid near the slat.

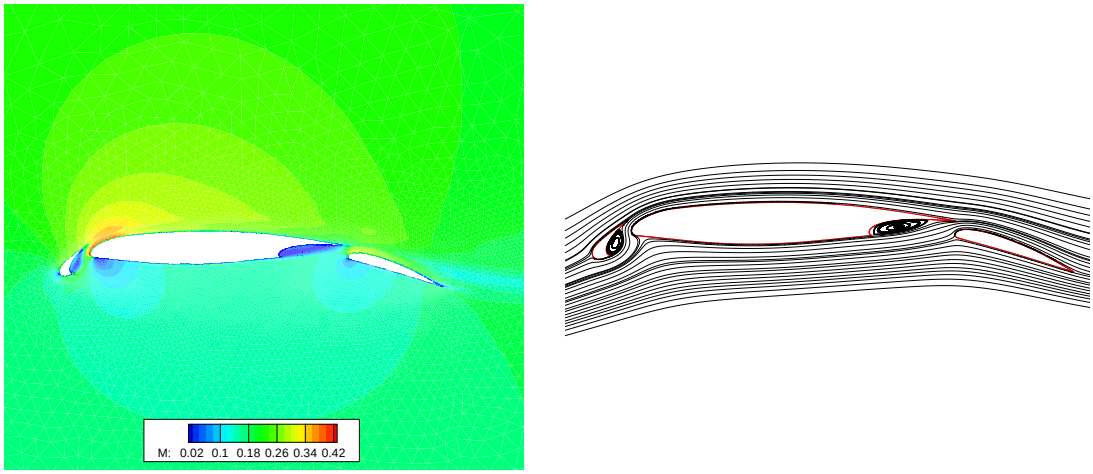
6.5.4 Subsonic Flow over a Multi-Element Airfoil

The flow over the three elements airfoil L1T2 is computed with a free-stream Mach number M = 0.197, angle of attack 4.01° and Reynolds number Re = 3.52×10^6, based on the free stream values and the chord of the retracted airfoil. The main difficulty of this problem is the flow complexity which involves strong interacting wakes. This test case has been computed with the linear scheme, using linear and quadratic approximation of the solution on an unstructured grid of 33 3384 elements, see Figure 6.15. The solution has been converged with the non linear LU-SGS method; the residual of the solution is reduced by ten orders of magnitude with linear elements and eight orders of magnitudes with quadratic elements.

In Figure 6.16 are depicted the Mach number contours for the simulation with quadratic elements and stream lines which shown the strong recirculation region in the flap-cove. In Figure 6.17 are depicted the contours of the turbulent working variable near the slat and the flap of the airfoil. Finally, in Figure 6.18 the values of the pressure coefficient, computed with linear and quadratic elements, are compared against experimental data, showing a good agreement between the numerical solutions and the experimental data.

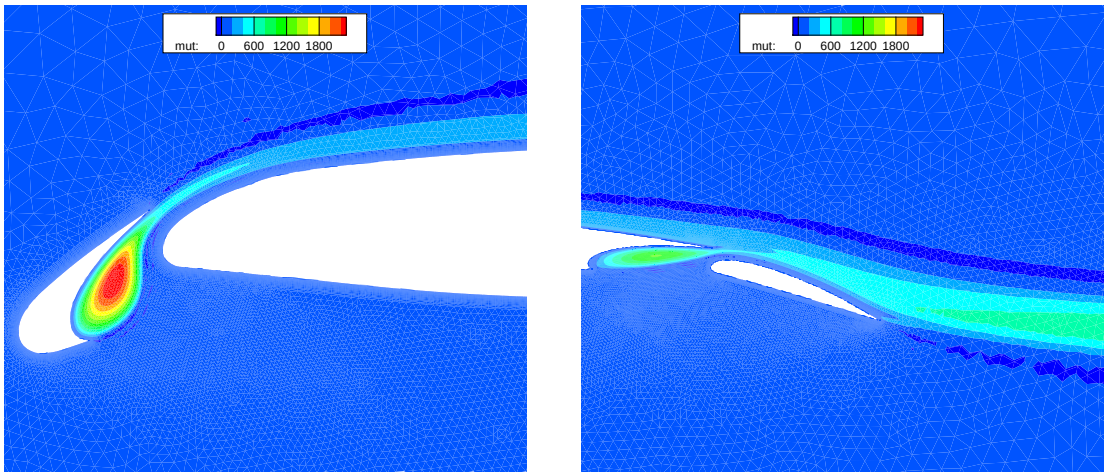
6.5.5 Transonic Flow around the ONERA M6 Wing

As last test case, the simulation of the transonic turbulent flow over the ONERA M6 wing is performed. The free-stream Mach number is M = 0.8395, the angle of incidence is $\alpha = 3.06°$ and the Reynolds number based of the free-stream values and the mean aerodynamic chord is Re = 11.72×10^6. The grid is composed by



(a) (b)

Figure 6.16: Mach number contours (a) and streamlines (b) over the L1T2 multi-element airfoil, for the solution with quadratic elements.

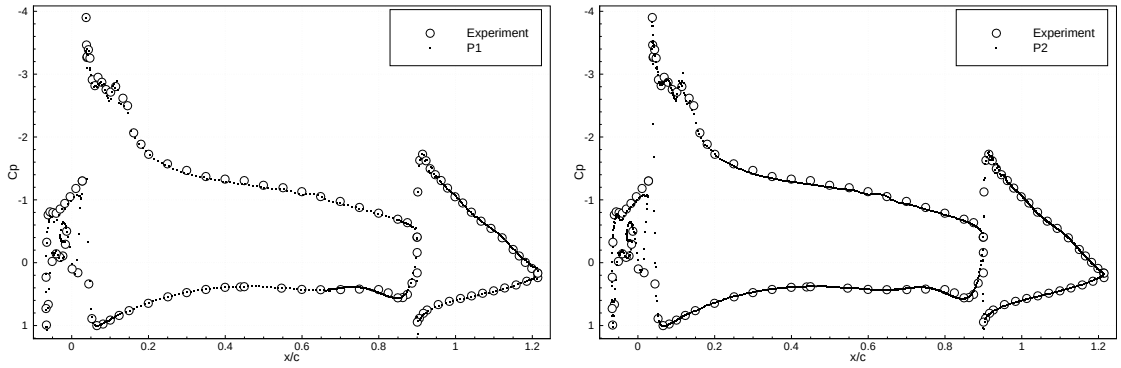


(a) (b)

Figure 6.17: Zoom of the contours of the turbulent working variable near the leading (a) and trailing (b) of the airfoil

123 444 tetrahedra, and the mean height of elements on the wing in dimensionless units is approximately $y_1^+ = 5$. Far-field boundary conditions are applied on the outer part of the domain and the wing surface is modeled as a non-slip adiabatic wall. In Figure 6.19 are reported the grid used for the simulation, and the pressure coefficient contours for the third order simulation.

The non-linear solver has been used to discretize the governing equations with the SPR-ZZ gradient reconstruction method, and the non-linear LU-SGS method has been used to make the scheme converge to the steady state solution. Convergence



(a) (b)

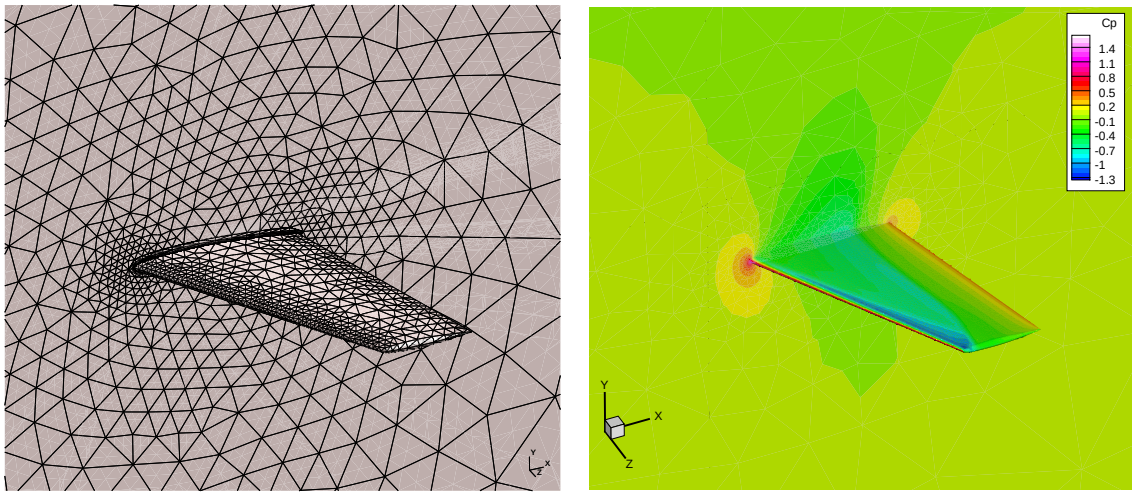
Figure 6.18: Pressure coefficient distribution over the L1T2 multi-element airfoil for the solution with linear (a) and quadratic (b) elements.

is considered achieved when the L^2 residual of all the equations is dropped by eight orders of magnitude. Simulations have been performed in parallel on 64 processors.

In figure Figure 6.20, the values of the pressure coefficient for the third order simulation at different stations along the spanwise direction of the wing are compared against the experimental data. Despite the quite coarse mesh, a good agreement with the experimental data is obtained and even the lambda shock is well represented. However, a finer grid is required to better capture the shock structure at wing tip and also to better resolve the complex shock-boundary layer interaction occurring here. Finally, in Table 6.2 are reported the values of lift and drag coefficients for the second and third order simulations.

	N. DoF	CL	CD
$\mathbb{P}^1$	22 276	0.268231	0.019002
$\mathbb{P}^2$	170 751	0.270758	0.018554

Table 6.2: Lift and drag coefficients for the second and third order turbulent simulations over the M6 wing.



(a) (b)

Figure 6.19: Computational mesh used for the simulation of the turbulent flow over the M6 wing (a) and pressure coefficient contours (b) for the third order simulation.

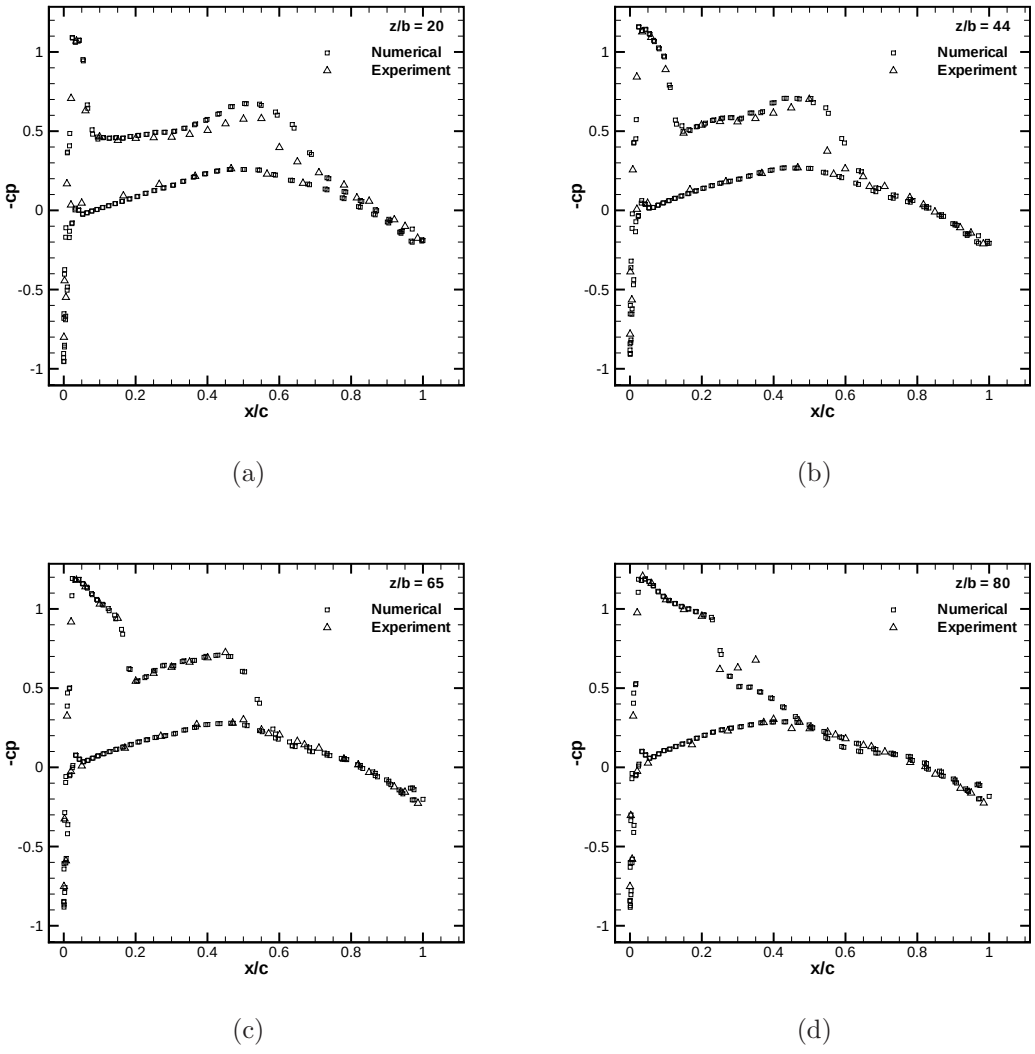


Figure 6.20: Pressure coefficient distribution at different spanwise locations over the M6 wing, for the third order simulation.

Chapter 7

Simulations of Dense Gas Flows

In this section inviscid numerical simulations of dense gas flows are reported. Dense gases are single-phase vapors whose properties deviate significantly from the ideal gas law, operating at temperatures and pressures of the order of magnitude of the critical ones. For a class of dense gases, known as BZT, non-classical gasdynamic behaviors are theoretically predicted: for particular values of pressure and temperature, compression shocks of the perfect gas theory violate the entropy inequality and are therefore inadmissible. Such non-classical phenomena may have some potential advantages in turbomachinery applications due the possibility to reduce losses caused by wave drag and shock/boundary layer interactions. The complexity of performing experimental studies with this class of fluids, motivates the development of accurate numerical solvers which make easier to isolate differences due to the physical modeling from numerical errors.

Here second and third order simulations with the non-linear RD scheme are performed, for two and three dimensional problems, with complex thermodynamic laws. The ability of the numerical solver to preserve the monotonicity of the solutions near strong shocks and the good convergence properties of the scheme for fluids which require more complex model that the ideal gas are verified.

This part of the work has been done in collaboration with P. M. Congedo and N. Razaaly.

7.1 Introduction to Dense Gas Flows

In numerical simulations of compressible flows for standard aerodynamic applications the ideal gas model, despite its simplicity, represents an acceptable approximation. However, the ideal gas approximation cannot be considered accurate when pressures and temperatures are of the order of magnitude of their liquid-vapor saturation curve. This thermodynamic region is generally called dense gas region, in contrast to the dilute gas region where the use of the ideal gas model can be retained

valid.

Fluids in dense regime may display so-called non-classical gas dynamic phenomena. In these cases dense gases may have significantly different properties from dilute gases, from a quantitative and qualitative point of view. For example, for fluids with complex molecules near the saturation curve, at high pressure and temperature, in an isentropic expansion, the speed of sound increases with decreasing density, differently from what happens in the ideal gas model [38]. The opposite trend in the variation of the speed of sound with the density may produce new effects like the impossibility of compression shocks in that region or the suppression of shock-induced separation [139].

The dynamic of dense gases is governed by a thermodynamic parameter known as the fundamental derivative of gasdynamics [122]

$$\Gamma = 1 + \frac{\rho}{c}\left(\frac{\partial c}{\partial \rho}\right)_s, \tag{7.1}$$

where ρ is the density, c is the speed of sound and s is the entropy. For ideal gas $\Gamma = (\gamma + 1)/2 > 1$. For some complex fluids, Γ may be lower that one, implying that $(\partial c/\partial \rho)_s < 0$. This means that the behavior of the speed of sound upon isentropic perturbations is reversed with respect to classical fluids. For a particular class of highly-complex heavy fluids Γ may have negative values in a subset of the dense gas region next to the saturation curve. Such fluids are usually referred to as Bethe-Zel'dovich-Thompson fluids, from the researchers who first postulated their existence; the thermodynamic region characterized by negative values of Γ is called the inversion zone. It has been theoretically shown that, for $\Gamma < 0$, compression waves are smoothed out. As a consequence, compression shocks within the inversion zone violate the entropy inequality, and are therefore inadmissible; conversely, rarefaction shocks are allowed [38, 39, 122, 139].

The interest in BZT fluids is motivated by the potential benefits in the use of such class of fluids in energy applications. For instance, in turbo-machinery flows the shock formation and the consequently loss of energy could be ideally suppressed if the turbine expansion could happen with the inversion zone.

Good candidates for BZT fluids are organic compounds with complex molecular structures. In particular, siloxanes are possibly the most suitable fluids from an engineering point of view, due to their nontoxicity, excellent thermal and chemical stability, and limited flammability. Siloxanes are a class of fluids composed of molecules containing alternate silicon and oxygen atoms in either a linear or cyclic arrangement, usually with one or two organic groups attached to each silicon atom. For example, cyclic molecules D4 (octamethylcyclotetrasiloxane, $C_8H_{24}O_4Si_4$) and D5 (decamethylcyclopentasiloxane, $C_{10}H_{30}O_5Si_5$) may be used for non-classical gasdynamic studies. In Table 7.1 are reported the relevant thermophysical property data of some cyclic siloxanes.

Fluid	MW [kg/kmol]	T_c [K]	P_c [kPa]	v_c [m^3/kmol]
D4	296.62	586.5	1332.0	0.958
D5	370.77	619.2	1160.0	1.267
D6	444.92	645.8	961.0	1.594

Table 7.1: Thermophysical property data for some cyclic siloxanes. MW is the molecular weight, T_c, P_c, and v_c the critical temperature, pressure, and volume, respectively [35].

7.2 Thermodynamic models

Several thermodynamic models, with different levels of complexity, have been proposed in literature. In particular, in this work two equations of state are considered: the cubic Peng-Robinson-Stryjeck-Vera (PRSV) [119] and the 12-parameter technical equation of state proposed by Span and Wagner (SW) [117].

The form of the governing equations is not changed with respect to the ideal gas model formulation. For instance, the form of the Euler equations is exactly the same presented in Eq. (5.1), the only difference is in the expression of the equations of state. Namely, in Eq. (5.3) other relations must be used according the thermodynamic model adopted. For dense gas flows, it is convenient to make the governing equations dimensionless by using the following set of reference quantities

$$l_r = l, \quad \rho_r = \rho_c, \quad P_r = P_c \quad \text{and} \quad T_r = T_c, \tag{7.2}$$

where ρ_c, P_c, T_c are the critical density, pressure and temperature, respectively, and l_r is some characteristic length of the problem.

It is worth noting that the Jacobian matrix of the advective flux function and the relative eigen-structure involve thermodynamic relations, thus the Jacobian matrix and its eigenvectors should be computed for a generic gas, to avoid model-depending implementations, see Appendix C.

7.2.1 PRSV equation

Peng and Robinson (1976) proposed a cubic equation of state of the form:

$$P(T, v) = \frac{\mathcal{R}T}{v - b} - \frac{a}{v^2 + 2bv - b^2}, \tag{7.3}$$

with P the pressure, v specific volume and T the temperature of the fluid, and where a and b are substance-specific parameters related to the fluid critical-point properties (P_c, T_c) and are representative of the attractive and repulsive molecular forces

$$a = 0.457235\frac{\mathcal{R}^2 T_c^2}{P_c}\alpha(T), \quad \text{and} \quad b = 0.077796\frac{\mathcal{R}T_c}{P_c}. \tag{7.4}$$

These parameters are not completely independent, since isothermal lines in the P–v plane should satisfy the thermodynamic stability conditions of zero curvature and zero slope at the critical point. Such conditions allow computing the critical compressibility factor $Z_c = (P_c v_c)/(\mathcal{R}T_c)$ as the solution of a cubic equation. The correction factor α is given by

$$\alpha(T) = \left(1 + K(1 - T_r^{1/2})\right)^2, \tag{7.5}$$

with

$$K = 0.378893 + 1.4897153\omega - 0.17131848\omega^2 + 0.0196554\omega^3, \tag{7.6}$$

where the parameter ω is the fluid acentric factor. The other needed information to complete the thermodynamic model, is approximated through a power law, namely

$$c_{v,\infty}(T) = c_{v,\infty}(T_c)\left(\frac{T}{T_c}\right)^n \tag{7.7}$$

with n a fluid-dependent parameter. The equation of the state for the internal energy, e, is computed by exploiting the compatibility relations, namely

$$e(T, v) = e_r + \int_{T_r}^{T} c_{v,\infty}(T)\mathrm{d}T - \int_{\rho_r}^{\rho} \frac{1}{\rho^2}\left(T\frac{\partial P}{\partial T} - P\right)\mathrm{d}\rho, \tag{7.8}$$

where $c_{v,\infty}$ is the low-pressure, *i.e.* ideal-gas, specific heat at constant volume, and the subscript r indicates a reference state.

7.2.2 SW equation

The state-of-the art multi-parameter thermodynamic model is represented by the Span-Wagner (SW) equation of state. The SW equation is defined as follows

$$\begin{aligned}\psi(\tau,\delta) &= \psi^0(\tau,\delta) + \psi^r(\tau,\delta)\\ &= \psi^0(\tau,\delta) + n_1\delta\tau^{0.25} + n_2\delta\tau^{1.125} + n_3\delta\tau^{1.5} + n_4\delta^2\tau^{1.375} +\\ &\quad n_5\delta^3\tau^{0.25} + n_6\delta^7\tau^{0.875} + n_7\delta^2\tau^{0.625} + n_8\delta^5\tau^{1.75} + n_9\delta^5\tau^{1.75} +\\ &\quad n_{10}\delta^4\tau^{3.625}e^{-\delta^2} + n_{11}\delta^3\tau^{14.5}e^{-\delta^3} + n_{12}\delta^4\tau^{12}e^{-\delta^3},\end{aligned} \tag{7.9}$$

where ψ is the reduced Helmoltz energy (i.e., normalized with $\mathcal{R}T$), ψ^0 is the ideal-gas contribution to the Helmholtz free energy, and the remaining terms represent a real-gas correction, which depend on the reduced density $\delta = \rho/\rho_c$ and on the inverse of the reduced temperature $\tau = T_c/T$. The terms $n_1, \ldots, n_{12}$ are substance-specific coefficients. The SW equation is valid for many classes of non-polar and weakly polar fluids and its parameters have been determined in the literature for different classes of fluids.

For the calculation of caloric properties, the SW model is supplemented by the ideal gas contribution to the specific heat at constant pressure, approximated by a polynomial function of the temperature

$$c_{p,\infty} = c_{v,\infty} + \mathcal{R} = \eta_1 + \eta_2 T + \eta_3 T^2 + \eta_4 T^3, \tag{7.10}$$

where the polynomial coefficients $\eta_1, \ldots, \eta_4$, depend on the substance under consideration. With the previous definition of the specific heat, the internal energy can be computed using Eq. (7.8), while the pressure is given by as following equation

$$\frac{Pv}{\mathcal{R}T} = 1 + \delta \left(\frac{\partial \psi^r}{\partial \delta} \right)_\tau . \tag{7.11}$$

7.3 Numerical Results

7.3.1 Transonic Flow over a NACA0012 Airfoil

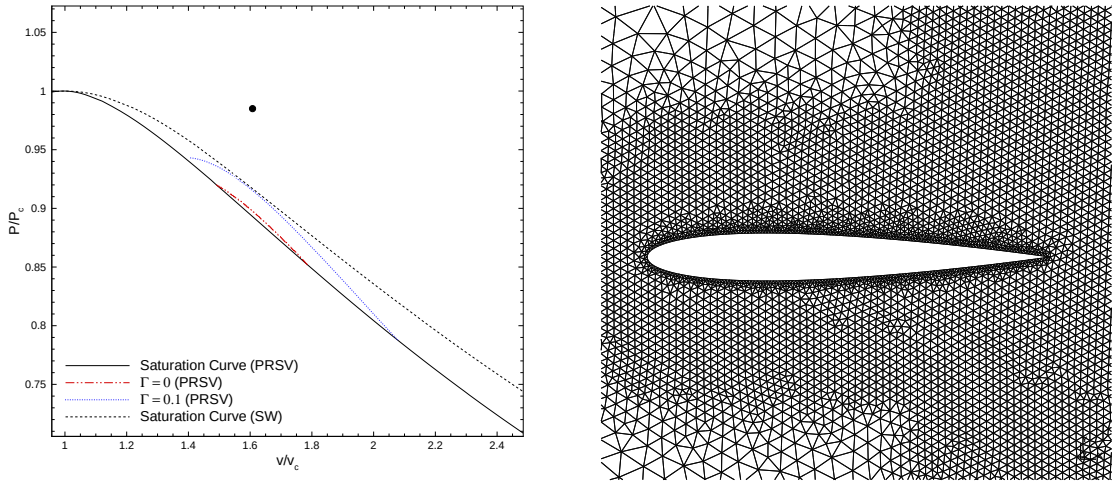
The transonic flow over the NACA-0012 airfoil at zero angle of incidence and free-stream Mach number M = 0.985 is now considered. The D5 siloxane working fluid is used with the PRSV and the SW functional forms, furthermore simulations with the ideal gas ($\gamma = 1.4$) are also performed. For dense gas simulations, the thermodynamic conditions at the free-stream are taken as $P_\infty/P_c = 0.985$ and $v_\infty/v_c = 0.622$, Figure 7.1-a.

In the PRSV model for the D5 siloxane, the acentric factor is $\omega = 0.6658$, the exponent and the ideal-gas specific heat in Eq. (7.7) are $n = 0.5208$ and $c_{v\infty}(T_c) = 76.0\mathcal{R}$, respectively. The parameters of the SW model for the D5 siloxane are listed in Table 7.2

$n_1, \ldots, n_6$	$n_7, \ldots, n_{12}$	$\eta_1, \ldots, \eta_4, \omega$ and Z_c
$n_1 = 1.40844725$	$n_7 = 0.82412481$	$\eta_1 = -34.898$
$n_2 = -2.29248044$	$n_8 = 0.15214274$	$\eta_2 = 1861.5 \times 10^{-3}$
$n_3 = 0.42851607$	$n_9 = -0.68495890$	$\eta 3 = -1403.4 \times 10^{-6}$
$n_4 = -0.73506382$	$n_{10} = -0.55703624 \times 10^{-1}$	$\eta_4 = 500.0 \times 10^{-9}$
$n_5 = 0.16103808$	$n_{11} = 0.13055391 \times 10^{-1}$	$\omega = 0.6658$
$n_6 = 0.29643278 \times 10^{-3}$	$n_{12} = -0.31853761 \times 10^{-1}$	$Z_c = 0.285883427$

Table 7.2: Parameters for the SW equation of state for the D5 siloxane [36].

The simulation are performed on an unstructured grid made of 22 784 triangles, (Figure 7.1-b), with the non-linear RD solver, with linear and quadratic approximation of the solution. The Jacobian-free approach with the LU-SGS preconditioner is used to make the simulations converge to the steady state, which is considered to be reached when the L^2 norm of the density is dropped at least by nine orders of magnitude. Convergence histories are reported in Figure 7.2 for the simulations with the three equations of state, at the second and third order.

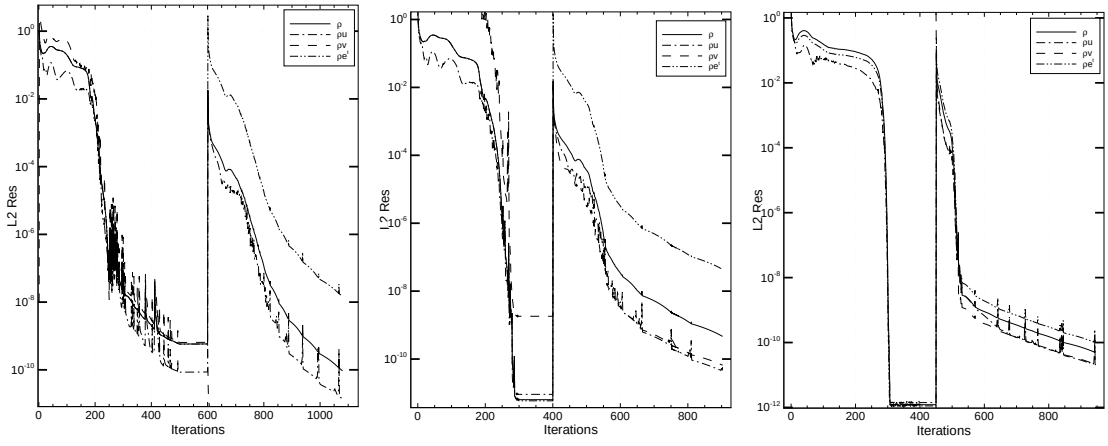


(a) (b)

Figure 7.1: Working point (a) and grid (b) used for the numerical simulations of dense gases flows over the NACA0012 airfoil.

In Figure 7.3, for the three equations of state, are reported the contours of the pressure coefficient and Mach number for the simulations with quadratic approximation of the solution. Note the wide shocks that appear at the trailing edge of the airfoils; the shocks are captured very sharply and within one single element. The profiles of the pressure coefficient over the airfoil are reported in Figure 7.4-a–c for the three equations of state and results with $\mathbb{P}^1$ and $\mathbb{P}^2$ approximation are compared against each other. Note that increasing the order of approximation the shock becomes sharper and remains essentially non oscillatory.

Finally, in Figure 7.4-d the pressure coefficient profiles with $\mathbb{P}^2$ elements are



(a) (b) (c)

Figure 7.2: Convergence histories of the second and third order simulations (order sequencing used) with the PRSV (a), SW (b), and perfect gas (c) models.

reported together for the three equations of state. Note that the shock obtained with the SW model is more intense of that given by the PRSV equations and is located forward, while in the case of ideal gas shock originates from the trailing edged of the airfoil. For completeness, in Table 7.3 are reported the values of the drag coefficients for the three equations of state and with linear and quadratic elements.

	SW	PRSV	PG
$\mathbb{P}^1$	0.111117	0.130794	0.112375
$\mathbb{P}^2$	0.113070	0.133935	0.109838

Table 7.3: Drag coefficients of the NACA0012 airfoil with the D5 fluid, using the PRSV and the SW equations, and for the perfect gas model.

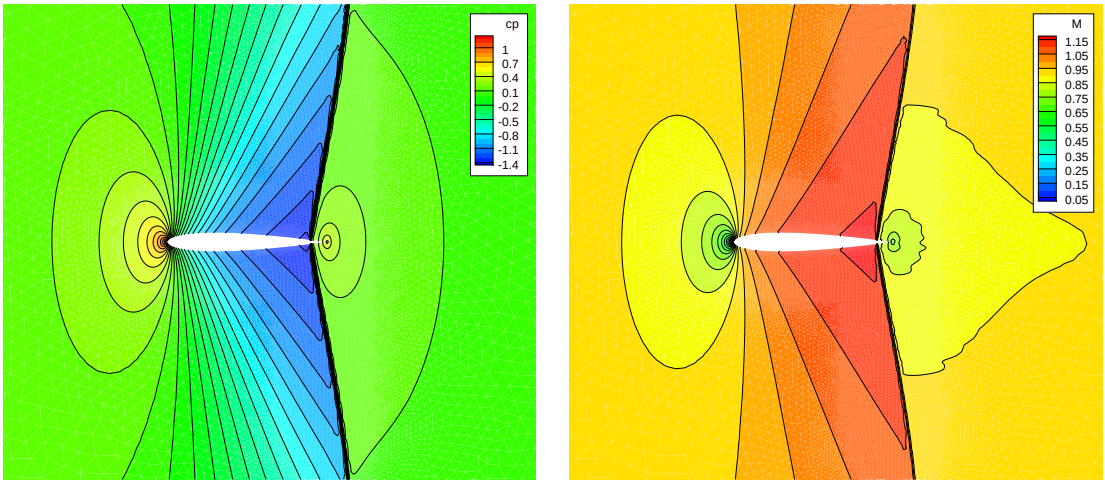
7.3.2 Gas Flows through Turbine Cascades

The flow over a two-dimensional turbine cascade with real gas effects is now considered. The configuration studied is the VKI LS-59 plane cascade of rotor blades, with the PRSV and the SW equations of state. Simulations have been performed on a sequence of three grids uniformly refined. The coarsest grid, made of 2 241 triangles, is shown in Figure 7.5, finer grids are obtained by splitting each triangle with four elements.

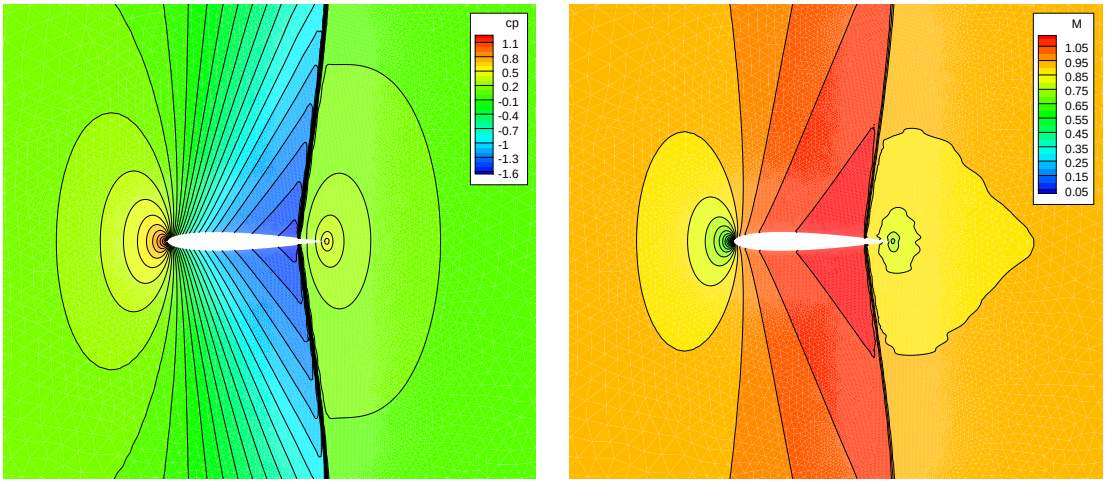
On the left boundary of the domain, inflow boundary conditions are imposed. Since the inflow boundary is subsonic, three quantities must be specified for a two dimensional problem. For turbomachines, it is common practice to impose stagnation thermodynamic conditions and the flow direction. Here, the imposed thermodynamic quantities are the inlet entropy and the inlet stagnation enthalpy. On the left boundary of the domain, outflow boundary conditions are imposed. For a subsonic flow, at the at the outflow boundary, one condition must be imposed, typically the static pressure, while the others are extrapolated from the interior. Here, the velocity components and the entropy are extrapolated. Note that the extrapolation of the entropy requires the solution of the following non-linear system of equations

$$\begin{cases} s = s(\rho, T) \\ P = P(\rho, T) \end{cases} \tag{7.12}$$

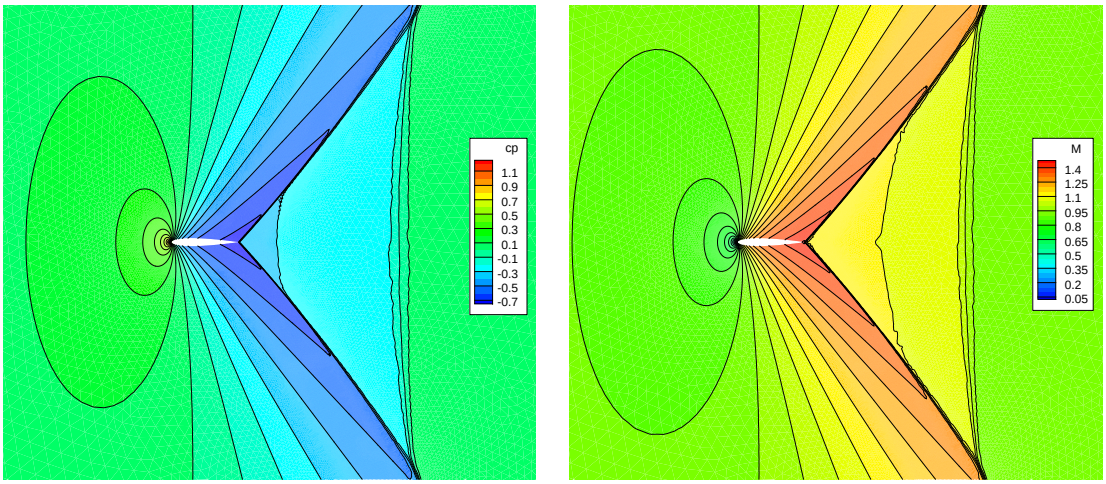
to computed the conservative variables at the boundary. If the density is extrapolated instead of the entropy, the determination of the conservative variables at the boundary is straightforward. In [37], with a FV solver, it was observed that when the density is extrapolated, spurious oscillations of the entropy and static enthalpy appear at the outflow boundary. With the present RD solver such oscillations were not observed, independently from the quantities extrapolated, but the entropy extrapolation is retained for a fair comparison with the results presented in [37]. For



(a) (b)

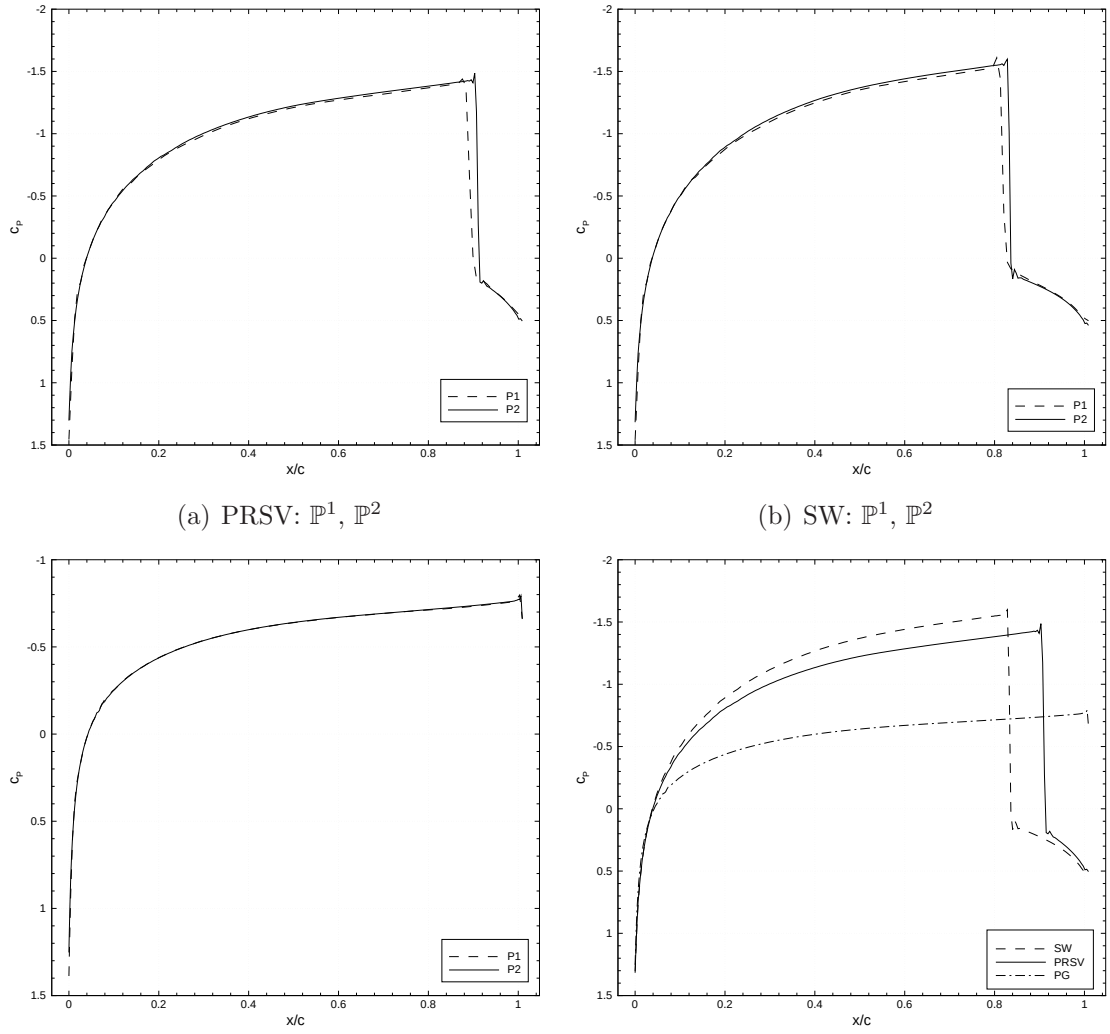


(c) (d)



(e) (f)

Figure 7.3: Contours of the pressure coefficient (left column) and of the Mach number (right) column for the NACA0012 simulation with quadratic elements. First row PRSV gas model for the D5 fluid, second row SW gas model for the D5 fluid, third row perfect gas model.



(a) PRSV: $\mathbb{P}^1$, $\mathbb{P}^2$

(b) SW: $\mathbb{P}^1$, $\mathbb{P}^2$

(c) PG: $\mathbb{P}^1$, $\mathbb{P}^2$

(d) PRSV, SW and PG: $\mathbb{P}^2$

Figure 7.4: Pressure coefficient distribution over the NACA0012 airfoil: (a,b,c) solutions with linear and quadratic elements for the PRSV, SW and perfect gas model, (d) comparison between the different thermodynamic models for the quadratic approximation of the solution.

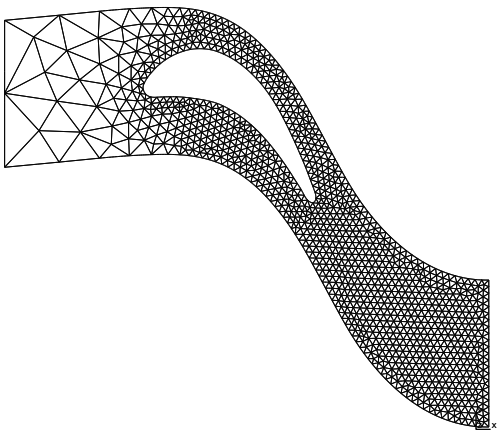


Figure 7.5: Coarsest grid used for the simulations of the turbine cascade.

the blade, no-slip wall boundary conditions are applied, while on the top and bottom boundaries of the domain, periodic boundary conditions are used.

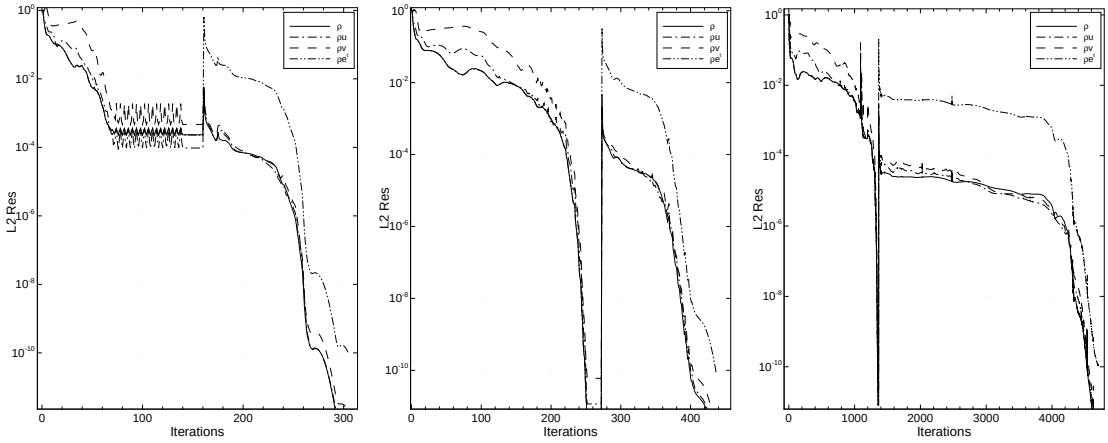
Simulations are performed with the PRSV and the SW equations of state, for the D5 working fluid. In all the cases, the operating condition is chosen such that at the inlet boundary $P/P_c = 0.6217864$ and $\rho/\rho_c = 0.2759043$ and the flow angle is 30°, at the outflow boundary the pressure ratio $P^t_{\text{in}}/P_{\text{out}} = 1.82$ is imposed, where P^t_{in} is the total pressure at the inflow and P_{out} is the static pressure at the outflow.

The Jacobian-free method with the LU-SGS preconditioner is used to make the scheme converge to the steady state, which is considered to be reach when the L^2 norm of the density residual is dropped at least by ten orders of magnitude. In Figure 7.6 and are reported the convergence histories on the three levels of grids for the PRSV model, a similar behavior is obtained with the SW equation of state. Note that the residual of the second order simulation on the coarsest grid could not be reduce more than four orders of magnitude.

In Figure 7.7 are reported the Mach number contours of the third order simulations on the finest grid, for the PRSV equations of state. From the inflow boundary on the left hand side of the domain, the fluid accelerates through the passage formed by the blade cascade to supersonic velocities. At the blade trailing edge, an over-expansion region can be observed and a weak shock appears downstream.

In Figure 7.8 are reported the distributions of the Mach number for the different grids and orders, with the PRSV and SW models. The oscillations of the Mach number observed at the trailing edged are due to the fact that the inviscid flow model cannot correctly capture the separation around the blunt trailing edge of the blade, as also observed in [37]. Increasing the order of approximations allows one to reach faster a grid independent solution than refining the mesh. Furthermore, the discontinuities are better resolved with the high-order approximation than the second order one, for the same number of degrees of freedom.

Finally, in Figure 7.9 are reported the distributions of Mach number and pressure over the blade, for the third order RD solutions on the different grids and the



(a) Coarse (b) Medium (c) Fine

Figure 7.6: Convergence histories of the second and third order simulations (order sequencing used) with the PRSV models for three uniformly refined girds.

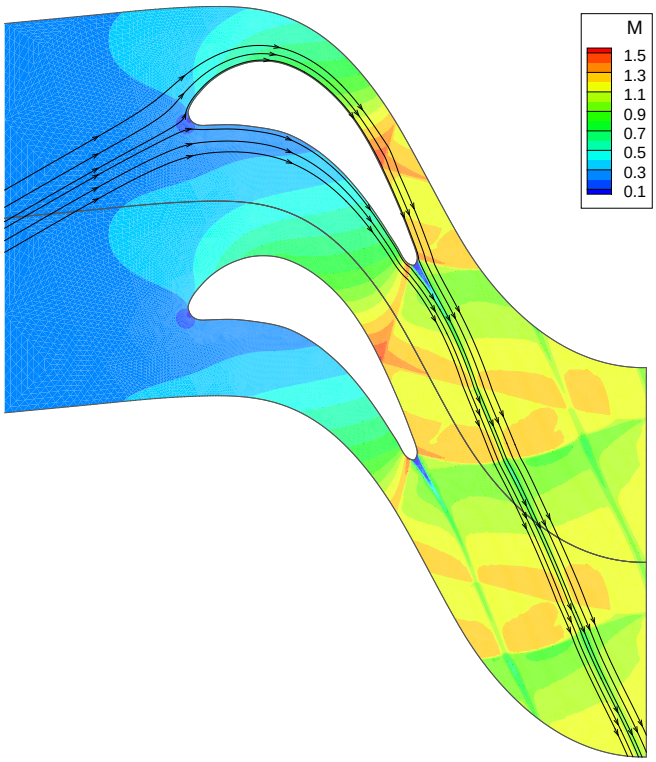


Figure 7.7: Mach number contours over the turbine cascade for the third order simulation on the finest grid. Streamlines are also reported.

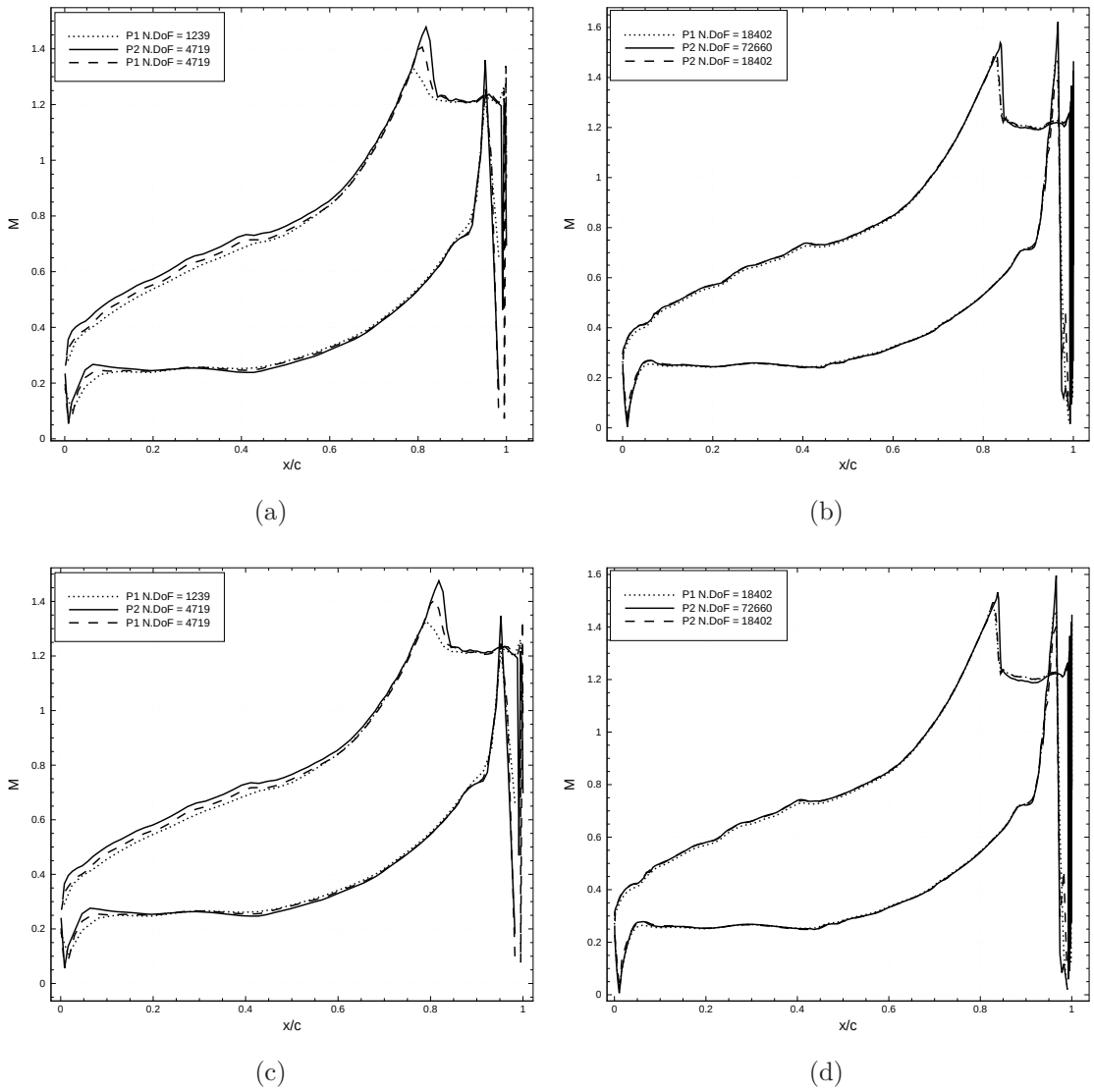
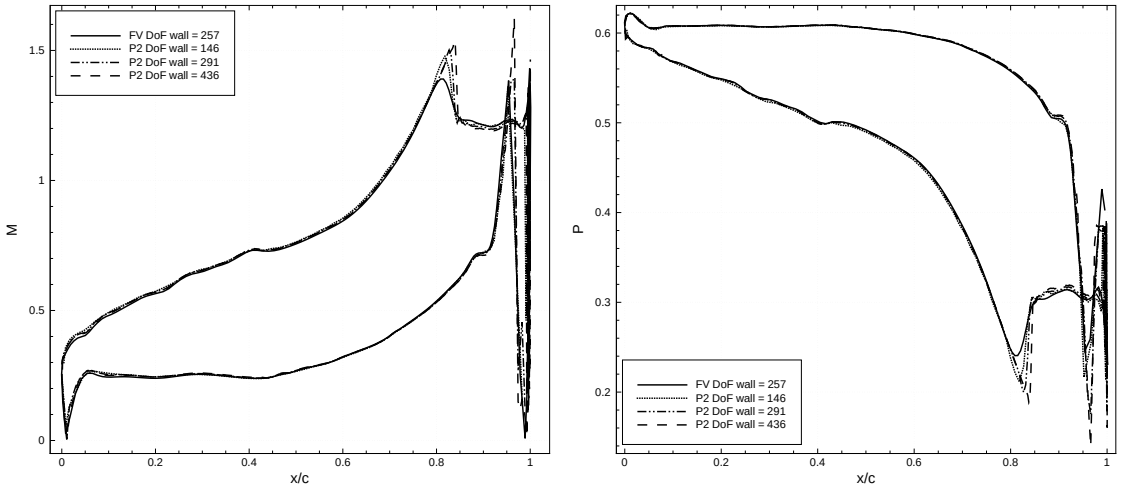


Figure 7.8: Mach number profiles over the blade on different grids with linear and quadratic approximation of the solution. Top row: PRSV, bottom row: SW models.

solutions obtained with the cell-centered FV solver of [37] are reported for comparison. Taking the third order simulation on the finest grid as reference, it can be observed that, for approximately the same number of degrees of freedom along the blade, the third order RD simulation is closer to the reference value with respect to the FV solution. Note also that the shock is always much better resolved with the third order RD scheme.



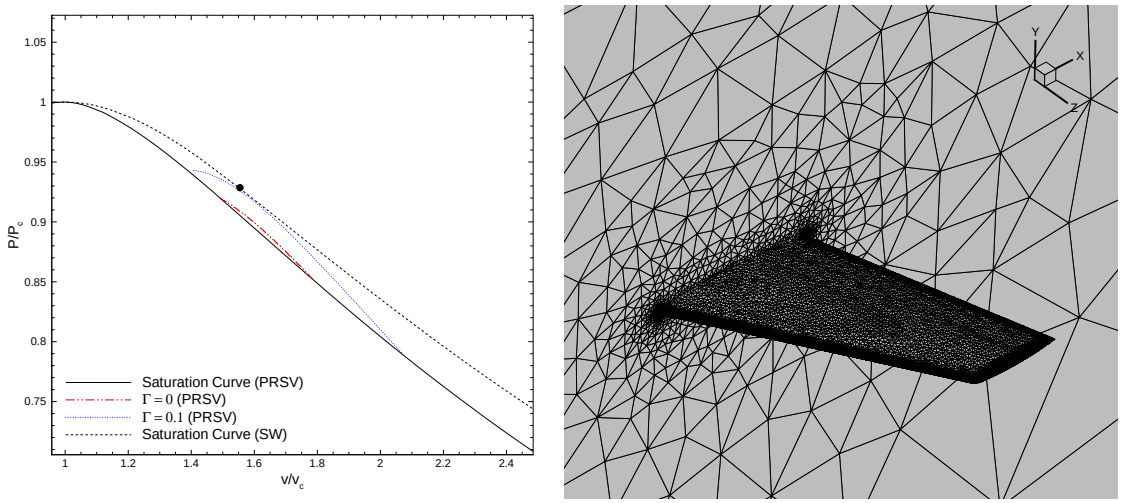
(a) (b)

Figure 7.9: Mach number (a) and pressure (b) distributions along the blade for the third order RD solution and the cell-centered FV solver. In the legend is reported the number of degrees of freedom along the blade of the different simulations.

7.3.3 Transonic Flow over the M6 Wing

The transonic flow over the ONERA M6 swept wing is now considered. The free-stream Mach number and the angle of attack are M = 0.8395 and $\alpha = 3.06°$, respectively, while the yaw angle is zero. The effect of using a real gas model for the simulation over the M6 wing is here considered. Transonic flows of dense gases over finite wings have been studied in reference [32] considering the Martin-Hou equation. Here the Peng-Robinson equation is considered for the D5 working fluid. The free-stream values of density and pressure are taken as $\rho_\infty/\rho_c = 0.6433$ and $P_\infty/P_c = 0.9285$, respectively, for which the free-stream value of the fundamental derivative of gasdynamics is $\Gamma_\infty \approx 0.1$, see Figure 7.10-a.

Results using linear and quadratic interpolation of the solution have been obtained on an unstructured grid of tetrahedra composed by 582 752 elements (108 396 nodes), see Figure 7.10-b. The non-linear RD scheme has been used in combination with the matrix-free method with the LU-SGS preconditioner to make the solution converge to the steady state; in the simulations the L^2 norm of the density residual has been reduce by ten orders of magnitudes.



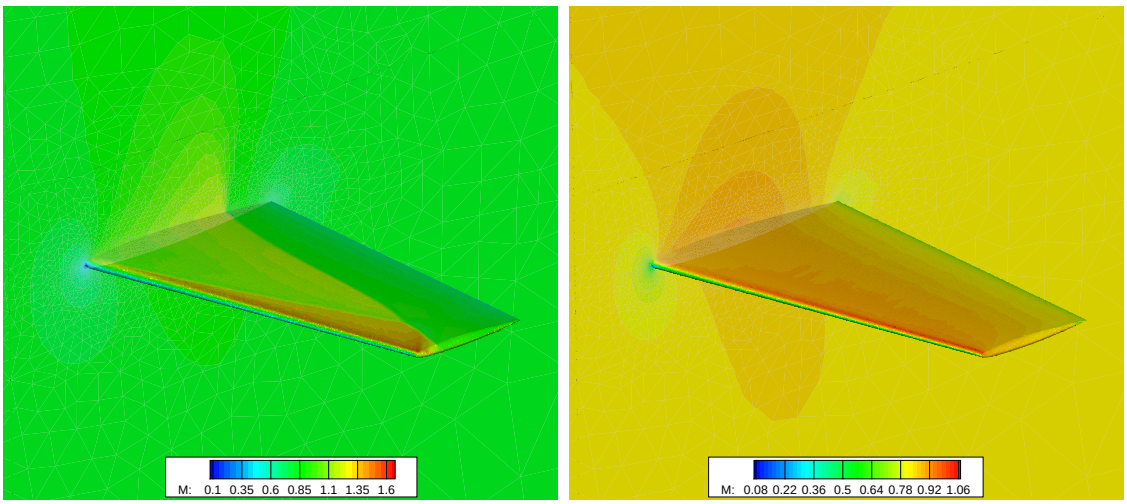
(a) (b)

Figure 7.10: Working point used in the flow simulation over the M6 wing for the Peng-Robinson with the D5 working fluid (a), and grid used for the numerical simulation (b).

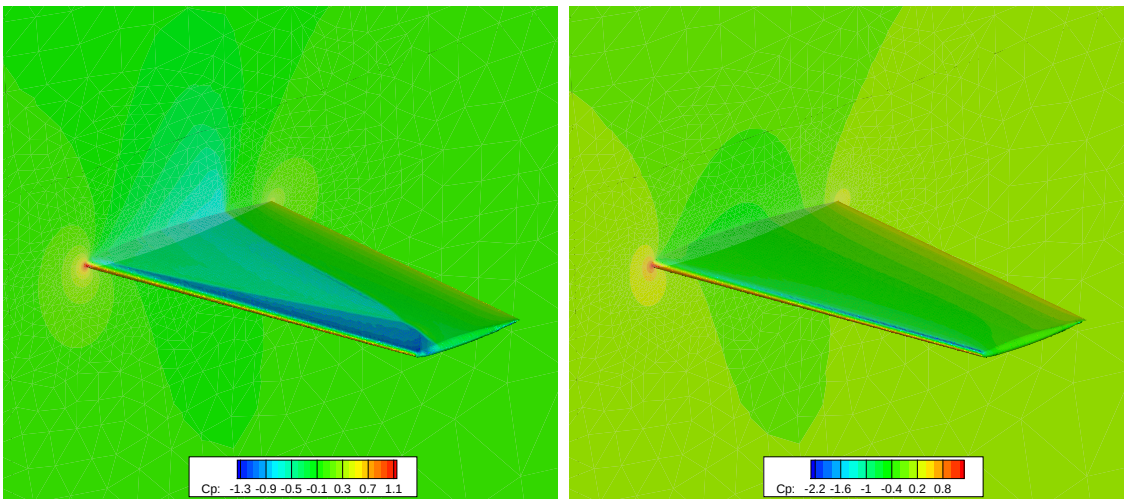
In Figure 7.11 are reported the contours of the Mach number and of pressure coefficient over the wing for the perfect gas and the D5 fluid using the Peng-Robinson equation, for the third order simulations. In the case of the perfect gas, the flow is supersonic at the leading edge of the wing upper surface, with a maximum value of the Mach number about M = 1.6 at the wing tip, the flow becomes subsonic in the rear part of the wing after passing across a λ-shock structure. In the case of the dense gas, the flow reaches the sonic point at the leading edge of the wing upper surface, but the flow over the wing remains shock-free. This behavior can be observed better in Figure 7.12, where the distributions of the pressure coefficient over the wing at different spanwise locations are reported. Note that for the real gas, although the pressure peak at the leading edge is higher, there is no shock formation. The fact that the flow in the case of real gas is shock-free can be justified by the fact that the value of the fundamental derivative of gasdynamics, Γ, remains always bounded between zero and one for the condition chosen, while with perfect gas Γ is constant and always bigger than one. In Figure 7.13 the contours of $\Gamma < 1$ over the wing are reported, and can be observed that, except for a region along the leading edge of the wing, Γ is always smaller than one. For completeness, in Table 7.4 are reported the values of the force coefficients and efficiency obtained with the perfect gas and the Peng-Robinson models.

	C_L	C_D	C_L/C_D
Perfect Gas $\mathbb{P}1$	0.28989	0.018142	15.978
Perfect Gas $\mathbb{P}2$	0.29185	0.011766	24.804
Peng-Robinson $\mathbb{P}1$	0.26448	0.013919	19.002
Peng-Robinson $\mathbb{P}2$	0.26823	0.0067518	39.728

Table 7.4: Values of lift and drag coefficients, and efficiency for the flow over the M6 wing with the perfect gas and D5 fluid using the Peng-Robinson equation.



(a) (b)



(c) (d)

Figure 7.11: Contours of the Mach number (a, b) and of the pressure coefficient (c, d) for the flow over the M6 wing with the perfect gas (a, c) and the Peng-Robinson equation (b, d), for the third order simulations.

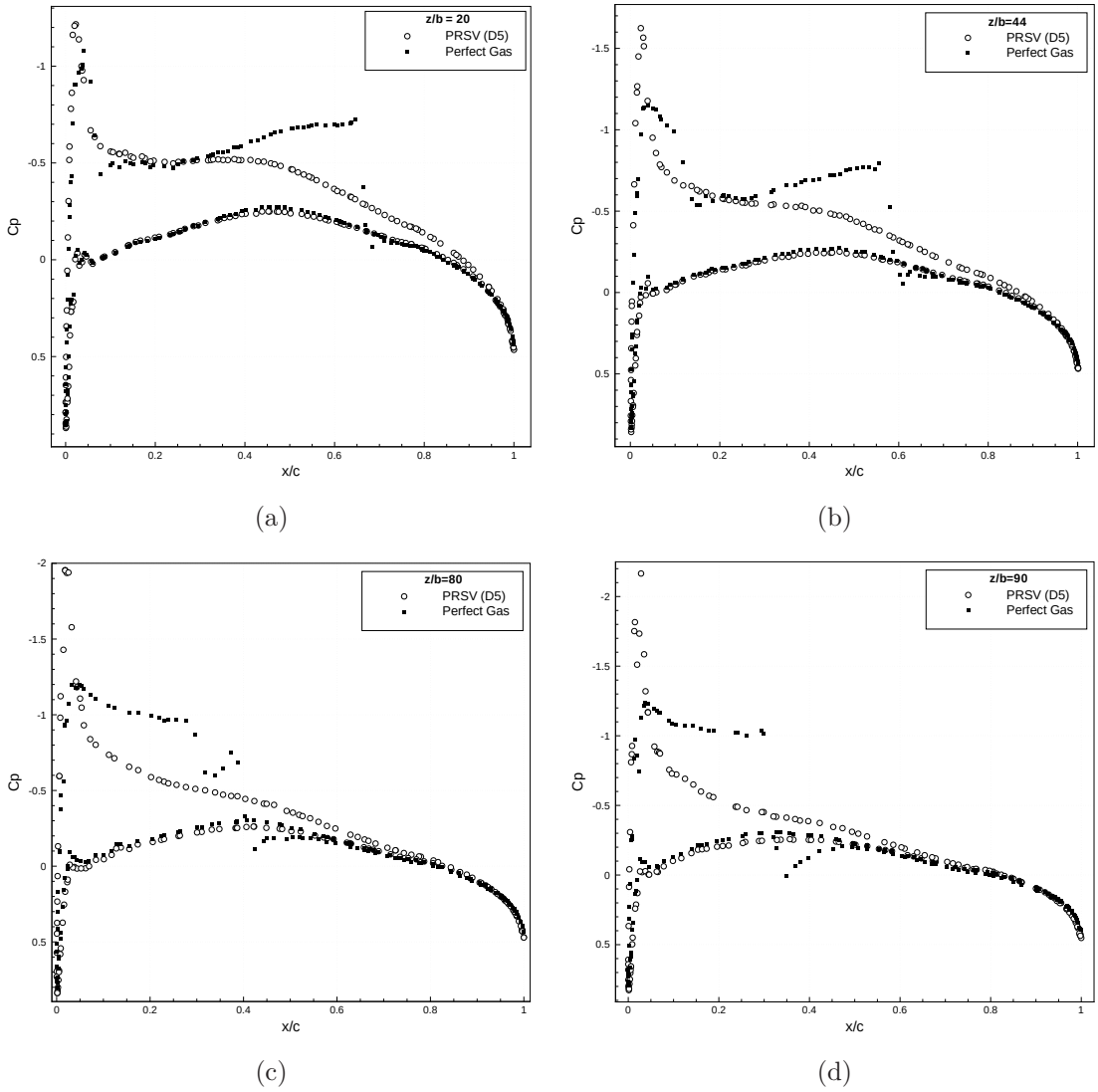


Figure 7.12: Pressure coefficient distribution over the M6 wing at different spanwise locations, with the perfect gas and the Peng-Robinson equation, for the third order simulations.

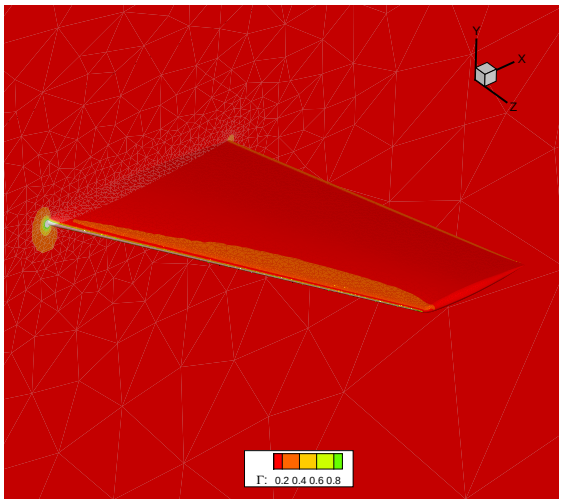


Figure 7.13: Contours of the fundamental derivative of gasdynamics for $0 < \Gamma < 1$ over the M6 wing with the Peng-Robinson equation.

Chapter 8

Conclusions and Perspectives

8.1 Summary

This thesis focuses on the development and the application of Residual Distribution (RD) schemes with second and third order accurate discretization of steady advection-diffusion problems, including scalar equations and the simulation of compressible viscous flows. Several numerical experiments are used to investigate the properties of the proposed approach and its applicability to complex fluid problems, like transonic turbulent flows.

This work has been motivated by the recent need to increase the predictive accuracy of simulations of complex flows over complex geometries or (in case of the same accuracy) to alleviate the computational cost compared to existing numerical schemes. In this respect, high-order methods seem to be potentially superior to Finite Volume (FV) schemes, which require extremely fine grids to compute the solution with a sufficiently small level of error, hence with high computational time and memory usage.

Among high-order schemes, the Discontinuous Galerkin (DG) method has drawn the attention of many researchers over the last decade, due to the capability of this approach to combine high-order discretization with an extreme flexibility. However, these advantages come at the price of a very high computational cost and memory requirement. Furthermore, when shocks are present in the flow field, the robustness of DG schemes may be seriously compromised if the discretization is not properly modified near the discontinuities; namely, some amount of additional dissipation must be added into high-order numerical schemes to avoid spurious oscillations of the solution. This approach may reduced the benefit of using high-order discretization and also introduce problem-dependent tuning parameters.

The RD approach based on the continuous formulation of the problem, introduces less degrees of freedom than DG methods. In addition, with the possibility to construct non-linear RD schemes, the discretization of continuous and discontinuous solutions within the same numerical scheme is straightforward, without the necessity

to add artificial viscosity or empirical shock capturing procedures.

RD methods have been largely applied to inviscid problems, but their use in viscous applications remains limited. In fact, differently from the continuous and discontinuous finite element-based approaches, which are on firm ground in the discretization of both advective and viscous terms, the discretization of advection-diffusion problems with RD schemes is still an issue. Moreover, the classical multidimensional RD approach is limited to simplexes and to the second order of accuracy. While the limitations of the multidimensional schemes can be overcome using high-order central schemes, the extension to viscous problems still represents a barrier that must be leaped over in order to apply RD method to practical problems. For this reason, the work considered first simple scalar advection-diffusion problems, in order to have in hand the strategy for the discretization of more complex problems.

For advection-diffusion problems, coupling RD schemes for advective terms with a Galerkin method for the diffusion should be avoided due to the loss of accuracy when the two schemes are combined. An accurate solution requires that advective and diffusive terms are discretized with the same RD method. In order to compute a total residual which includes both advective and diffusive contributions, a continuous value of the gradient must be reconstructed for each element, due to the fact that the normal component of the gradient of the numerical solution is discontinuous at the faces of the elements. This approach requires that the gradient has to be reconstructed with the same order of accuracy of the solution, otherwise the accuracy of the method is spoiled by the poor approximation of the diffusive terms. On the other hand, reconstruction procedures which involves large stencils should be avoided to preserve the compactness of the numerical scheme.

In this work different gradient reconstruction procedures are tested. Second order of accuracy, for advection-diffusion problems, is obtained with all the approaches, but actual third order discretization is obtained only with the so-called Super-convergent Patch Recovery (SPR) method. The SPR allows to reconstruct the gradient of the numerical solution with the same accuracy of the solution, this guarantees an uniform order of accuracy of the numerical method for all the applications, ranging from the pure diffusive to the pure advective limits, including advection-diffusion problems. The SPR method is as compact as possible since, for each node of the grid, it involves only the neighboring elements that share that node, and it can be used with structured, unstructured and hybrid grids, regardless the number of spatial dimensions, with the same effort of a classical least-square approach. The proposed approach is also more efficient than the First Order System formulation, giving the same level of accuracy with a largely reduced computational cost.

The extension of the proposed RD approach to Navier-Stokes equations is straightforward. Through the use of the manufactured solutions, it has been observed that linear and non-linear RD schemes are able to approximate the governing equations with the optimal order of accuracy when the SPR method is used for the gradient reconstruction. In addition, the gradient of the numerical solution is computed with

the same order of accuracy of the solution, this means that important quantities like the shear and thermal stresses can be computed with a higher level of accuracy compared to other approaches.

The high-order discretization of the Navier-Stokes equations introduces additional complications, like the necessity to have a high-order representation of the boundaries and the need to use implicit solvers.

Differently from low-order methods, high-order schemes are sensible to the boundary discretization of the domain, and if a high-order description of the geometry is not adopted, spurious oscillations of the solution may appear at the boundary. In this work an isoparametric formulation is adopted, this means that solution and geometry are described with the same order of accuracy. Although also a second order RD schemes may benefit from high-order representation of the geometry, the use of isoparametric elements simplifies the code implementation of the scheme. For high-order elements, the nodes on the faces of the elements belonging to the boundary are placed on the real geometry, however some curved elements must be generated also at the interior of the domain in order to have non overlapping elements. The Gmsh library has been used to generate most of the high-order grids used in this work.

Implicit schemes are required to accelerate the convergence of the solution to the steady state. The backward Euler method is generally used for the solution of steady problems, due to its good stability properties. The solution of the resulting non-linear problem with a Newton's method requires the calculation and the inversion of the Jacobian matrix. Since for typical applications the Jacobian matrix is very large and sparse, iterative methods are preferred over direct methods for the matrix inversion. In order to make the non-linear scheme converge, the linearization of the residuals should be as accurate as possible, however an accurate linearization may be very expensive if not impossible, especially for non-linear schemes. In this work a matrix-free approach is used to avoid the computation of the Jacobian matrix. This approach exploits the fact that, when the GMRES approach is used to invert the Jacobian matrix, the matrix-vector products, used to construct the Krylov vectors, are approximated with a finite difference-like method. However, for practical applications a preconditioning matrix is still required, and the LU-SGS method has been used here for this purpose, since it is much cheaper compared to other preconditioners and is still very effective.

The resulting solver has been successfully used to perform simulations of several problems, including a three dimensional flow over a delta wing and the laminar shock-boundary layer interaction. The work shows that from a robustness point of view, high-order RD schemes are able to capture shock waves using high discretization orders and the implicit scheme is able to make the solution converge quickly to the steady state with a reduced memory usage. In terms of accuracy, it has been shown that employing RD schemes with the SPR method for the gradient reconstruction gives an optimal order of accuracy for the solution, and in addition, a super-optimal accuracy for the gradient is obtained with respect to other

schemes. Furthermore, it has been shown that the use of a high-order approximation of the solution allows not only to increase the accuracy respect to low-order schemes, but, for the same number of degrees of freedom, high-order schemes have smaller discretization errors than low-order ones. In addition, the use of non-linear RD schemes shows that a monotone approximation of discontinuous solutions can be obtained without the necessity to add further stabilization or shock capturing terms; boundary layer and shock related phenomena are handled within the same numerical method without any special treatment.

The proposed RD scheme has been used also for the simulations of compressible turbulent flows, at the second and third order of accuracy, with the RANS approach. In this work, the one equation Spalart-Allmaras turbulence model has been adopted as closure model for the eddy viscosity. Despite the fact that the Spalart-Allmaras model has been widely used for aerodynamic applications, it is well known that this model induces spurious oscillations of the turbulent working variables at the edges of the boundary layers and wakes, where the grid is usually not sufficiently resolved, and negative values of the turbulent working variables may appear. Although the eddy viscosity used in the RANS equation is always forced to remain positive, negative values of the turbulent variable in the Spalart-Allmaras equation can compromise the robustness and the iterative convergence of the numerical scheme. This aspect becomes even more critical for high-order methods, since they are less robust than classical FV schemes, and if this issue is not properly addressed, numerical simulations will very likely blow up. In this work, particular effort has been put into the construction of a robust solver for turbulent flows, and some modifications of the Spalart-Allmaras equation introduced in the contest of DG methods have been adopted. In particular, the modifications aim to reduce to zero the source term of the turbulence model when the turbulent working variable becomes negative, preserving at the same time the differentiability of the equation in order to avoid a deterioration of the iterative convergence.

By discretizing the turbulence equation fully-coupled with the RANS equations, the extension of the RD spatial discretization from laminar to turbulent flows is straightforward. However, simulations of turbulent flows have additional difficulties, due to the high stiffness of the problems, caused by high values of the gradient of the solution in combination with stretched grids used in the simulations of turbulent flows, and by the turbulence equation itself. In the construction of an implicit solver for the simulations of turbulent flows, it has been found that the Jacobian-free approach is not as robust as observed for laminar flows, especially when transonic flows are considered. The non-linear LU-SGS method, on the other hand, although more expensive than the Jacobian-free approach, has showed to be very robust for turbulent flows simulations with a good iterative convergence for both the mean flow and turbulence model equations. Several applications have been considered to test the accuracy and the robustness of the numerical scheme for the turbulent flows. It has been found that the solver is able to compute complex flows over complex geometries, with high accuracy and a reasonable level of robustness. In particular, it

has been shown that using a high-order discretization gives more accurate solutions than the low-order one, for the same number of degrees of freedom, even for non smooth solutions, like turbulent flows.

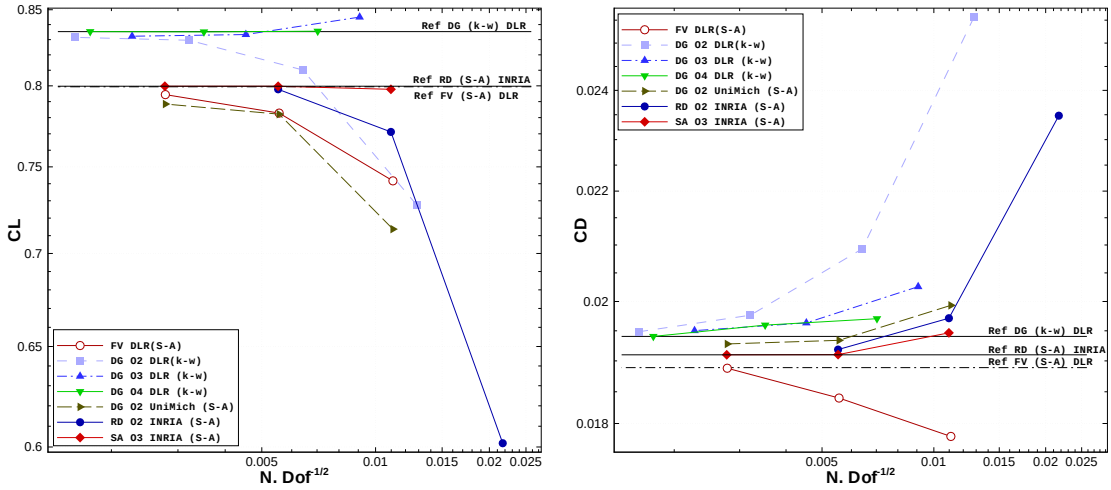
As first step to extend the numerical solver to more complex flow phenomena, dense gas flows have been simulated. For working conditions in which pressures and temperatures are of the order of magnitude of their liquid-vapor critical point values, the ideal gas model is no longer valid and more complex thermodynamic laws have to be considered to take into account real gas effects. Although, for typical aerodynamic applications using the ideal gas model can be retained reasonable, there are several cases for which the work conditions are not in the range of validity of the ideal gas model. These include organic Rankine cycle engine which utilizes high-density working fluids and hypersonic vehicles entering planetary atmospheres. In this work the Peng-Robinson-Stryjeck-Vera and the Span-Wagner equations are used for simulations with the D5 siloxane working fluid, of two and three dimensional inviscid cases. The RD scheme confirmed to be robust and accurate even when more complex thermodynamic phenomena are considered. In particular, in flows with intense shocks, the discontinuities are sharply computed and the monotonicity of the solution is maintained.

8.2 Conclusions

In this work the issue of the discretization of advection-diffusion problems within the RD framework has been addressed. First, the discretization of scalar problems has been considered, and than the resulting schemes have been applied to the discretization, at the second and third order of accuracy, of steady laminar and turbulent compressible flows, as well as inviscid problems.

The work shows that, by using a proper reconstruction procedure for the gradient of the numerical solution, a high-order discretization of the problems with the RD method can be obtained. The proposed approach guarantees that the order of accuracy of the numerical scheme is preserved independently on the relative importance of the advection and diffusion, thus overcoming the problem of loss of accuracy in the advection-diffusion regime observed with blended RD-Galerkin schemes. In addition, the preservation of the monotonicity of the solution over discontinuities has been another important issue addressed in this work. In this respect, RD methods seem to have a great potential respect to other high-order methods, like DG schemes for example, due to the possibility to construct non-linear schemes which avoid the need to introduce artificial viscosity terms for the high-order and robust discretization of discontinuous solutions without spurious oscillations.

The subject of the discretization of the viscous terms leaves a somewhat less definitive conclusion when RD methods are compared to other high-order methods. Although it has been shown that the SPR method for the gradient reconstruction



(a) (b)

Figure 8.1: Convergence of the lift (a) and drag (b) coefficient as function of the number of degrees of freedom, for the transonic turbulent flow over the RAE-2822 airfoil. Contribution of different research groups to the 2nd High-order Workshop. DLR: DG code with the $k-\omega$ turbulence with transition, and FV code with the Spalart-Allmaras turbulence model fully turbulent. INRIA: RD code with the Spalart-Allmaras turbulence model fully turbulent. University of Michigan: DG code with the Spalart-Allmaras turbulence model fully turbulent.

is accurate and robust enough for simulations of complex flows, the reconstruction procedure makes the numerical scheme more sensitive to the quality of the grid. In fact, a degradation of the accuracy of the reconstruction procedure is unavoidable in regions with high curvature and stretched elements, unless the stencil of the reconstruction procedure is enlarged. Thus, the present approach results less appealing respect to continuous or discontinuous finite element methods. On the other hand, fair comparisons between different approaches for complex test cases have been missing. A first attempt to compare high-order methods for complex applications has been done in the 1st and 2nd International Workshop on High-Order CFD Methods [135]. For example, in Figure 8.1 are reported the variations of the lift and drag coefficients with the number of degrees of freedom, for the transonic turbulent flow over the RAE-2822 airfoil (see Section 6.5.3); the results have been produced by different research groups who contributed to this test case at the 2nd High-Order Workshop. Although the same type of grids has been used by all the partners, different turbulence models have been adopted, making difficult to isolate the differences due to the numerical discretization only.

An other important aspect to consider for a numerical scheme is the computational effort, both in terms of CPU time and memory requirement. Continuous approaches introduce less degrees of freedom and have fewer non-zero entries in the Jacobian matrix than discontinuous methods, for the same degree of approximation

of the solution. However, RD schemes suffer of the serious drawback of a poor iterative convergence, due to the fact that the evaluation of an accurate Jacobian matrix remains a very difficult task. For non-linear RD schemes, the exact linearization of the residual is made impossible by the use of the limiting technique (done in the space of the characteristic variables), but also for linear schemes, an accurate linearization cannot be computed due to gradient reconstruction procedure. These difficulties requires approaches that circumvent the explicit calculation of the Jacobian entries, like for example matrix-free or non-linear LU-SGS methods. However, the price to pay for these strategies is an increment of the number of evaluations of the residual of the scheme, with an increment of the computational cost. These approaches remain effective only if the number of non-linear iterations is drastically reduced respect to a matrix-based formulation.

In summary, this work has shown that RD methods can be made robust and efficient for CFD applications, as demonstrated by the application of the numerical method to a variety of flow problems. The approach has potential advantages, but the efficiency and the benefits respect to others current approaches still need to be evaluated.

8.3 Perspectives

There exists a number of directions for future work focusing on the improvement and extension of the methods developed in this thesis. A few ideas for future research in the short term are listed here.

Extension to unsteady problems

Extension of RD methods to unsteady problems has been always a challenging task. Achieving higher order of accuracy in time dependent computations requires the time derivative to be consistently introduced in the element residual. A consistent discretization in space, leads to the appearance of a mass matrix multiplying the time derivative. The existence of a mass matrix introduces additional complexity to the method. First of all, it requires at each time-step the solution of an implicit system of equations, even if an explicit time discretization is used, and lumping the matrix may cause loss in spatial accuracy.

An alternative approach makes use of space-time discretization, however some kind of decoupling in time has to be introduced, otherwise the discrete problem would involve all the degrees of freedom of all the time levels at once. In the classical space-time finite element approach this is achieved by assuming that the numerical solution is discontinuous in time. In a different approach used to decouple the solution in time, for space-time meshes composed of linear space-time elements, the time coordinate is treated as an additional spatial

coordinates and decoupling is achieved naturally thanks to multidimensional upwinding. The levels of the space-time mesh are decoupled under certain constraints for the time-step, (past-shield condition), of the same form of the CFL condition.

All these approach are valid for second order in space and in time, but their extension to higher orders is still unclear.

Improvement of the implicit solver

To improve the efficiency of the numerical scheme it is important to speed up the iterative convergence of the solver. The implicit approaches adopted in this work are quite satisfactory, but further acceleration techniques should be considered to increase the performances of the solver. The first approach that should be used the accelerate the convergence of the implicit solver is based on the line-implicit method. The main idea of the line-implicit solver is to couple together elements connected along directions of strong convection or grid anisotropy. Much faster convergence rate is obtained than an classical implicit schemes since the flow solutions on the lines are solved in a coupled manner, furthermore, the resultant block tri-diagonal system can be solved efficiently. Such approach is expected to be very effective in alleviating the stiffens generated by the use of stretched grids used to resolve the boundary layer. Subsequently, a geometric multigrid algorithm could be used to further accelerate the convergence to the steady state, with a line-implicit method as smoother.

Improvement of the parallel scalability

Although some of the numerical simulations presented here have been performed in parallel, the development of an effective parallel solver has been out of the scope of the present work. Nevertheless, the construction of a highly scalable solver is a very important aspect for practical applications. The first issue that need to be considered is the mesh partition. In the present approach the Scotch [93] library is used to partition the grid considering only the graph associated to degrees of freedom of the linear representation of the solution. The extra degrees of freedom introduced by the high-order representation of the solution are then added to the obtained partitions. The approach is not optimal since non well balanced partitions are obtained and the parallel scalability is compromised. A more efficient approach should considered the partition of the graph associated to the whole set of degrees of freedom. In addition, since the scalability of the solver is also affected by the linear solver and the preconditioners used, the implicit solver should be constructed also in the perspective of a good parallel scalability.

Extension to very higher-order discretization

After the improvement of the parallel performances and efficiency of the implicit scheme, numerical discretization with higher orders (more than three)

could be considered. Although the extension to arbitrary orders is straightforward, in practice the code implementation requires particular care, otherwise the computational cost could become not affordable increasing the polynomial order. Another aspect that should be considered is the decoupling of the order of accuracy of the solution from that of the geometry. Even this aspect does not require theoretical effort, but it is only matter of implementation.

Grid adaptation

Adaptive mesh refinement is a well know strategy for minimizing the cost of a computational simulation while achieving a given level of accuracy. Several adaptation techniques have been developed to refine and de-refine portions of the computational grid. Adaptation criteria based on the residual or the gradient of physical quantities may not be optimal for complex flows, while adjoint-based approaches could be more powerful, as shown for (continuous and discontinuous) finite elements and finite volume methods. In the contest of the RD methods, however, it is not clear if the adjoint-based grid adaptation could effective, due to the non-differentiability of the numerical schemes.

More complex physical models

The extension of the numerical scheme to more complex physical models is a another area of future research. This includes the implementation of more sophisticated turbulence models, like $k-\omega$ for example, or even different approaches for the turbulence modeling, like detached eddy simulation. Another very challenging research field could be the simulations of non-equilibrium hypersonic flows, in this case the use of a non-linear RD method could be advantageous due to the robustness and the monotonicity preserving character of this approach.

Appendix A

Homogeneity Tensor of the Navier-Stokes Viscous Flux Function

The element of the homogeneity tensor of the Navier-Stokes flux function are computed by applying the following definition

$$K_{ijrs}(\mathsf{u}) = \frac{\partial f_{ij}^{v}(\mathsf{u}, \boldsymbol{\nabla}\mathsf{u})}{\partial\left(\dfrac{\partial u_r}{\partial x_s}\right)}$$

with $i, r = 1, \ldots, N_{\text{eq}}$ and $j, s = 1, \ldots, N_{\text{dim}}$ and where f_{ij}^{v} are the components of the viscous flux function for the Navier-Stokes equations.

For sake of clarity, the elements of the tensor $\mathbb{K}$ are grouped in the following matrices:

$$\begin{aligned}
K_{11} &= \mathbb{K}_{\substack{j=1\\s=1}}, \quad K_{12} = \mathbb{K}_{\substack{j=1\\s=2}}, \quad K_{13} = \mathbb{K}_{\substack{j=1\\s=3}} \\
K_{21} &= \mathbb{K}_{\substack{j=2\\s=1}}, \quad K_{22} = \mathbb{K}_{\substack{j=2\\s=2}}, \quad K_{23} = \mathbb{K}_{\substack{j=2\\s=3}} \\
K_{31} &= \mathbb{K}_{\substack{j=3\\s=1}}, \quad K_{32} = \mathbb{K}_{\substack{j=3\\s=2}}, \quad K_{33} = \mathbb{K}_{\substack{j=3\\s=3}}
\end{aligned}$$

The following notation is assumed: $x_1 = x, x_2 = y, x_3 = z$, the velocity vector is $\boldsymbol{v} = (v_x, v_y, v_z)^{\mathrm{T}}$, the momentum is $\boldsymbol{m} = (m_x, m_y, m_z)^{\mathrm{T}}$. The components of the stress tensor are

$$\sigma_{x_i x_j} = \mu\left(\frac{\partial v_{x_i}}{\partial x_j} + \frac{\partial v_{x_j}}{\partial x_i} - \frac{2}{3}\frac{\partial v_{x_k}}{\partial x_k}\delta_{ij}\right),$$

where μ is the viscosity coefficient, δ_{ij} is Kronecker delta and the summations convention on repeated indices is used. The components of the heat flux are

$$q_{x_i} = -\kappa\frac{\partial T}{\partial x_i},$$

with κ the thermal conductivity.

The components of the tensor $\mathbb{K}$ are now explicitly derived for the x-component of the flux function, namely

$$\mathsf{f}_1^v = \begin{pmatrix} 0 \\ \sigma_{xx} \\ \sigma_{xy} \\ \sigma_{xz} \\ \frac{m_x}{\rho}\sigma_{xx} + \frac{m_y}{\rho}\sigma_{xy} + \frac{m_y}{\rho}\sigma_{xz} - q_x \end{pmatrix},$$

such that $f_{i1}^v = K_{i1rs}\dfrac{\partial u_r}{\partial x_s}$, $i, r = 1, \ldots, N_{\text{eq}}$ and $s = 1, \ldots, N_{\text{dim}}$. The elements in the vector f_{i1}^v can be written in terms of the conservative variables as follows

$$\begin{aligned}
f_{11}^v &= 0 \\
f_{21}^v &= \sigma_{xx} = \frac{\mu}{\rho}\left[\frac{4}{3}\left(\frac{\partial m_x}{\partial x} - \frac{m_x}{\rho}\frac{\partial \rho}{\partial x}\right) - \frac{2}{3}\left(\frac{\partial m_y}{\partial y} - \frac{m_y}{\rho}\frac{\partial \rho}{\partial y}\right) - \frac{2}{3}\left(\frac{\partial m_z}{\partial z} - \frac{m_z}{\rho}\frac{\partial \rho}{\partial z}\right)\right] \\
f_{31}^v &= \sigma_{xy} = \frac{\mu}{\rho}\left[\frac{\partial m_x}{\partial y} - \frac{m_x}{\rho}\frac{\partial \rho}{\partial y} + \frac{\partial m_y}{\partial x} - \frac{m_y}{\rho}\frac{\partial \rho}{\partial x}\right] \\
f_{41}^v &= \sigma_{xz} = \frac{\mu}{\rho}\left[\frac{\partial m_x}{\partial z} - \frac{m_x}{\rho}\frac{\partial \rho}{\partial z} + \frac{\partial m_z}{\partial x} - \frac{m_z}{\rho}\frac{\partial \rho}{\partial x}\right]
\end{aligned}$$

$$\begin{aligned}
f_{41}^v &= \frac{m_x}{\rho}\sigma_{xx} + \frac{m_y}{\rho}\sigma_{xy} + \frac{m_y}{\rho}\sigma_{xz} - q_x \\
&= \mu\frac{m_x}{\rho}\left[\frac{4}{3}\left(\frac{\partial m_x}{\partial x} - \frac{m_x}{\rho}\frac{\partial \rho}{\partial x}\right) - \frac{2}{3}\left(\frac{\partial m_y}{\partial y} - \frac{m_y}{\rho}\frac{\partial \rho}{\partial y}\right) - \frac{2}{3}\left(\frac{\partial m_z}{\partial z} - \frac{m_z}{\rho}\frac{\partial \rho}{\partial z}\right)\right] + \\
&\quad \mu\frac{m_y}{\rho}\left[\frac{\partial m_x}{\partial y} - \frac{m_x}{\rho}\frac{\partial \rho}{\partial y} + \frac{\partial m_y}{\partial x} - \frac{m_y}{\rho}\frac{\partial \rho}{\partial x}\right] + \\
&\quad \mu\frac{m_z}{\rho}\left[\frac{\partial m_x}{\partial z} - \frac{m_x}{\rho}\frac{\partial \rho}{\partial z} + \frac{\partial m_z}{\partial x} - \frac{m_z}{\rho}\frac{\partial \rho}{\partial x}\right] + \\
&\quad \kappa\frac{\gamma - 1}{\mathcal{R}\rho}\left[\left(\frac{\partial E^t}{\partial x} - \frac{E^t}{\rho}\frac{\partial \rho}{\partial x}\right) - \right. \\
&\qquad \left.\left(\frac{m_x}{\rho}\frac{\partial m_x}{\partial x} - \frac{m_x^2}{\rho^2}\frac{\partial \rho}{\partial x} + \frac{m_y}{\rho}\frac{\partial m_y}{\partial x} - \frac{m_y^2}{\rho^2}\frac{\partial \rho}{\partial x} + \frac{m_z}{\rho}\frac{\partial m_z}{\partial x} - \frac{m_z^2}{\rho^2}\frac{\partial \rho}{\partial x}\right)\right]
\end{aligned}$$

Regarding at the previous expressions it is easy to see that the matrices K_{11}, K_{12},

and K_{13} read

$$K_{11} = \frac{\mu}{\rho}\begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ -\frac{4}{3}\frac{m_x}{\rho} & \frac{4}{3} & 0 & 0 & 0 \\ -\frac{m_y}{\rho} & 0 & 1 & 0 & 0 \\ -\frac{m_z}{\rho} & 0 & 0 & 1 & 0 \\ K_{5111} & \left(\frac{4}{3}-\frac{\kappa(\gamma-1)}{\mu\mathcal{R}}\right)\frac{m_x}{\rho} & \left(1-\frac{\kappa(\gamma-1)}{\mu\mathcal{R}}\right)\frac{m_y}{\rho} & \left(1-\frac{\kappa(\gamma-1)}{\mu\mathcal{R}}\right)\frac{m_z}{\rho} & \frac{\kappa(\gamma-1)}{\mu\mathcal{R}} \end{pmatrix},$$

with

$$K_{5111} = -\frac{1}{3}\frac{m_x^2}{\rho^2} - \mu\frac{\|\boldsymbol{m}\|^2}{\rho^2} - \frac{\kappa(\gamma-1)}{\mu\mathcal{R}}\left(\frac{E^t}{\rho} - \frac{\|\boldsymbol{m}\|^2}{\rho^2}\right)$$

$$K_{12} = \frac{\mu}{\rho}\begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ -\frac{2}{3}\frac{m_y}{\rho} & 0 & -\frac{2}{3} & 0 & 0 \\ -\frac{m_x}{\rho} & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ -\frac{1}{3}\frac{m_x m_y}{\rho^2} & \frac{m_y}{\rho} & -\frac{2}{3}\frac{m_x}{\rho} & 0 & 0 \end{pmatrix},$$

$$K_{13} = \frac{\mu}{\rho}\begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ \frac{2}{3}\frac{m_z}{\rho} & 0 & 0 & -\frac{2}{3} & 0 \\ 0 & 0 & 0 & 0 & 0 \\ -\frac{m_x}{\rho} & 1 & 0 & 0 & 0 \\ -\frac{1}{3}\frac{m_x m_z}{\rho^2} & \frac{m_z}{\rho} & 0 & -\frac{2}{3}\frac{m_x}{\rho} & 0 \end{pmatrix}.$$

By considering also the derivatives of the viscous flux function along the y and z directions it is possible to obtain the other components of the tensor $\mathbb{K}$, namely

$$K_{21} = \frac{\mu}{\rho}\begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ -\frac{m_y}{\rho} & 0 & 1 & 0 & 0 \\ \frac{2}{3}\frac{m_x}{\rho} & -\frac{2}{3} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ -\frac{1}{3}\frac{m_x m_y}{\rho^2} & -\frac{2}{3}\frac{m_y}{\rho} & \frac{m_x}{\rho} & 0 & 0 \end{pmatrix},$$

$$K_{22} = \frac{\mu}{\rho}\begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ -\frac{m_x}{\rho} & 1 & 0 & 0 & 0 \\ -\frac{4}{3}\frac{m_y}{\rho} & 0 & \frac{4}{3} & 0 & 0 \\ -\frac{m_z}{\rho} & 0 & 0 & 1 & 0 \\ K_{5212} & \left(1-\frac{\kappa(\gamma-1)}{\mu\mathcal{R}}\right)\frac{m_x}{\rho} & \left(\frac{4}{3}-\frac{\kappa(\gamma-1)}{\mu\mathcal{R}}\right)\frac{m_y}{\rho} & \left(1-\frac{\kappa(\gamma-1)}{\mu\mathcal{R}}\right)\frac{m_z}{\rho} & \frac{\kappa(\gamma-1)}{\mu\mathcal{R}} \end{pmatrix},$$

with

$$K_{5212} = -\tfrac{1}{3}\tfrac{m_y^2}{\rho^2} - \mu\tfrac{\|\boldsymbol{m}\|^2}{\rho^2} - \tfrac{\kappa(\gamma-1)}{\mu\mathcal{R}}\left(\tfrac{E^t}{\rho} - \tfrac{\|\boldsymbol{m}\|^2}{\rho^2}\right)$$

$$K_{23} = \frac{\mu}{\rho}\begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ \frac{2}{3}\frac{m_z}{\rho} & 0 & 0 & -\frac{2}{3} & 0 \\ -\frac{m_y}{\rho} & 0 & 1 & 0 & 0 \\ -\frac{1}{3}\frac{m_y m_z}{\rho^2} & 0 & \frac{m_z}{\rho} & -\frac{2}{3}\frac{m_y}{\rho} & 0 \end{pmatrix},$$

$$K_{31} = \frac{\mu}{\rho}\begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ -\frac{m_z}{\rho} & 0 & 0 & \mu & 0 \\ 0 & 0 & 0 & 0 & 0 \\ \frac{2}{3}\frac{m_x}{\rho} & -\frac{2}{3} & 0 & 0 & 0 \\ -\frac{1}{3}\frac{m_x m_z}{\rho^2} & -\frac{2}{3}\frac{m_z}{\rho} & 0 & \frac{m_x}{\rho} & 0 \end{pmatrix},$$

$$K_{32} = \frac{\mu}{\rho}\begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ -\frac{m_z}{\rho} & 0 & 0 & \mu & 0 \\ \frac{2}{3}\frac{m_y}{\rho} & 0 & -\frac{2}{3} & 0 & 0 \\ -\frac{1}{3}\frac{m_y m_z}{\rho^2} & 0 & -\frac{2}{3}\frac{m_z}{\rho} & \frac{m_y}{\rho} & 0 \end{pmatrix},$$

$$K_{33} = \frac{\mu}{\rho}\begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ -\frac{m_x}{\rho} & 1 & 0 & 0 & 0 \\ -\frac{m_y}{\rho} & 0 & 1 & 0 & 0 \\ -\frac{4}{3}\frac{m_z}{\rho} & 0 & 0 & \frac{4}{3} & 0 \\ K_{5313} & \left(1 - \frac{\kappa(\gamma-1)}{\mu\mathcal{R}}\right)\frac{m_x}{\rho} & \left(1 - \frac{\kappa(\gamma-1)}{\mu\mathcal{R}}\right)\frac{m_y}{\rho} & \left(\frac{4}{3} - \frac{\kappa(\gamma-1)}{\mu\mathcal{R}}\right)\frac{m_z}{\rho} & \frac{\kappa(\gamma-1)}{\mu\mathcal{R}} \end{pmatrix},$$

with

$$K_{5313} = -\tfrac{1}{3}\tfrac{m_z^2}{\rho^2} - \mu\tfrac{\|\boldsymbol{m}\|^2}{\rho^2} - \tfrac{\kappa(\gamma-1)}{\mu\mathcal{R}}\left(\tfrac{E^t}{\rho} - \tfrac{\|\boldsymbol{m}\|^2}{\rho^2}\right).$$

The reduction to the two dimensional case is trivial, it requires only to discard the contributions $K_{13}, K_{23}, K_{31}, K_{32}, K_{33}$ and in the remaining matrices the fourth rows and columns must be eliminated.

Appendix B

Structure of the RANS Equations Fully Coupled with the Spalart-Allmaras Model

When the RANS equations are fully coupled with the Spalart-Allmaras turbulence model, the system of the governing equations reads

$$\frac{\partial \mathsf{u}}{\partial t} + \boldsymbol{\nabla} \cdot \mathbf{f}^a(\mathsf{u}) - \boldsymbol{\nabla} \cdot \mathbf{f}^v(\mathsf{u}, \boldsymbol{\nabla}\mathsf{u}) = \mathsf{S}(\mathsf{u}, \boldsymbol{\nabla}\mathsf{u})$$

with

$$\mathsf{u} = \begin{pmatrix} \rho \\ \boldsymbol{m} \\ E^t \\ \mu_t^\star \end{pmatrix}, \qquad \mathbf{f}^a(\mathsf{u}) = \begin{pmatrix} \boldsymbol{m} \\ \dfrac{\boldsymbol{m} \otimes \boldsymbol{m}}{\rho} + P\mathbb{I} \\ \big(E^t + P\big)\dfrac{\boldsymbol{m}}{\rho} \\ \mu_t^\star \dfrac{\boldsymbol{m}}{\rho} \end{pmatrix}, \qquad \mathbf{f}^v(\mathsf{u}, \boldsymbol{\nabla}\mathsf{u}) = \begin{pmatrix} 0 \\ \mathbb{S} \\ \mathbb{S} \cdot \dfrac{\boldsymbol{m}}{\rho} + \kappa \boldsymbol{\nabla} T \\ \dfrac{\eta}{\sigma_a} \boldsymbol{\nabla}\left(\dfrac{\mu_t^\star}{\rho}\right) \end{pmatrix},$$

and $\mathsf{S}(\mathsf{u}, \boldsymbol{\nabla}\mathsf{u}) = (0, \mathbf{0}, 0, S_{SA})^{\mathrm{T}}$, where $\mathbf{f}^a(\mathsf{u})$, $\mathbf{f}^v(\mathsf{u}, \boldsymbol{\nabla}\mathsf{u})$ are the advective and the viscous flux functions respectively and $\mathsf{S}(\mathsf{u}, \boldsymbol{\nabla}\mathsf{u})$ is the source term.

The Jacobian matrix of the advective flux function is defined as $\boldsymbol{A} = \partial \mathbf{f}^a / \partial \mathsf{u}$. Introducing for convenience the vector of the primitive variables density, velocity, enthalpy and turbulent working variable, $\mathsf{w} = (\rho, \boldsymbol{v}, h_t, \nu_t^\star)^{\mathrm{T}}$, the explicit expression of the Jacobian matrix is found by direct differentiation

$$\boldsymbol{A}(\mathsf{u}(\mathsf{w}), \boldsymbol{n}) = \begin{pmatrix} 0 & n_x & n_y & n_z & 0 & 0 \\ (\gamma-1)qn_x - v_x v_n & v_n - (\gamma-2)un_x & un_y - (\gamma-1)v_x n_x & un_z - (\gamma-1)v_z n_x & (\gamma-1)n_x & 0 \\ (\gamma-1)qn_y - v_y v_n & un_x - (\gamma-1)v_x n_y & v_n - (\gamma-2)v_y n_y & un_z - (\gamma-1)v_z n_y & (\gamma-1)n_y & 0 \\ (\gamma-1)qn_z - v_z v_n & wn_x - (\gamma-1)v_x n_y & wn_y - (\gamma-1)v_y n_z & v_n - (\gamma-2)v_z n_z & (\gamma-1)n_y & 0 \\ ((\gamma-1)q - h_t)v_n & h_t n_x - (\gamma-1)v_x v_n & h_t n_y - (\gamma-1)v_y v_n & h_t n_z - (\gamma-1)v_z v_n & \gamma v_n & 0 \\ -v_n \nu_t^\star & \nu_t^\star n_x & \nu_t^\star n_y & \nu_t^\star n_z & 0 & v_n \end{pmatrix},$$

where $h_t = \frac{E^t+P}{\rho}$, $\nu_t^\star = \frac{\mu_t^\star}{\rho}$, $q = \frac{1}{2}\|\boldsymbol{v}\|^2$ and $v_n = \boldsymbol{v}\cdot\boldsymbol{n}$.

The eigenvalues of $\boldsymbol{A}(\mathsf{u},\boldsymbol{n})$ are

$$\lambda_1 = v_n - c, \quad \lambda_2 = v_n, \quad \lambda_3 = v_n + c, \quad \lambda_4 = v_n, \quad \lambda_5 = v_n, \quad \lambda_6 = v_n,$$

with c is the speed of sound. The matrix of the associated right eigenvectors is

$$\mathsf{R}(\mathsf{u}(\mathsf{w}),\boldsymbol{n}) = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ u - cn_x & u & u + cn_x & n_y & -n_z & 0 \\ v - cn_y & v & v + cn_y & -n_x & 0 & 0 \\ w - cn_z & w & w + cn_z & 0 & n_x & 0 \\ h_t - cv_n & q & h_t + cv_n & un_y - vn_x & wn_x - un_z & 0 \\ \nu_t^\star & 0 & \nu_t^\star & 0 & 0 & 1 \end{pmatrix}.$$

and the matrix of the left eigenvectors is given by

$$\mathsf{L}(\mathsf{u}(\mathsf{w}),\boldsymbol{n}) = \frac{1}{2c^2}\begin{pmatrix} (\gamma-1)q+cv_n & (1-\gamma)u-cn_x & (1-\gamma)v-cn_y & (1-\gamma)w-cn_z & \gamma-1 & 0 \\ 2((1-\gamma)q+c^2) & 2(\gamma-1)u & 2(\gamma-1)v & 2(\gamma-1)w & 2(1-\gamma) & 0 \\ (\gamma-1)q-cv_n & (1-\gamma)u-cn_x & (1-\gamma)v+cn_y & (1-\gamma)w+cn_y & 1-\gamma & 0 \\ 2c^2\frac{(v-v_n n_y)}{n_x} & 2c^2 n_y & 2c^2\frac{n_y^2-1}{n_x} & 2c^2\frac{n_y n_z}{n_x} & 0 & 0 \\ 2c^2\frac{(v_n n_z-w)}{n_x} & -2c^2 n_z & -2c^2\frac{n_y n_z}{n_x} & -2c^2\frac{1-n_z^2}{n_x} & 0 & 0 \\ 2q(1-\gamma)\nu_t^\star & 2(\gamma-1)\nu_t^\star u & 2(\gamma-1)\nu_t^\star v & 2(\gamma-1)\nu_t^\star w & 2(1-\gamma)\nu_t^\star & 2c^2 \end{pmatrix},$$

such that $\mathsf{RL} = \mathsf{LR} = \mathbb{I}$.

In the matrix of the left eigenvectors there are singular terms when $n_x = 0$ (as in the Euler equations), however the singularity can be avoided remembering that the eigenvectors of repeated eigenvalues λ_4 and λ_5 are not distinct, thus any linear combination of the 4th and 5th column of the matrix R is itself an eigenvector. A new set of right eigenvectors and the corresponding left eigenvector can be written to avoid the singularity [105]. Note that in the case of two spatial dimensional, the are no singular terms.

The homogeneity tensor of the viscous flux function can be easily calculated from its definition, the matrices K_{11}, K_{22} and K_{33} reads

$$K_{11} = \frac{\mu}{\rho}\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{4}{3}\frac{m_x}{\rho} & \frac{4}{3} & 0 & 0 & 0 & 0 \\ -\frac{m_y}{\rho} & 0 & 1 & 0 & 0 & 0 \\ -\frac{m_z}{\rho} & 0 & 0 & 1 & 0 & 0 \\ K_{5111} & \left(\frac{4}{3}-\frac{\kappa(\gamma-1)}{\mu\mathcal{R}}\right)\frac{m_x}{\rho} & \left(1-\frac{\kappa(\gamma-1)}{\mu\mathcal{R}}\right)\frac{m_y}{\rho} & \left(1-\frac{\kappa(\gamma-1)}{\mu\mathcal{R}}\right)\frac{m_z}{\rho} & \frac{\kappa(\gamma-1)}{\mu\mathcal{R}} & 0 \\ -\frac{\eta}{\rho^2\sigma_{SA}}\mu_t^\star & 0 & 0 & 0 & 0 & \frac{\eta}{\rho\sigma_{SA}} \end{pmatrix},$$

$$K_{22} = \frac{\mu}{\rho}\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{m_x}{\rho} & 1 & 0 & 0 & 0 & 0 \\ -\frac{4}{3}\frac{m_y}{\rho} & 0 & \frac{4}{3} & 0 & 0 & 0 \\ -\frac{m_z}{\rho} & 0 & 0 & 1 & 0 & 0 \\ K_{5212} & \left(1-\frac{\kappa(\gamma-1)}{\mu\mathcal{R}}\right)\frac{m_x}{\rho} & \left(\frac{4}{3}-\frac{\kappa(\gamma-1)}{\mu\mathcal{R}}\right)\frac{m_y}{\rho} & \left(1-\frac{\kappa(\gamma-1)}{\mu\mathcal{R}}\right)\frac{m_z}{\rho} & \frac{\kappa(\gamma-1)}{\mu\mathcal{R}} & 0 \\ -\frac{\eta}{\rho^2\sigma_{SA}}\mu_t^\star & 0 & 0 & 0 & 0 & \frac{\eta}{\rho\sigma_{SA}} \end{pmatrix},$$

$$K_{33} = \frac{\mu}{\rho}\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{m_x}{\rho} & 1 & 0 & 0 & 0 & 0 \\ -\frac{m_y}{\rho} & 0 & 1 & 0 & 0 & 0 \\ -\frac{4}{3}\frac{m_z}{\rho} & 0 & 0 & \frac{4}{3} & 0 & 0 \\ K_{5313} & \left(1-\frac{\kappa(\gamma-1)}{\mu\mathcal{R}}\right)\frac{m_x}{\rho} & \left(1-\frac{\kappa(\gamma-1)}{\mu\mathcal{R}}\right)\frac{m_y}{\rho} & \left(\frac{4}{3}-\frac{\kappa(\gamma-1)}{\mu\mathcal{R}}\right)\frac{m_z}{\rho} & \frac{\kappa(\gamma-1)}{\mu\mathcal{R}} & 0 \\ -\frac{\eta}{\rho^2\sigma_{SA}}\mu_t^\star & 0 & 0 & 0 & 0 & \frac{\eta}{\rho\sigma_{SA}} \end{pmatrix},$$

while the remaining matrices are simple obtained by framing the right and the bottom parts of the matrices computed for the Navier-Stokes equations with a row and a column of zeros.

Appendix C

Jacobian Matrix and Eigenstructure of the Euler Equations for a Generic Gas

In this appendix the Jacobian matrix and the eigen-structure of the Euler flux function for a generic gas are reported. The Jacobian matrix reads

$$\boldsymbol{A}(\mathsf{u}(\mathsf{w}),\boldsymbol{n}) = \begin{pmatrix} 0 & n_x & n_y & n_z & 0 \\ -v_x v_n + n_x P_{\partial\rho} & v_n + v_x(1-P_{\partial E^t})n_x & v_x n_y - v_y n_x P_{\partial E^t} & v_x n_z - v_z n_x P_{\partial E^t} & n_x P_{\partial E^t} \\ -v_y v_n + n_y P_{\partial\rho} & v_y n_x - v_x n_y P_{\partial E^t} & v_n + v_y(1-P_{\partial E^t})n_y & v_y n_z - v_z n_y P_{\partial E^t} & n_y P_{\partial E^t} \\ -v_z v_n + n_z P_{\partial\rho} & v_z n_x - v_x n_z P_{\partial E^t} & v_z n_y - v_y n_z P_{\partial E^t} & v_n + v_z(1-P_{\partial E^t})n_z & n_z P_{\partial E^t} \\ -v_n(h^t - P_{\partial\rho}) & h^t n_x - v_x v_n P_{\partial E^t} & h^t n_y - v_y v_n P_{\partial E^t} & h^t n_z - v_z v_n P_{\partial E^t} & v_n(1+P_{\partial E^t}) \end{pmatrix},$$

where the notation $P_{\partial(.)} = \partial P/\partial(.)$ has been used. The matrices of the right and left eigenvectors read respectively

$$\mathsf{R}(\mathsf{u}(\mathsf{w}),\boldsymbol{n}) = \begin{pmatrix} 1 & n_x & 1 & n_y & n_z \\ v_x - c n_x & v_x n_x & v_x + c n_x & v_x n_y - n_z & v_x n_z + n_y \\ v_y - c n_y & v_y n_x + n_z & v_y + c n_y & v n_y & v_y n_z - n_x \\ v_z - c n_z & v_z n_x - n_y & v_z + c n_z & v_z n_y + n_x & v_z n_z \\ h^t - c v_n & \xi n_x + v_y n_z - v_z n_y & h^t + c v_n & \xi n_y + v_z n_x - v_x n_z & \xi n_z + v_x n_y - v_y n_x \end{pmatrix}.$$

$$\mathsf{L}(\mathsf{u}(\mathsf{w}),\boldsymbol{n}) = \begin{pmatrix} \frac{1}{2c^2}(P_{\partial\rho} + c v_n) & -\frac{1}{2}v_x(P_{\partial E^t} + c n_x) & -\frac{1}{2}v_y(P_{\partial E^t} + c n_y) & -\frac{1}{2}v_z(P_{\partial E^t} + c n_z) & \frac{1}{2}P_{\rho E^t} \\ -\Phi n_x - c^2(v n_z - w n_y) & v_x n_x P_{\partial E^t} & v_y n_x P_{\partial E^t} + c^2 n_z & w n_x P_{\partial E^t} - c^2 n_y & -n_x P_{\partial E^t} \\ \frac{1}{2}P_{\partial\rho} - c v_n & \frac{1}{2}(-v_x P_{\partial E^t} + c n_x) & \frac{1}{2}(-v_y P_{\partial E^t} + c n_y) & \frac{1}{2}(-v_z P_{\partial E^t} + c n_z) & \frac{1}{2}P_{\partial E^t} \\ -\Phi n_y - c^2(v_z n_x - v_x n_z) & v_x n_y P_{\partial E^t} - c^2 n_z & v_y n_y P_{\partial E^t} & v_z n_y P_{\partial E^t} + c^2 n_x & -n_y P_{\partial E^t} \\ -\Phi n_z - c^2(u n_y - v n_x) & v_x n_z P_{\partial E^t} + c^2 n_y & v_y n_z P_{\partial E^t} - c^2 n_x & v_z n_z P_{\partial E^t} & -n_z n_z P_{\partial E^t} \end{pmatrix}.$$

where the following auxiliary variables have been introduced

$$\Phi = P_{\partial\rho} - c^2 \quad \text{and} \quad \xi = \|\boldsymbol{v}\|^2 - P_{\partial\rho}/P_{\partial E^t}.$$

The corresponding eigenvalues of the Jacobian matrix, $\boldsymbol{A}$, are

$$\lambda_1 = v_n - c, \quad \lambda_2 = v_n, \quad \lambda_3 = v_n, \quad \lambda_4 = v_n, \quad \lambda_5 = v_n + c.$$

Bibliography

[1] R. Abgrall. On essentially non-oscillatory schemes on unstructured meshes: analysis and implementation. *Journal of Computational Physics*, 114:45–58, 1994.

[2] R. Abgrall. Toward the ultimate conservative scheme: Following the quest. *Journal of Computational Physics*, 167(2):277 – 315, 2001.

[3] R. Abgrall. Essentially non-oscillatory residual distribution schemes for hyperbolic problems. *Journal of Computational Physics*, 214:773–808, 2006.

[4] R. Abgrall and T. Barth. Residual distribution schemes for conservation laws via adaptive quadrature. *SIAM Journal on Scientific Computing*, 24:719–731, 2003.

[5] R. Abgrall, G. Baurin, A. Krust, D. De Santis, and M. Ricchiuto. Numerical approximation of parabolic problems by means of residual distribution schemes. *International Journal for Numerical Methods in Fluids*, 71:1191–1206, 2013.

[6] R. Abgrall and D. De Santis. High order residual distribution scheme for Navier-Stokes equations. In *20th AIAA Computational Fluid Dynamics Conference*. AIAA 2011-3231, 2011.

[7] R. Abgrall, C. Dobrzynski, and A. Froehly. A method for computing curved 2d and 3d meshes via the linear elasticity analogy: preliminary results. Technical Report RR-8061, INRIA, 2012.

[8] R. Abgrall, A. Larat, and M. Ricchiuto. Construction of very high order residual distribution schemes for steady inviscid flow problems on hybrid unstructured meshes. *Journal of Computational Physics*, 230:4103–4136, 2011.

[9] R. Abgrall and K. Mer. A Lax-Wendroff type theorem for distribution schemes. Technical Report 98010, INRIA, Le Chesnay, France, 1999.

[10] R. Abgrall and M. Mezine. Construction of second-order accurate monotone and stable residual distribution schemes for unsteady problem. *Journal of Computational Physics*, 195:474–507, 2004.

[11] R. Abgrall and P. L. Roe. High-order fluctuation schemes on triangular meshes. *Journal of Scientific Computing*, 19:3–36, 2003.

[12] R. Abgrall. and J Trefilík. An example of high order residual distribution scheme using non-lagrange elements. *Journal of Scientific Computing*, 45:3–25, 2010.

[13] Y. Allaneau and A. Jameson. Connections between the filtered discontinuous Galerkin method and the flux reconstruction approach to high order discretization. *Computer Methods in Applied Mechanics and Engineering*, 200:3628–3636, 2011.

[14] S. R. Allmaras, F. T. Johnson, and P. R. Spalart. Modifications and clarifications for the implementation of the spalart–allmaras turbulence model. In *Seventh International Conference on Computational Fluid Dynamics*. ICCFD7-1902, 2012.

[15] W.K. Anderson, R. D. Rausch, and D. L. Bonhaus. Implicit/multigrid algorithms for incompressible turbulent flows on unstructured grids. *Journal of Computational Physics*, 128(2):391 – 408, 1996.

[16] T. J . Barth and S. W. Linton. An unstructured mesh Newton solver for compressible fluid flow and its parallel implementation. In *33rd AIAA Aerospace Sciences Meeting and Exhibit*. AIAA–95–0221, 1995.

[17] F. Bassi, A. Crivellini, S. Rebay, and M. Savini. Discontinuous Galerkin solution of the reynolds-averaged Navier–Stokes and $k - \omega$ turbulence model equations. *Computers & Fluids*, 34:507 – 540, 2005.

[18] F. Bassi and S. Rebay. A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier–Stokes equations. *Journal of Computational Physics*, 131:267 – 279, 1997.

[19] M. Blanco and D. W. Zingg. Newton–Krylov algorithm with a loosely coupled turbulence model for aerodynamic flows. *AIAA Journal*, 45(5):980–987, 2007.

[20] A. N. Brooks and T. J.R. Hughes. Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 32(1–3):199 – 259, 1982.

[21] P.N. Brown and Y. Saad. Hybrid Krylov methods for nonlinear systems of equations. *SIAM Journal on Scientific and Statistical Computing*, 11:450–481, 1990.

[22] A. Burbeau, P. Sagaut, and Ch. H. Bruneau. A problem-Independent limiter for high-order Runge-Kutta discontinuous Galerkin Methods. *Journal of Computational Physics*, 169:111–150, 2011.

[23] N. K. Burgess and D. J. Mavriplis. Robust computation of turbulent flows using a discontinuous Galerkin method. In *50th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*. AIAA 2012-0457, 2012.

[24] C. Hu C and C-W. Shu. Weighted essentially non-oscillatory schemes on triangular meshes. *Journal of Computational Physics*, 150:97–127, 1999.

[25] D. Caraeni and L. Fuchs. Compact third-order multidimensional upwind scheme for Navier–Stokes simulations. *Theoretical and Computational Fluid Dynamics*, 15:373–401, 2002.

[26] D. Caraeni and L. Fuchs. Compact third-order multidimensional upwind discretization for steady and unsteady flow simulations. *Computers & Fluids*, 34(4-5):419–441, 2005.

[27] P. Castonguay, P. E. Vincent, and A. Jameson. A new class of high-order energy stable flux reconstruction schemes. *Journal of Scientific Computing*, 51:224 – 256, 2012.

[28] P. Castonguay, D.M. Williams, P.E. Vincent, and A. Jameson. Energy Stable Flux Reconstruction Schemes for Advection-Diffusion Problems. *Computer Methods in Applied Mechanics and Engineering*, 2013.

[29] R. F. Chen and Z. J. Wang. Fast, block lower-upper symmetric gauss–seidel scheme for arbitrary grids. *AIAA Journal*, 38(12):2238 – 2245, 2000.

[30] T. T. Chisholm and D. W. Zingg. A Jacobian-free Newton–Krylov algorithm for compressible turbulent fluid flows. *Journal of Computational Physics*, 228(9):3490 – 3507, 2009.

[31] C.S. Chou and C.W. Shu. High order residual distribution conservative finite difference WENO schemes for convection-diffusion steady state problems on non-smooth meshes. *Journal of Computational Physics*, 224(2):992 – 1020, 2007.

[32] P. Cinnella. Transonic flows of dense gases over finite wings. *Physics of Fluids*, 20(4):046103, 2008.

[33] B. Cockburn, G.E. Karniadakis, and C.W. Shu. *Discontinuous Galerkin methods: theory, computation and application*. Lecture notes in computational science and engineering. Springer, Berlin, 2000.

[34] B. Cockburn, S. Y. Lin, and CW. Shu. TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws III: One dimensional systems. *Journal of Computational Physics*, 84:90–113, 1989.

[35] P. Colonna, A. Guardone, and N. R. Nannan. Siloxanes: A new class of candidate Bethe-Zel'dovich-Thompson fluids. *Physics of Fluids*, 19(8):086102, 2007.

[36] P. Colonna, N.R. Nannan, A. Guardone, and E.W. Lemmon. Multiparameter equations of state for selected siloxanes. *Fluid Phase Equilibria*, 244(2):193 – 211, 2006.

[37] P.M. Congedo, C. Corre, and P. Cinnella. Numerical investigation of dense-gas effects in turbomachinery. *Computers & Fluids*, 49:290–301, 2011.

[38] M. S. Cramer and A. Kluwick. On the propagation of waves exhibiting both positive and negative nonlinearity. *Journal of Fluid Mechanics*, 12:9–37, 1984.

[39] M. S. Crammer. Nonclassical dynamics of classical gase. In A. Kluwick, editor, *Nonlinear Waves in Real Fluids*, pages 91–145. Springer, New York, 1991.

[40] A. Crivellini, V. D'Alessandro, and F. Bassi. A Spalart-Allmaras turbulence model implementation in a discontinuous Galerkin solver for incompressible flows. *Journal of Computational Physics*, 241:388 – 415, 2013.

[41] Á. Csík, H. Deconinck, and S. Poedts. Monotone residual distribution schemes for the ideal magnetohydrodynamics equations on unstructured grids. *AIAA Journal*, 39(8):1532 – 1541, 2001.

[42] Á. Csík, M. Ricchiuto, and H. Deconinck. A conservative formulation of the multidimensional upwind residual distribution schemes for general nonlinear conservation laws. *Journal of Computational Physics*, 179(1):286 – 312, 2002.

[43] David W. Levy and Tom Zickuhr and John Vassberg and Shreekant Agrawal and Richard A. Wahls and Shahyar Pirzadeh and Michael J. Hemsch. Data Summary from First AIAA Computational Fluid Dynamics Drag Prediction Workshop. *Journal of Aircraft*, 40:875–882, 2003.

[44] K. Van den Abeele, T. Broeckhoven, and C. Lacor. Dispersion and dissipation properties of the 1D spectral volume method and application to a p-multigrid algorithm. *Journal of Computational Physics*, 224:616 – 636, 2007.

[45] K. Van den Abeele and C. Lacor. An accuracy and stability study of the 2D spectral volume method. *Journal of Computational Physics*, 226:1007 – 1026, 2007.

[46] L. J. Durlofsky, B. Engquist, and S. Osher. Triangle based adaptive stencils for the solution of hyperbolic conservation laws. *Journal of Computational Physics*, 98:64–73, 1992.

[47] S.C. Eisenstat and H.F. Walker. Choosing the forcing terms in an inexact newton method. *SIAM Journal on Scientific and Statistical Computing*, 17:16–32, 1996.

[48] J. A. Ekaterinaris. High-order accurate, low numerical diffusion methods for aerodynamics. *Progress in Aerospace Sciences*, 41(3–4):192–300, 2005.

[49] O. Friedrich. Weighted Essentially Non-Oscillatory Schemes for the Interpolation of Mean Values on Unstructured Grids. *Journal of Computational Physics*, 150:194–212, 1998.

[50] C. Geuzaine and J.F. Remacle. Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities. *International Journal for Numerical Methods in Engineering*, 79(11):1309–1331, 2009.

[51] P. Geuzaine. Newton–Krylov strategy for compressible turbulent flows on unstructured meshes. *AIAA Journal*, 39(3):528–531, 2000.

[52] SK. Godunov. A finite-difference method for the numerical computation of discontinuous solutions of the equations of fluid dynamics. *Mat. Sb*, 47:271–290, 1959.

[53] A. Harten, B. Engquist, S. Osher, and S. Chakravarthy. Uniformly high-order essentially non-oscillatory schemes III. *Journal of Computational Physics*, 131:3–47, 1997.

[54] R. Hartmann. Adaptive discontinuous Galerkin methods with shock-capturing for the compressible Navier–Stokes equations. *International Journal for Numerical Methods in Fluids*, 51(6–10):1131–1156, 2006.

[55] J.C.C. Henriques and L.M.C. Gato. A multidimensional upwind matrix distribution scheme for conservative laws. *Computer and Fluids*, 33:755 – 769, 2004.

[56] L. R. Herrmann. Interpretation of finite element procedure as stress error minimization procedure. *Journal of the Engineering Mechanics Division*, 98(5):1330–1336, 1972.

[57] T. J.R. Hughes. Recent progress in the development and understanding of SUPG methods with special reference to the compressible Euler and Navier-Stokes equations. *International Journal for Numerical Methods in Fluids*, 7:1261–1275, 2005.

[58] T. J.R. Hughes, L. P. Franca, and G. M. Hulbert. A new finite element formulation for computational fluid dynamics: VIII. The Galerkin/least-squares method for advective-diffusive equations. *Computer Methods in Applied Mechanics and Engineering*, 73:173–189, 1989.

[59] T. J.R. Hughes, L.P. Franca, and M. Mallet. A new finite element formulation for computational fluid dynamics: I. Symmetric forms of the compressible Euler and Navier-Stokes equations and the second law of thermodynamics. *Computer Methods in Applied Mechanics and Engineering*, 54(2):223 – 234, 1986.

[60] T. J.R. Hughes and M. Mallet. A new finite element formulation for computational fluid dynamics: Iii. the generalized streamline operator for multidimensional advective-diffusive systems. *Computer Methods in Applied Mechanics and Engineering*, 58(3):305 – 328, 1986.

[61] H.T. Huynh. A flux reconstruction approach to high-order schemes including discontinuous Galerkin methods. AIAA 2009-4079, 2009.

[62] J. C. Vassberg and E. N. Tinoco and M. Mani and O. P. Brodersen and B. Eisfeld and R. A. Wahls and J. H. Morrison and T. Zickuhr and K. R. Laflin and D. J. Mavriplis. Data Summary from Third AIAA Computational Fluid Dynamics Drag Prediction Workshop. *Journal of Aircraft*, 45(3):781–793, 2008.

[63] A. Jameson. Computational fluid dynamics: Past, present and future. Future Directions in CFD Research National Institute for Aerospace. August 6–8, Hampton, VA, 2012.

[64] A. Jameson and S. Yoon. Lower-upper implicit schemes with multiple grids for the Euler equation. *AIAA Journal*, 25:929–935, 1987.

[65] B. Jiang. *The Least-Squares Finite Element Method – Theory and Applications in Computational Fluid Dynamics and Electromagnetics*. Springer, Berlin, 1998.

[66] G-S. Jiang and C-W. Shu. Efficient Implementation of Weighted ENO Schemes. *Journal of Computational Physics*, 126:202–228, 1996.

[67] A. Johnen, J.F. Remacle, and C. Geuzaine. Geometrical validity of curvilinear finite elements. In WilliamRoshan Quadros, editor, *Proceedings of the 20th International Meshing Roundtable*, pages 255–271. Springer Berlin Heidelberg, 2012.

[68] Kelly Laflin and Olaf Brodersen and Mark Rakowitz and John Vassberg and Richard Wahls and Joseph Morrison and Edward Tinoco and Jean-Luc Godard. Data Summary from Second AIAA Computational Fluid Dynamics Drag Prediction Workshop. *Journal of Aircraft*, 42(5):1165–1178, 2003.

[69] D. A. Knoll and D. Keyes. Jacobian-free newton method: a survey of approaches and application. *Journal of Computational Physics*, 193:357–397, 2004.

[70] D. A. Kopriva and H. Kolias J. A Conservative staggered-grid Chebyshev multidomain method for compressible flows. *Journal of Computational Physics*, 125:244 – 261, 1996.

[71] L. Krivodonova and M. Berger. High-order accurate implementation of solid wall boundary conditions in curved geometries. *Journal of Computational Physics*, 211(2):492 – 512, 2006.

[72] S. Lee and D. Whan Choi. On coupling the Reynolds-averaged Navier–Stokes equations with two-equation turbulence model equations. *International Journal for Numerical Methods in Fluids*, 50(2), 2006.

[73] T. Leicht and R. Hartmann. Error estimation and anisotropic mesh refinement for 3D laminar aerodynamic flow simulations. *Journal of Computational Physics*, 229(19):7344 – 7360, 2010.

[74] R. J. LeVeque. *Numerical methods for conservation laws*. Birkhauser, 1992.

[75] Z. Lilek and M. Perić. A fourth-order finite volume method with colocated variable arrangement. *Computers & Fluids*, 24(3):239 – 252, 1995.

[76] X-D. Liu, S. Osher, and T. Chan. Weighted Essentially Non-oscillatory Schemes. *Journal of Computational Physics*, 115:200–212, 1994.

[77] Y. Liu, Vinokur M M, and Z. J. Wang. Discontinuous spectral difference method for conservation laws on unstructured grids. *Journal of Computational Physics*, 216:780 – 801, 2006.

[78] R. Lohner, K. Morgan, and OC. Zienkiewicz. An adaptive finite element procedure for compressible high speed flows. *Computer Methods in Applied Mechanics and Engineering*, 51:441–465, 1989.

[79] D. Moro, N. C. Nguyen, and J. Peraire. Navier-Stokes solution using hybridizable discontinuous Galerkin methods. In *20th AIAA Computational Fluid Dynamics Conference*. AIAA Paper 2011-3407, 2011.

[80] NASA LaRC, Hampton, VA. FUN3D 12.3-66687 Manual. `http://fun3d.larc.nasa.gov`, May 2013.

[81] R.H. Ni. A multiple grid scheme for solving the Euler equations. In *5th Computational Fluid Dynamics Conference*, pages 257–264, 1981.

[82] H. Nishikawa. A first-order system approach for diffusion equation. I: Second-order residual-distribution scheme. *Journal of Computational Physics*, 227:315–352, 2007.

[83] H. Nishikawa. A first-order system approach for diffusion equation. II: Unification of advection and diffusion. *Journal of Computational Physics*, 229:3889–4016, 2010.

[84] H. Nishikawa. Robust and accurate viscous discretization via upwind scheme – I: Basic principle. *Computers & Fluids*, 49:62–86, 2011.

[85] H. Nishikawa and P. L. Roe. On high-order fluctuation-splitting schemes for Navier–Stokes equations. In *Computational Fluid Dynamics 2004: Proceedings of the Third International Conference on Computational Fluid Dynamics, ICCFD, Toronto, 12-16 July 2004*. Springer 2006, 2004.

[86] T. A. Oliver. *High-Order, Adaptive, Discontinuous Galerkin Finite Element Method for the Reynolds-Averaged Navier-Stokes Equations*. PhD thesis, Massachusetts Institute of Technology, 2008.

[87] C. F. Ollivier-Gooch. Quasi-ENO Schemes for Unstructured Meshes Based on Unlimited Data-Dependent Least-Squares Reconstruction. *Journal of Computational Physics*, 133:6–17, 1997.

[88] H. Paillère, J. Boxho, G. Degrez, and H. Deconinck. Multidimensional upwind residual distribution schemes for the convection-diffusion equation. *International Journal for Numerical Methods in Engineering*, 23:923–936, 1996.

[89] H. Paillère and H. Deconinck. Multidimensional upwind residual distribution schemes for the 2D Euler equations. In H. Deconinck and B. Koren, editors, *Euler and Navier-Stokes Solvers Using Multi-Dimensional Upwind Schemes and Multigrid Acceleration*, pages 51–112, 1997.

[90] Henri Paillere. *Multidimensional upwind residual distribution schemes for the Euler and Navier-Stokes equations on unstructured grids*. PhD thesis, von Karman Institure for Fluid Dynamics, 1995.

[91] P. De Palma, G. Pascazio, D.T. Rubino, and M. Napolitano. Residual distribution schemes for advection and advection-diffusion problems on quadrilateral cells. *Journal of Computational Physics*, 218(1):159–199, 2006.

[92] M. Parsani, G. Ghorbaniasl, C. Lacor, and E. Turkel. An implicit high-order spectral difference approach for large eddy simulation. *Journal of Computational Physics*, 229(14):5373 – 5393, 2010.

[93] F. Pellegrini. Scotch and libscotch 5.1 user's guide. Technical Report Co A8720, IPB & LaBRI, UMR CNRS 5800, 2008.

[94] M. Pernice and H.F. Walker. Nitsol: A newton iterative solver for nonlinear systems. *SIAM Journal on Scientific and Statistical Computing*, 19:302–318, 1998.

[95] P.O. Persson and J. Peraire. Sub-cell shock capturing for discontinuous Galerkin methods. In *44th AIAA Aerospace Sciences Meeting and Exhibit*. AIAA 2006-0112, 2006.

[96] P.O. Persson and J. Peraire. Curved mesh generation and mesh refinement using Lagrangian solid mechanics. In *47th AIAA Aerospace Sciences Meeting Including The New Horizons Forum and Aerospace Exposition*. AIAA 2009-949, 2009.

[97] S. B. Pope. *Turbulent Flows*. Cambridge University Press, 2000.

[98] M. Ricchiuto, N. Villedieu, R. Abgrall, and H. Deconinck. On uniformly high-order accurate residual distribution schemes for advection-diffusion. *Journal of Computational and Applied Mathematics*, 215(2):547 – 556, 2008.

[99] Mario Ricchiuto. *Construction and analysis of compact residual discretizations for conservation laws on unstructured meshes*. PhD thesis, von Karman Institure for Fluid Dynamics, 2005.

[100] P. J. Roache. *Verification and Validation in Computational Science and Engineering*. Hermosa Publishers, 1998.

[101] P. L. Roe. *Fluctuations and signals - a framework for numerical evolution problems*. K.W. Morton and M.J. Baines, editors, Numerical Methods for Fluids Dynamics. Academic Press, 1982.

[102] P.L. Roe. Linear advection schemes on triangular meshes. Technical Report Co A8720, Cranfield Institute of Technology, 1986.

[103] P.L. Roe. Optimum upwind advection on a triangular mesh. Technical report, ICASE, NASA Langley R.C., 1990.

[104] P.L. Roe and D. Sidilkover. Optimum positive linear schemes for advection in two and three dimensions. *SIAM Journal on Numerical Analysis*, 29(4-5):1542–1568, 1992.

[105] A. Rohde. Eigenvalues and eigenvectors of the euler equations in general geometries. In *15th AIAA Computational Fluid Dynamics Conference*. AIAA 2001-2609, 2001.

[106] C.J. Roy, C. C. Nelson, and T. M. Smith. Verification of Euler/Navier-Stokes codes using the method of manufactured solutions. *International Journal for Numerical Methods in Fluids*, 44(6):599–620, 2004.

[107] C. Rumsey. Langley research center: Turbulence modeling resource. `http://turbmodels.larc.nasa.gov`, 2013.

[108] C. Rumsey. SA Expected Results - 2D Zero Pressure Gradient Flat Plate. `http://turbmodels.larc.nasa.gov/flatplate_sa.html`, 2013.

[109] T. Saad and M.H. Schultz. Gmres: a generalized minimum residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing*, 7:856–869, 1986.

[110] Hermann Schlichting and Klaus Gersten. *Boundary-layer theory*. Springer, 2000.

[111] K. Sermeus and H. Deconinck. Solution of the steady Euler and Navier-Stokes equations using residual distribution schemes. In *VKI LS 2003-05, 33rd Computational Fluid dynamics Course*. von Karman Institute for Fluid Dynamics, 2003.

[112] D. Sharov and K. Nakahashi. Reordering of 3D Hybrid Unstructured Grids for Vectorized LU-SGS Navier–Stokes Computations. AIAA Paper 79-2102, 1997.

[113] C-W. Shu. *Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws*. Cockburn B, Johnson C, Shu C-W, Tadmor E, Quarteroni A, editors. Advanced Numerical Approximation of Nonlinear Hyperbolic Equations, Lecture notes in mathematics. Springer, Berlin, 1998.

[114] D. Sidilkover and P.L. Roe. Unification of some advection schemes in two dimensions. Technical Report 95-10, ICASE, 1995.

[115] T. Sonar. On the construction of essentially non-oscillatory finite volume approximations to hyperbolic conservation laws on general triangulations: polynomial recovery, accuracy and stencil selection. *Computer Methods in Applied Mechanics and Engineering*, 140:157–181, 1997.

[116] P. R. Spalart and S. R. Allmaras. A one-equation turbulence model for aerodynamic flows. *Recherche Aerospatiale*, 1:2244–2553, 1994.

[117] R. Span and W. Wagner. Equations of state for technical applications. i. simultaneously optimized functional forms for nonpolar and polar fluids. *International Journal of Thermophysics*, 24:1–39, 2003.

[118] R. Struijs, H. Deconinck, and P. L. Roe. Fluctuation splitting schemes for the 2d Euler equations. In *Von Karman Institute for Fluid Mechanics Lecture Series on Computational Fluid Mechanics*, 1991.

[119] R. Stryjek and J. H. Vera. PRSV: An improved Peng–Robinson equation of state for pure compounds and mixtures. *The Canadian Journal of Chemical Engineering*, 64:323–333, 1986.

[120] Y. Sun, Z.J. Wang, and Y. Lun. Efficient implicit non-linear lu-sgs approach for compressible flow computation using high-order spectral difference method. *Communications in computational physics*, 5(2-4):760–778, 2009.

[121] K. R. Jackson T. F. Chan. Nonlinearly preconditioned Krylov subspace methods for discrete newton algorithms. *SIAM Journal on Scientific and Statistical Computing*, 5:533–4542, 1984.

[122] P. A. Thompson. A fundamental derivative in gasdynamics. *Physics of Fluids*, 14:1843 – 1849, 1971.

[123] M.D. Tidiriri. Preconditioning techniques for the Newton-Krylov solution of compressible flows. *Journal of Computational Physics*, 132:51–61, 1997.

[124] E. van der Weide, H. Deconinck, and G. Degrez. A parallel, implicit, multidimensional upwind, residual distribution method for the Navier-Stokes equations on unstructured grids. *Computational Mechanics*, pages 199–208, 1999.

[125] B. van Leer. Towards the ultimate conservative difference scheme II. Monotonicity and conservation combined in a second-order scheme. *Journal of Computational Physics*, 14:361–370, 1974.

[126] B. van Leer. Towards the ultimate conservative difference scheme V. A second order sequel to Godunov's method. *Journal of Computational Physics*, 32:101–136, 1979.

[127] V. Venkatakrishnan and D. J. Mavriplis. Implicit solvers for unstructured meshes. *Journal of Computational Physics*, 105(1):83 – 91, 1993.

[128] N. Villedieu, T. Quintino, R. Abgrall, and H. Deconinck. High-order residual distribution schemes on quadrilateral meshes. *International Journal for Numerical Methods in Fluids*, 56(8):1559–1566, 2008.

[129] J.B. Vos, A. Rizzi, D. Darracq, and E.H. Hirschel. Navier-Stokes solvers in European aircraft design. *Progress in Aerospace Sciences*, 38(8):601–697, 2002.

[130] M. Vymazal, T. Quintino, N. Villedieu, and H. Deconinck. High-order upwind residual distribution schemes on isoparametric curved elements. *Journal of Computational Physics*, 230:890 – 906, 2011.

[131] Z. J Wang and Y. Liu. Spectral (finite) volume method for conservation laws on unstructured grids I: basic formulation. *Journal of Computational Physics*, 178:210–251, 2002.

[132] Z. J Wang and Y. Liu. Spectral (finite) volume method for conservation laws on unstructured grids II: extension to two-dimensional scalar equation. *Journal of Computational Physics*, 199:665–697, 2002.

[133] Z. J Wang and Y. Liu. Spectral (finite) volume method for conservation laws on unstructured grids VI: extension to viscous flow. *Journal of Computational Physics*, 215:51–58, 2006.

[134] Z. J. Wang, Y. Liu, G. May, and A. Jameson. Spectral difference method for unstructured grids II: extension to the Euler equations. *Journal of Scientific Computing*, 32:45 – 71, 2007.

[135] Z.J. Wang, Krzysztof Fidkowski, Remi Abgrall, Francesco Bassi, Doru Caraeni, Andrew Cary, Herman Deconinck, Ralf Hartmann, Koen Hillewaert, H.T. Huynh, Norbert Kroll, Georg May, Per-Olof Persson, Bram van Leer, and Miguel Visbal. High-order cfd methods: current status and perspective. *International Journal for Numerical Methods in Fluids*, 72(8):811–845, 2013.

[136] K. Wieghardt and W. Tillman. On the turbulent friction layer for rising pressure. Technical Report 1314, NACA, 1951.

[137] D. C. Wilcox. *Turbulence Modeling for CFD*. DCW Industries, Inc., 1994.

[138] W.A. Wood and W.L. Kleb. Diffusion characteristics of finite volume and fluctuation splitting schemes. *Journal of Computational Physics*, 153:353–377, 1999.

[139] C. Zamfirescu, A. Guardone, and P. Colonna. Admissibility region for rarefaction shock waves in dense gases. *Journal of Fluid Mechanics*, 599:363–381, 2008.

[140] O. C. Zienkiewicz and J. Z. Zhu. A simple error estimator and adaptive procedure for practical engineerng analysis. *International Journal for Numerical Methods in Engineering*, 24(2):337–357, 1987.

[141] O. C. Zienkiewicz and J. Z. Zhu. The superconvergent patch recovery and a posteriori error estimates. part 2: Error estimates and adaptivity. *International Journal for Numerical Methods in Engineering*, 33(7):1365–1382, 1992.

[142] O.C. Zienkiewicz and R.L.Taylor. *Finite Element Method (5th Edition) Volume 1 - The Basis*. Elsevier, 2000.