NATIONAL TECHNICAL
UNIVERSITY OF
ATHENS

Aerodynamics Laboratory

A thesis submitted for the degree of Doctor of Philosophy

Development of a hybrid compressible vortex particle method and application to external problems including helicopter flows

Giorgos Papadakis

December 2014

Development of a hybrid compressible vortex particle method and application to external problems including helicopter flows

Ph.D. Thesis

Giorgos Papadakis

Examination Committee:

1. G. Bergeles *, Professor, NTUA, School of Mechanical Engineering
2. A. Boudouvis, Professor, NTUA, School of Chemical Engineering
3. D. Bouris, Assistant Professor, NTUA, School of Mechanical Engineering
4. K. Giannakoglou *, Professor, NTUA, School of Mechanical Engineering
5. G. Tzabiras, Professor, NTUA, School of Naval Architecture and Marine Engineering
6. S. Tsaggaris, Professor, NTUA, School of Mechanical Engineering
7. S. Voutsinas (Supervisor) *, Associate Professor, NTUA, School of Mechanical Engineering

* Member of the Advisory Committee

Acknowledgements

Even though the completion of a Ph. D. requires personal commitment, this work wouldn't be complete without the help (direct and indirect) of other people.

First of all, I would like to thank Professor Spyros Voutsinas. I can't express my gratitude and appreciation for his guidance and support the last 7 years. His help through the years was invaluable and I hope that he will continue to guide and teach me in the future. He is like a second father to me.

I would also like to thank the members of the advisory committee for their useful comments that helped me writing this thesis. I also express my gratitude to the members of the examination committee for reading the thesis and making constructive comments on the presented work.

Special thanks to Dimitris Manolas. We are working with Dimitris since 2001 when we entered NTUA. For 14 years (almost)we have been close friends and for the last 7 years we shared the same office in the Laboratory. Nothing would be the same without him. Together we evolved as people and as scientists. I wish him the best.

Moving on, the next people that come in my mind are Dr. Marinos Manolesos and Aris Kapelonis. I met them as co-workers but now they are close friends. They are both excellent scientists and the conversations we had on scientific matters were always enlightening. Special thanks, for the wonderful times we had outside the laboratory. I am most grateful to Marinos not only for proof-reading the thesis but also making important remarks on the context.

Giannis Prospathopoulos, Vasilis Riziotis and Petros Chasapogiannis are the three pillars of support of the Aerodynamics Laboratory. Having long experience in scientific and computational matters they helped and continue to help not only me but all the students affiliating with the Lab.

Many thanks to Konstantinos Diakakis for the help he provided with grid construction. His help was invaluable.

I would also like to thank Vasilis Tsarsitalidis for the long and winding conversations we had. My deep respect to Vaggelis Fillipas and Stelios Polyzos for being wonderful neighbors in work.

Setting co-workers aside, I would like to thank my friends Ago, Thodoris, Vaggelis, Alkis, Nikitas, Andreas and Giorgos for their continuous and long term support all those years. They generously took all my troubles and anxieties away

and helped me anyway they could. I am counting on their support for the years to come. With their help any issue can be dealt with. I acknowledge Ago for the nickname "Papis" that follows me through the years.

As usual, family comes last. I would like to thank them for their deep support all those years. I wouldn't be who I am without them.

Finally, I would like to mention Maria. She put up with me all those years and encouraged me to continue endlessly. I can't thank her because it is simply not enough. I hope we will have a happy and prosperous life together.

This page intentionally left blank.

Contents

This page intentionally left blank.

List of Figures

This page intentionally left blank.

List of Tables

This page intentionally left blank.

Abstract

A new hybrid CFD methodology is formulated for external aerodynamic problems. The work was motivated by the need to efficiently simulate complex engineering problems that are dominated by strong vortex interactions and involve several independently moving bodies. To this end standard (Eulerian & grid-based) CFD is strongly coupled with compressible vortex particle method (Lagrangian & grid free). More specifically,

The Eulerian CFD Solver (MaPFlow): A compressible unstructured Finite Volume URANS solver was developed. MaPFlow is equipped with: Low Mach Preconditioning to handle flows in the incompressible regime; the Spalart-Almaras and k-ω SST turbulence modeling; the options of near wall treatment and wall functions; Arbitrary Lagrangian Eulerian formulation for moving of deformable grids; and multi-block domain decomposition. The parallelization of the code follows an MPI implementation.

The Lagrangian Solver: A compressible Lagrangian Vortex Particle solver was formulated and developed. The formulation uses mass, dilatation, vorticity, energy and volume as working variables and can handle compressible flows including shocks. To mitigate the computational cost, a multi-block Particle Mesh (PM) technique was implemented using the James-Lackner Algorithm.

Coupling of the two solvers in one tool (HoPFlow): The Eulerian and Lagrangian solvers were strongly coupled in a single hybrid methodology. Specific coupling conditions have been formulated that guarantee continuity, consistency and conservation.

The results presented in this work basically aim at verifying the codes that were developed. Initially, the CFD solver is validated by comparing its predictions to experimental and numerical data in several two and three dimensional cases. Next, the hybrid solver is validated by comparing its predictions to purely Eulerian results. Validation cases include: two dimensional flows around stationary and moving airfoils in a wide range of Reynolds and Mach Numbers and three dimensional flows around stationary and rotating wings (Wind Turbine and Helicopter Rotors).

The extensive validation conducted showed that both the CFD solver (MaPFlow) and the hybrid one (HoPFlow) perform well. It is shown that the hybrid solver operates at significantly lower diffusion rate and because of that gives better results compared to standard CFD when the flow is dominated by strong vortex structures as in the case of a helicopter in hover. Most of the test cases are less complex than that for which standard CFD can provide satisfactory predictions.

However, it was necessary to validate the new method against these simpler cases before considering more challenging engineering flows.

Chapter 1

Scope and Content

1.1 State of the Art in CFD and current Trends

Computational Fluid Dynamics (CFD) is a well-established and highly valued tool. It has been widely used in the last decades for addressing various engineering problems, from basic aerodynamic to complex aeroelastic problems. CFD has evolved through the years from purely potential inviscid flows to the solution of the Navier-Stokes equations on grids composed of millions cells [1].

Following the amazing progress in computer technology, large scale simulations of the flow around complex configurations are now feasible. Key drivers to that, were the ability of CFD to model the flow mechanisms in detail and to accurately predict the flow development in space and time [2].

From a conceptual point of view, CFD modeling can be divided in two main categories: the grid based methods and the grid-free ones. In grid-free or mesh-less methods, the flow is described in Lagrangian coordinates while in the grid-based ones the description is Eulerian. In the Lagrangian or material context, fluid markers (particles) are introduced to describe the flow and evolution is accounted for by following the trajectories of the particles. On the contrary, grid based methods describe the flow by recording the values of the flow quantities at grid nodes.

Eulerian (grid-based) Solvers

Eulerian CFD is by far the most popular simulation method in aerodynamics. Wall boundary conditions are accurately introduced while the flow equations can be solved with high accuracy, provided that the grid is sufficiently dense. The grids can be structured or unstructured while the formulation of the discrete flow equations can be formulated based on one of the following: the Finite Difference Method (FDM) [3], the Finite Volume Method (FVM) [4], the recently suggested Discontinuous Galerkin Method (DGM) [5] and finally, the Spectral Method [6].

Current state of the art CFD software includes features such as: domain decomposition together with parallel processing for cost reduction [7]; local grid refinement for higher accuracy and gradient capturing [8]; multi-grid methods for faster convergence [9]; and mixed grids for handling boundary layers with accuracy [10]. Additionally, in order to simulate flows around complex configurations containing several independently moving bodies, the technique of Over-set (Chimera) grids [11] and that of Sliding Grids [12] have been defined.

Regarding turbulence, great steps have been made, from Reynolds Averaged Navier Stokes (RANS) [13] to Large Eddy Simulation (LES)[14] and Direct Numerical Simulation (DNS) [15]. In RANS, the spectrum of turbulent fluctuations is averaged in frequency. By applying this kind of averaging to the flow equations, the so-called Reynolds stress terms appear in the momentum and energy equations leading to a closure problem. There are several closures ranging from simple algebraic up to multi equation differential models. On the other end, in DNS turbulence modeling is not needed while in LES, that stands in between, instead of averaging over the frequency range, space filtering is carried out that eliminates the unresolved scales. The grid requirements in DNS are very demanding while in LES more manageable sizes are needed. Still, in High Reynolds flows the grid requirements are very high and hence hybrid RANS/LES models have been developed [16].

Despite the vast development of the computational resources and the numerical tools, there are still some open issues to be addressed. Most of the CFD codes are Finite Volume codes which can handle complex flows. However, the usual accuracy is only 2nd order. Additionally, when external flows are considered, the flow field must be truncated at some finite distance. Over the distant boundary, far-field conditions are needed which in most cases approximate a fully developed flow. Depending on the approximation chosen, far-field conditions could introduce errors that are sensed as reflections in the simulations [17]. Furthermore, domain truncation is usually combined with gradual grid coarsening which increases numerical diffusion and adds errors. In fact, 2nd order LES would require a very dense grid, in order to control numerical diffusion. In [18] it is reported that the low order discretization errors might be of the same magnitude with the sub-grid scale contribution. On top of that, in many applications unsteady simulations are needed which are still very expensive especially when time accuracy is important (as for example in aeracoustics or aeroelasticity). Furthermore, when considering flows around independently moving bodies, sliding or over-set grids are needed in order to properly handle the involved interactions. This introduces additional errors and complexity. Typical examples in this respect are the blade-tower interaction problem for Horizontal Axis Wind Turbines and the flow around Vertical Axis Wind Turbines. Finally, when the evolution of wakes over large distances is needed as in the case of wind farms and helicopter rotors, standard CFD simulations could easily exhaust the computer capacity available, due to excessively and eventually prohibitively large grids. In certain cases this problem can be overcome

using Actuator Disk or Actuator Line methods [19] but this adds uncertainties. Especially in wind farms (which is also relevant to wind turbine aerodynamics) additional complexity is brought by the atmospheric inflow characteristics that contains large eddies [19]. These eddies play an important role and must be taken into account. An alternative approach in view of keeping the size of the grid reasonable but reduce diffusion is the Vorticity Confinement method first introduced in [20] and further developed in [21].

Lagrangian (mesh-free) Solvers

Alternatively, one may solve the flow equations in Lagrangian formulation as particle methods do. They are grid-free and self-adaptive, having (in theory) zero numerical diffusion; domain truncation is not needed and true velocity conditions at infinity are exactly embedded in the formulation [22]. There are two main options in the context of Lagrangian methods: the Vortex Particle Method (VPM) [22] and Smooth Particle Hydrodynamic method (SPH) [23, 24]. They differ in the quantities that particles carry. However, regardless of the choice, in particle methods wall boundary conditions constitute a major challenge, involving costly convolution operations and large number of particles [25, 26].

Vortex Particle Methods have been widely used for capturing the 3D wake behind rotating bodies like Wind Turbines [27], Helicopter Rotors [28] but also Aircrafts [29]. They are usually combined with Boundary Element Methods [30]. Their non-diffusive properties makes VPMs ideal for wake modeling because, as they are grid-free, the cost of keeping the wake for several diameters is significantly reduced. However, VPMs have certain shortcomings. Firstly in the VPM formulation, the flow velocity is defined in convolution form over the set of flow particles [31, 32]. Convolution can be computationally expensive as the number of particles increases. To account for the excessive computational cost, Particle Mesh (PM) methods [33, 34, 35] and multi-pole expansions [36, 37] have been introduced. Moreover, the topology of the particles gets distorted as time evolves. Clustering or spreading of particles can deteriorate the accuracy of the method [22, 38]. To mitigate this, the re-meshing technique is employed. Re-meshing is equivalent to a periodic spatial redistribution of the particle information at ordered positions [22, 39, 40, 41].

Vortex Particle Methods were originally formulated for inviscid incompressible flow in free space without boundaries. Concerning viscosity, several attempts have been made to include viscous effects in particle simulations, basically in laminar flow conditions [42, 43] concerning free space flows. Extension to flows with boundaries has been done but again in low Re problems [25, 26]. Particles are generated close to the solid boundaries in order to satisfy the no slip condition, therefore going higher in Reynolds number would require resolution of a thinner boundary layer requiring an ever increasing number of particles. Simulations of this type have not been yet attempted.

Likewise, the extension of VPMs to compressible flows is limited. Pure particle methods have been formulated by [44, 45] while in [46, 47] using Particle Mesh based techniques has been proposed. The two PM based methods are an extension of the "incompressible" VPM in which compressibility is accounted for through a properly defined scalar potential. In [46] the scalar potential equation has the form of Hamilton-Jacobi equation using dilatation as an auxiliary field, while in [47] the scalar potential was treated in a wave form equation for pressure. The pure particle approach presented in [44], used Lagrangian markers that carried vorticity, density, dilatation, entropy, and enthalpy while all equations are solved in a Lagrangian manner.

Despite, the multi-directional research carried out in view of including viscous and compressibility effects in particle methods, the high number of particles needed close to solid boundaries prevented a wide application of purely particle methods to real life engineering problems [48].

Eulerian vs Lagrangian Methods

Summarizing the positive (+) and negative (-) characteristics of the two basic CFD methodologies the following comparison can be made:

Eulerian Solvers

+ Solid Boundary Conditions are accurately satisfied
+ They have been widely used in a wide variety of problems
+ With proper grid resolution, high fidelity calculations are possible.
- The far-field must be truncated at a finite distance.
- Gradual grid coarsening adds excessive diffusion making wake aerodynamics a really challenging task.
- Multi-body simulations require special techniques which add additional errors and complexity to the problem.

Lagrangian Solvers

+ They are mesh free and self adaptive.
+ They have zero diffusion in theory.
+ They satisfy exactly the far-field conditions.
+ They have been widely used for wake aerodynamics
+ Multi-body simulations are easily handled.
- Convolutions can make calculations computationally prohibitive
- Solid Wall boundary conditions constitute a major challenge.

Possible Alternatives

While in principle CFD can solve any problem, the computational cost remains an important bottleneck. This explains why CFD is always in search of methods that reduce cost without compromising accuracy. In this respect, domain decomposition is a widely used technique [7] which has significantly reduced the user time of CFD simulations. In standard domain decomposition, the same formulation is used in all sub-domains. However, in certain methodologies, named *hybrid*, different formulations are used. The strategy in this case is to choose for every sub-domain the best performing methodology.

Since particle methods appear so complementary to Eulerian CFD, it is natural to combine the two in one single package following a domain decomposition approach. The option of combining domain decomposition with essentially different formulations has taken several forms. From a topological point of view, the sub-domains can be either overlapping or not. Strong viscous-inviscid interaction models [49, 50] and RANS-Vortex coupled ones [51, 52] are examples of overlapping hybrid methodologies. They combine an inviscid Lagrangian formulation for the entire flow with a viscous layer close to the solid boundaries and define coupling conditions for the pressure and the velocity on the outer boundary of the viscous layer. The coupling in this case is approximate, not only because the two formulations are not equivalent, but also because the sub-domains overlap. Exact coupling consists of imposing continuity to all flow quantities which in principle refers to non-overlapping sub-domains. In this case, the coupling conditions take the form of integral equations (also known as Neumann-to-Dirichlet map) defined on the boundary interface [22, 53, 54]. In 2D problems the extra cost is manageable but in 3D the penalty can be substantial. This explains why certain hybrid methods prefer to have at least some degree of overlapping, in the form of a buffer area as in [55], while others directly approximate the coupling conditions. In [56, 57] compressible RANS solvers were coupled with viscous vortex methods. They both partially implement non-overlapping decomposition. Vortex particles are defined outside the Eulerian domain based on the flux of vorticity across the interface while the calculation of the velocity on the interface includes the vorticity within the Eulerian domain. Another particularity, is that density and pressure are not explicitly matched at the boundary.Recently, in [58] a 2D hybrid solver was developed that uses a compressible particle formulation that was combined with an Euler code and promising results in moderate Mach numbers were achieved.

1.2 Scope of the Thesis and Main Objectives

1.2.1 The Basic Concept

The motivation behind this work, was to formulate a hybrid method for two and three dimensional external flows by combining an Eulerian Solver and a Lagrangian

one in a strongly coupled context. The requirements set for the hybrid solver were:

- To simulate flows at all Mach numbers,
- To have Accuracy that is at least comparable to that of standard Eulerian CFD
- To have substantially lower numerical diffusion as compared to Eulerian CFD
- To easily handle multi-body geometries
- To allow aeroelastic and aeracoustic couplings.

A hybrid flow solver with these characteristics would constitute an alternative methodology in the context of multi-physics computational simulations.

The basic concept behind the proposed hybrid methodology consists of the following items:

1. Replacement of the large CFD grid with a restricted one. Instead of using grids spanning several chords or diameters, the key idea is to use grids only close to solid walls. This way wall boundary conditions can be accurately and easily satisfied. By restricting the CFD to the near wall region, the handling of multi-body geometries becomes easier. Each body will have a separate grid and thus complex geometries with moving and stationary parts will be easily treated. Additionally, having grids of limited extent more points can be packed in the boundary layer region and accuracy could be increased.

2. Formulation of accurate far-field conditions at the outer boundary of the CFD grid. Since the CFD grid is truncated to a short distance from the solid walls, far-field conditions can no longer be used on the outer boundary of the grid. Instead a correct account of the flow outside the CFD grid is needed. To this end,

   - The entire space is covered with particles and the flow equations are solved in Lagrangian form,
   - In order to be compatible with the CFD solver, particles are defined to carry mass, dilatation, vorticity and energy.

   In this way the domains (and the solutions) of the two solvers will overlap. This means that if the coupling between the two is conservative and consistent the corresponding solutions should match at all times.

3. Implementation of advanced CFD tools like multi-grid solution strategies and grid adaptation.

It should be stressed that a hybrid solver is not proposed as a replacement of Eulerian CFD solvers. Hybrid methods are far too complicated for the simulations of steady flow problems. In fact, the idea it to enhance standard CFD and provide efficient means to solve really complex problems. Using small grids near solid bodies and Lagrangian markers in the rest of the flow-field can make feasible simulations of complex flows such as Vertical Axis Wind Turbines flow, Wind Farms flow, Vortex induced Vibration, Aerodynamics of Wakes and Blade Vortex Interaction. Problems that can easily be handled with standard CFD, are not really suited for a hybrid solver.

1.2.2 Modular Description of a Hybrid Methodology

In the course of the hybrid solver development, the following stand-alone tools were needed that were finalized and tested.

An Eulerian CFD Solver

For the CFD part the MaPFlow solver was developed. MaPFlow solves the unsteady compressible Reynolds-Averaged Navier Stokes equations on unstructured grids using the Finite Volume method. There are two options regarding the modeling of turbulence: the k-ω SST model as formulated by Menter [59] and the Spalart Allmaras one equation model [60]. In order to extent its use to low Mach flows, preconditioning has been added. MaPFlow, is 2nd order in time and space and can handle moving or deforming grids. Finally, it has been parallelized using the Message Passing Interface (MPI).

The Lagrangian Solver

Aiming at an all Mach hybrid flow solver, the choice of a compressible particle formulation is mandatory. Initially, following the work by Oxley [58], and Mas-Gallic's [46] a (ρ, ω, ϕ) formulation was used. As previously mentioned, in this formulation, dilatation is indirectly obtained by solving a Hamilton-Jacobi equation for the scalar potential ϕ on the PM grid using an Essentially Non-Oscillatory (ENO) difference scheme [61, 62, 63, 64]. However, the underlying equation for ϕ is formulated under the assumption of isentropic flow variations and thus such a solver is restricted up to moderate Mach numbers by construction. So this formulation was abandoned and a general Lagrangian description was chosen. It consists of adding to the particle flow description the total energy carried by the particles together with mass, dilatation and vorticity.

The Particle Mesh Poisson Solver

In the proposed formulation the flow-field is filled with particles. This results in prohibitively high computational cost when particle interactions are calculated

directly. To override this heavy computational cost, the Particle Mesh (PM) acceleration technique was implemented. The James-Lackner Algorithm [65, 66] was used in combination with Local Corrections (MLC) [67]. The PM solver was also parallelized in a domain decomposition context [68].

Coupling

With the two solvers at at hand, what remains is the coupling procedures. The Eulerian and Lagrangian were strongly coupled in a single hybrid methodology. The idea of the coupling is simple.

- The Eulerian solver in order to evolve, needs a starting solution and the proper boundary conditions. Solid wall boundary conditions are accurately handled by the CFD solver itself while the far-field boundary conditions are provided by the Lagrangian Solver.
- The presence of bodies is not taken into account by the Lagrangian solver. Solid walls exist only in the Eulerian context. However, during the coupling procedure Eulerian particles are generated by the CFD solver that replace the Lagrangian ones within the CFD domain. In this way the wall boundary condition is taken into account by the Lagrangian Solver.

In the way the coupling is done, any CFD solver could be used regardless of the its internal particularities.

1.3 Novelties

The present work has delivered:

- A compressible Finite Volume URANS Solver (MaPFlow)
- A compressible Lagrangian Solver
- A hybrid solver (HoPFlow)

MaPFlow can be regarded as a standard CFD solver and has been developed as one of the basic modules in HoPFlow. There is no real novelty in MaPFlow except perhaps in certain implementation details which in principle would at most increase robustness, lower the cost and eventually the accuracy of the solver.

The novelties brought by this work concern the other two solvers. With respect to Lagrangian solvers, a new formulation of compressible particle methods is given. As compared to previous works, the present method does not assume isentropic flow variations as in [46, 47, 69], while with respect to the work by Eldredge [44], particles carry energy instead of enthalpy and entropy. The latter difference is not regarded essential and is viewed as an alternative formulation of the same set of

equations. It is noted however that the Lagrangian solver was developed as part of the hybrid one and consequently it was important from the beginning to define compressible particles in conformity with the CFD solver. The challenging issue was to also include solid boundaries and this is done here for the first time at least in a fully consistent way. In addition to that, the present work provides for the first time a detailed validation of compressible particle methods that covers a wide variety of applications (non-conventional airfoils, airfoils in deep stall, wind turbines, helicopter rotors) . The implementation of the highly efficient parallel Particle Mesh method developed by James & Lackner is of special practical importance. Substantial cost reduction has been achieved indicating the applicability of the hybrid method in large scale simulations.

1.4 Thesis Outline

The thesis is divided in 5 chapters.

- Chapter 1 gives an overview of the current state of art in CFD and by that identifies the aspects that remain open, presents the basic idea behind the hybrid solver that has been developed and outlines the novelties of the thesis.
- Chapter 2 describes the Eulerian Solver (MaPFlow).
- In Chapter 3 the Lagrangian and the hybrid solver (HoPFlow) are presented. This Chapter is basically split in three main sections. In the first section, a compressible Lagrangian solver is formulated. Then in the second section the theoretical and numerical aspects of the Particle Mesh solver that has been implemented are described. Finally in the third section, the coupling procedure between the Eulerian and Lagrangian solvers is detailed.
- Chapter 4 contains the validation and verification of both MaPFlow and HoPFlow. First MaPFlow is validated against experimental and other numerical data in a number of two and three dimensional flows. Next, HoPFlow is verified against MaPFlow and experiments in cases of two and three dimensional flows. Based on this verification the chapter concludes with a discussion on the properties and the performance characteristics of the hybrid solver.
- Chapter 5 contains the conclusions drawn from the present work along with recommendations for future work.

This page intentionally left blank.

Chapter 2

The Eulerian solver

In this chapter the Eulerian solver (MaPFlow) that was developed is described. MaPFlow solves the Unsteady Reynolds Averaged Navier Stokes equations (URANS) in a multiprocessor environment using the MPI protocol. The solver is 2nd order in space and time; makes use of the Roe approximate Riemann solver for reconstruction; is equipped with Preconditioning to handle Low Mach number flows and uses the Spalart Allmaras and the $k-\omega\, SST$ models for turbulence closure.

2.1 Governing Equations

2.1.1 Conservative form

Let D denote a volume of fluid and ∂D its boundary. By integrating the Governing equations over D, the following integral form is obtained:

$$\int_D \frac{\partial \vec{U}}{\partial t} dD + \oint_{\partial D} (\vec{F}_c dS - \vec{F}_v) dS = \int_D \vec{Q} dD \tag{2.1}$$

In (2.1) $\vec{U}$, is the vector of the Conservative Flow Variables,

$$\vec{U} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho E \end{pmatrix} \tag{2.2}$$

where ρ denotes the density, (u, v, w) the three components of the velocity field and E the total energy. $\vec{F}_c$ and $\vec{F}_v$ denote the Convective and Viscous Fluxes

respectively,

$$\vec{F_c} = \begin{pmatrix} \rho V \\ \rho u V + n_x p \\ \rho v V + n_y p \\ \rho w V + n_z p \\ \rho(E + \frac{p}{\rho})V \end{pmatrix} \tag{2.3}$$

$$\vec{F_v} = \begin{pmatrix} 0 \\ n_x \tau_x x + n_y \tau_x y + n_z \tau_x z \\ n_x \tau_y x + n_y \tau_y y + n_z \tau_y z \\ n_x \tau_z x + n_y \tau_z y + n_z \tau_z z \\ n_x \Theta_x + n_y \Theta_y + n_z \theta_z \end{pmatrix} \tag{2.4}$$

where V is the contravariant velocity, $V = \vec{u} \cdot \vec{n}$ and

$$\begin{aligned} \Theta_x &= u\tau_{xx} + v\tau_{xy} + w\tau_{xz} + k\frac{\partial T}{\partial x} \\ \Theta_y &= u\tau_{yx} + v\tau_{yy} + w\tau_{yz} + k\frac{\partial T}{\partial y} \\ \Theta_z &= u\tau_{zx} + v\tau_{zy} + w\tau_{zz} + k\frac{\partial T}{\partial z} \end{aligned} \tag{2.5}$$

The above system is completed with the equation of state for perfect gas:

$$p = (\gamma - 1)\rho \left[E - \frac{u^2 + v^2 + w^2}{2} \right] \tag{2.6}$$

2.1.2 Variable Transformations

CFD solvers are formulated using the governing equations (2.1) written in primitive ($\vec{V}$) or characteristic ($\vec{V}_{ch}$) variables,

$$\vec{V} = \begin{pmatrix} \rho \\ u \\ v \\ w \\ p \end{pmatrix} \tag{2.7}$$

Primitive Variables

Starting from the differential form of the governing equations,

$$\frac{\partial \vec{U}}{\partial t} + \frac{\partial \vec{F_c}}{\partial \vec{x}} - \frac{\partial \vec{F_v}}{\partial \vec{x}} = \frac{\partial \vec{Q}}{\partial \vec{x}} \tag{2.8}$$

and neglecting the viscous terms and using the chain rule:

$$
\begin{aligned}
&\frac{\partial \vec{U}}{\partial t} + \frac{\partial \vec{F}_c}{\partial \vec{U}} \frac{\partial \vec{U}}{\partial \vec{x}} = \frac{\partial \vec{Q}}{\partial \vec{x}} \\
&\frac{\partial \vec{U}}{\partial t} + A_c \frac{\partial \vec{U}}{\partial \vec{x}} = \frac{\partial \vec{Q}}{\partial \vec{x}}
\end{aligned} \tag{2.9}
$$

the Jacobian of the convective fluxes $A_c = \partial \vec{F}_c / \partial \vec{U}$ is obtained. By introducing the transformation matrix $M = \partial \vec{U} / \partial \vec{V}$, the system of equations is written in Primitive Variables form [3]:

$$
\begin{aligned}
&\frac{\partial \vec{U}}{\partial \vec{V}} \frac{\partial \vec{V}}{\partial t} + A_c \frac{\partial \vec{U}}{\partial \vec{V}} \frac{\partial \vec{V}}{\partial t} = \frac{\partial \vec{Q}}{\partial \vec{x}} \\
&M \frac{\partial \vec{V}}{\partial t} + A_c M \frac{\partial \vec{V}}{\partial t} = \frac{\partial \vec{Q}}{\partial \vec{x}} \\
&\frac{\partial \vec{V}}{\partial t} + M^{-1} A_c M \frac{\partial \vec{V}}{\partial t} = M^{-1} \frac{\partial \vec{Q}}{\partial \vec{x}} \\
&\frac{\partial \vec{V}}{\partial t} + A_p \frac{\partial \vec{V}}{\partial t} = \frac{\partial \vec{Q}_v}{\partial \vec{x}}
\end{aligned} \tag{2.10}
$$

The transformation matrix M is defined as:

$$
M = \frac{\partial \vec{U}}{\partial \vec{V}} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ u & \rho & 0 & 0 & 0 \\ v & 0 & \rho & 0 & 0 \\ w & 0 & 0 & \rho & 0 \\ \frac{u^2+v^2+w^2}{2} & \rho u & \rho v & \rho w & \frac{1}{\gamma-1} \end{bmatrix} \tag{2.11}
$$

and its inverse as:

$$
M^{-1} = \frac{\partial \vec{V}}{\partial \vec{U}} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ -\frac{u}{\rho} & \frac{1}{\rho} & 0 & 0 & 0 \\ -\frac{v}{\rho} & 0 & \frac{1}{\rho} & 0 & 0 \\ -\frac{w}{\rho} & 0 & 0 & \frac{1}{\rho} & 0 \\ \frac{\gamma-1}{2}\left(u^2+v^2+w^2\right) & -u(\gamma-1) & -v(\gamma-1) & -w(\gamma-1) & \gamma-1 \end{bmatrix} \tag{2.12}
$$

Equation (2.10) has the same form as (2.9) but the convective flux Jacobian A_p is in primitive variables.

Characteristic Variables

Diagonalization of A_c ($A_c = R\Lambda L$, where R,L contain the right and left eigenvectors respectively and Λ the eigenvalues) enables to transform (2.9) in characteristic

variables and by that to decouple the system of equations.

$$\frac{\partial \vec{U}}{\partial t} + L^{-1}\Lambda L\frac{\partial \vec{U}}{\partial \vec{x}} = \frac{\partial \vec{Q}}{\partial \vec{x}}$$
$$L\frac{\partial \vec{U}}{\partial t} + \Lambda L\frac{\partial \vec{U}}{\partial \vec{x}} = L\frac{\partial \vec{Q}}{\partial \vec{x}} \tag{2.13}$$

By defining $M_{ch} \equiv \partial\vec{U}/\partial\vec{V}_{ch} = L$ the decoupled system is acquired:

$$\frac{\partial \vec{V}_{ch}}{\partial t} + \Lambda\frac{\partial \vec{V}_{ch}}{\partial \vec{x}} = \frac{\partial \vec{Q}_{ch}}{\partial \vec{x}} \tag{2.14}$$

with:

$$\Lambda = \begin{bmatrix} V & 0 & 0 & 0 & 0 \\ 0 & V & 0 & 0 & 0 \\ 0 & 0 & V & 0 & 0 \\ 0 & 0 & 0 & V+c & 0 \\ 0 & 0 & 0 & 0 & V-c \end{bmatrix} \tag{2.15}$$

Depending on the variables chosen the diagonalization of the Jacobian Matrix (A_c for conservative variables,A_p for primitive) will lead to different eigenvectors. Of course, using the appropriate transformation matrix the eigenvectors can be transformed to the variables desired. For example in between primitive and characteristic variables the following hold:

$$A_p = M^{-1}A_cM = \left(M^{-1}R\right)\Lambda\left(LM\right) = R_p\Lambda L_p \tag{2.16}$$

where $R_p = M^{-1}R$ and $L_p = LM$ are the right and left eigenvectors in primitive variables. Using the inverse transformation matrix enables the transformation from primitive eigenvectors to conservative eigenvectors.

The right eigenvectors in primitive variables are :

$$R_p = \begin{bmatrix} n_x & 0 & n_z & -n_y & -\frac{n_x}{c^2} \\ n_y & -n_z & 0 & n_x & -\frac{n_y}{c^2} \\ n_z & n_y & -n_x & 0 & -\frac{n_z}{c^2} \\ 0 & n_x & n_y & n_z & -\frac{\lambda_1-\lambda_4}{\rho c^2} \\ 0 & -n_x & -n_y & -n_z & \frac{\lambda_1-\lambda_5}{\rho c^2} \end{bmatrix} \tag{2.17}$$

where $\vec{n} = (n_x, n_y, n_z)$ is the unit normal vector, $\lambda_1 = \lambda_2 = \lambda_3 = V$, $\lambda_4 = V + c$ and $\lambda_5 = V - c$.

The left eigenvectors in primitive variables are :

$$L_p = R_p^{-1} = \begin{bmatrix} n_x & n_y & n_z & \frac{\rho}{\lambda_4-\lambda_5} & \frac{\rho}{\lambda_4-\lambda_5} \\ 0 & -n_z & n_y & \frac{\lambda_1-\lambda_5}{\lambda_4-\lambda_5}n_x & \frac{\lambda_1-\lambda_4}{\lambda_4-\lambda_5}n_x \\ n_z & 0 & -n_x & \frac{\lambda_1-\lambda_5}{\lambda_4-\lambda_5}n_y & \frac{\lambda_1-\lambda_4}{\lambda_4-\lambda_5}n_y \\ -n_y & n_x & 0 & \frac{\lambda_1-\lambda_5}{\lambda_4-\lambda_5}n_z & \frac{\lambda_1-\lambda_4}{\lambda_4-\lambda_5}n_z \\ 0 & 0 & 0 & \frac{\rho c^2}{\lambda_4-\lambda_5} & \frac{\rho c^2}{\lambda_4-\lambda_5} \end{bmatrix} \tag{2.18}$$

2.1.3 Low Mach Number Preconditioning

In cases the local Mach number approaches zero there is large disparity in the wave propagation speeds. The speed of sound (c) becomes very large compared to the flow velocity (V) and completely deteriorates the stability and convergence properties of the system. In such cases the equations should be modified. Low Mach Preconditioning is applied that acts on the time derivatives of the equations and basically modifies the speed of sound to make the two velocities comparable. By that, the convergence and stability characteristics of the system are improved.

Neglecting the viscous terms, and using the Preconditioning Matrix Γ, the system of equations (2.9) takes the following general form:

$$\begin{aligned} &\Gamma^{-1}\frac{\partial \vec{U}}{\partial t} + A_c\frac{\partial \vec{U}}{\partial \vec{x}} = \frac{\partial \vec{Q}}{\partial \vec{x}} \Rightarrow \\ &\frac{\partial \vec{U}}{\partial t} + \Gamma A_c\frac{\partial \vec{U}}{\partial \vec{x}} = \Gamma\frac{\partial \vec{Q}}{\partial \vec{x}} \Rightarrow \\ &\frac{\partial \vec{U}}{\partial t} + A_\Gamma\frac{\partial \vec{U}}{\partial \vec{x}} = \Gamma\frac{\partial \vec{Q}}{\partial \vec{x}} \end{aligned} \tag{2.19}$$

Various forms of the Preconditioning Matrix Γ have been proposed [70, 71]. In the present work Eriksson's Preconditioning Matrix [72] has been implemented, based on its successful use in [73]. In primitive variables,

$$\Gamma_p = \begin{bmatrix} 1 & 0 & 0 & 0 & \beta \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & \alpha \end{bmatrix} \tag{2.20}$$

where $\alpha = min(1, M_{local}^2, \kappa_p M_\infty^2)$ and $\beta = (1-\alpha)/c^2$. The amount of preconditioning is controlled by α in the sense that it follows the local Mach Number (Ma_{local}) variations. The parameter κ_p takes values from 3-5. On one hand, its role is to prevent α from approaching zero in stagnat regions and on the other to assure constant α within the boundary layer [74]. It is noted that zero preconditioning and recovery of the original form of the equations is obtained for $\alpha = 1$.

In order to express the preconditioning matrix in conservative or characteristic variables, the appropriate transformation matrix is applied. For example,

$$\Gamma_c = M^{-1}\Gamma_p M, \qquad M = \frac{\partial \vec{U}}{\partial \vec{V}} \tag{2.21}$$

When transforming the preconditioned system in characteristic variables, a new set of modified eigenvalues is obtained,

$$\Lambda_\Gamma = \begin{bmatrix} V & 0 & 0 & 0 & 0 \\ 0 & V & 0 & 0 & 0 \\ 0 & 0 & V & 0 & 0 \\ 0 & 0 & 0 & V'+c' & 0 \\ 0 & 0 & 0 & 0 & V'-c' \end{bmatrix} \tag{2.22}$$

in which the "modified" speed of sound c' and velocity V' are defined as follows:

$$\begin{aligned} V' &= \frac{1}{2}(1+\alpha)V \\ c' &= \frac{1}{2}\sqrt{\left[(1-\alpha)\,V\right]^2 + 4\alpha c^2} \end{aligned} \tag{2.23}$$

leading to the following expressions for the eigenvalues:

$$\begin{aligned} \lambda_1 &= \lambda_2 = \lambda_3 = V \\ \lambda_4 &= \frac{1}{2}\left[(1+\alpha)\,V - \sqrt{\left[(1-\alpha)\,V\right]^2 + 4\alpha c^2}\right] \\ \lambda_5 &= \frac{1}{2}\left[(1+\alpha)\,V + \sqrt{\left[(1-\alpha)\,V\right]^2 + 4\alpha c^2}\right] \end{aligned} \tag{2.24}$$

Since preconditioning changes the eigenvalues, the left and right eigenvectors will also change. The right eigenvectors in primitive variables become:

$$R_{\Gamma p} = \begin{bmatrix} n_x & 0 & n_z & -n_y & -\frac{n_x}{c^2} \\ n_y & -n_z & 0 & n_x & -\frac{n_y}{c^2} \\ n_z & n_y & -n_x & 0 & -\frac{n_z}{c^2} \\ 0 & n_x & n_y & n_z & -\frac{\lambda_1-\lambda_4}{\alpha\rho c^2} \\ 0 & -n_x & -n_y & -n_z & \frac{\lambda_1-\lambda_5}{\alpha\rho c^2} \end{bmatrix} \tag{2.25}$$

while the left eigenvectors in primitive variables become :

$$L_{\Gamma p} = R_{\Gamma p}^{-1} = \begin{bmatrix} n_x & n_y & n_z & \frac{\alpha\rho}{\lambda_4-\lambda_5} & \frac{\alpha\rho}{\lambda_4-\lambda_5} \\ 0 & -n_z & n_y & \frac{\lambda_1-\lambda_5}{\lambda_4-\lambda_5}n_x & \frac{\lambda_1-\lambda_4}{\lambda_4-\lambda_5}n_x \\ n_z & 0 & -n_x & \frac{\lambda_1-\lambda_5}{\lambda_4-\lambda_5}n_y & \frac{\lambda_1-\lambda_4}{\lambda_4-\lambda_5}n_y \\ -n_y & n_x & 0 & \frac{\lambda_1-\lambda_5}{\lambda_4-\lambda_5}n_z & \frac{\lambda_1-\lambda_4}{\lambda_4-\lambda_5}n_z \\ 0 & 0 & 0 & \frac{\alpha\rho c^2}{\lambda_4-\lambda_5} & \frac{\alpha\rho c^2}{\lambda_4-\lambda_5} \end{bmatrix} \tag{2.26}$$

2.1.4 Moving Grids

In case D changes in time, the time derivative in (2.1) will also act on $D(t)$. In order to pass time derivation into the integral, Reynold's transport theorem is applied:

$$\begin{aligned} \frac{\partial}{\partial t}\int_{D(t)} \vec{U} dD &= \int_{D(t)} \frac{\partial \vec{U}}{\partial t} + \int_{D(t)} \nabla\left(\vec{U}\cdot\vec{u}_{vol}\right) dD \\ &= \int_{D(t)} \frac{\partial \vec{U}}{\partial t} + \oint_{\partial D(t)} \vec{U}\cdot\left(\vec{u}_{vol}\,\vec{n}\right) dS \end{aligned} \tag{2.27}$$

where $\vec{u}_{vol}$ is the velocity that defines the time evolution of $D(t)$. If $\vec{u}_{grid}$ denotes the grid velocity then $\vec{u}_{vol} = \vec{u}_{grid}$, hence,

$$\int_{D(t)} \frac{\partial \vec{U}}{\partial t} = \frac{\partial}{\partial t}\int_{D(t)} \vec{U} dD - \oint_{\partial D(t)} \vec{U}\cdot\left(\vec{u}_{vol}\vec{n}\right) dS \tag{2.28}$$

Substituting the above expression in (2.1), provides the Arbitrary Lagrangian-Eulerian (ALE) formulation of the governing equations:

$$\frac{\partial}{\partial t}\int_{D(t)} \vec{U} dD + \oint_{\partial D(t)} \left(\vec{F_c} - V_g\vec{U} dS - \vec{F_v}\right) dS = \int_{D(t)} \vec{Q} dD \tag{2.29}$$

where $V_g = \vec{u}_{grid}\cdot\vec{n}$. In the case of moving grids the eigenvalues of the system must change in order to account for the grid velocity and become:

$$\Lambda = \begin{bmatrix} V-V_g & 0 & 0 & 0 & 0 \\ 0 & V-V_g & 0 & 0 & 0 \\ 0 & 0 & V_g & 0 & 0 \\ 0 & 0 & 0 & V-V_g+c & 0 \\ 0 & 0 & 0 & 0 & V-V_g+c \end{bmatrix} \tag{2.30}$$

The above formulation is consistent for all types of motions since no assumptions about the velocity have been made. Thus, it can be used both for rigid body motions (rotation, translation) and for deforming grids where the control volume is stretched and twisted.

Rotating frame of reference

In several applications, the solid bodies are rotating. Often it is preferred to formulate the problem in the rotating frame of reference, especially in cases where the relative flow attains a steady state.

The governing equations in a rotating frame take the following form (see [3] for details):

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{u}_r) = 0 \tag{2.31}$$

$$\frac{\partial \rho \vec{u}_r}{\partial t} + \nabla \cdot \left(\rho \vec{u}_r \vec{u}_r + p\vec{I} - \vec{\tau}\right) = -2\rho \left(\vec{\Omega} \times \vec{u}_r\right) - \rho \vec{\Omega} \times (\vec{\Omega} \times \vec{R}) \tag{2.32}$$

$$\frac{\partial E_r}{\partial t} + \nabla \cdot \left[\vec{u}_r \left(E_r + p\right) + \vec{\tau} \cdot \vec{u}_r\right] = 0 \tag{2.33}$$

where $\vec{u}_r = \vec{u} - (\vec{\Omega} \times \vec{R})$ is the relative velocity and $E_r = p/(\gamma - 1) + 1/2\rho u_r^2 + 1/2\rho|(\vec{\Omega} \times \vec{R})|^2$ is the total energy per unit volume in the rotating frame. Also in the momentum equation the Coriolis and Centrifugal forces are added.

A variant of the above formulation is easily obtained by introducing the absolute flow velocity as expressed in the rotating frame:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \rho \left(\vec{u} - \vec{\Omega} \times \vec{R}\right) = 0 \tag{2.34}$$

$$\frac{\partial \rho \vec{u}}{\partial t} + \nabla \cdot \left[\rho \vec{u}(\vec{u} - \vec{\Omega} \times \vec{R}) + p\vec{I} - \vec{\tau}\right] = -\rho \left(\vec{\Omega} \times \vec{u}\right) \tag{2.35}$$

$$\frac{\partial E}{\partial t} + \nabla \cdot \left[(\vec{u} - \vec{\Omega} \times \vec{R})\left(E + p\right) + \left(\vec{\Omega} \times \vec{R}\right) p + \vec{\tau} \cdot \vec{u}\right] = 0 \tag{2.36}$$

According to [75, 76] the absolute velocity formulation results in a more accurate evaluation of the fluxes in Finite Volume solvers.

2.2 Spatial Discretization

In MaPFlow the flow variables are calculated and stored at cell centers. Assuming that the cell volume remains unchanged:

$$\frac{\partial}{\partial t} \int_D \vec{U} dD = D \frac{\partial \vec{U}}{\partial t} \tag{2.37}$$

where:

$$\vec{U} = \frac{1}{D} \int_D \vec{U}_{exact} dD \tag{2.38}$$

Thus equation (2.1) becomes:

$$\frac{\partial \vec{U}}{\partial t} = -\frac{1}{D}[\oint_{\partial D}(\vec{F}_c - \vec{F}_v)dS - \int_D \vec{Q}dD] \tag{2.39}$$

The surface integral is approximated using piecewise constant fluxes over the cell faces that are calculated at their centers. For cell I,

$$\frac{d\vec{U}_I}{dt} = -\frac{1}{D_I}\underbrace{[\sum_{m=1}^{N_f}(\vec{F}_c - \vec{F}_v)_m \Delta S_m) - (\vec{Q}D)_I]}_{R_I} = -\frac{1}{D_I}\vec{R}_I \tag{2.40}$$

where N_f is the number of faces the cell has and ΔS_m is the area of face "m". The terms $(\vec{F}_c)_m, (\vec{F}_v)_m$ represent the convective and viscous fluxes through face m.

2.2.1 Reconstruction of variables

In order to calculate the fluxes appearing in the right hand side of (2.40), the values of all flow variables at the face centers are needed. This information is absent, since all flow variables are defined at the cell centers. Passing the flow information from the cell centers to the faces is carried out by means of *variable reconstruction.*

Consider two cells *I,J* being in contact over face *f*. Variable reconstruction on *f* can be defined either starting from cell *I* or cell *J*. For compressible solvers it is assumed that across the face the flow experiences a jump defined by the left *L* and right *R* states. The L/R specification depends on the normal to *f* which directs from L to R.

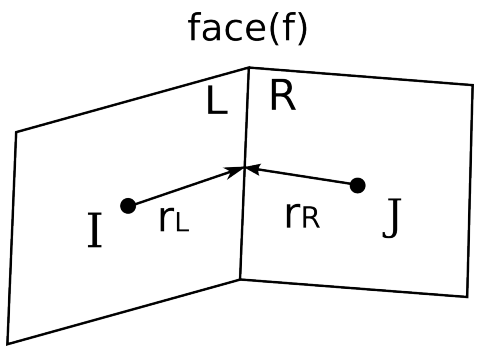


Figure 2.1: Reconstruction of variables on a face (f).

In MaPFlow, the Piecewise Linear Reconstruction (PLR) is used which is formally second order on regular grids [77]. PLR approach implies that the flow variables are linearly distributed over the control volume. Thus the Left and Right reconstructed states are defined as follows:

$$\vec{V}_L = \vec{V}_I + \Psi_I(\nabla\vec{V}_I \cdot \ \vec{r}_L) \tag{2.41}$$

$$\vec{V}_R = \vec{V}_J - \Psi_J(\nabla\vec{V}_J \cdot \ \vec{r}_R) \tag{2.42}$$

where $\vec{r}_L, \vec{r}_R$ denote the distance vectors pointing from the cell centers to the face center(Fig. 2.1) and Ψ a limiter function. In the above expression, the gradients are calculated at the corresponding cell centers using the Green-Gauss formulation:

$$\nabla\vec{V} \approx \frac{1}{D}\int_{\partial D} \vec{V}\vec{n}dS \tag{2.43}$$

which in the Cell-Centered scheme takes the form:

$$\nabla\vec{V}_I \approx \frac{1}{D}\sum_{J=1}^{N_f}\frac{1}{2}(\vec{V}_I + \vec{V}_J)\vec{n}_{IJ}\Delta S_{IJ} \tag{2.44}$$

2.2.2 Limiters

Function Ψ appearing in (2.41) and (2.42) is a limiter function that reduces the gradients $\nabla\vec{V}_I, \nabla\vec{V}_J$. Limiter functions are widely used in compressible solvers in order to ensure convergence in areas with strong gradients. In the present work, the Venkatakrishnan limiter is used due to its good convergence properties [78, 79].

$$\Psi_i = min_j \begin{cases} \frac{1}{\Delta_2}[\frac{\Delta_{1,max}^2+\epsilon^2)\Delta_2+2\Delta_2^2\Delta_{1,max}}{\Delta_{1,max}^2+2\Delta_2^2+\Delta_{1,max}\Delta_2+\epsilon^2}] & \text{if } \Delta_2 > 0 \\ \frac{1}{\Delta_2}[\frac{\Delta_{1,min}^2+\epsilon^2)\Delta_2+2\Delta_2^2\Delta_{1,min}}{\Delta_{1,min}^2+2\Delta_2^2+\Delta_{1,min}\Delta_2+\epsilon^2}] & \text{if } \Delta_2 < 0 \\ 1 & \text{if } \Delta_2 = 0 \end{cases} \tag{2.45}$$

where

$$\Delta_2 = \nabla\vec{V}_i \cdot \vec{r}_i \tag{2.46}$$

$$\Delta_{1,max} = \vec{V}_{max} - \vec{V}_i \tag{2.47}$$

$$\Delta_{1,min} = \vec{V}_{min} - \vec{V}_i \tag{2.48}$$

$\vec{V}_{max}, \vec{V}_{min}$ refer,to the maximum and minimum of $\vec{V}$ of all neighboring cells. The parameter ϵ^2 defines the amount of limiting. In practice ϵ is proportional to the length scale of the grid (Δh),

$$\epsilon^2 = (K\Delta h)^3 \tag{2.49}$$

where K is a free parameter. Small values of K make the limiter strict rendering the PLR first order, while $K = \infty$ leads to an unlimited scheme. Typically the value of $K = 5$ is used.

2.2.3 Convective Fluxes

The discretization of the convective fluxes can be based on central, flux-vector or flux-difference schemes. Central schemes calculate the convective fluxes across faces as the arithmetic average of the values obtained at the two sides of the face plus an artificial dissipation term added to enhance stability [80]. Flux-vector schemes are based on *upwinding* which respects the direction of propagation of waves [81, 82]. Finally, flux-difference schemes calculate convective fluxes at cell faces solving the Riemann problem for the Left and Right states defined on the face [83].

The present work uses Roe's approximate Riemann solver [83], which is a flux-difference scheme. Roe's scheme consists of constructing the convective flux as a sum of wave contributions:

$$(\vec{F_c})_{I+\frac{1}{2}} = \frac{1}{2}[\vec{F_c}(\vec{V_R}) + \vec{F_c}(\vec{V_L}) - |A_{Roe}|_{I+\frac{1}{2}}(\vec{V_R} - \vec{V_L})] \tag{2.50}$$

where the Left and Right states $(\vec{V_L}, \vec{V_R})$ are calculated using (2.41) and (2.42) respectively. The Roe matrix A_{Roe} has the same form as the convective flux Jacobian but instead of formally averaged values, the following Roe-averaged variables are used:

$$\begin{aligned}
\tilde{\rho} &= \sqrt{\rho_L \rho_R} \\
\tilde{u} &= \frac{u_L\sqrt{\rho_L}+u_R\sqrt{\rho_R}}{\sqrt{\rho_L}+\sqrt{\rho_R}} \\
\tilde{v} &= \frac{v_L\sqrt{\rho_L}+v_R\sqrt{\rho_R}}{\sqrt{\rho_L}+\sqrt{\rho_R}} \\
\tilde{w} &= \frac{w_L\sqrt{\rho_L}+w_R\sqrt{\rho_R}}{\sqrt{\rho_L}+\sqrt{\rho_R}} \\
\tilde{H} &= \frac{H_L\sqrt{\rho_L}+H_R\sqrt{\rho_R}}{\sqrt{\rho_L}+\sqrt{\rho_R}} \\
\tilde{c} &= \sqrt{(\gamma-1)(\tilde{H} - \tilde{q}^2/2)} \\
\tilde{q} &= \tilde{u}^2 + \tilde{v}^2 + \tilde{w}^2
\end{aligned}$$

In (2.50), $|A_{Roe}|$ is constructed using the absolute values of the eigenvalues and the the right eigenvector matrix R:

$$|A_{Roe}| = R^{-1}|\Lambda|R \tag{2.51}$$

In case the preconditioned system of equation is used , the eigenvalues and the eigenvectors of the preconditioned system must be used$(\Lambda_\Gamma, R_\Gamma)$. Hence, $|A_{Roe}|$ is changed to (according to [84]) :

$$\begin{aligned}
|A_{\Gamma Roe}| &= |\Gamma^{-1}\Gamma A_{Roe}| \\
&\simeq \Gamma^{-1}|\Gamma A_{Roe}| \\
&\simeq \Gamma^{-1}R_\Gamma^{-1}|\Lambda_\Gamma|R_\Gamma
\end{aligned} \tag{2.52}$$

2.2.4 Viscous Fluxes

For the calculation of the Viscous Fluxes, variable values and space derivatives are needed. For the face in between cells I and J, variable values are obtained from simple averaging:

$$\vec{V}_{IJ} = \frac{1}{2}\left(\vec{V}_I + \vec{V}_J\right) \tag{2.53}$$

while for the gradients, the Green-Gauss formula is applied using the face averaged values $\vec{V}_{IJ}$ as defined in (2.53) but supplemented with a directional derivative[85]:

$$\nabla\vec{V}_{IJ} = \overline{\nabla\vec{V}_{IJ}} - \left[\overline{\nabla\vec{V}_{IJ}} \cdot \vec{t_{IJ}} - \left(\frac{\partial\vec{V}}{\partial l}\right)_{IJ}\right] \cdot \vec{t_{IJ}} \tag{2.54}$$

where,

$$\overline{\nabla\vec{V}_{IJ}} = \frac{1}{2}\left(\nabla\vec{V}_I + \nabla\vec{V}_J\right) \tag{2.55}$$

is the mean gradient,

$$\left(\frac{\partial\vec{V}}{\partial l}\right)_{IJ} \approx \frac{\vec{V}_J - \vec{V}_I}{l_{IJ}} \tag{2.56}$$

and l_{IJ} is the distance between cell centers I and J and $\vec{t_{IJ}}$ is the unit vector pointing from cell center I to cell center J.

2.3 Temporal Discretization

For the temporal discretization the method of lines is used. This means that temporal and spatial discretization are done separately leading for every control volume to the following equation:

$$\frac{d\left(D_I\vec{U}_I\right)}{dt} = -R_I \tag{2.57}$$

In comparison to (2.39) the form of equation (2.57)is more general in the sense that the control volume can vary with time.

Temporal discretization can be either explicit or implicit. Explicit methods use the $\vec{U}^n$ known solution and march in time using the corresponding residual $\vec{R}^n$ to obtain solution at $(t + \Delta t)$. On the other hand the implicit schemes use $R(\vec{U}^{n+1}) = \vec{R}^{n+1}$ to obtain the new solution and are favored because they allow larger time-steps. Since $\vec{R}^{n+1}$ is unknown, the following linear approximation is used:

$$\vec{R}^{n+1} \approx \vec{R}^n + \left(\frac{\partial\vec{R}}{\partial\vec{U}}\right)_n \cdot \Delta\vec{U}^n, \quad \Delta\vec{U}^n = \vec{U}^{n+1} - \vec{U}^n \tag{2.58}$$

In MaPFlow a finite difference scheme is used for the time derivative (see [86]):

$$\frac{1}{\Delta t}\left[\phi_{n+1}\left(D\vec{U}\right)^{n+1}+\phi_n\left(D\vec{U}\right)^{n}+\phi_{n-1}\left(D\vec{U}\right)^{n-1}+\phi_{n-2}\left(D\vec{U}\right)^{n-2}+\dots\right]=-R^{n+1} \tag{2.59}$$

Depending on the choice of ϕ_n the corresponding backwards difference formulae (BDF) of the temporal scheme is defined. $BDF2OPT$, refers to a class of optimized, second-order, backward difference methods with an error constant half as large as the conventional 2nd order scheme [87].

Table 2.1: Backwards Difference Schemes

order	ϕ_{n+1}	ϕ_n	ϕ_{n-1}	ϕ_{n-1}
1st	1	-1	0	0
2nd	3/2	-2	1/2	0
3rd	11/6	-3	3/2	$-1/3$
$BDF2OPT$	$3/2-\phi_{n-2}$	$-2+3\phi_{n-2}$	$1/2-3\phi_{n-2}$	$-0.58/3$

2.3.1 Steady State Computations

Even when steady state simulations are considered a pseudo-unsteady technique is followed. For steady state simulations 1st order scheme is chosen to march the solution in pseudo-time until convergence is reached. At 1st order, after linearizing R^{n+1}, (2.59) becomes:

$$\frac{\left(D_I\Delta\vec{U}_I^n\right)}{\Delta t_I}=\vec{R}_I^n+\left(\frac{\partial\vec{R}}{\partial\vec{U}}\right)_I\Delta\vec{U}_I^n \tag{2.60}$$

By rearranging the terms the final system of discrete equations is obtained in which the system matrix defines the *implicit operator* of the scheme:

$$\underbrace{\left[\frac{(D)_I}{\Delta t_I}+\left(\frac{\partial\vec{R}}{\partial\vec{U}}\right)_I\right]}_{\text{Implicit Operator}}\Delta\vec{U}_I^n=-\vec{R}_I^n \tag{2.61}$$

Local Time Stepping

In order to facilitate convergence, the Local Time Step technique is used [88]. The time step for steady state calculation can be defined using the spectral radii of each cell. For every cell, a different time step is defined by:

$$\Delta t = CFL \frac{D_I}{\left(\hat{\Lambda}_c + C\hat{\Lambda}_v\right)_I} \tag{2.62}$$

where $\hat{\Lambda}_c, \hat{\Lambda}_v$ is the sum of convective and viscous eigenvalues over all cell faces. The convective spectral radii defined by:

$$(\hat{\Lambda}_c)_I = \sum_{J=1}^{N_f} (|\vec{u}_{IJ} \cdot \vec{n}_{IJ}| + c_i j)\Delta S_{IJ} \tag{2.63}$$

and the viscous spectral radii by:

$$(\hat{\Lambda}_v)_I = \frac{1}{D_I} \sum_{J=1}^{N_f} [max(\frac{3}{3\rho_{IJ}}, \frac{\gamma_{IJ}}{\rho_{IJ}})(\frac{\mu_L}{Pr_L} + \frac{\mu_T}{Pr_T})_{IJ}(\Delta S_{IJ})^2] \tag{2.64}$$

2.3.2 Time True Computations

When making Time True computations, temporal discretization is crucial because any remaining numerical error will propagate in the flow as a disturbance. In order to minimize temporal errors higher order schemes should be used in conjunction with the Dual Time-Step technique [89].

Dual Time-Stepping

The Dual Step approach adds an extra time-like derivative in the transport equation that refers to a different "time variable" τ, called "pseudo-time". The conservative variables in the pseudo-time problem are denoted by $U^\star$ because until convergence they don't satisfy the original unsteady problem.

Using this approach the unsteady problem is transformed into a steady one. In every true time-step the following problem is solved in the pseudo-time (τ) :

$$\frac{\partial(D^{n+1}\vec{U}^\star)}{\partial\tau} + \vec{R}^\star = 0 \tag{2.65}$$

Setting,

$$\vec{R}^\star = \frac{\partial(D\vec{U}^\star)}{\partial t} + R(\vec{U}^\star) \tag{2.66}$$

the following final form is obtained:

$$\frac{\partial(D^{n+1}\vec{U}^\star)}{\partial\tau} + \frac{\partial(D\vec{U}^\star)}{\partial t} = -R(\vec{U}^\star) \tag{2.67}$$

When (2.65) converges $R^\star = 0, \vec{U}^\star = \vec{U}$ which satisfies the original unsteady problem.

The discretized form of (2.67) can be written as:

$$\frac{D^{n+1}\Delta U^{\star k}}{\Delta\tau} + \frac{1}{\Delta t}\left[\phi_{n+1}\left(D\vec{U}^{\star}\right)^{n+1} + \phi_n\left(D\vec{U}\right)^{n} + \phi_{n-1}\left(D\vec{U}\right)^{n-1}\right] = -R^{k+1} \tag{2.68}$$

or:

$$\frac{D^{n+1}\Delta U^{\star k}}{\Delta\tau} = -R^{\star k+1} \tag{2.69}$$

with $\Delta U^{\star k} = U^{\star k+1} - U^{\star k}$ and:

$$R^{\star k+1} = R^{k+1} - \frac{1}{\Delta t}\left[\phi_{n+1}\left(D\vec{U}^{\star}\right)^{n+1} + \phi_n\left(D\vec{U}\right)^{n} + \phi_{n-1}\left(D\vec{U}\right)^{n-1}\right] \tag{2.70}$$

where k denotes the steady state problem sub-iteration.

In order to apply an Implicit Scheme in the Dual Time-Step procedure we must linearize the unsteady residual $R^{\star k+1}$:

$$\vec{R}^{\star k+1} \approx \vec{R}^{\star k} + \left(\frac{\partial\vec{R}}{\partial\vec{U}^{\star}}\right)_k \cdot \Delta\vec{U}^{\star k} \tag{2.71}$$

or,

$$\vec{R}^{\star k+1} \approx \vec{R}^{k} - \frac{1}{\Delta t}\left[\phi_{n+1}\left(D\vec{U}^{\star}\right)^{n+1} + \phi_n\left(D\vec{U}\right)^{n} + \phi_{n-1}\left(D\vec{U}\right)^{n-1}\right] + \frac{\partial\vec{R}}{\partial\vec{U}^{\star}}\Delta\vec{U}^{\star} - \phi_{n+1}\frac{D^{n+1}}{\Delta t}\Delta\vec{U}^{\star} \tag{2.72}$$

The correction $\Delta U^{\star}$ refers to the steady problem defined in pseudo-time. Thus, when the steady problem converges $\Delta U^{\star} = 0$. However, this does not mean $U^{n+1} - U^{n} = 0$ but $U^{k+1,n+1} - U^{k,n+1} = 0$. The ϕ coefficients change according to the time discretization scheme desired (see Table 2.1).

Substituting in (2.69) the final form is obtained:

$$\left[\frac{D^{n+1}}{\Delta\tau} + \phi_{n+1}\frac{D^{n+1}}{\Delta t} + \frac{\partial\vec{R}}{\partial\vec{U}^{\star}}\right]\Delta U^{\star k} = -R^{k} - Q^{k}_{dual} \tag{2.73}$$

where the dual step unsteady source-like terms Q^k_{dual} are given by:

$$Q^k_{dual} = \frac{1}{\Delta t}\left[\phi_{n+1}\left(D\vec{U}^{\star}\right)^{n+1} + \phi_n\left(D\vec{U}\right)^{n} + \phi_{n-1}\left(D\vec{U}\right)^{n-1}\right] \tag{2.74}$$

It is noted here that the pseudo time step $\Delta\tau$ is defined as in the Steady state computations using local timestepping (2.62).

2.4 Boundary Conditions

In external aerodynamics the following boundary conditions are needed:

- Far-field Boundaries
- Solid wall Boundaries
- Symmetry Boundaries
- Periodic Boundaries

Before analyzing each one of the boundary condition types, it is important to discuss the concept of dummy cells. Dummy cells are additional virtual cells that extend the computational domain. Their purpose is to provide assistance in calculating the flow variables at the computational domain boundaries. Far-field and solid wall conditions are defined exactly at the boundary face while the other two are applied at the center of the dummy cell.

2.4.1 Far-field Boundaries

In the far-field, it is important to comply with the hyperbolic character of the problem as expressed when formulated in characteristic variables. The information provided is related to the sign of the eigenvalues of the flow state at the far-field boundary and the associated Riemann invariants. Both must be respected and so the far-field boundary conditions must be accordingly defined. The approach followed is based on the characteristics of the 1D Euler equations along the normal direction to the boundary [3], $u = \vec{V} \cdot \vec{n}$. The sign of each of the three eigenvalue $u, u+c, u-c$, defines the direction of propagation while along the corresponding characteristic the associated Riemann invariants $R, R+, R-$ (see Fig. 2.2).

$$R^{\pm} = u \pm \frac{2c}{\gamma - 1}, R = s \tag{2.75}$$

remain constant.

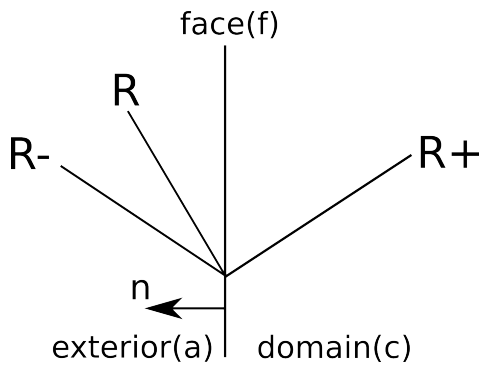


Figure 2.2: The case of a subsonic inlet face. Note that on an inlet face and the normal defined to point outwards, the normal to the boundary velocity component $u = \vec{V} \cdot \vec{n} < 0$. This means that in reality the flow information associated to $R, R-$ is provided by the state defined in (a).

Thus on an inflow face (and similarly for an outflow face), using the invariants:

$$\begin{aligned}
&\text{from } R^+ && u_f + \frac{2c_f}{\gamma - 1} = u_c + \frac{2c_c}{\gamma - 1} \\
&\text{from } R^- && u_f - \frac{2c_f}{\gamma - 1} = u_a - \frac{2c_a}{\gamma - 1} \\
&\text{isentropic assumption} && s_f = s_a
\end{aligned} \tag{2.76}$$

Although the flow variables at boundary faces can be obtained as linear combinations of these invariants, in the present formulation, the characteristic equations are used instead:

$$\begin{aligned}
d\rho - \frac{1}{c^2}dp = 0 \qquad &\text{along} \qquad \lambda_1 = u \\
du + \frac{1}{\rho c}dp = 0 \qquad &\text{along} \qquad \lambda_2 = u + c \\
du - \frac{1}{\rho c}dp = 0 \qquad &\text{along} \qquad \lambda_3 = u - c
\end{aligned} \tag{2.77}$$

Supersonic Inlet-Outlet

(a) Supersonic Inlet (b) Supersonic Outlet

Figure 2.3: Riemann Invariants on a far-field supersonic boundary

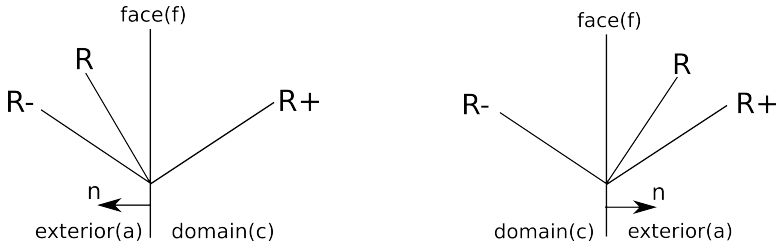
In the supersonic case all eigenvalues are positive since $V > c$. This means that in the case of inflow, all flow information is propagating from outside into the domain (Fig. 2.3) and therefore all flow variables must be given as input:

$$\vec{V}_{inlet} = \vec{V}_{\infty} \tag{2.78}$$

On the contrary in the case of outflow all flow information propagates from inside of the domain:

$$\vec{V}_{outlet} = \vec{V}_{domain} \tag{2.79}$$

Subsonic inlet-outlet



(a) Subsonic Inlet (b) Subsonic Outlet

Figure 2.4: Riemann Invariants on a far-field subsonic boundary

For subsonic inflow and along the normal to the boundary, the two characteristics propagate information from outside of the domain while the third propagates flow information from inside of the domain (see Fig. 2.4). The situation reverses in case of subsonic outflow where two characteristics propagate information from inside the domain while the third propagates information from outside into the flow domain. Note that the three Riemann invariants, associated with the eigenvalues,

are defined with respect to the normal direction. Thus for the subsonic inlet with the normal vector pointing outwards $u - c < 0$, $u < 0$, $u + c > 0$ while for the subsonic outlet $u - c < 0$, $u > 0$, $u + c > 0$.

Based on (2.77), with the normal pointing outwards of the inlet face:

$$\begin{aligned} \frac{p_f}{\rho_f^\gamma} &= \frac{p_a}{\rho_a^\gamma} && \text{along} && \lambda_1 = u, (R) \\ p_f - p_c + \frac{1}{\rho_a c_a}(u_f - u_c) &= 0 && \text{along} && \lambda_2 = u + c, (R+) \\ p_f - p_a - \frac{1}{\rho_a c_a}(u_f - u_a) &= 0 && \text{along} && \lambda_3 = u - c, (R-) \end{aligned} \tag{2.80}$$

By combining R^+ and R^- pressure and velocity at the boundary are determined. Density can be retrieved from the isentropic relation. As reference state at the inlet, that at the exterior of the domain is used. Similarly, under the assumption that the normal direction is pointing outside of the domain, at the outlet boundary,

$$\begin{aligned} \frac{p_f}{\rho_f^\gamma} &= \frac{p_c}{\rho_c^\gamma} && \text{along} && \lambda_1 = u, (R) \\ p_f - p_c + \frac{1}{\rho_c c_c}(u_f - u_c) &= 0 && \text{along} && \lambda_2 = u + c, (R+) \\ p_f - p_a - \frac{1}{\rho_c c_c}(u_f - u_a) &= 0 && \text{along} && \lambda_3 = u - c, (R-) \end{aligned} \tag{2.81}$$

in which the reference state is defined from inside of the computational domain.

Preconditioned characteristic equations

As already discussed in Low Mach Preconditioning (Section 2.1.3) , the eigenvalues and eigenvectors of the system change (2.24). Hence, it is expected that the characteristic equations change too. When Eriksson's Preconditioning Matrix is used, like in the present case, the 1-D characteristic equations take the form [90]:

$$\begin{aligned} d\rho - \frac{1}{c^2}dp &= 0 && \text{along} && \lambda_1 = u \\ du + \frac{c' - \alpha_m u}{\rho \alpha c^2}dp &= 0 && \text{along} && \lambda_2 = u' + c' \\ du - \frac{c' + \alpha_m u}{\rho \alpha c^2}dp &= 0 && \text{along} && \lambda_3 = u' - c' \end{aligned} \tag{2.82}$$

with,

$$\begin{aligned} \alpha_m &= \frac{(1-\alpha)}{2} \\ u' &= \frac{1}{2}(1+\alpha)\, u \\ c' &= \frac{1}{2}\sqrt{[(1-\alpha)\, u]^2 + 4\alpha c^2} \end{aligned} \tag{2.83}$$

As in the un-preconditioned case, the characteristic equations are combined to obtain $\rho, \vec{u}, p$ at the boundary face,

$$
\begin{aligned}
\rho_f &= \rho_o \left(\frac{p_b}{p_o}\right)^{\gamma} \\
V_{nf} &= V_{nc} + \left(\frac{1}{\rho c'}\right)\left(\frac{p_c - p_a}{2}\right) - \left(1 - \frac{\alpha_m V_{no}}{c'}\right)\left(\frac{V_{nc} - V_{na}}{2}\right) \\
p_f &= p_a + \left(1 - \frac{\alpha_m V_{no}}{c'}\right)\left(\frac{p_c - p_a}{2}\right) + \left(\frac{\rho \alpha c^2}{c'}\right)\left(\frac{V_{nc} - V_{na}}{2}\right) \\
u_f &= u_o + (V_{nf} - V_{no}) \cdot n_x \\
v_f &= v_o + (V_{nf} - V_{no}) \cdot n_y \\
w_f &= w_o + (V_{nf} - V_{no}) \cdot n_z
\end{aligned}
\tag{2.84}
$$

where $V_{n(\cdot)} = u_{(\cdot)} \cdot n_x + v_{(\cdot)} \cdot n_y + w_{(\cdot)} \cdot n_z$. The reference state, here denoted by subscript o, is decided by the isentropic wave, λ_1. At a subsonic inlet face it corresponds to the outer flow data whereas at a subsonic outlet face the values from inside of the computational domain are used.

2.4.2 Wall Boundary Conditions

Inviscid Wall

When the fluid is assumed inviscid on solid boundaries,

$$(\vec{u} - \vec{u}_g) \cdot \vec{n} = 0 \tag{2.85}$$

where $\vec{u}_g$ denotes the grid velocity. Density and pressure are set equal to their values at the cell center next to the wall,

$$p_w = p_I,\ \rho_w = \rho_I \tag{2.86}$$

Viscous Wall

In the general case, the fluid is viscous and the no slip wall condition is applied,

$$\vec{u} = \vec{u}_g \tag{2.87}$$

Density and pressure are treated as in the inviscid case. Regardless the assumptions made for the fluid, the convective fluxes take the form,

$$\vec{F}_{cwall} = \begin{pmatrix} 0 \\ n_x p_w \\ n_y p_w \\ n_z p_w \\ p_w V_g \end{pmatrix} \tag{2.88}$$

where $V_g = \vec{u}_g \cdot \vec{n}$

2.4.3 Symmetry and Periodic Boundary Conditions

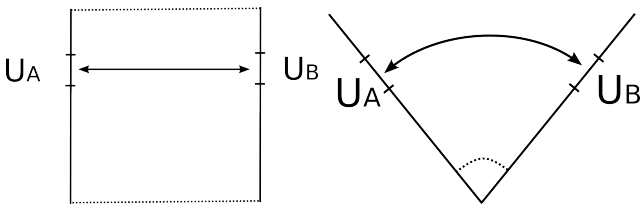
Symmetry

When the flow is symmetric with respect to a plane, the simulation can only concern half of the flow domain. Over the symmetry plane, the condition imposed resembles that applied to an Inviscid Wall. In both cases there is no flux across the boundary. However, in symmetry conditions gradients normal to the boundary must also vanish [85]. So,

$$\begin{aligned} \vec{u} \cdot \vec{n} &= 0, \qquad \text{zero flux} \\ \nabla \vec{U} \cdot \vec{n} &= 0, \qquad \text{zero normal gradients} \end{aligned} \tag{2.89}$$

Periodic Flows

If the flow is periodic, the simulation is conducted on one single period. In such cases, parts of the outer boundary of the grid will correspond to the periodic surfaces on which explicit boundary data are not available. Closure of the problem is done by the Periodic boundary conditions that basically transfer the flow variables from one periodic surface to the next one. Periodicity can be either translational (e.g. as in the case of a wing equipped with an array of vortex generators) or rotational (e.g. as in the case of multi-bladed rotors) depending on the motion required for collapsing a periodic surface to its associate. The implementation is based on ghost cells [85].



(a) Translational Periodicity (b) Rotational Periodicity

Figure 2.5: Periodic Boundary Conditions

Let A, B denote the two associated periodic surfaces (Fig. 2.5). For every cell in contact with A, it's corresponding ghost cell is associated to a cell in contact with B. Scalar quantities are transferred from B to the ghost cells of A as they are, while vector (and tensor) variables follow the motion required for collapsing A on B. The motion is defined as a transformation matrix R_A, and so,

$$\begin{aligned} U_A &= U_B \\ \vec{U}_A &= R_A \vec{U}_B \end{aligned} \tag{2.90}$$

In translational periodicity $R_A = I$ while in rotational periodicity R_A is the corresponding rotation matrix.

2.5 Turbulence Modeling

In order to account for turbulence modeling, flow variables are split in their mean:

$$\overline{u_i} = \lim_{T\to\infty} \frac{1}{T} \int_t^{t+T} u_i dt \tag{2.91}$$

and fluctuating parts u_i':

$$u_i = \overline{u}_i + u_i' \qquad \text{with} \qquad \overline{u_i'} = 0 \qquad \text{but} \qquad \overline{u_i' u_j'} \neq 0 \tag{2.92}$$

The above is known as Reynold's time averaging and is suitable for statistically stationary turbulence. In compressible flows due to the fluctuation of density, the Favre (Mass) Averaging is applied:

$$\widetilde{u_i} = \frac{1}{\overline{\rho}} \lim_{T\to\infty} \frac{1}{T} \int_t^{t+T} \rho u_i dt \tag{2.93}$$

$$u_i = \widetilde{u_i} + u_i'' \quad \text{with} \quad \widetilde{\rho u_i} = \overline{\rho}\tilde{u}_i, \quad \overline{\rho u''} = 0 \quad \text{but} \quad \overline{u_i''} \neq 0 \tag{2.94}$$

Favre's averaging is similar to the Reynold's one but not identical. Again, $\widetilde{u_i''} = 0$ and $\widetilde{u_i'' u_j''} \neq 0$.

Application of Favre's averaging to the governing equations, leads to a considerably more complex system. Thus, Reynold's averaging is only applied to density and pressure while Favre's averaging to all other variables [85]. Dropping the bar and the tilde sign for the averaged variables:

$$\begin{aligned}
\frac{\partial \rho}{\partial t} + \frac{\partial \rho u_i}{\partial x_i} &= 0 \\
\frac{\partial \rho u_i}{\partial t} + \frac{\partial \rho u_i u_j}{\partial x_j} &= -\frac{\partial p}{\partial t} + \frac{\partial}{\partial x_j}\left(\tau_{ij} - \rho \widetilde{u_i'' u_j''}\right) \\
\frac{\partial \rho E}{\partial t} + \frac{\partial \rho u_j H}{\partial x_j} &= \frac{\partial}{\partial x_j}\left(k\frac{\partial T}{\partial x_j} - \rho \widetilde{u_j'' h''} + \widetilde{\tau_{ij} u_i''} - \rho \widetilde{u_j'' K}\right) \\
&\quad + \frac{\partial}{\partial x_j}\left[u_i\left(\tau_{ij} - \rho \widetilde{u_i'' u_j''}\right)\right]
\end{aligned} \tag{2.95}$$

where $\rho K = 1/2 \rho \widetilde{u_i'' u_i''}$ denotes the Turbulent Kinetic Energy.

The above system defines the compressible Reynolds-averaged Navier-Stokes equations or the Favre-averaged Navier-Stokes equations. By introducing the Favre-Averaged Reynolds-stress tensor as:

$$\tau_{ij}^F = -\overline{\rho} \widetilde{u_i'' u_j''} \tag{2.96}$$

and by neglecting temperature variations, molecular diffusion of K and turbulent transport, the equations (2.95) become:

$$\begin{aligned}
\frac{\partial \rho}{\partial t} + \frac{\partial \rho u_i}{\partial x_i} &= 0 \\
\frac{\partial \rho u_i}{\partial t} + \frac{\partial \rho u_i u_j}{\partial x_j} &= -\frac{\partial p}{\partial t} + \frac{\partial}{\partial x_j}\left(\tau_{ij} + \tau_{ij}^F\right) \\
\frac{\partial \rho E}{\partial t} + \frac{\partial \rho u_j H}{\partial x_j} &= \frac{\partial}{\partial x_j}\left[u_i\left(\tau_{ij} + \tau_{ij}^F\right)\right]
\end{aligned} \tag{2.97}$$

Eddy-Viscosity Hypothesis

The turbulence models implemented in MaPFlow are first order closures based on the Boussinesq approximation for the Reynold's stresses:

$$\tau_{ij}^F = 2\mu_T\left(S_{ij} - \frac{1}{3}\frac{\partial u_k}{\partial x_k}\right) - \frac{2}{3}K\delta_{ij} \tag{2.98}$$

where:

$$S_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) \tag{2.99}$$

and μ_T denotes the turbulent molecular viscosity. Depending on the specific turbulence model, turbulent kinetic energy is either used or neglected in the state equation,

$$p = (\gamma - 1)\left[\rho E - \rho\frac{u^2 + v^2 + w^2}{2} - \rho K\right] \tag{2.100}$$

2.5.1 Spalart-Allmaras Turbulence Model

The Spalart-Allmaras model [60] solves the following transport equation for $\tilde{\nu}$:

$$\mu_T = \rho\tilde{\nu} f_{v1} \tag{2.101}$$

$$\begin{aligned}
\frac{\partial \rho\tilde{\nu}}{\partial t} + \frac{\partial}{\partial x_j}(\rho\tilde{\nu}u_j) =& \rho C_{b1}\left[1 - f_{t2}\right]\tilde{S}\tilde{\nu} - \rho\left[C_{w1}f_w - \frac{C_{b1}}{\kappa^2}f_{t2}\right]\left(\frac{\tilde{\nu}}{d}\right)^2 + \\
& \frac{\rho}{\sigma}\left[\frac{\partial}{\partial x_j}\left((\nu + \tilde{\nu})\frac{\partial\tilde{\nu}}{\partial x_j}\right) + C_{b2}\frac{\partial\tilde{\nu}}{\partial x_i}\frac{\partial\tilde{\nu}}{\partial x_i}\right]
\end{aligned} \tag{2.102}$$

in which,

$$\tilde{S} = \Omega + \frac{\tilde{\nu}}{\kappa^2 d^2}f_{v2} \qquad \text{where} \qquad \Omega = \sqrt{2W_{ij}W_{ij}} \qquad \text{with} \tag{2.103}$$

$$W_{ij} = \frac{1}{2}\left(\frac{\partial u_j}{\partial x_i} - \frac{\partial u_i}{\partial x_j}\right) \tag{2.104}$$

d is defined as the distance from the cell center to the nearest wall and

$$f_{v1} = \frac{\chi^3}{\chi^3 + C_{v1}}, \qquad \chi = \frac{\tilde{\nu}}{\nu} \tag{2.105}$$

$$f_{v2} = 1 - \frac{\chi}{1 + \chi f_{v1}}, \qquad f_w = g\left[\frac{1 + c_{w3}^6}{g^6 + c_w 3}^6\right]\frac{1}{6} \tag{2.106}$$

$$g = r + c_{w2}(r^6 - r), \qquad r = min\left[\frac{\tilde{\nu}}{\tilde{S}\kappa^2 d^2}, 10\right] \tag{2.107}$$

$$f_{t2} = c_{t3}\exp(-c_{t4}\chi^2) \tag{2.108}$$

$$\begin{aligned} &C_{b1} = 0.1355 \qquad \sigma = 2/3 \qquad C_{b2} = 0.622 \qquad k = 0.41 \\ &C_{w2} = 0.3 \qquad C_{w3} = 2 \qquad C_{v1} = 7.1 \qquad C_{t3} = 1.2 \qquad C_{t4} = 0.5 \\ &C_{w1} = \frac{Cb1}{\kappa^2} + \frac{1 + C_{b2}}{\sigma} \end{aligned} \tag{2.109}$$

In order to avoid numerical problems, the production term $\tilde{S}$ must be kept positive. Thus, $\tilde{S} = max(\tilde{S}, 10^-20)$. Also, in case the equation tends to yield a negative solution for $\tilde{\nu}$ then it is recommended to set $\tilde{\nu}$ equal to a small positive number.

Boundary Conditions

The boundary conditions as suggested in [60] are:

$$\tilde{\nu}_{wall} = 0, \qquad \tilde{\nu}_\infty = [3 - 5]\,\nu\infty \tag{2.110}$$

In case transition from laminar to turbulent flow is considered, the boundary conditions become:

$$\tilde{\nu}_{wall} = 0, \qquad \tilde{\nu}_\infty = 0.01\nu\infty \tag{2.111}$$

2.5.2 Menter $k - \omega$ SST Turbulence Model

Menter's Shear Stress Transport (SST) turbulence model [91] is a modification of Wilcox's two equation Eddy-Viscosity model [13] defined for the turbulence kinetic energy K and the specific dissipation rate ω. The transport equations of k and ω are given below:

$$\frac{\partial \rho K}{\partial t} + \frac{\partial}{\partial x_j}(\rho K u_j) = P - \beta^\star \rho \omega K + \frac{\partial}{\partial x_j}\left[(\mu + \sigma_K \mu_T)\frac{\partial K}{\partial x_j}\right] \tag{2.112}$$

$$\frac{\partial \rho \omega}{\partial t} + \frac{\partial}{\partial x_j}(\rho \omega u_j) = \frac{\gamma}{\nu_T} P - \beta^\star \rho \omega^2 + \frac{\partial}{\partial x_j}\left[(\mu + \sigma_\omega \mu_T)\frac{\partial K}{\partial x_j}\right] + 2(1 - F_1)\frac{\rho \sigma_{\omega 2}}{\omega}\frac{\partial K}{\partial x_j}\frac{\partial \omega}{\partial x_j} \tag{2.113}$$

where:

$$P = \tau_{ij}\frac{\partial u_i}{\partial x_j}, \qquad \tau_{ij} = \mu_T\left(2S_i - \frac{2}{3}\frac{\partial u_k}{\partial x_k}\right) - \frac{2}{3}\rho K \delta_{ij} \tag{2.114}$$

S_{ij} is the stress tensor given by (2.99) and $\nu_T = \mu_T/\rho$. Eddy viscosity μ_T is given by:

$$\mu_T = \frac{\rho \alpha_1 K}{max(\alpha_1 \omega, \Omega F_2)} \tag{2.115}$$

with Ω being the vorticity magnitude as in (2.104).

The constants for Menter's SST turbulence model are a blend of inner (1) and outer (2) constants defined by the following weighted expression:

$$\phi = F_1 \phi_1 + (1 - F_1)\phi_2 \tag{2.116}$$

with F_1 being defined by:

$$F_1 = tanh(\arg_1^4), \qquad \arg_1 = min\left[max\left(\frac{\sqrt{K}}{\beta^\star \omega d}, \frac{500\nu}{d^2 \omega}\right), \frac{4\rho \sigma_{\omega 2} K}{CD_{K\omega} d^2}\right] \tag{2.117}$$

$$CD_{k\omega} = max\left(2\rho \sigma_{\omega 2}\frac{1}{\omega}\frac{\partial K}{\partial x_j}\frac{\partial \omega}{\partial x_j}, 10^{-20}\right) \tag{2.118}$$

$$F_2 = tanh(\arg_2), \qquad \arg_2 = max\left(2\frac{\sqrt{K}}{\beta^\star \omega d}, \frac{500\nu}{d^2 \omega}\right) \tag{2.119}$$

In the above, d is the distance from the cell center to the viscous wall boundary.

The inner constants (those denoted with subscript 1) are:

$$\gamma_1 = \frac{\beta_1}{\beta^\star} - \frac{\sigma_{\omega 1}\kappa^2}{\sqrt{\beta^\star}}, \tag{2.120}$$

$$\sigma_{k1} = 0.85, \qquad \sigma_{\omega 1} = 0.5, \qquad \beta_1 = 0.075$$

$$\beta^\star = 0.09, \qquad \kappa = 0.41, \qquad \alpha_1 = 0.31$$

The outer constants (those with subscript 2) are:

$$\gamma_2 = \frac{\beta_2}{\beta^\star} - \frac{\sigma_{\omega 2}\kappa^2}{\sqrt{\beta^\star}} \tag{2.121}$$
$$\sigma_{k2} = 1.0, \qquad \sigma_{\omega 2} = 0.856, \qquad \beta_2 = 0.0828$$

In [59], a limiter on the production term in the K equation is recommended. Hence, the production term in the K-equation is replaced by:

$$P = min(P, 20\beta^\star \rho K \omega) \tag{2.122}$$

Boundary Conditions

The boundary conditions as defined in [91] are:

$$\frac{U_\infty}{L} < \omega_\infty < 10\frac{U_\infty}{L}, \qquad \frac{10^{-5}U_\infty^2}{Re_L} < K_\infty < \frac{10^{-1}U_\infty^2}{Re_L} \tag{2.123}$$
$$\omega_{wall} = \frac{6\nu}{\beta_1 d_1^2}, \qquad K_{wall} = 0 \tag{2.124}$$

where L is the approximate length of the computational domain and d_1 the distance to the next point off the wall.

2.5.3 Remarks on the formulation of turbulence equations

Even though both models were presented in their differential form, their integral form is actually used. The transport equation for a generic turbulence model variable U_T, takes the form,

$$\int_D \frac{\partial U_T}{\partial t} dD + \int_D \frac{\partial}{\partial x_j}\left(\rho u_j U_T\right) dD - \int_D \rho \frac{\partial}{\partial x_j}\left[\dots \frac{\partial U_T}{\partial x_j}\right] dD = \int_D \rho(S_{Prod} + S_{Destr}) dD \tag{2.125}$$

which by means of the Green Gauss formula is transformed into:

$$\int_D \frac{\partial U_T}{\partial t} dD + \oint_{\partial D} \left(\rho u_j U_T\right) \vec{n} dS - \oint_{\partial D} \rho(\dots) \frac{\partial U_T}{\partial x_j} \vec{n} dS = \int_D \rho(S_{Prod} + S_{Destr}) dD \tag{2.126}$$

where $F_{Tc} = \left(\rho u_j U_T\right)\vec{n}$, $F_{Tv} = \rho(\dots)\frac{\partial U_T}{\partial x_j}\vec{n}$ denote the convective and viscous fluxes respectively and S_{Prod}, S_{Destr} the Production and Destruction source terms.

From (2.126) the following differential form is obtained,

$$\frac{D\,U_T}{Dt} = S_{Prod} + Diff - S_{Destr} \tag{2.127}$$

$$\frac{DU_T}{Dt} = \frac{\partial U_T}{\partial t} + u_j \frac{\partial U_T}{\partial x_j} = \frac{\partial U_T}{\partial t} + \frac{\partial}{\partial x_j}(u_j U_T) - U_T \frac{\partial u_j}{\partial x_j} \tag{2.128}$$

$$Diff = \frac{\partial}{\partial x_j}\left[\dots \frac{\partial (U_T)}{\partial x_j}\right] \tag{2.129}$$

For incompressible flows, $\partial u_j / \partial x_j = 0$ and thus:

$$\frac{\partial U_T}{\partial t} + u_j \frac{\partial U_T}{\partial x_j} = \frac{\partial U_T}{\partial t} + \frac{\partial}{\partial x_j}(u_j U_T) \tag{2.130}$$

However, when the compressible equations are solved $\partial u_j / \partial x_j \neq 0$. In order to bring (2.128) in conservative form, the continuity equation is used,

$$\frac{\partial U_T}{\partial t} + u_j \frac{\partial U_T}{\partial x_j} + U_T \left(\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_j}(\rho u_j)\right) =$$
$$\frac{1}{\rho}\frac{\partial}{\partial t}(\rho U_T) + \frac{1}{\rho}\frac{\partial}{\partial x_j}(\rho u_j U_T) \tag{2.131}$$

leading to the final conservative form of the turbulence model equation:

$$\frac{\partial}{\partial t}(\rho U_T) + \frac{\partial}{\partial x_j}(\rho u_j U_T) - \rho \frac{\partial}{\partial x_j}\left[\dots \frac{\partial U_T}{\partial x_j}\right] = \rho \cdot (S_{Prod} - S_{Destr}) \tag{2.132}$$

$$\tag{2.133}$$

2.5.4 Discretization

Convective terms:

For the convective term first order up-winding is applied:

$$\rho u_j n_j U_T = max(\rho_L V_{Ln} U_{TL}, 0) + min(\rho_R V_{Rn} U_{TR}, 0) \tag{2.134}$$

in which the left and right states are the values at the cell, centers I, J and the corresponding contravariant velocity.

$$U_L = U_I, \qquad U_R = U_J, \qquad V_{(\cdot)n} = \vec{u}_{(\cdot)} \cdot \vec{n} \tag{2.135}$$

Diffusion terms

The discretization of the diffusion terms consists of first order central differencing. The value of the diffusion terms at a face is taken as an arithmetic mean of the values at the center of the cells sharing the face. The gradient that appears on that term is calculated in the same manner as the gradients of flow variables that contribute to the governing equation diffusion terms (2.54).

Finally the discretization of the temporal and source terms is done in the same manner as for the governing equations.

2.6 Solution of the System

The final form of the discrete equations corresponds to a linear system

$$AX = B \tag{2.136}$$

of large dimension. The above system can be either solved directly or iteratively. Direct solvers are accurate but have demanding memory requirements and are not easily parallelised. On the contrary, iterative solvers might need many iterations to converge but are suitable for parallel coding and have limited memory requirements. Therefore in the present work, an iterative solver was chosen. Following the work of [92], the Jacobi iterative solver was implemented.

Noting that all but one terms in the discrete form of the equations for cell I,

$$\left[\frac{(D)_I}{\Delta t_I} + \left(\frac{\partial \vec{R}}{\partial \vec{U}}\right)_I\right] \Delta \vec{U}_I^n = -\vec{R}_I^n \tag{2.137}$$

refer to the cell in consideration, the following splitting

$$D_I \Delta \vec{U}_I^n + O_I \sum \Delta \vec{U}_J^n = -\vec{R}_I^n \tag{2.138}$$

is introduced. In (2.138), the first term is block diagonal and is linked to cell I, while the second contains the off diagonal contributions in (2.137) that are linked to $\left(\frac{\partial \vec{R}}{\partial \vec{U}}\right)_I$. This term involves the cells that surround I.

Jacobi iterative solver

Equation 2.138 can be easily solved iteratively using the Jacobi method:

$$D_I \Delta \vec{U}_I^{n,k+1} = -\vec{R}_I^n - O_I \sum \Delta \vec{U}_J^{n,k} \tag{2.139}$$

where k is the Jacobi iterations index.

Gauss-Seidel iterative solver

As an alternative, the Gauss-Seidel method can be used. It is similar to Jacobi solver, except the fact that the off diagonal terms are calculated using the current update for $\vec{U}$,

$$D_I \Delta \vec{U}_I^{n,k+1} = -\vec{R}_I^n - O_I \sum \Delta \vec{U}_L^{n,k+1} - O_I \sum \Delta \vec{U}_J^{n,k} \tag{2.140}$$

where U_L concerns the cell values that have been updated in $k+1$ iteration.

The performance of the Gauss-Seidel method strongly depends on the type of the matrix A in (2.136). If A is banded, the matrix can be split in an Upper and Lower part and thus Gauss-Seidel becomes:

$$D_I \Delta \vec{U}_I^{n,k+1} = -\vec{R}_I^n - O_I \sum \Delta \vec{U}_L^{n,k+1} - O_I \sum \Delta \vec{U}_R^{n,k} \tag{2.141}$$

However if the sparsity of A is substantial, the Gauss-Seidel solver has the same convergence properties as the Jacobi method [93].

In case the grid is structured, the matrix is indeed banded and the Gauss-Seidel solver will behave well. On the contrary, if an unstructured grid is used, because the band width of the matrix depends on the cell numbering, good performance is directly linked to proper renumbering. In this respect the Reverse Cuthill-Mckee (RCM) reordering scheme [93] substantially reduces the band-width and therefore the Gauss-Seidel methods outperforms the Jacobi solver.

It is important to note here that in a parallel environment even if Gauss-Seidel iterative solver is used the update ΔU_J^n must remain in the k iteration if U_J is a multi-block ghost cell. The reason behind this constraint is to ensure that the solution will be continuous across the blocks at all times.

2.7 Deforming Grids

Often the grid must deform, as in the case of a deformable trailing edge flap [94] or fluid-structure interaction. In such cases, on one hand the grid deformation must ensure that the grid lines do not overlap and that the change of the cell volume is taken into account.

For grid deformation, the work by Zhao [95] was followed. The idea in Zhao's scheme is to propagate the displacements of the solid boundaries into the grid without changing the far-field boundary while keeping the same grid topology. This is carried out at nodal level as follows:

$$\vec{dr}(node) = f(node)\vec{dr}(node_{wall}) \tag{2.142}$$

where $\vec{dr}$ is the displacement of the any grid node, $\vec{dr}(node_{wall})$ is the displacement of a node on the solid boundary and f is the propagation function. For a two-dimensional problem:

$$f(x) = \frac{ly^2(x)}{lx^2(x) + ly^2(y)} \tag{2.143}$$

$$\begin{aligned} lx(x) &= \frac{1 - \exp(-d(x)/d_{max})}{(e-1)/e} \\ ly(x) &= \frac{1 - \exp(1 - d(x)/d_{max})}{(e-1)/e} \end{aligned} \tag{2.144}$$

where $d(x)$, is the distance of the node to the nearest solid node and d_{max} is the maximum distance of all nodes from the solid boundary.

Grid deformation will render the cell volume $D(t)$ time dependent. Thomas and Lombard [96] proposed the so called Geometric Conservation Law (GCL)

$$\frac{d}{dt}\int_{D(t)} dD = \oint_{\partial D(t)} \vec{V}_g \cdot \vec{n} dS \tag{2.145}$$

The principle of GCL is that a uniform flow solution must remain unchanged regardless of the grid motion.

Various numerical implementations of the GCL are found in the literature (e.g [97]). In the present work the implementation in [98] is adopted, which consists of adding a source term to the original equations. Starting from the integral form of the equations and assuming volume averaged approximation,

$$\frac{d}{dt}(\vec{U}D) + R = 0 \tag{2.146}$$

it follows that

$$\frac{d\vec{U}}{dt}D + \frac{dD}{dt}\vec{U} + R = 0 \tag{2.147}$$

So by introducing (2.145),

$$\frac{d\vec{U}}{dt}D + R = -\vec{U}\oint_{\partial D(t)} \vec{V}_g \cdot \vec{n}dS \tag{2.148}$$

and applying (2.59), the following discrete formulation is obtained,

$$\frac{1}{\Delta t}\left[\phi_{n+1}\vec{U}^{n+1} + \phi_n\vec{U}^n + \phi_{n-1}\vec{U}^{n-1} + \phi_{n-2}\vec{U}^{n-2} + \dots\right]\cdot D^{n+1} = -R^{n+1} - \vec{U}\oint_{\partial D(t)} \vec{V}_g\cdot\vec{n}dS \tag{2.149}$$

It is noted that for rigid body motions $\oint_{\partial D(t)} \vec{V}_g \cdot \vec{n}dS \approx 0$.

2.8 The jBAY Model for Vortex Generators

As an add-on to the capabilities of MaPFlow, the modeling of Vortex Generators (VG) has been implemented as described by the jBAY model [99]. The principle of the specific model is to approximate the effect induced by the presence of the VGs through cell distributed body forces. This gives additional source terms in the momentum and energy equations. Gradually, these source terms force the flow to get aligned with the VG vane. Upon convergence the additional source terms vanish leading to the problem solution. The implementation that was carried out, closely follows [100].

The force added in the momentum equation is of the following form:

$$\vec{L}_i = c_{VG}S_{VG}\frac{D_i}{\sum D_i}\alpha\rho|\vec{u}|^2\hat{l} \tag{2.150}$$

where c_{VG} is a model constant (usually $c_{VG} = 10$), S_{VG} is the VG planform area, D_i is the grid cell Volume in which the force is applied, $\sum D_i$ is the total volume of the cells approximating the VG vane, ρ is the fluid density and $\vec{u}$ is the flow velocity at the cell center. The remaining terms $\alpha, \hat{l}$ refer to the geometrical representation of the VG vane.

Let $\hat{b}$, $\hat{t}$ and $\hat{n}$ define a local coordinate system attached to the VG as shown in Figure 2.6. Assuming that the angle of attack is small, linear aerodynamic theory can be applied. In this case, the direction of the force $\hat{l}$ will lay in the $\hat{t} - \hat{n}$ plane and will be perpendicular to the direction of the local velocity $\hat{u} = \frac{\vec{u}}{|\vec{u}|}$,

$$\hat{l} = \hat{u} \times \hat{b} \tag{2.151}$$

while the force coefficient will be proportional to the angle of attack α,

$$\alpha \approx sin\alpha = \cos(\frac{\pi}{2} - \alpha) = \hat{u} \cdot \hat{n} \tag{2.152}$$

Finally in order to account for the loss of side forces in high angles of attack, the force in 2.150 is multiplied by $\hat{u} \cdot \hat{t}$ [100],

$$\vec{L}_i = c_{VG} S_{VG} \frac{V_i}{\sum V_i} \rho |\vec{u}|^2 \hat{(}\hat{u} \cdot \hat{n})(\hat{u} \times \hat{b})(\hat{u} \cdot \hat{t}) \tag{2.153}$$

The extra source term in the energy equations is simply $\vec{u} \cdot \vec{L}_i$ and corresponds to the work done by the force on the VG.

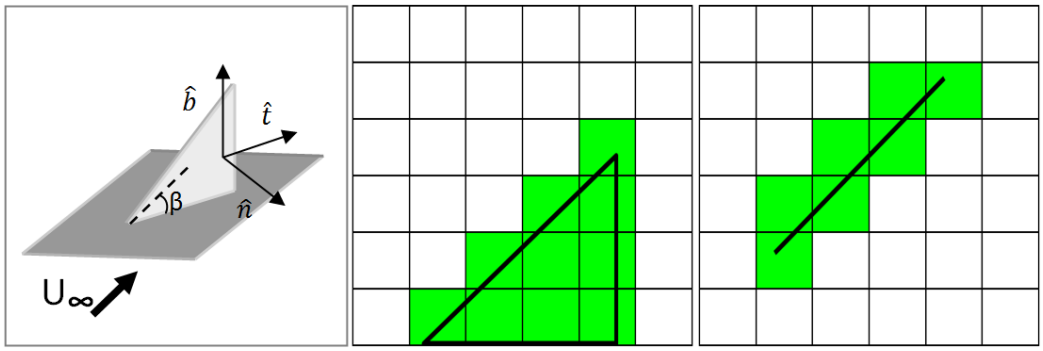


Figure 2.6: (Left) The local coordinate system. (Center) Side view of the cells on which the VG body forces are acting. (Right) top view of the cells affected by the presence of the VGs (taken from [101])

This page intentionally left blank.

Chapter 3

The Lagrangian and Hybrid solvers

In this chapter the formulation of the proposed hybrid method is described in three parts. First, the Lagrangian part of the solver is formulated, then the Particle-Mesh method for numerically solving the Poisson equation is presented and finally the coupling procedures leading to the hybrid solver are detailed.

3.1 The Lagrangian solver

3.1.1 Particle Approximations

In the present work particle approximations are used in order to solve the governing equations in Lagrangian form. A particle is assumed to represent the mass of fluid M_p, contained in volume $V_p(t)$ centered at $\vec{Z}_p(t)$. In addition to mass, particles are carrying momentum and energy $\Pi_p(t)$. Using Helmholtz's decomposition, the velocity and hence the momentum, is expressed through dilatation $\Theta_p(t)$ and vorticity $\vec{\Omega}_p(t)$. All properties carried by a particle are defined as volume integrals; so,

$$M_p = \int_{V_p(t)} \rho(\vec{x},t) dV_p(t) \; = \rho_p(t)\, V_p(t) \tag{3.1}$$

$$\Theta_p(t) = \int_{V_p(t)} \theta(\vec{x},t) dV_p(t) \; = \theta_p(t)\, V_p(t) \; = (\nabla \vec{u})_p\, V_p \tag{3.2}$$

$$\vec{\Omega}_p(t) = \int_{V_p(t)} \vec{\omega}(\vec{x},t) dV_p(t) = \vec{\omega}_p(t) V_p(t) \; = (\nabla \times \vec{u})_p\; V_p(t) \tag{3.3}$$

$$\Pi_p(t) = \int_{V_p(t)} \varepsilon(\vec{x},t) dV_p(t) \; = \varepsilon_p(t)\, V_p(t) \; = (\rho E)_p\,(t) V_p(t) \tag{3.4}$$

in which, $\rho(\vec{x},t)$, $\theta(\vec{x},t)$, $\omega(\vec{x},t)$, $\varepsilon(\vec{x},t)$, correspond to continuous field properties of the flow.

Numerical particles can be regarded as flow markers approximating the flow field at discrete positions and should not be confused with physical particles. Error analysis of particle methods indicates that numerical particles must overlap while the physical ones do not [102, 103]. This is an important remark since it allows to recover the continuous flow properties by means of discrete convolutions,

$$\rho(\vec{x},t) = \sum_p \rho_p(t) V_p(t) \zeta_\epsilon(\vec{x} - \vec{Z}_p(t)) \tag{3.5}$$

$$\theta(\vec{x},t) = \sum_p \theta_p(t) V_p(t) \zeta_\epsilon(\vec{x} - \vec{Z}_p(t)) \tag{3.6}$$

$$\vec{\omega}(\vec{x},t) = \sum_p \vec{\omega_p}(t) V_p(t) \zeta_\epsilon(\vec{x} - \vec{Z}_p(t)) \tag{3.7}$$

$$\varepsilon(\vec{x},t) = \sum_p \varepsilon_p(t) V_p(t) \zeta_\epsilon(\vec{x} - \vec{Z}_p(t)) \tag{3.8}$$

where ζ_ϵ represents a distribution function.

In the original formulation of particle methods, the Dirac function was used in place of ζ_ϵ which corresponds to "point particles" [104]. Lack of smoothness and singularities in the field derivatives led to the introduction of smooth particle approximations [105, 106, 22]. In this respect, ζ_ϵ is considered as a smooth approximation of the Dirac function.

3.1.2 Lagrangian formulation of flow equations

In Lagrangian description, the inviscid flow equations take the form:

$$\text{Mass Conservation: } \frac{DM_p}{Dt} = 0 \tag{3.9}$$

$$\text{Momentum Conservation: } \frac{D\vec{\Omega}_p}{Dt} = V_p \left((\vec{\omega} \cdot \nabla)\vec{u} + \frac{1}{\rho^2} \nabla\rho \times \nabla p \right)_p \tag{3.10}$$

$$\frac{D\Theta_p}{Dt} = V_p \left(2\|\nabla\vec{u}\| - \nabla \cdot \frac{\nabla p}{\rho} \right)_p \tag{3.11}$$

$$\text{Energy Conservation: } \frac{D\Pi_p}{Dt} = -V_p \nabla (\vec{u}p) \tag{3.12}$$

where $(\cdot)_p$ denotes evaluation at the particle position $\vec{Z}_p$. The derivation of these equations can be found in Appendix A. Extension to viscous flows involves extra terms in the Θ, $\vec{\Omega}$ and Π equations. For the first two, the extra terms are derived from the divergence $(\nabla\cdot)$ and curl $(\nabla\times)$ of $\nabla\sigma/\rho$, while for the energy equation the extra term is $\nabla(\vec{u}\cdot\sigma)/\rho$. In these terms σ denotes the viscous stress tensor.

The above system is supplemented with the Helmholtz velocity decomposition:

$$\vec{u} = \vec{U}_\infty + \vec{u}_\phi + \vec{u}_\omega, \quad \vec{u}_\phi = \nabla\phi, \quad \vec{u}_\omega = \nabla \times \vec{\psi} \tag{3.13}$$

that splits the velocity field into an irrotational (curl free) and a solenoidal component (divergence free). The irrotational part is defined through the scalar potential ϕ while the vortical part through the stream function $\vec{\psi}$ (also known as vector potential). By taking the divergence and curl of (3.13),

$$\nabla \cdot \vec{u} = \nabla^2 \phi \tag{3.14}$$

$$\nabla \times \vec{u} = \nabla \times \nabla \times \vec{\psi} = -\nabla^2 \vec{\psi} \tag{3.15}$$

Thus, the velocity field can be determined from ϕ and $\vec{\psi}$ after solving the following two Poisson equations,

$$\begin{aligned} \nabla^2 \phi &= \theta \\ \nabla^2 \vec{\psi} &= -\vec{\omega} \end{aligned} \tag{3.16}$$

to which appropriate boundary conditions are assigned.

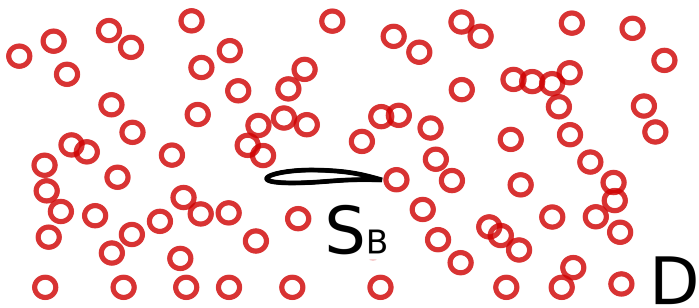


Figure 3.1: The flow domain in the case of an external flow around a solid body. Red circles denote the particles.

Let S collectively denote the boundaries of the flow field D. Making use of Green's theorem, integral representations for ϕ and ψ can be obtained. Let $\vec{n}$, $\vec{\tau}_\alpha, \alpha = 1, 2$ denote the normal and tangent to S unit vectors, and $\partial_n(\cdot) = \vec{n} \cdot \nabla$, $\partial_\alpha(\cdot) = \vec{\tau}_\alpha \cdot \nabla$ the corresponding directional derivatives. Then,

$$\begin{aligned} \phi(\vec{x}) &= \int_D \nabla^2 \phi(\vec{y})\, G(\vec{r})\, dD(\vec{y}) + \int_S \left(\partial_n \phi(\vec{y})\, G(\vec{r}) - \phi(\vec{y}) \partial_n G(\vec{r})\right)\, dS(\vec{y}) \\ \psi_i(\vec{x}) &= \int_D \nabla^2 \psi_i(\vec{y})\, G(\vec{r})\, dD(\vec{y}) + \int_S \left(\partial_n \psi_i(\vec{y})\, G(\vec{r}) - \psi_i(\vec{y}) \partial_n G(\vec{r})\right)\, dS(\vec{y}) \end{aligned} \tag{3.17}$$

where $\vec{r} = \vec{x} - \vec{y}$ and G is the Green's function for the Laplace operator. Taking the ∇ of the first and the $\nabla\times$ of the second, the integral representation of $\vec{u}$ is obtained:

$$\begin{aligned} \vec{u}(\vec{x}) = \vec{U}_\infty + &\int_D \left(\theta(\vec{y})\, \vec{K}(\vec{r}) + \vec{\omega}(\vec{y}) \times \vec{K}(\vec{r})\right) dD(\vec{y}) + \\ &\int_S \left(u_n(\vec{y})\, \vec{K}(\vec{r}) + \vec{u_\tau}(\vec{y}) \times \vec{K}(\vec{r})\right) dS(\vec{y}) \end{aligned} \tag{3.18}$$

In the above expression, u_n, $\vec{u}_\tau$ denote the normal and tangential disturbance velocity components on S and $\vec{K}(\vec{r}) = \nabla G(\vec{r})$. The volume term in (3.18) is directly obtained from the volume integrals in (3.17). Also, the surface terms that are associated to ∂_n in (3.17) directly contribute their shares in $u_n, \vec{u}_\tau$. In order to recover the contribution of ϕ, ψ_i appearing in 3.17 into 3.18, integration by parts is carried out combined with Stokes' theorem applied to second surface terms in (3.17) [107].

If there are no boundaries, the way to proceed is rather straightforward. For given initial conditions, the transport equations (3.9) are integrated in time and the particles are transported using the velocity field defined in (3.18). Cost is dominated by the convolution integral in (3.18). For N particles, the associated cost is proportional to N^2, which can easily explode as N becomes large and the intended duration of the simulation is long. This makes matters worse, when boundaries are present since the surface convolutions in (3.18) must be also evaluated.

In (3.18), S is the boundary of D and therefore contains not only the solid boundaries S_B but also any other interface, as in the case of non-overlapping domain decomposition. On S_B, u_n along with $\vec{u}_\tau$ are determined so that $\vec{u}$ satisfies the boundary conditions of the flow problem. On interfaces, they are determined by the corresponding jump conditions. This involves the solution of two integral equations derived from (3.18) for $\vec{x} \in S$ [53, 54]. These equations will either determine u_n, $\vec{u}_\tau$ on S for given θ, ω or specify the production of dilatation and vorticity for given u_n, $\vec{u}_\tau$. However, to solve these integral equations, can significantly penalize the cost and therefore should be avoided.

Decisive cost reduction can be achieved with Particle Mesh (PM) methods. The idea is to solve the equations for ϕ and $\vec{\psi}$ on a Cartesian grid using Fast Poisson solvers [108]; a cost-saving issue of critical importance, especially in 3D simulations. In this way, the expensive velocity calculations (3.18) are avoided. It is noted that in 2D flows two Poisson equations are solved while in 3D the number rises to four since the stream function is a vector field. The cost reduces to $NlogN$ which for large N increases almost linearly.

3.2 The Particle Mesh method

In Particle Mesh (PM) methods the basic idea is to obtain the solution of the Poisson equations on a structured uniform grid using Fast Fourier Transforms (FFT). This kind of solvers are known as Fast Poisson solvers.

3.2.1 The James-Lackner algorithm for the Poisson problem in full space

For the Poisson problem in full space,

$$\nabla^2 w = f(\vec{x}), \quad \vec{x} \in R^n \tag{3.19}$$

let D denote a compact subset of R^n which contains the support of the forcing f. Then, the above problem can take the following form:

$$\begin{aligned} \nabla^2 w &= f(\vec{x}), \qquad \vec{x} \in D \\ w &= w_B, \qquad \vec{x} \in \partial D \end{aligned} \tag{3.20}$$

The boundary (Dirichlet) data w_B, can be readily obtained in convolution form using Green's function G,

$$w_B(\vec{x}_o) = \int_{supp\{f\}} f(\vec{x})\, G(\vec{x}_o - \vec{x}) dD(\vec{x}) \tag{3.21}$$

Even if tree algorithms are applied, the above calculation remains expensive. In order to mitigate the cost, James and Lackner [65, 66] separated the space and boundary forcing.

Let,

$$w = w_0 + w_1 \tag{3.22}$$

so that, the problem is split in two parts:

a)The Homogeneous Problem

$$\begin{aligned} \nabla^2 w_0 &= f(\vec{x}), \quad \vec{x} \in D \\ w_0 &= 0, \qquad \vec{x} \in \partial D \end{aligned} \tag{3.23}$$

b) The Single Layer Problem

$$\begin{aligned} \nabla^2 w_1 &= 0, \qquad \vec{x} \in D \\ w_1 &= w_B, \quad \vec{x} \in \partial D \end{aligned} \tag{3.24}$$

which can be given an explicit single layer (source) expression:

$$w_1(\vec{x}_o) = \int_{\partial D} \sigma(\vec{x})\, G(\vec{x}_o - \vec{x}) dS(\vec{x}) \tag{3.25}$$

where,

$$\sigma = [\frac{\partial w_1}{\partial n}]$$

denotes the jump in the normal derivative on ∂D.

By requiring continuity of w across ∂D,

$$[\frac{\partial w}{\partial n}] = [\frac{\partial w_0}{\partial n}] + [\frac{\partial w_1}{\partial n}] = 0, \quad \text{on} \quad \partial D \tag{3.26}$$

it follows that,

$$\sigma = -[\frac{\partial w_0}{\partial n}] \tag{3.27}$$

So in summary, the space convolution integral (3.21) is substituted by a surface one (3.25). More specifically, depending on the calculation point $\vec{x}$:

$$\vec{x} \in D, \quad w = w_0 + w_1 \tag{3.28}$$

$$\vec{x} \in R^n - D, \quad w = w_1 \tag{3.29}$$

This means that within D two Poisson problems are solved: the first is the auxiliary problem for w_0 that determines σ and by that w_B is defined. Consequently, the second problem solved the original one (3.20), for the boundary data w_B defined.

In the next section, application of the James-Lackner algorithm, in case the original domain is split in sub-domains will be described.

3.2.2 Domain Decomposition

By splitting D into N non-overlapping blocks, the following decomposition is introduced for the domain, the forcing and the unknown field (3.2).:

$$D_h = \cup_{i=1}^{N} D_i \tag{3.30}$$

$$f(\vec{x}) = \sum_{i=1}^{N} f_i(\vec{x}) \tag{3.31}$$

$$w(\vec{x}) = \sum_{i=1}^{N} w_i(\vec{x}) \tag{3.32}$$

In the above, f_i and w_i are considered as zero extensions of the restrictions in D_i. For a linear operator, it follows that,

$$\nabla^2 w_i = f_i(\vec{x}), \; \vec{x} \in D_i \tag{3.33}$$

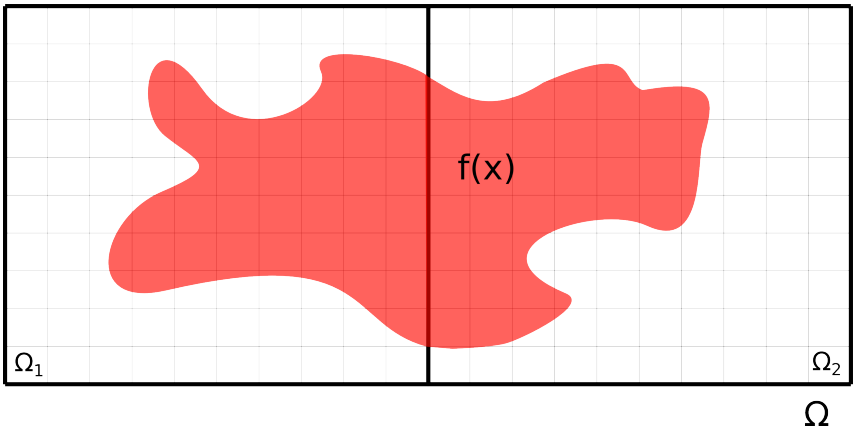
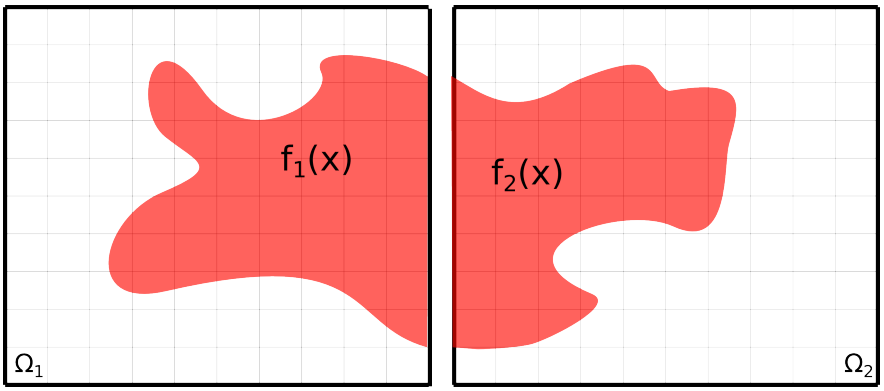


Figure 3.2: The full problem definition. Example of 2 sub-domains

In order to ensure that w_i will match the solution in the whole domain w, appropriate boundary conditions must be applied to the sub-domain boundaries ∂D_i.

3.2.3 The James-Lackner Algorithm Applied at Sub-domains

Let D be split in N sub domains (see Fig.3.3).



(a) Domain 1 (b) Domain 2

Figure 3.3: The decomposition of the domain in blocks. Two blocks are used for better illustration.

The first step is to solve the homogeneous problem in every sub-domain separately as defined in the free space solver already described. Because the specific solver assumes space data of compact support, D_i is extended to $D_{ei} \supset D_i$.

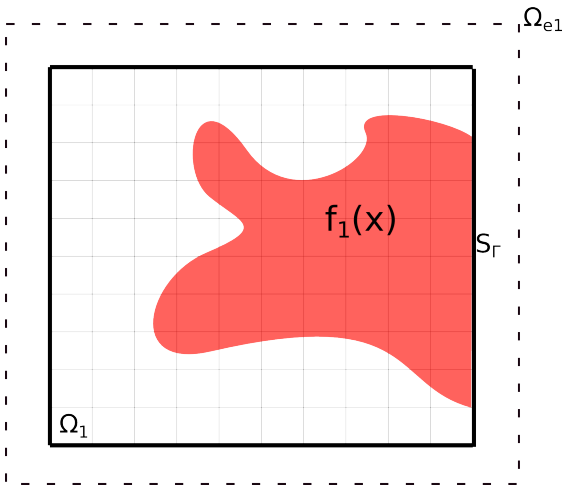


Figure 3.4: The extended domain, D_{e1}.

Thus the following problems are first solved:

$$\begin{aligned} \nabla^2 w_{0i} &= f_i(\vec{x}), \quad \vec{x} \in D_{ei},\ i = 1, N \\ w_{0i} &= 0, \qquad \vec{x} \in \partial D_{ei},\ i = 1, N \end{aligned} \tag{3.34}$$

that determine their corresponding source distributions,

$$\sigma_i = \frac{\partial w_{0i}}{\partial n}, \qquad \vec{x} \in \partial D_{ei} \tag{3.35}$$

By combining the w_{0i} and σ_i, $\tilde{w}_i$ is defined as follows :

$$\begin{aligned} \nabla^2 \tilde{w}_i &= f_i(\vec{x}), \quad \vec{x} \in D_{ei},\ i = 1, N \\ \tilde{w}_i &= \sigma_i, \qquad \vec{x} \in \partial D_{ei},\ i = 1, N \end{aligned} \tag{3.36}$$

Since ∂D_{ei} is different from the interfacing boundaries $S_i = \partial D_i$ (Fig. 3.4) the information given by the source distributions must be appropriately transferred so that continuity holds across S_i (here given for the case of two sub-domains):

$$w_1 = w_2, \vec{x} \in S_\Gamma \tag{3.37}$$

$$[\frac{\partial w_1}{\partial n}] = -[\frac{\partial w_2}{\partial n}], \vec{x} \in S_\Gamma \tag{3.38}$$

The ultimate objective is to define the correct boundary conditions w_{Bi} on ∂D_i. This is done by superimposing all contributions on S_i according to (3.29). More specifically, for the example with two domains:

$$w_{Bi}(\vec{x_o}) = \tilde{w}_1 + \tilde{w}_2 + \sum_{\vec{x}_m \notin \partial D_{ei}} G * \sigma_m(\vec{x_m}) \tag{3.39}$$

Note that the sum on σ's refers to all sub-domains and that in the multi-domain case:

$$G * \sigma_m(\vec{x_o}) = \int_{\partial D_e m} \sigma_m(\vec{x})\, G(\vec{x_o} - \vec{x}) dS(\vec{x}),\ m = 1, N$$

So finally for every sub-domain the problem to be solved takes the form:

$$\begin{aligned} \nabla^2 w_i &= f_i, & \vec{x} \in D_i,\ i = 1, N \\ w_i(\vec{x}) &= \sum_{\vec{x} \in \partial D_k} \tilde{w}_k(\vec{x}) + \sum_{\vec{x}_m \notin \partial D_{ei}} G * \sigma_m(\vec{x}), & \vec{x} \in \partial D_i,\ i = 1, N \end{aligned} \tag{3.40}$$

While the above formulation provides the theoretical basis on which domain decomposition can be applied to a general Poisson problem, the computational cost of the surface convolutions in (3.40) can still be very high. In fact as the number of sub-domains N increases, the cost of calculating the boundary conditions rises and becomes dominant. For this reason, the Method of Local Corrections (MLC) [67] is applied.

The basic idea of MLC is to use a global coarse grid that covers the whole domain and obtain an approximate solution of the full problem (w_c). Avoiding the calculation of the source surface integrals, reduces the cost radically. However the coarse grid solution will be definitely approximate. Taking into account that Green's function $G(r)$ decays fast, the error in every sub-domain will basically concern the neighboring blocks. In fact, the coarse grid is used to replace the sources term $\sum_{\vec{x}_m \notin \partial D_{ei}} G * \sigma_m(\vec{x_m})$ in (3.40). However, care must be taken so that no contributions are included twice.

A way to do this, is to get the background information already contained in $\tilde{w}_i$, at the level of the coarse grid. Let $\tilde{w}_{ci}$ be the interpolation of $\tilde{w}_i$ on the coarse grid. Then for block i

$$\sum_{\vec{x}_m \notin \partial D_{ei}} G * \sigma_m(\vec{x}) = w_c - \sum_{\vec{x} \in \partial D_k} \tilde{w}_{ck}(\vec{x}) \tag{3.41}$$

which leads to following final form of the problem the Fast Poisson solver will solve in every domain:

$$\begin{aligned} \nabla^2 w_i &= f_i, & \vec{x} \in D_i,\ i = 1, N \\ w_i(\vec{x}) &= \sum_{\vec{x} \in \partial D_k} \tilde{w}_k(\vec{x}) + \left[w_c - \sum_{\vec{x} \in \partial D_k} \tilde{w}_{ck}(\vec{x}) \right], & \vec{x} \in \partial D_i,\ i = 1, N \end{aligned} \tag{3.42}$$

The above described procedure can be split in three parts:

- Step 1a: Solve the free space problem in every extended sub-domain D_{ei} and obtain w_{0i}, σ_i and $\tilde{w}_i$, i.e. (3.34) , (3.36).

- Step 1b: Solve the full problem using a coarse grid (see Fig. 3.5) and obtain $w_i c$.
- Step 2: Combine $\tilde{w}_i$ and w_c and define the Dirichlet boundary data w_{Bi} on ∂D_i for every sub-domain D_i.
- Step 3: Solve the Poisson Dirichlet problem in every sub-domain and obtain w_i, i.e. (3.42).

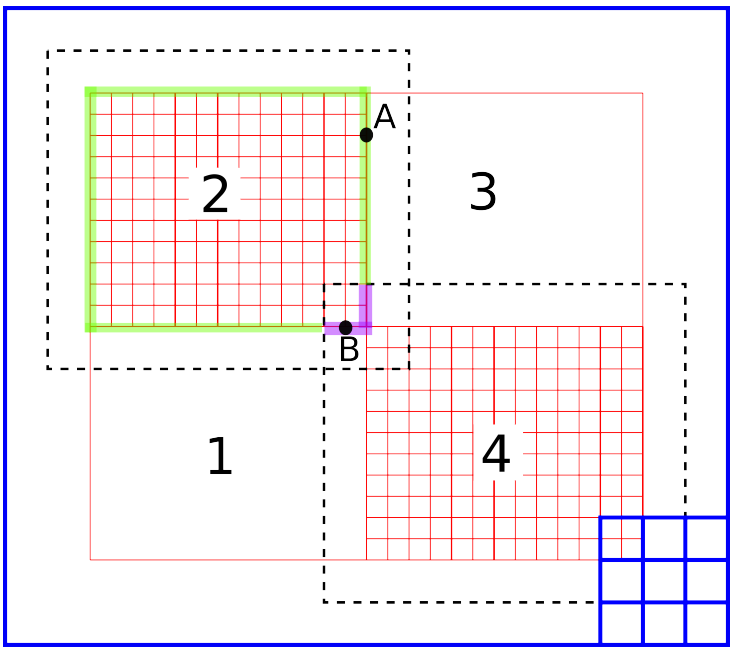


Figure 3.5: Two blocks are shown in red together with their corresponding extensions marked by the black dashed line. In blue a portion of the overlaid coarse grid that covers the complete domain is shown. Corresponding to the domains depicted, the green boundary is treated using the coarse grid while the purple boundary using the local solution.

For example regarding point A in Fig. 3.5:

$$w_A = \tilde{w}_2 + \tilde{w}_3 + \left[w_c - \tilde{w}_{c1} - \tilde{w}_{c2} \right] \tag{3.43}$$

while for point B:

$$w_b = \tilde{w}_2 + \tilde{w}_3 + \tilde{w}_1 + \tilde{w}_4 \tag{3.44}$$

The way the domain is split might also affect the solution. Locally large gradients might disappear when information is transferred to a coarser grid and

consequently their contribution in the distant blocks boundary. This is the reason why the use of arbitrary order Poisson solvers is suggested in [68].

3.2.4 Projection & Interpolation Operators

In order to apply the multi-domain Poisson solver, the space data must be first defined. This involves the projection of the particle information on the corresponding grids. Furthermore, once the solution of the Poisson solver is obtained, this information must be back-transferred (interpolated) to the particles. The required projection and interpolation operations are defined next.

Let $Q_p = q_p \cdot V_p$ denote any flow quantity carried by the particles. Then on the grid the corresponding density is obtained by the following projection operation:

$$q_{i,j,k} \equiv \mathrm{Proj}_{\mathrm{PM}}(q_p; V_p) = \frac{\sum_p q_p V_p W(\vec{x}_{i,j,k} - \vec{Z}_p)}{\sum_p V_p W(\vec{x}_{i,j,k} - \vec{Z}_p)} \tag{3.45}$$

where $\vec{x}_{i,j,k}$ denotes the position of the i, j, k grid node (also denoted as I),

$$W(\vec{r}) = W_1(r_x/h) W_1(r_y/h) W_1(r_z/h)$$

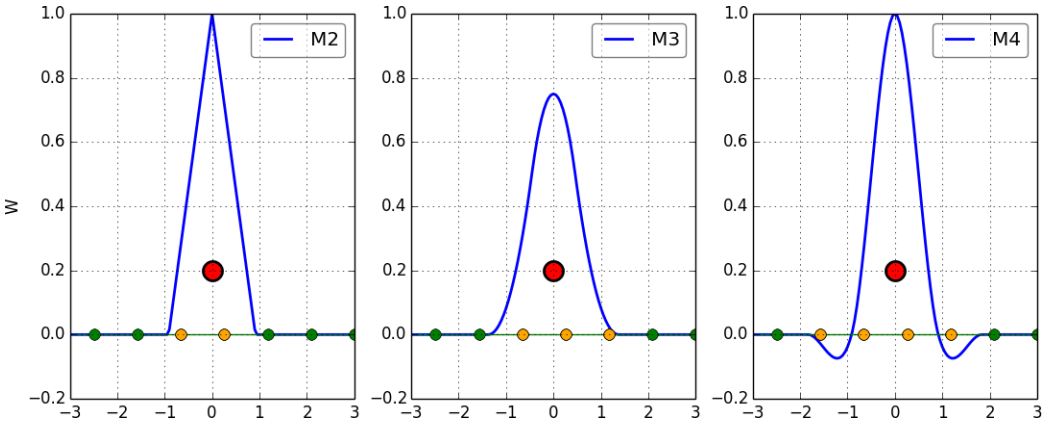
$\vec{r} = (r_x, r_y, r_z)$, h denotes the grid spacing and W_1 the 1D interpolation function used. In the present work the $M_4{}'$ interpolation function is used which conserves the moments up to 3rd order (for other options see [22]) .

Inversely, any quantity defined on the grid, can be interpolated back to the particles positions using the same interpolation function:

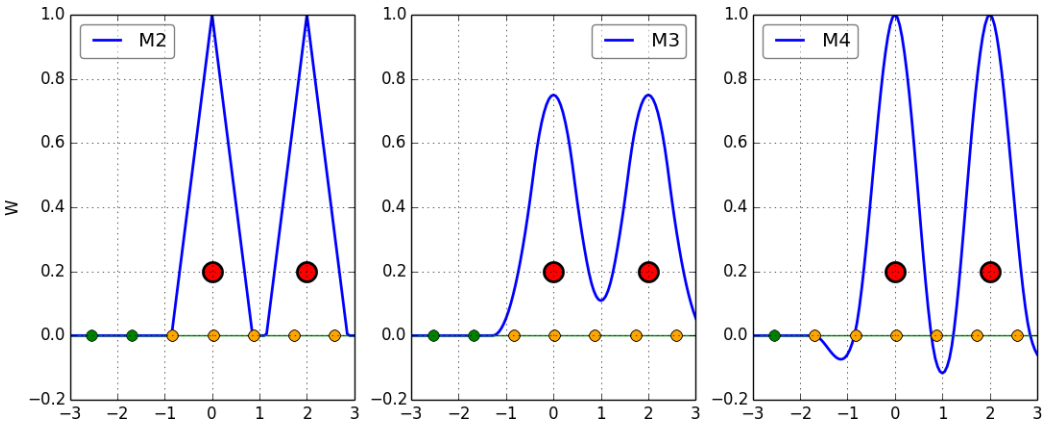
$$q_p \equiv \mathrm{Interp}(q_{i,j}) = \sum_{i,j,k} q_{i,j,k} W(\vec{x}_{i,j,k} - \vec{Z}_p) \tag{3.46}$$

A well known problem in particle methods concerns the gradual loss of regularity in time. Spreading of Particles can result in loss of accuracy while high particle concentration can lead to numerical instabilities [22]. This is analogous to grid irregularity or stretching in standard CFD solvers that is corrected with grid refinement. Of course in principle the grid is generated so that there are no such issues. This is not always easy or guaranteed. In flow problems involving the evolution and/or propagation of steep gradients, the original grid might turn out to be inadequate. Then grid refinement is needed. A similar procedure is also needed and applied in particle methods. Now the role of the grid is taken over by the particles themselves and grid refinement corresponds to the so called *remeshing*. Remeshing aims at recovering the conditions set by error analysis so that consistency and accuracy of the numerical solution is best served. More specifically, it is required that particles overlap and that they are regularly distributed in space. When a Particle Mesh method is applied as in the present case, the above requirements are related to the size of the grid in combination to the interpolation function W used in (3.45) and (3.46).

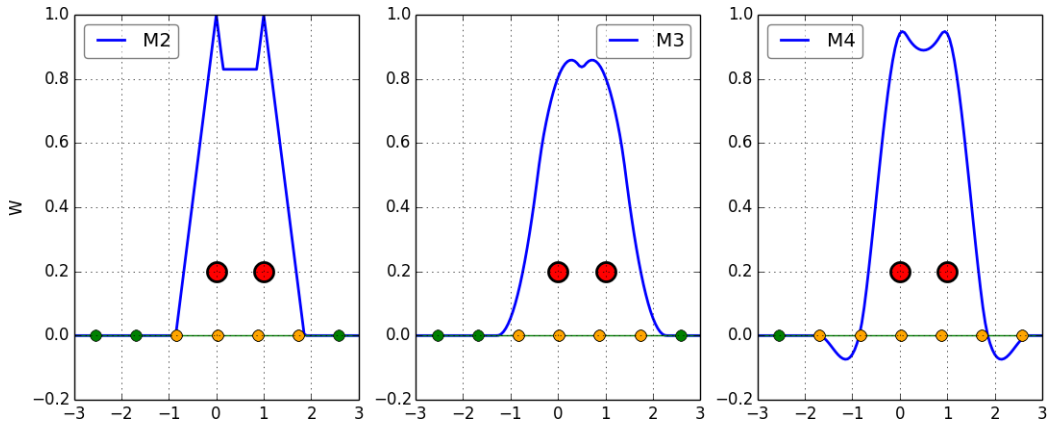
For a given PM grid, the projection will spread the particle information over a stencil that depends on the degree of W. Fig. 3.6a shows in yellow the activated nodes for 1st, 2nd and 3rd order interpolation functions. If the particles are not densely populated with respect to the grid size as in Fig. 3.6b, the volume projected on the nodes will become irregular. Any deficit in volume will deteriorate the quality of the information recorded on the PM grid and lead to significant errors. By increasing the density of the particles, as in Fig. 3.6c, the interpolation quality is clearly improved.



(a) Projected information on the grid for $1_{st}, 2_{nd}$ and 3_{rd} order interpolation functions. The red circle gives the position of the particle. The nodes affected are marked in yellow.



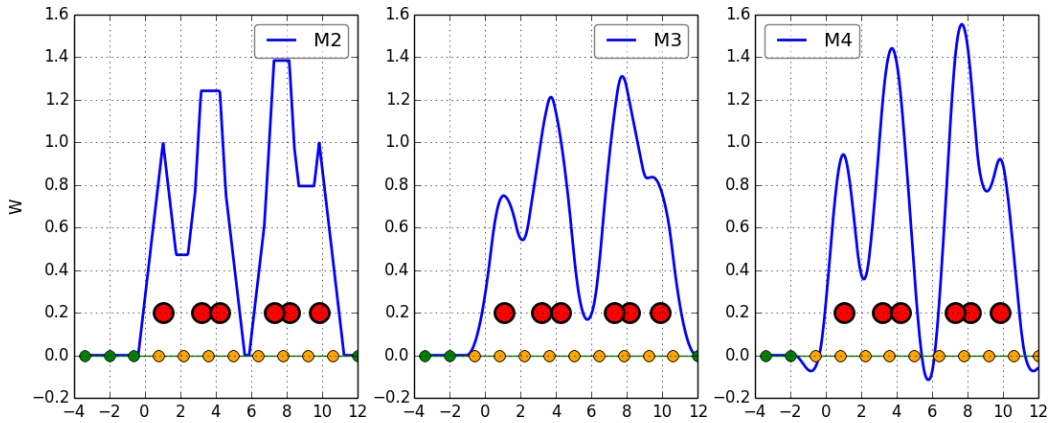
(b) Projected information on the grid when the distance between the particles is greater than the grid size.



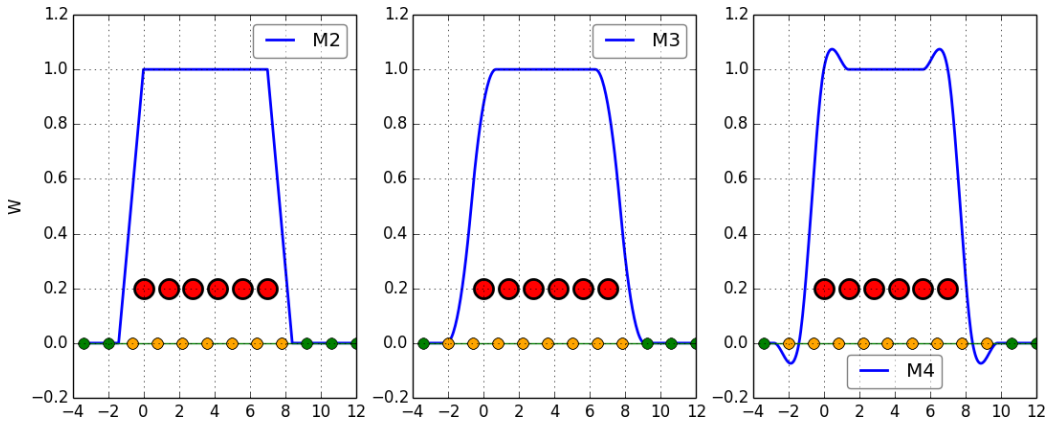
(c) Projected information on the grid when the distance between the particles is the same order of the grid size.

Figure 3.6: Importance of the density of particles.

However, increasing the particle density does not guarantee smoothness of the flow information as shown in Fig. 3.7. Having a regular distribution of particles in space largely improves the quality of the projected information. The effect of the spacial regularity of particle distribution on the flow solution becomes even more profound when the fluid is compressible. In this case mass and volume that are projected on the PM grid play a crucial role. Spreading of Particles will create "holes" of mass in the PM grid which will lead to unphysical density fluctuations. Similarly if particles get closely clustered, the projected volume will exceed that occupied by a PM cell and therefore again density will get unrealistic values.



(a) Projected information on the grid in the case of an irregular distribution of particles.



(b) Projected information on the grid in the case of a regular distribution of particles.

Figure 3.7: Importance of the regularity of the distribution of particles in space.

Re-meshing is the action of re-positioning the particles at regularly ordered positions every N time steps. The quantities carried by the relocated particles are interpolated from the PM grid at the new particle positions using (3.46). A point of concern in this respect, is that re-meshing adds numerical diffusion. For example $\operatorname{supp}\vec{\omega}$ will spread out along its boundary and points that previously had zero vorticity will receive at least a small amount of $\vec{\omega}$. A way to limit numerical diffusion is to perform re-meshing on a denser grid than the PM one. Another way, is to apply one sided interpolation function along the boundary of $\operatorname{supp}\vec{\omega}$ [22]. Note that the latter must not concern mass.

The volume assigned to the re-meshed Particles is usually the *geometrical* volume (i.e. the volume of a PM cell, DV_{PM}) in order to keep the scheme conservative. Therefore special care is needed in order to ensure conservation of the flow properties carried by the particles. When changing the volume in the re-meshing procedure, any quantity Q carried by the particles must be conserved. Thus, when changing the volume of a particle V_i to the geometrical V_{ref}, conservation of quantities at particle level are ensured with the following scaling:

$$\begin{aligned} Q_{ref} = Q_i \Rightarrow \quad & q_{ref}V_{ref} = q_iV_i \Rightarrow \\ q_i = \frac{Q_{ref}}{V_i} \quad \text{or} \quad & q_{ref} = \frac{Q_i}{V_{ref}} \end{aligned} \tag{3.47}$$

The above relation implies that the volume used in the projection operation (3.45) is the volume the Particles will get after they are placed in their new (ordered) positions.

3.2.5 The Lagrangian solver

The sub-steps taken in every Lagrangian time step can be summarized as follows:

For a given collection of Particles: $\{\vec{Z}_p^n,\, m_p^n,\, V_p^n,\, \vec{\Omega}_p^n,\, \Theta_p^n,\, \Pi_p^n\}$ at $t = n\Delta t$:

Step 1: Project $\{m_p^n,\, \Theta_p^n, \vec{Omega}_p^n,\, \Pi_p^n\}$ on the PM grid and get: $\rho_{i,j}^n,\, \theta_{i,j}^n,\, \vec{\omega}_{i,j}^n,\, \varepsilon_{i,j}^n$.
Step 2: Solve $\nabla^2\phi = \theta, \nabla^2\vec{\psi} = -\vec{\omega}$ and obtain: $\phi_{i,j}^n,\, \psi_{i,j}^n,\, \vec{u}_{i,j}^n,\, \vec{w}_{i,j}^n$
Step 3: Calculate on the PM grid the terms in the RHS of (3.9), e.g. $\nabla\rho_{ij}^n,\, \nabla p_{ij}^n,\, \nabla\vec{u}_{ij}^n$
Step 4: Interpolate all grid based data q_{ij}^n at the particle positions:
$q_p^n = \sum_{ij} q_{ij}^n\, W(\vec{x}_{i,j} - \vec{Z}_p)$
Step 5: Update all particle properties (integrate (3.9) in time)
Step 6: Re-mesh if needed

In Step 3, the Right Hand Side of (3.9) are obtained by means of Finite Differences on the Particle Mesh grid.

Dilatation Damping

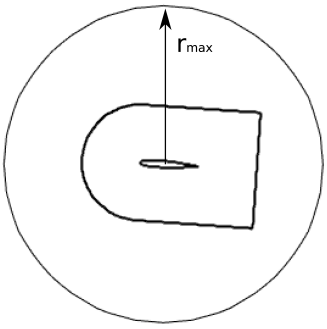
Dilatation is directly related to acoustic waves. So when a finite region is used, these waves can be reflected on its outer boundary. In order to prevent the occurrence of reflections, a Gaussian damping function,

$$\zeta(r) = \left\{ \begin{array}{ll} e^{-\sigma(r-r_{max})^2} & , r > r_{max} \\ 1 & , \text{else} \end{array} \right\} \tag{3.48}$$

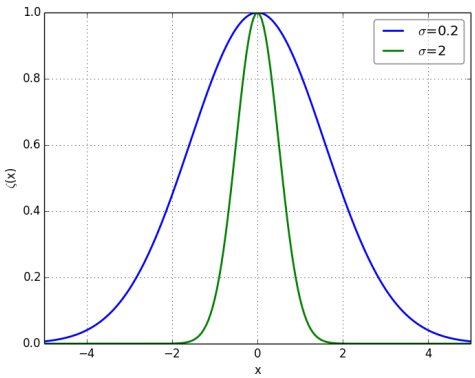
is applied to the evolution of dilatation:

$$\theta = \theta \cdot \zeta(r) \tag{3.49}$$

Damping is activated outside a circle or a sphere of radius r_{max} while σ defines its rate. The parameters, σ, r_{max} are chosen so that $\theta = 0$ outside the PM domain. An example is shown in Figure 3.9.

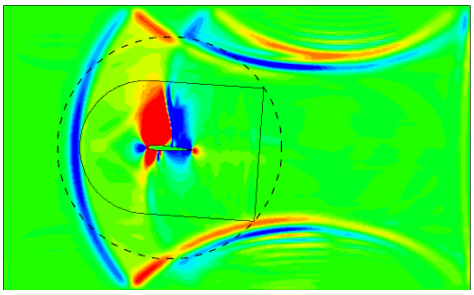


(a) Outside the circle of radius r_{max} damping of dilatation is applied.

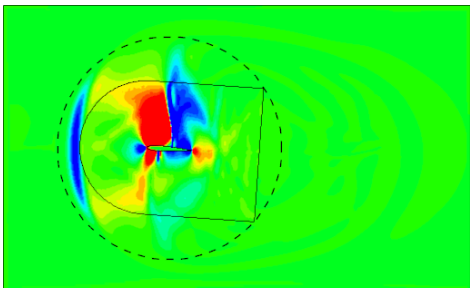


(b) The damping function for various values of σ

Figure 3.8: Definition of the damping of dilatation.



(a) Reflections of acoustic waves on the PM boundary.



(b) Application of damping cancels reflections on S_o

Figure 3.9: Effect of dilatation damping on the artificial reflections of acoustic waves. The dotted line outlines the circle of radius r_{max}. Here the damping is very close to the source of the waves for better illustration

3.3 The Hybrid Solver

The hybrid solver aims at coupling an Eulerian solution (from hereon referenced as CFD) near the solid boundaries with a Lagrangian one (from here on referenced as PM) that satisfies the true far field conditions. In the specific formulation the Lagrangian domain covers the entire space and therefore overlaps the Eulerian domain. By choosing to use the PM method in the Lagrangian solver, the Lagrangian domain is divided in two parts: the PM domain D_{PM} and the pure particle one $D_P = R^n - D_{PM}$ under the understanding that D_{PM} will contain all solid boundaries.

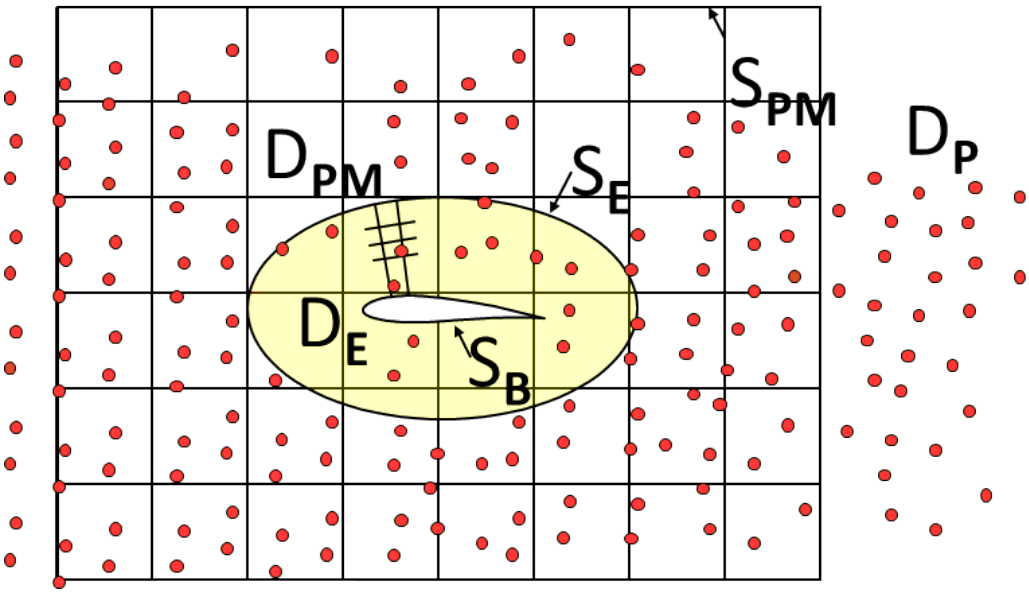


Figure 3.10: Definition of the Computational Domains

In Fig.3.10 the above arrangement is shown in the case of an airfoil. D_E defines the CFD domain and S_E the interface with the PM solution. Note that the PM grid does not respect the solid boundary S_B similarly to the *immersed boundary condition* implementation [109, 110]. This allows avoiding the solution of any integral equation since (3.18) is approximately satisfied. The accuracy of such an approach depends on the PM grid resolution and the way the boundary data on S_E are defined.

In (3.18) $\theta\, dD$ & $u_n\, dS$ as well as $\vec{\omega}\, dD$ & $\vec{u}_\tau\, dS$ are associated to the same kernel and hence they can be treated as particles without distinction. However, it is important to respect conservation and therefore when transforming a surface quantity q_s into a volume quantity Q_p; so $Q_p = q_s\, \Delta S$, where ΔS denotes the support of q_s. Then, as regards accuracy, the fact that the Eulerian solver is used close to the boundaries considerably reduces our requirements. In fact the quick decay of $\vec{K}(r)$ for increasing r assures that the error will also decay quickly.

3.3.1 Coupling Procedures

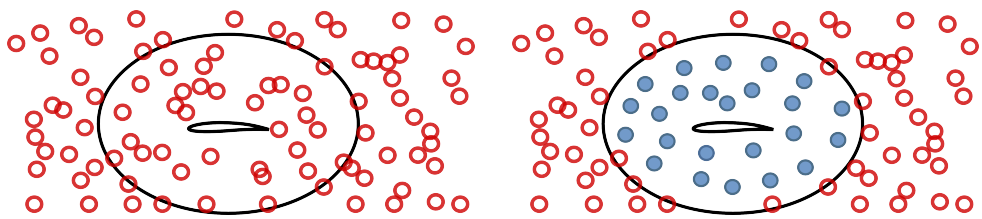
The most important aspect of a hybrid method is the coupling procedure. The CFD and PM solutions are coupled in two ways:

- The CFD flow information U_E is used in order to update the PM solution Q_{PM} in D_E.
- The PM solution is used in order to define the boundary conditions for the CFD solver on S_E.

The key idea is to ensure that at the end of each time step, the PM solution is a smooth extension of the CFD one outside D_E. To this end, an iterative procedure is formulated that replaces the PM particles with those defined by the CFD solution in the course of the PM-CFD iterations. In this sense, particles are not emitted nor absorbed at the boundaries but the flow information is corrected throughout D_E. An important feature of this coupling procedure is that at every PM-CFD iteration, the set of boundary conditions on S_E are updated making use of the latest update of the flow information. This means that the coupling is strong and, as explained next, conservative.

3.3.2 Correction of Q_{PM} in D_E

The purpose of the correction next presented, is to transfer the correct CFD flow information to the Lagrangian solver. This is done in particle form. Over D_E, CFD particles are defined that replace the existing PM particles (Figure 3.11).



(a) The PM particles as obtained at the end of a time step.

(b) Within D_E the PM particles are substituted by the CFD particles here marked in blue.

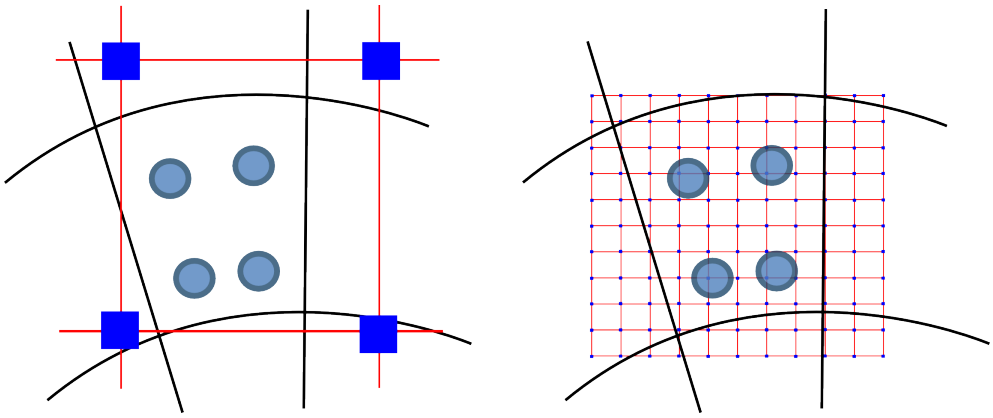
Figure 3.11: Correction of Q_{PM} in D_E.

Definition of the CFD particles

In order to obtain the CFD flow information in particle form, a set of markers is defined over the CFD grid. The number of markers within a CFD grid cell depends on its size with respect to that of the PM grid. It is important to assure on one hand full coverage of the PM grid and on the other adequate spacial density of particles

as already explained. As a rule, more than one CFD particles should be contained in every PM cell as shown in Fig. 3.12. If the coverage is not good, certain PM nodes will receive no flow information and therefore the resulting solution will be irregular. In the present work 4 and 8 particles per cell were found adequate in 2D and 3D problems respectively.

The CFD particles P_E are defined at regular positions within every CFD cell. First U_E is transformed into $q_E = \{\rho, \vec{\omega}, \theta, \Pi\}_E$ at P_E and then projected on the PM grid. The U_E solution is defined at the CFD grid cell centers and contains $\rho, \rho\vec{u}, \rho E$ as well as the spatial velocity derivatives which are needed in (2.43)). So all necessary information (including dilatation and vorticity) at the CFD cell centers q_c can be defined. Next q_c is interpolated at P_E using iso-parametric finite element approximations which are also applied in order to determine the associated volume. Note that if the CFD grid is not moving/deforming, the positions of the CFD particles P_E and their volumes are fixed.



(a) Correct coverage of the PM grid with CFD particles.

(b) Inadequate coverage. In this exaggerated example more than 4 particles per CFD cell should be defined.

Figure 3.12: Spacial distribution of CFD particles for full coverage. The PM grid is in red while the CFD grid in black.

Definition of the correction scheme

By substituting the PM particles with CFD ones within D_E, the presence of the solid boundaries is taken into account in the PM solution. While this correction is valid well within D_E, close to S_E special care is needed. Since the support of any particle is finite, the particles that are within a strip adjacent to S_E will have contributions on both of its sides. More specifically, when projecting PM particles that are outside D_E but close to S_E, certain PM grid points within D_E will be activated. Also the CFD particles that are close to S_E will contribute information outside D_E. Such a procedure will on one hand violate conservation

and on the other generate discontinuity across S_E. To correct this inconsistency, the associated *error* is subtracted.

Let $\hat{q}_{PM}$ denote the PM solution at the beginning of every time step. In $\hat{q}_{PM}$ there are two errors: the solution error and the projection error. The solution error is associated to the fact that the PM solver does not take into account the presence of S_B. This part of the error is corrected by substituting the PM particles with the CFD particles. However the positions of the PM and CFD particles are not the same and therefore a projection error is generated because of that.

The fact that the P_E positions are fixed, allows defining the error of that substitution. This is accomplished by first interpolating the flow information at the CFD particle positions P_E and then back projecting this information to the PM grid nodes (see Fig. 3.13).

$$error\{q_{PM}\} = \mathrm{Proj}_{\mathrm{PM}}\{\mathrm{Interp}_{P_E}(\hat{q}_{\mathrm{PM}})\} \tag{3.50}$$

The corrected PM solution is readily obtained by subtracting $error\{q_{PM}\}$ from $\hat{q}_{PM}$ and adding q_E,

$$correct\{q_{\mathrm{PM}}\} = \hat{q}_{\mathrm{PM}} - error\{q_{\mathrm{PM}}\} + q_E \tag{3.51}$$

$$q_E = \mathrm{Proj}_{\mathrm{PM}}\,(Q_E; V_E) \tag{3.52}$$

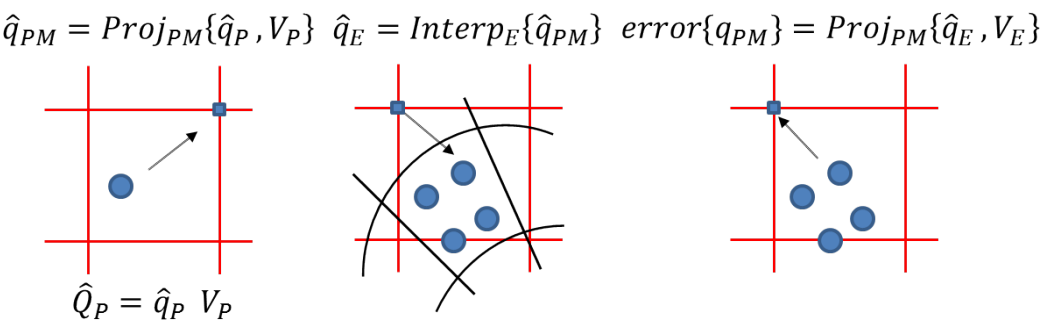


Figure 3.13: Definition of $error\{q_{PM}\}$.

The *error* as defined above is expected to be significant only within a strip centered at S_E and therefore,

$$\begin{aligned} q_{\mathrm{PM}} &= \hat{q}_{\mathrm{PM}} && \text{outside the Eulerian Domain} \\ q_{\mathrm{PM}} &= q_E + \underbrace{\hat{q}_{\mathrm{PM}} - error\{q_{\mathrm{PM}}\}}_{\text{orrection}} && \text{inside the Eulerian Domain} \end{aligned} \tag{3.53}$$

The above correction procedure is displayed in Fig. 3.14. Technically, the Lagrangian particles are never replaced by the CFD ones. Instead, the values at

the PM nodes change. Once q_{PM} is finalized, it is interpolated at the original Lagrangian Particles Positions.

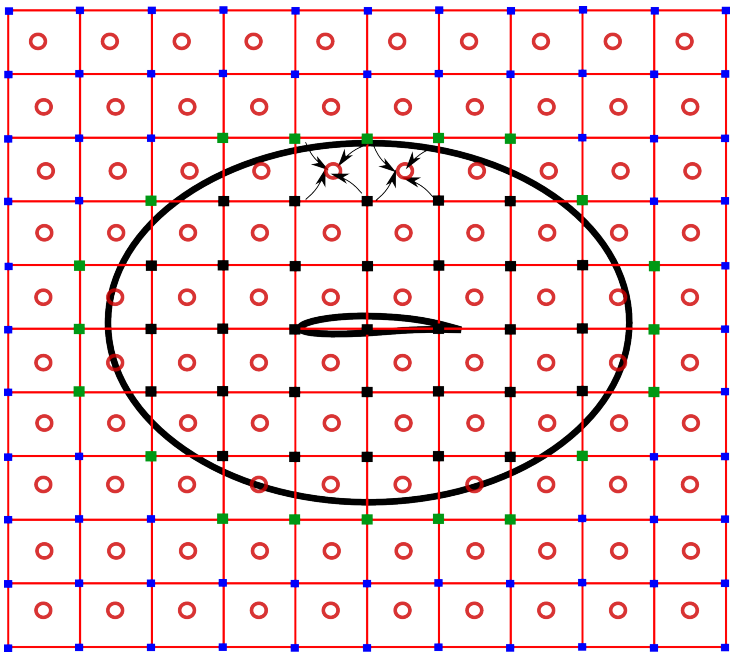


Figure 3.14: Correction procedure: The flow information on Blue nodes corresponds to $\hat{q}_{PM}$; that on Black nodes to $\hat{q_E}$; that on Green nodes to $q_E + \hat{q}_{\mathrm{PM}} - error\{q_{\mathrm{PM}}\}$.

Special care is needed in the part of the PM grid that contains solid boundaries. This necessity arises from two facts: a) there will be no CFD particles in the interior of S_B, and b) convection will eventually make PM particles to cross S_B (see Fig. 3.15). Considering (3.53), lack of CFD particles in a region, implies that $\hat{q}_{\mathrm{PM}} = \hat{q}_{\mathrm{PM}}$ which is physically wrong. In order to circumvent this, at the PM grid points within S_B the following correction are applied:

$$\begin{aligned} q_{PM} &= 0 \\ error\{q_{\mathrm{PM}}\} &= 0 \end{aligned} \tag{3.54}$$

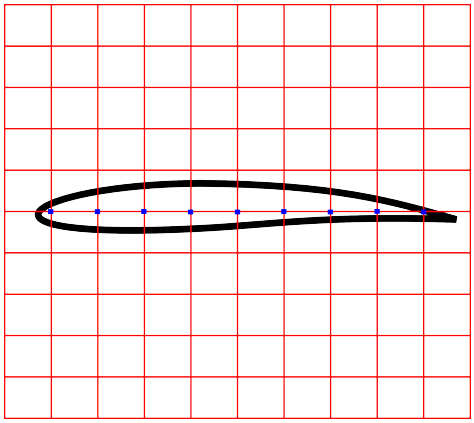


Figure 3.15: Correction procedure: At the points inside the airfoil (blue dots) zero values are assigned to q_{PM} and $error\{q_{\mathrm{PM}}\}$

.

CFD boundary conditions on S_E

The outer (far-field) boundary conditions of the CFD solver are obtained by interpolating the particle information at the boundary faces. This is done using (3.46). Thus the state outside the CFD domain D_E at the center of the ghost cells U_o is defined (Figure 3.16). In order to obtain U_B on S_E, the Riemann invariants associated to U_o and U_E on the two sides of the corresponding face, are used.

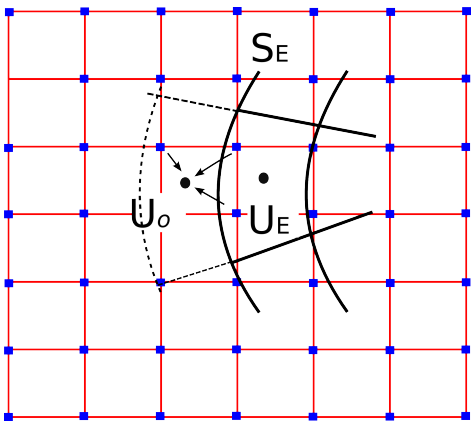


Figure 3.16: Definition of the boundary conditions on S_E. The state U_o at the center of the ghost cell (here marked by a dashed line) is obtained by interpolating the PM data defined at the nodes (here marked in blue).

Buffer Zone

Another issue that needs to be addressed, is the non-physical entropy waves that occur at the early stages of the simulation. Since the initial conditions do not actually satisfy the boundary conditions, non-physical entropy/vorticity waves are generated which travel upstream of the flow.

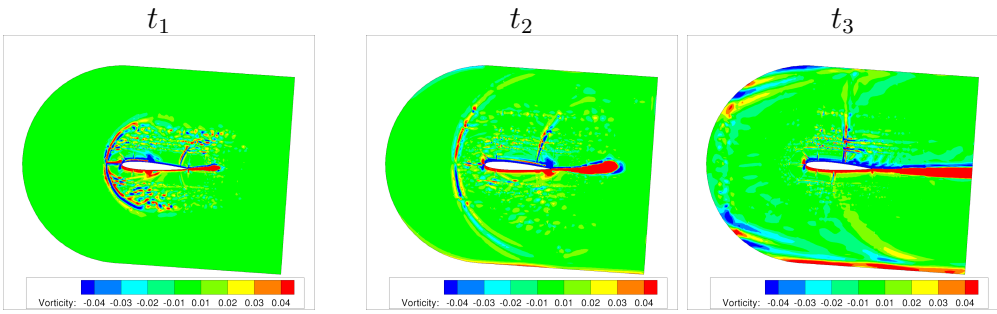


Figure 3.17: Impulsively started Flow around an airfoil. Snapshots of vorticity contours during the transient state of the convergence history show the development generation of a non physical entropy/vorticity close to the wall boundary that travels upstream($t_1 < t_2 < t_3$). Upon arrival at the S_E they are back reflected.

In Fig. 3.17 a vorticity wave is generated that travels in the opposite flow direction. This wave is non-physical since vorticity is convected with the local velocity. Such waves are generated regardless which solver is used and have been also observed in standard CFD.

When the CFD grid is extended, the wave will be gradually diffused before reaching the outer boundary. So any reflection on the outer boundary will have negligible effect on the solution. If instead the CFD grid is limited, the wave will be still strong when arriving at S_E and therefore the reflections will be significant. Because such waves correspond to errors and have no physical origin, the Lagrangian solver will not "absorb" them. Neither will the 1D Riemann Invariants which lose their "transparency' when the boundary conditions are time-dependent or the flow is highly multi-dimensional [111].

The problem of defining non-reflecting boundary conditions has been addressed especially in the context of computational aeroacoustics. In this respect, two schemes have been reported in the literature. The first is purely numeric and is based on introducing a buffer or as is called "sponge" zone [112] and the other is based on the LODI relations formulated in [17] on the basis of the full equations in characteristic form. In this respect the LODI approach is a generalization of the 1D expressions used in standard CFD. However, in the present context, a rigorous non-reflecting boundary condition in the CFD part will not prevent the waves to cross the S_E boundary and by that propagate in the PM domain. Once a flow

disturbance, even an artifact, enters the PM domain it will be convected until it exits. This will just result convergence deceleration. In order to circumvent this, in the sponge zone technique was employed.

The sponge zone is defined around the CFD far-field S_E within which CFD particles are not generated. Thus, the artificial vorticity waves are blocked from being identified by the PM solver and consequently their effect on the far-field conditions of the CFD part is diminished. It should be noted that the vorticity levels contained in these waves are three orders of magnitude lower than the maximum vorticity level (Fig. 3.18). It should be stressed that due to the small magnitude of vorticity these waves carry, the solution is not actually affected [113]. In fact, the level of vorticity carried by these waves is comparable to that generated by grid stretching in standard CFD (see Fig.3.18). Tests have shown that the buffer zone size should cover at least two layers of CFD cells. In practice, a buffer zone covering approximately a five-cell layer was used.

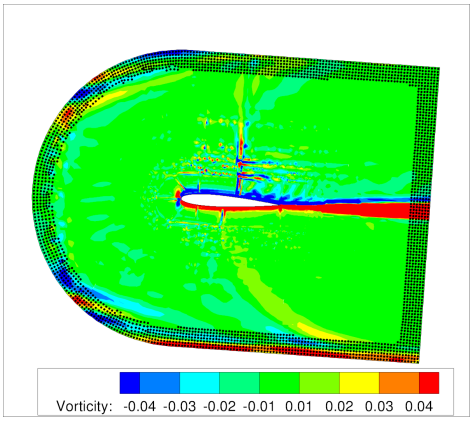


Figure 3.18: The buffer zone, is shown with black dots. In this region, CFD particles are not generated.

3.3.3 The HoPFlow algorithm

For a given collection of Particles: $\{\vec{Z}_p^n,\, m_p^n,\, V_p^n,\, \vec{\Omega}_p^n,\, \Theta_p^n,\, \Pi_p^n\}$ at $t = n\Delta t$, in every time step of the hybrid solver the following sub-steps are performed:

Step 1:	Project $\{m_p^n,\, \Theta_p^n,\, \vec{\Omega}_p^n,\, \Pi_p^n\}$ on the PM grid and get: $\rho_{ijk}^n,\, \theta_{ijk}^n,\, \vec{\omega}_{ijk}^n,\, \varepsilon_{ijk}^n$.
Step 2:	Solve $\nabla^2\phi = \theta, \nabla^2\vec{\psi} = -\vec{\omega}$ and obtain: $\phi_{ijk}^n,\, \vec{\psi}_{ijk}^n,\, \vec{u}_{ijk}^n,\, \vec{w}_{ijk}^n$
Step 3a:	Provide Boundary Conditions for the CFD Solution
Step 3b:	Update the CFD solution (dual step subiterations)
Step 3c:	Update the flow information at the CFD particles.
Step 3d:	If the dual step convergence is reached continue; else go to 2
Step 4:	Calculate on the PM grid the terms in the RHS of (3.9), e.g. $\nabla\rho_{ijk}^n,\, \nabla p_{ijk}^n,\, \nabla\vec{u}_{ijk}^n$
Step 5:	Interpolate all grid based data q_{ij}^n at the particle positions: $q_p^n = \sum_{ijk} q_{ijk}^n\, W(\vec{x}_{ijk} - \vec{Z}_p)$
Step 6a:	Integrate (3.9) in time and obtain $\{\vec{Z}_p^{n+1},\, m_p^{n+1},\, V_p^{n+1},\, \vec{\Omega}_p^{n+1},\, \Theta_p^{n+1},\, \Pi_p^{n+1}\}$
Step 6b:	Re-mesh if needed

This page intentionally left blank.

Chapter 4

Validation and Verification

In this chapter validation and verification test cases are presented for the CFD code, MaPFlow, as well as for the hybrid method proposed, HoPFlow. Verification refers to the procedure of testing whether a code has implementation errors while validation refers to the investigation whether the results of a code accurately describe the physics of the problem considered. Concerning the CFD solver, the simulations are compared to experimental data as well as to other numerical results. The list of cases starts with steady state 2D flows and then proceeds to 3D cases. The hybrid solver is directly compared to CFD results. Again, the presentation starts with simple 2D cases and ends with more complex 3D cases. It is important to stress that validation and verification is a never ending procedure. In every test case certain aspects of the simulation procedure are evaluated. As the number and versatility of the test cases increases. the more confident one becomes in the simulations a code can produce. In this respect, the present chapter is considered adequate but not exhaustive.

4.1 Validation and Verification of the standard CFD solver (MaPFlow)

4.1.1 Zero Pressure Gradient Flow over a Flat Plate

This first test case has been taken from [114] and concerns a zero pressure gradient flow over a flat plate. Results are compared to those obtained with the structured compressible code CFL3D [115] and the unstructured compressible code FUN3D [116].

For the specific test case, the grid provided by the reference [114] is used. The grid size is 545x385 which results in 208896 cells. Grid details are given in Fig. 4.1. The infinite Mach number (Ma) is 0.2 and the Reynolds Number used(Re) is $5 \cdot 10^6$ at $x = 1$ (see Table 4.1). Simulations were performed using both the

Table 4.1: Flatplate flow conditions

Mach	Reynolds
0.2	$5 \cdot 10^6$at $x = 1$

Spalart-Allmaras (SA) and the $k - \omega$ Shear Stress Transport (SST) turbulence model.

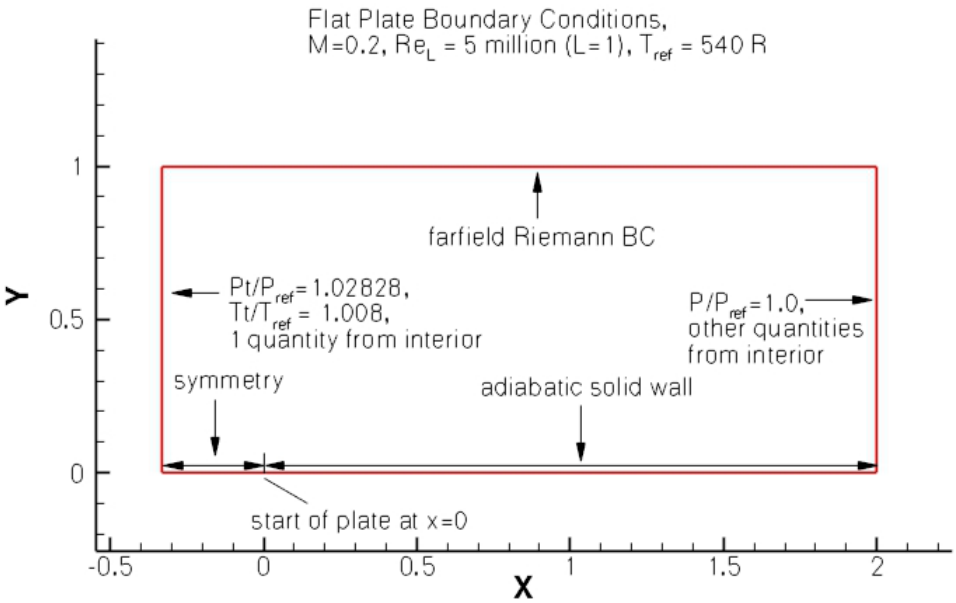
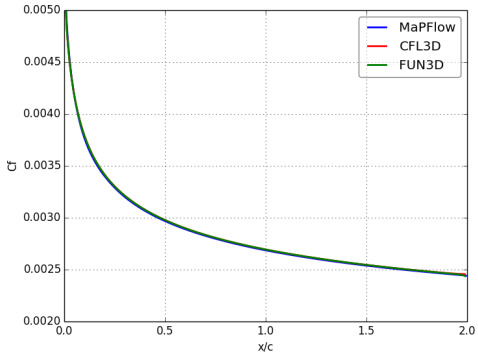
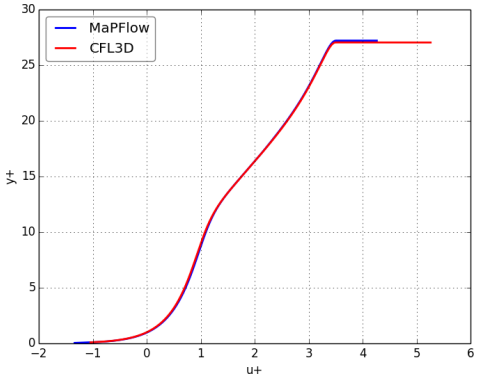


Figure 4.1: The Computational Set-up (taken from [114])

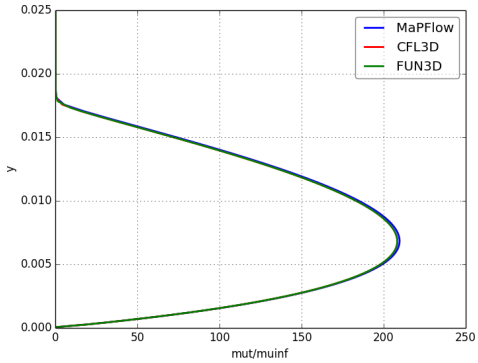
The agreement between the codes is very good as shown in Fig. 4.2, 4.3, where the friction coefficient, eddy viscosity in the boundary layer and velocity profiles from different codes are compared. This a very simple case used for verification purpose and results are positive.



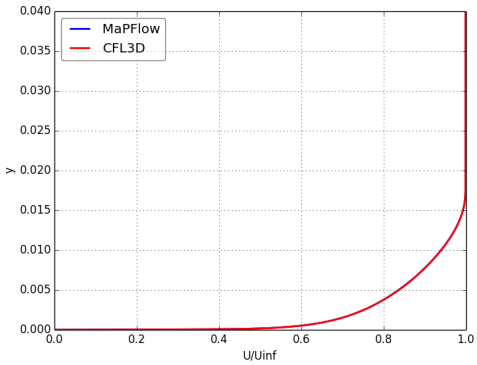
(a) Surface Friction Coefficient



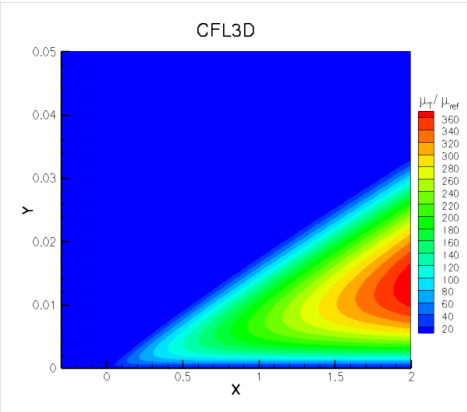
(b) The law of the wall at x=0.97



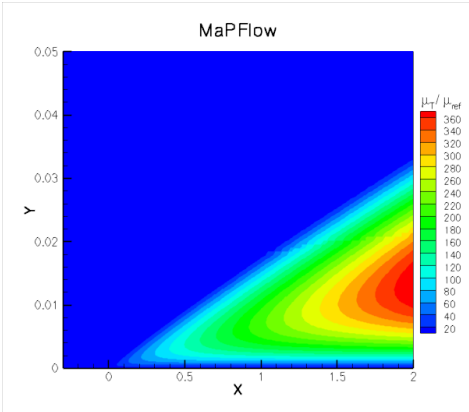
(c) Eddy viscosity at x=0.97



(d) Velocity profile

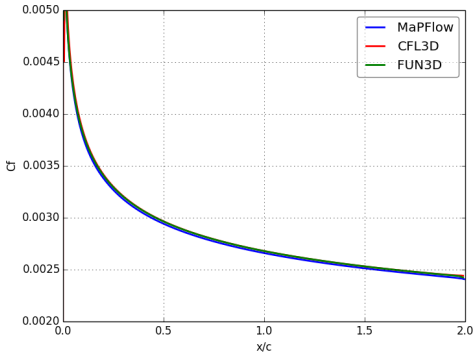


(e) Contour of non-dimensional eddy viscosity, CFL3D

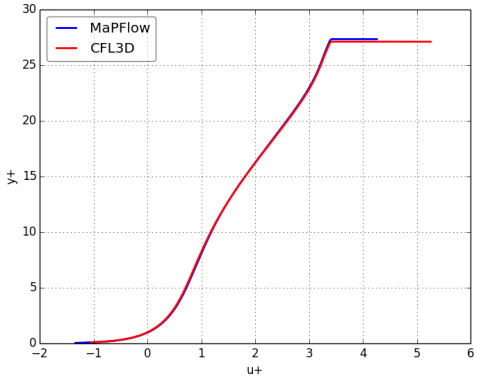


(f) Contour of non-dimensional eddy viscosity, MaPFlow

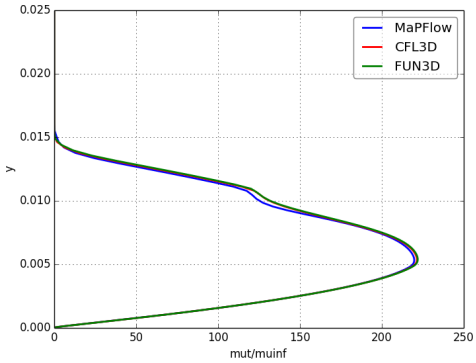
Figure 4.2: SA model results. The Codes predict almost identical results.



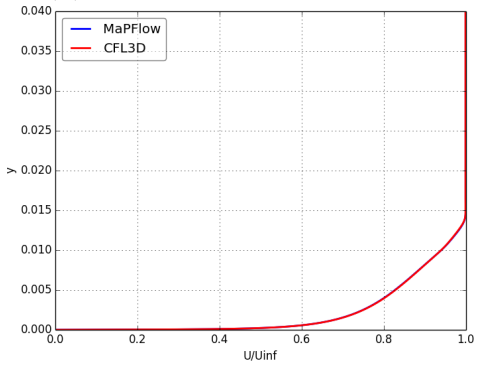
(a) Friction Coefficient



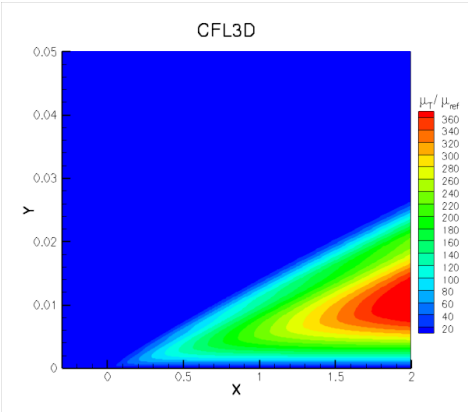
(b) Boundary layer at x=0.97



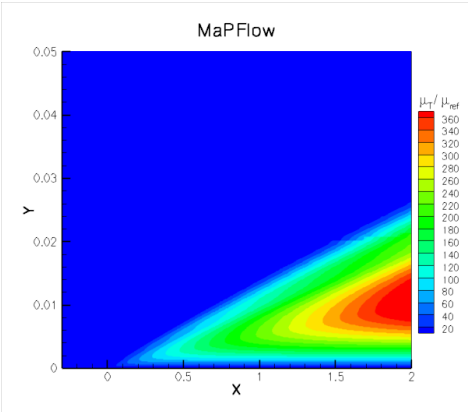
(c) Eddy viscosity at x=0.97



(d) Velocity profile



(e) Contour of non-dimensional eddy viscosity, CFL3D



(f) Contour of non-dimensional eddy viscosity ,MaPFlow

Figure 4.3: SST model results. The Codes predict almost identical results

4.1.2 NACA0012 Airfoil Validation case

The next test case was also taken from [114]. It concerns the flow around a NACA0012 airfoil in attached flow conditions at 0.15 Mach Number and 6 million Reynolds(see Table 4.2). Comparison is made with results obtained with the CFL3D code as well as with experimental data from [117]. The structured grid of 897x257 points (514 points on the airfoil) provided in [114] was used.

Table 4.2: NACA0012 flow conditions

Mach	Reynolds	Angle of Attack
0.15	$6 \cdot 10^6$	$0^o, 10^o, 15^o$

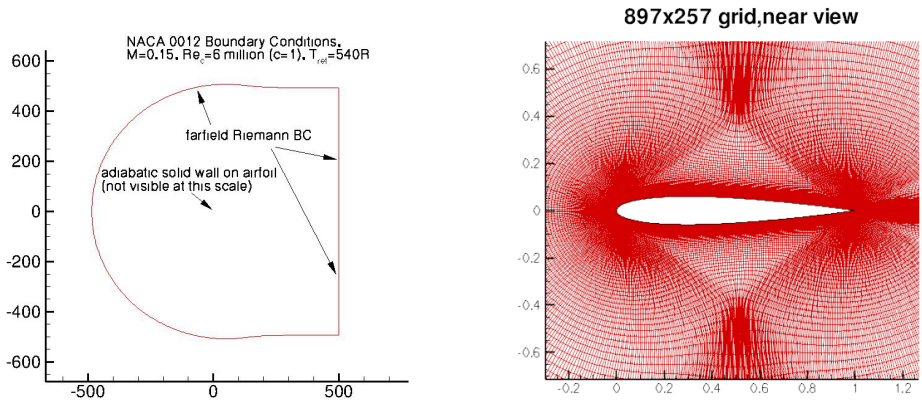


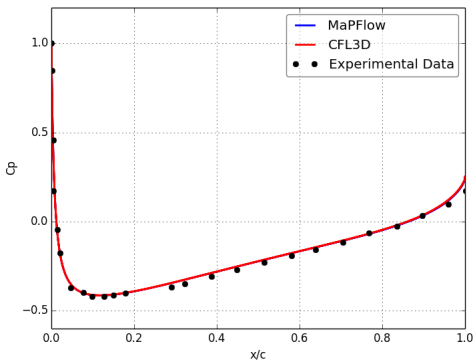
Figure 4.4: The Computational Grid (taken from [114])

Results were obtained at three Angles of Attack(AoA): $0^o, 10^o$ and 15°. The comparisons made include the lift and drag coefficients as well as the pressure (Cp) and friction (Cf) distributions. In Table 4.3 the relative error for Cl and Cd between MaPFlow and CFL3D is presented. The maximum relative error for Cl is 0.4% while for Cd the maximum error is 5.5% (at 15° using SST model). At $\alpha = 0^o$ the error in Cl is higher but this is because the lift itself is very close to zero.

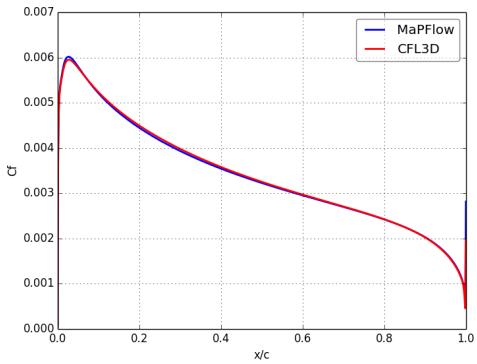
Table 4.3: Cl,Cd Comparison

	0°		10°		15°	
Code	Cl	Cd	Cl	Cd	Cl	Cd
MaPFlow(SA)	-6.85E-06	8.19E-03	1.090E+00	1.23E-02	1.546E+00	2.12E-02
CFL3D (SA)	-5.68E-06	8.34E-03	1.091E+00	1.25E-02	1.548E+00	2.21E-02
Δ %(SA)	**1.70E+01**	**-1.90E+00**	**-5.878E-02**	**-1.76E+00**	**-1.455E-01**	**-4.13E+00**
MaPFlow(SST)	-7.62E-06	8.09E-03	1.077E+00	1.236E-02	1.50E+00	2.21E-02
CFL3D (SST)	-5.15E-06	8.07E-03	1.078E+00	1.217E-02	1.51E+00	2.09E-02
Δ %(SST)	**3.24E+01**	**2.17E-01**	**-3.984E-02**	**1.535E+00**	**-4.09E-01**	**5.51E+00**

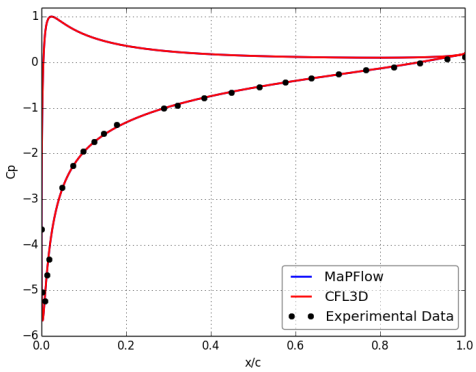
Good agreement was also found in terms of Cp and Cf for the three angles of attack considered (Fig. 4.5 for SA and Fig. 4.6 for SST turbulence model). Some minor differences between MaPFlow and CFL3D can be noted in the Cf results, Fig. 4.5 and 4.6. Deviations of that order are expected in quantities like Cf that are very sensitive to numerical implementation details.



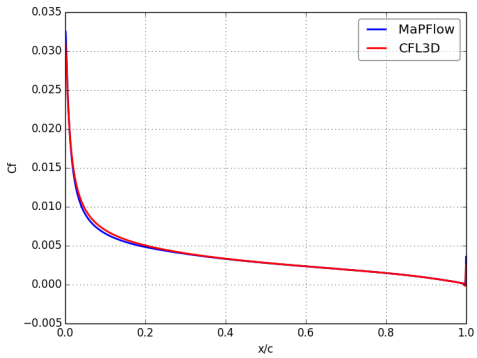
(a) Pressure Coefficient at 0 degrees



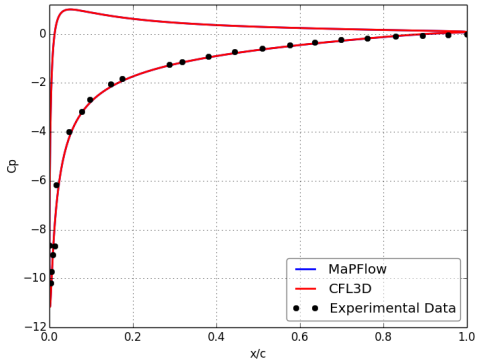
(b) Friction Coefficient at 0 degrees



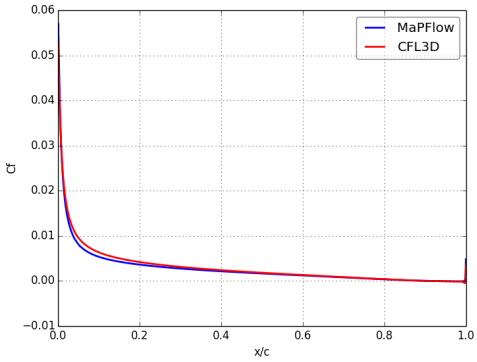
(c) Pressure Coefficient at 10 degrees



(d) Friction Coefficient at 10 degrees

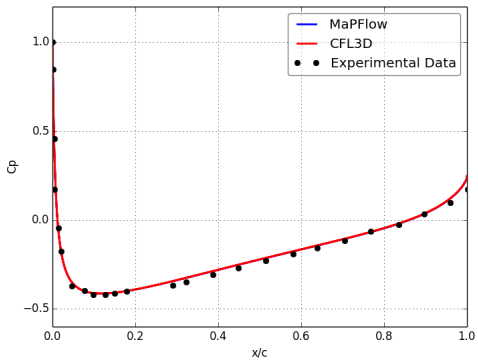


(e) Pressure Coefficient at 15 degrees

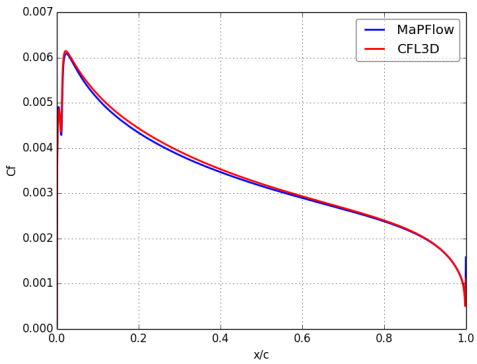


(f) Friction Coefficient at 15 degrees

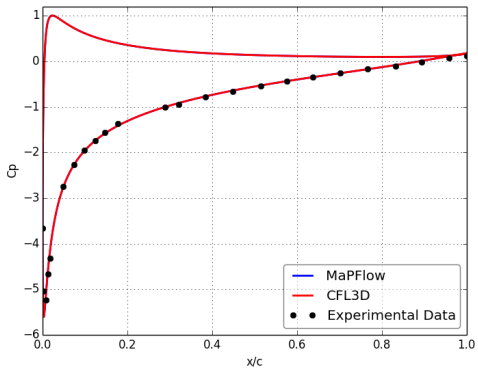
Figure 4.5: SA model results. The two Codes and the Experiment are in good agreement. Minor differences are present in the Friction Coefficient predictions.



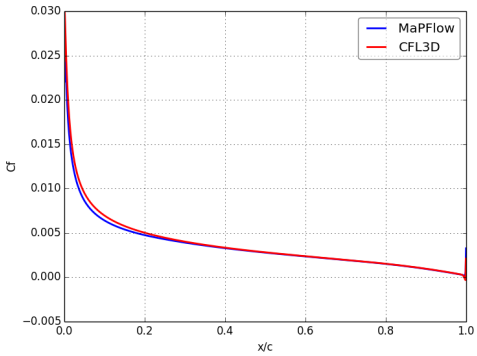
(g) Pressure Coefficient at 0 degrees



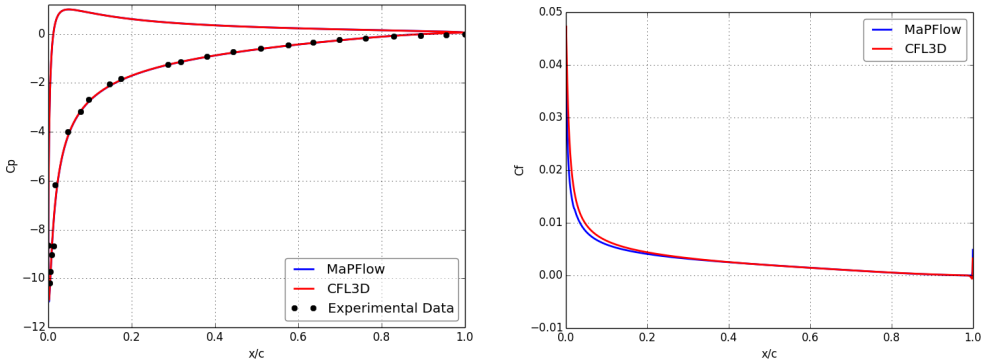
(h) Friction Coefficient at 0 degrees



(i) Pressure Coefficient at 10 degrees



(j) Friction Coefficient at 10 degrees



(a) Pressure Coefficient at 15 degrees (b) Friction Coefficient at 15 degrees

Figure 4.6: Menter SST model results. The two Codes and the Experiment are in good agreement. Minor differences are present in the Friction Coefficient predictions.

4.1.3 3D Separation on an Airfoil

The next case is related to an experiment conducted at NTUA [101]. The flow over a rectangular wing model was examined in the low speed closed loop wind tunnel at the Laboratory of Aerodynamics. The airfoil shape was optimized for use on wind turbine blades [118]. The measurements were combined with 2D and 3D CFD simulations using MaPFlow and have been presented in detail in [101]. Here some selected results are given in view of supporting the validation of MaPFlow.

Table 4.4: NTUA experiment flow conditions

Mach	Reynolds	Angle of Attack
0.08	$0.87 \cdot 10^6$	$7^o, 16^o$

A grid of approximately $1 \cdot 10^6$ cells (Fig. 4.8) was used for one half of the model. The simulations implemented inviscid wall condition on the side fences (Fig. 4.7) at the model extremities and symmetry conditions at the center of the wing span.

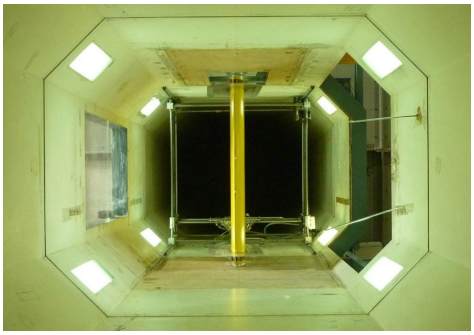


Figure 4.7: Experimental Setup. Rectangular Wing in NTUA Wind Tunnel.

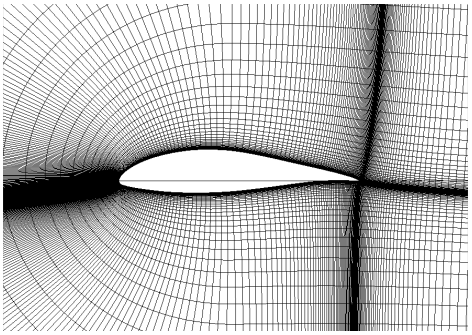
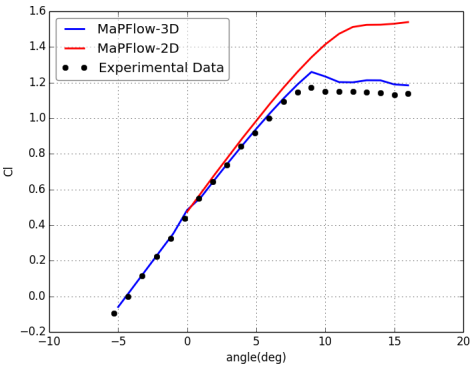


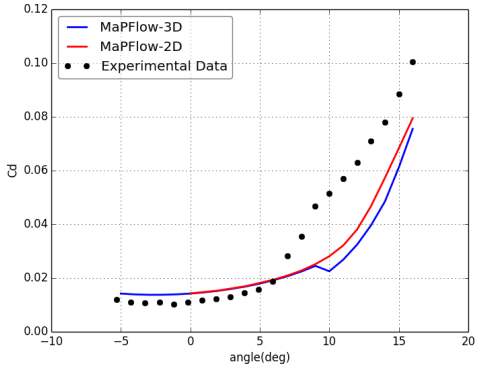
Figure 4.8: Slice of the 3D Computational Grid

Figure 4.9: NTUA experimental and Computational setup

As detailed in [101] the flow exhibits 3D separation which explains the reason for conducting 3D simulations in a nominally 2D flow. The 3D effects on the flow are clearly seen in the polars given in Fig. 4.10. The deviation between the 2D and 3D predictions increases with α while only the 3D results follow the experimental results..

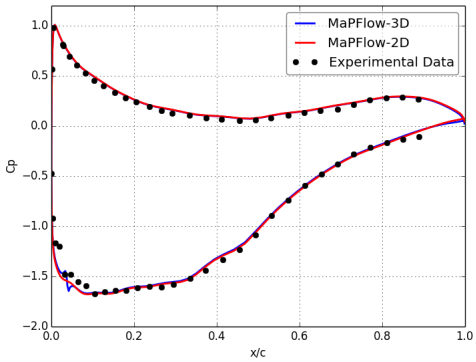


(a) Lift Coefficient vs Angle of Attack

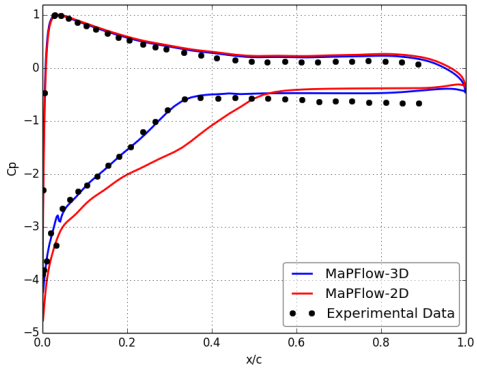


(b) Drag Coefficient at vs Angle of Attack

Figure 4.10: NTUA experiment: Lift and drag polars



(a) Pressure Coefficient at 7 degrees



(b) Pressure Coefficient at 16 degrees

Figure 4.11: NTUA experiment: Comparison of pressure distributions at 7 and 16 degrees of incidence

This is also confirmed by the pressure distributions in Fig 4.11. At $\alpha = 7^o$ (Fig 4.11a), the flow is attached and because of that it keeps its 2D character. However, at $\alpha = 16^o$, 2D simulations fail to accurately predict the extent of separation ($x/c = 0.5$). On the contrary the 3D predictions accurately reproduce the experimental data. This difference in behavior is due to the onset of large three dimensional structures that form as Angle of Attack(α) enters stall and because of that they are known in the literature as "Stall Cells". In Fig. 4.13 the surface flow visualization clearly shapes the stall cell while in Fig. 4.12 MaPFlow predictions are compared to Stereo PIV measurements as an indication of the quality of the simulations mentioned here. Further details can be found in [119, 120, 121].

From a validation point of view, the 3D results agree well with the measurements especially in the linear region of angles of attack. In stalled conditions MaPFlow predicts larger lift and lower drag as compared to the experimental data. More specifically, MaPFlow predicts the onset of separation with approximately a 3° delay. However, once separation takes place the deviation from the measurements remains constant.

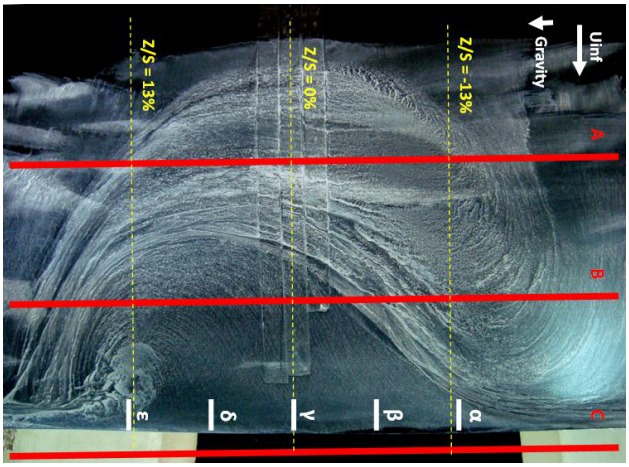


Figure 4.12: Shape of the stall cell using flow visualization. Taken from [101]

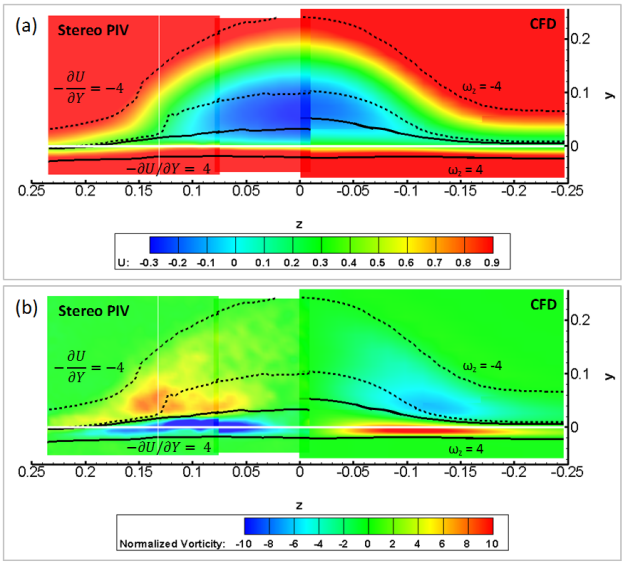
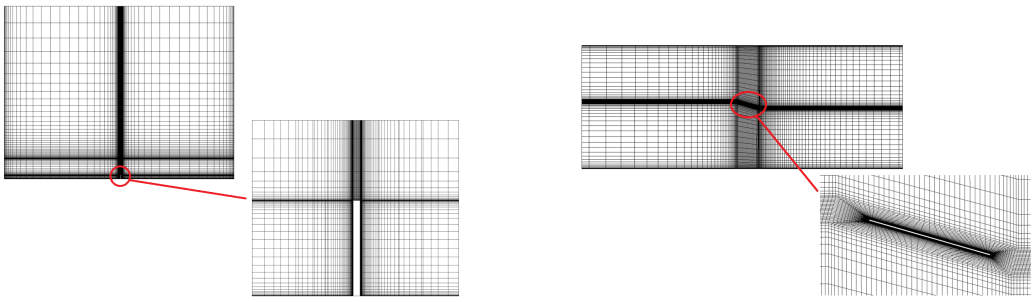


Figure 4.13: Contours of normalized streamwise vorticity. Fair comparison of the experimental with the numerical data. Taken from [101]

4.1.4 3D Flat Plate with Vortex Generator

In this case the flow over a zero pressure gradient flat plate in the presence of a Vortex Generator (VG) is considered. A comparison is made between two CFD codes, MaPFlow and WIND-US [122], and with respect to experimental data [123]. The aim was to evaluate the performance of the jBAY model [99] which was implemented in MaPFlow [101, 124] in comparison to simulations that fully resolve the VGs. The computational setup was the same as in [122]. Fig. 4.14 shows the grid when the VG is fully resolved. The VG was considered as a rectangular flat plate placed at $\alpha = 16^o$ of inclination while the grid amounted to approximately $3 \cdot 10^6$ cells.



(a) Computational Grid-Front View (b) Computational Grid-Top View

Figure 4.14: Fully resolved VG grid

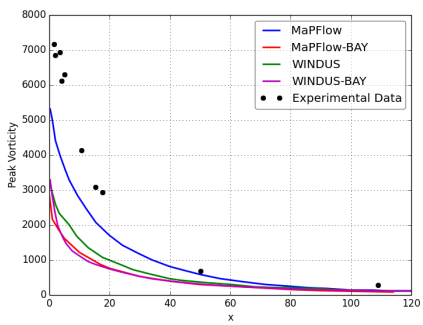


Figure 4.15: Variation of Peak Vorticity. The Fully resolved simulation predicts higher Peak Vorticity than that based on the BAY Model. Yet, the experimental data show consistently higher values.

In Fig. 4.15 the evolution of the peak vorticity is considered. The agreement of the two implementations of the BAY model is good but both fail to predict the high peak values close to the VG. When the VG is fully resolved, MaPFlow predicts much higher peak values as compared to the WIND-US results, which is

probably due to differences in the grid. While full resolution of the VGs significantly improves the quality of the predicted peak vorticity, it still deviates from the experimental data.

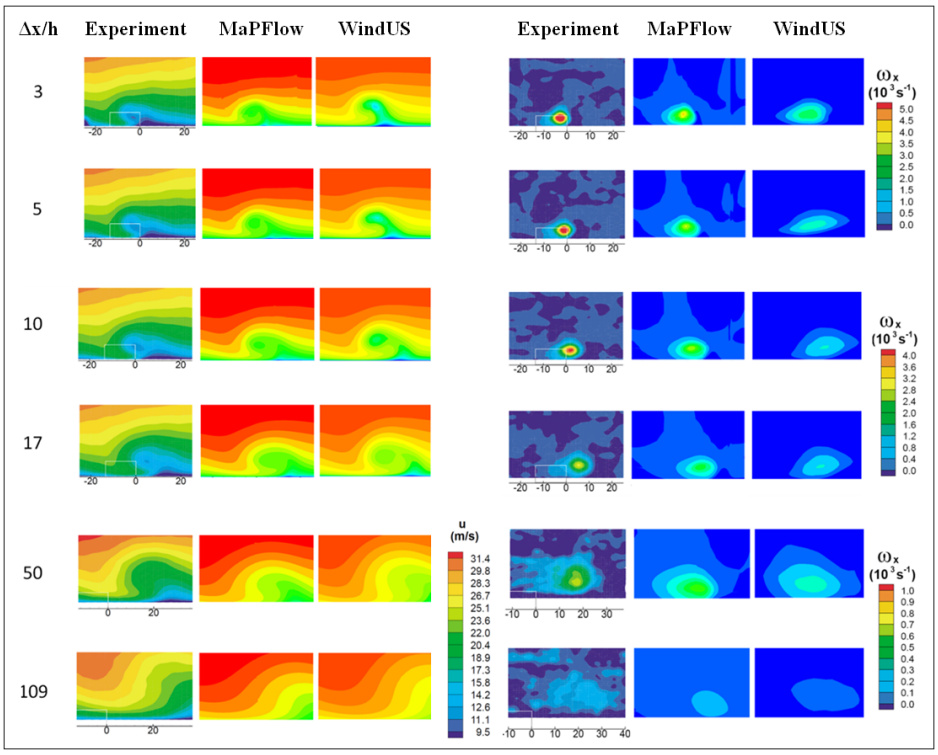


Figure 4.16: Comparison of Velocity (left) and Vorticity (right) field at six stations downstream of the VG. In the simulation data diffusion is more pronounced than in the experimental data results.

Next in Fig 4.16 flow contours are presented at six positions downstream of the Vortex Generator. On the left velocity contours are depicted while on the right vorticity along the x-axis is displayed. Both codes fail to accurately reproduce the experimental data. The vortex in the simulations appears to diffuse much faster. Besides that, already at the VG trailing edge, the predicted vortex is significantly weaker than that shown in the measurements. Limited tests with more dense grids up to the size our computer facilities allow did not show appreciable improvement in this respect, but further analysis should be carried out.

Another point of concern, is the type of turbulence model used. Eddy viscosity models have difficulty in reproducing flows with strong secondary vorticity as is the case along the tip of a VG which operates with a wall boundary layer. However investigation in this direction was outside of the scope of this work. So from this test case, the conclusion is that MaPFlow behaves at least similarly to other existing CFD codes.

4.1.5 ONERA M6 validation case

This test case was taken from [125] as reference for 3D steady compressible flows. The flow is at Ma=0.8395 so on the wing a supersonic area is formed. The flow conditions are summarized in table 4.5.

Table 4.5: ONERA M6 flow conditions

Mach	Reynolds	Angle of Attack
0.8395	$11.72 \cdot 10^6$	3.06^o

Two different grids were used, a coarse with $y+ \approx 30$ and a denser one with $y+ \approx 1$. The coarse grid was taken from [125] (Fig. 4.17) while the fine from [126]. The coarse grid consists of $3 \cdot 10^5$ cells and was used along with wall functions at the wall region. It is the same grid used by the WIND-US code where also wall functions were used. The finer one consists of $9 \cdot 10^5$ cells and was used when the boundary layer is fully resolved. Finally the experimental data were taken from [127].

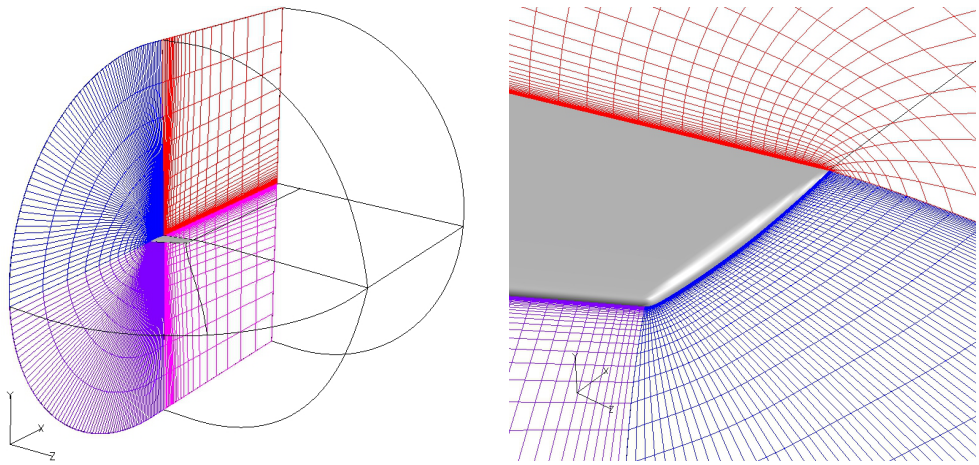
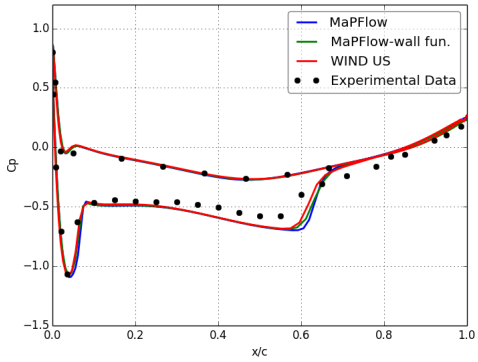
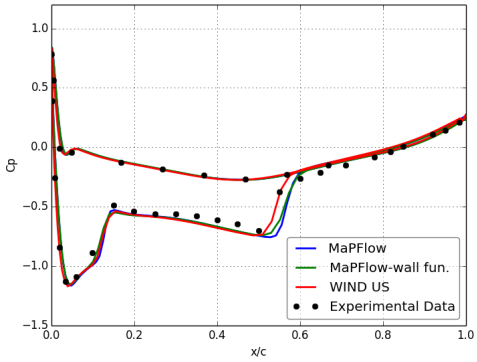


Figure 4.17: The Computational Grid (taken from [125])

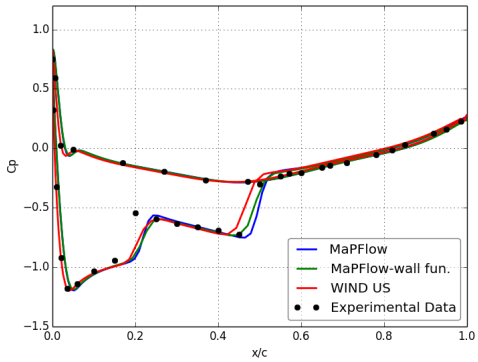
In Fig. 4.18 comparisons are shown in terms of sectional pressure distributions at six spanwise locations. There is good agreement at all stations. There is slight shift in the MaPFlow results as compared to the other data. This is an artifact of results post-processing (WIND-US uses CFPOST to post-process the data and extract the Cp curves [125]). The comparisons indicate that in this case, the modeling of the boundary layer with wall functions is adequate.



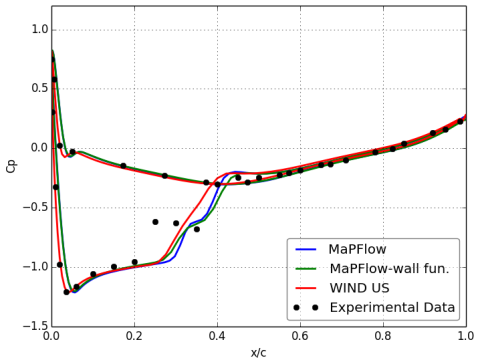
(a) Pressure Coefficient at $y/(b/2)$=20%



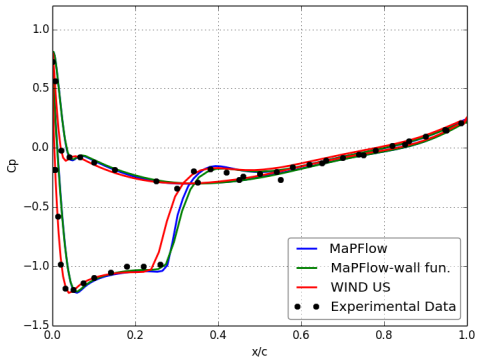
(b) Pressure Coefficient at $y/(b/2)$=44%



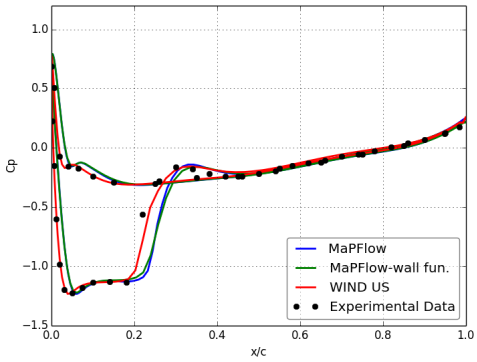
(c) Pressure Coefficient at $y/(b/2)$=65%



(d) Pressure Coefficient at $y/(b/2)$=80%



(e) Pressure Coefficient at $y/(b/2)$=90%



(f) Pressure Coefficient at $y/(b/2)$=95%

Figure 4.18: Onera M6: Chordwise pressure distribution at various spanwise stations (Ma=0.84, $\alpha = 3.06^o$).

4.1.6 The MEXICO Experiment

The next test case concerns the MEXICO Rotor, which was tested in the open jet wind tunnel at the German Dutch Wind Tunnel Organization (DNW) [128]. CFD results are compared to experimental data and predictions obtained with the Free Wake code GENUVP [30]. Flow conditions are defined in Table 4.6.

Table 4.6: Flow conditions in the MEXICO experiment

U_∞	Ω (RPM)	Blade Pitch
10 m/s	425	-2.3^o
15 m/s	425	-2.3^o
24 m/s	425	-2.3^o

Fig 4.19 gives an outline of the grid. Due to periodicity only one blade is considered. Three grids were used of $5 \cdot 10^6$, $8 \cdot 10^6$ and $11 \cdot 10^6$ cells all having y+$\approx$ 1. As explained next, the specific refinement was dictated by the inadequacy of the less coarse grid to correctly predict the flow field in the wake of the rotor. The intermediate grid was produced in this direction while for the denser, a global refinement was carried out. All grids were constructed with ANSYS ICEM CFD grid generator.

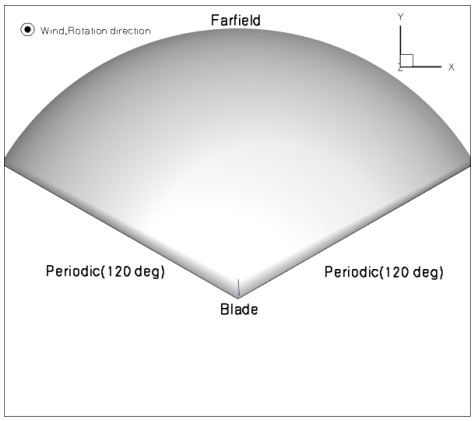
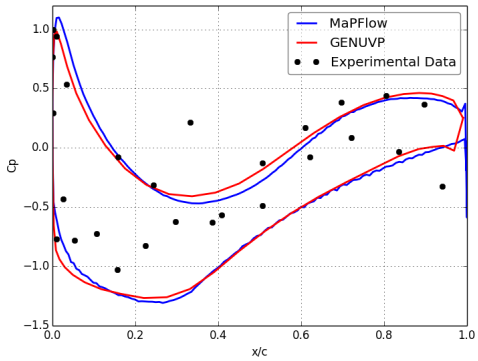


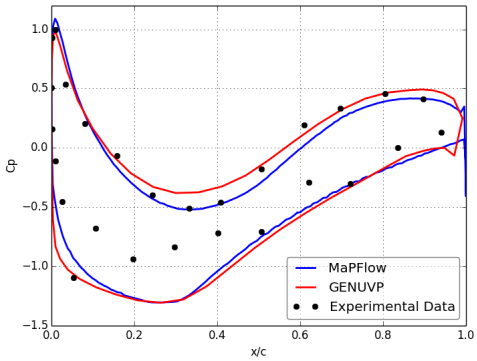
Figure 4.19: The Computational Domain

In Fig. 4.20, 4.21 and 4.22 pressure data are compared, for three free-stream velocities at five spanwise locations. All three grids produced approximately the same distributions Cp curves. Also, as expected the GENUVP predictions over estimate the blade loading as compared to CFD. However despite the good overall agreement between the two methodologies, the results indicate that both methods give higher loading compared to the experimental data.

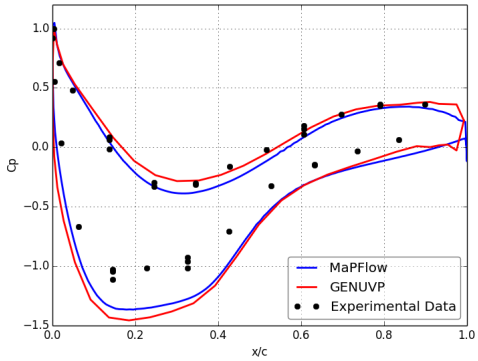
The $10m/s$ case



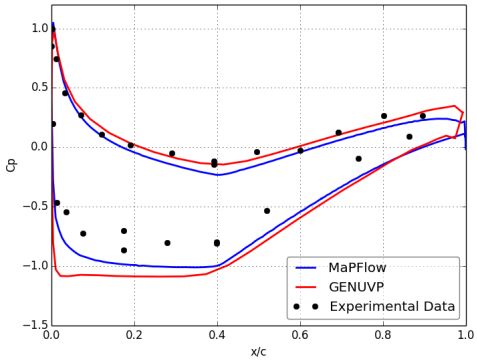
(a) Pressure Coefficient at r/R=25%



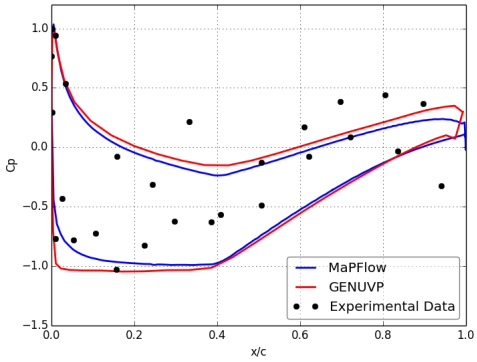
(b) Pressure Coefficient at r/R=35%



(c) Pressure Coefficient at r/R=60%



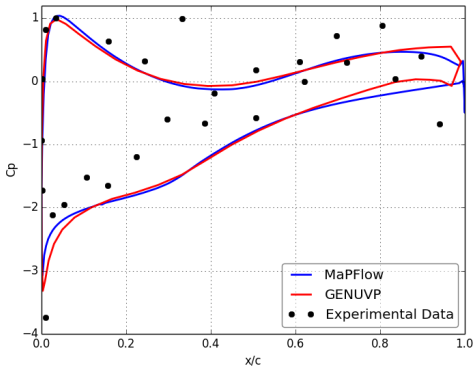
(d) Pressure Coefficient at r/R=82%



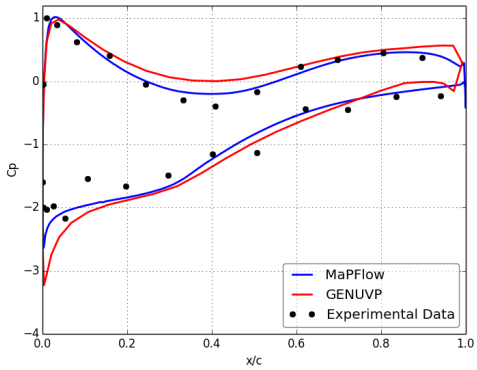
(e) Pressure Coefficient at r/R=92%

Figure 4.20: Mexico Rotor at $10m/s$ inflow. Chordwise pressure distribution comparison at different radial positions

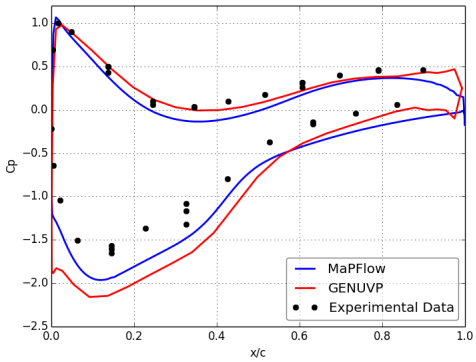
The $15m/s$ case



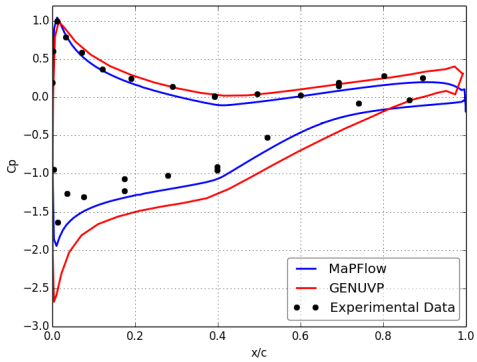
(a) Pressure Coefficient at r/R=25%



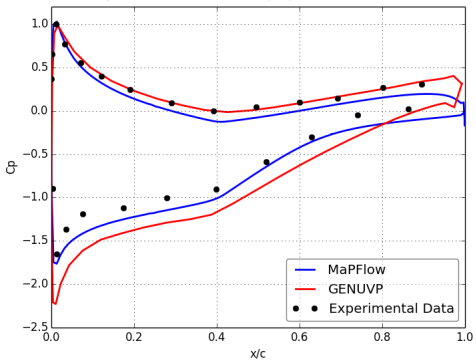
(b) Pressure Coefficient at r/R=35%



(c) Pressure Coefficient at r/R=60%



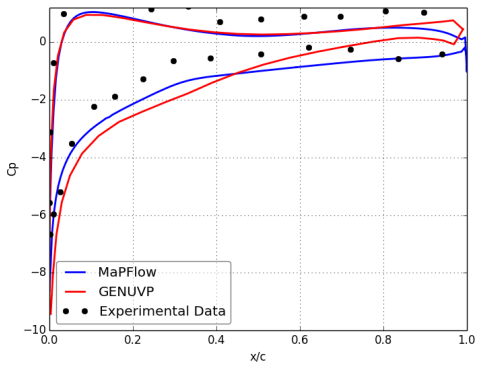
(d) Pressure Coefficient at r/R=82%



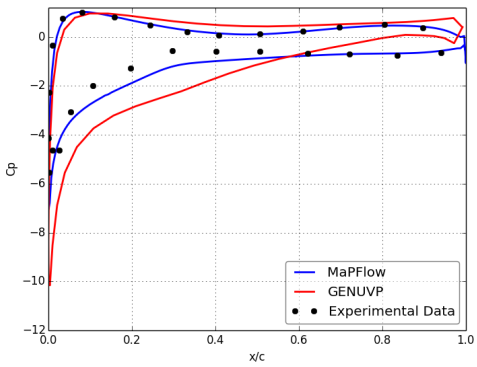
(e) Pressure Coefficient at r/R=92%

Figure 4.21: Mexico Rotor at $15m/s$. Chordwise pressure distribution comparison at different radial positions

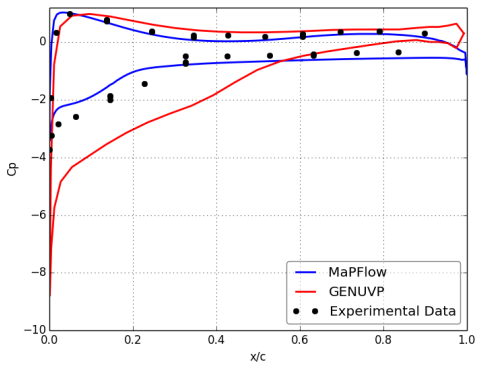
The $24m/s$ case



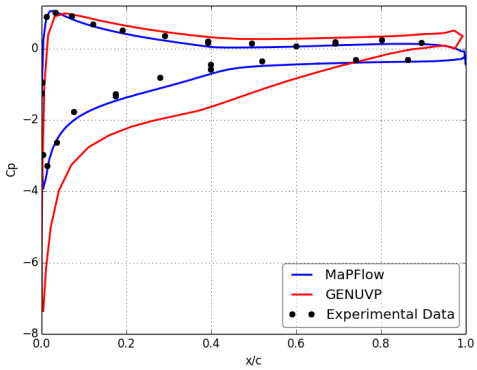
(a) Pressure Coefficient at r/R=25%



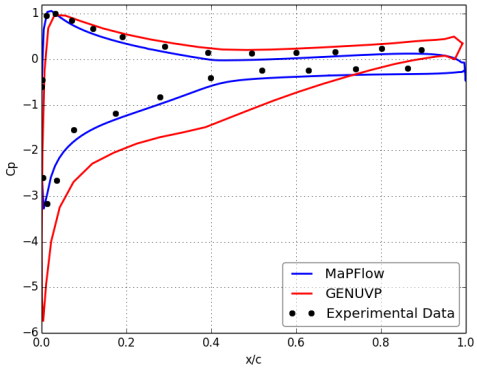
(b) Pressure Coefficient at r/R=35%



(c) Pressure Coefficient at r/R=60%

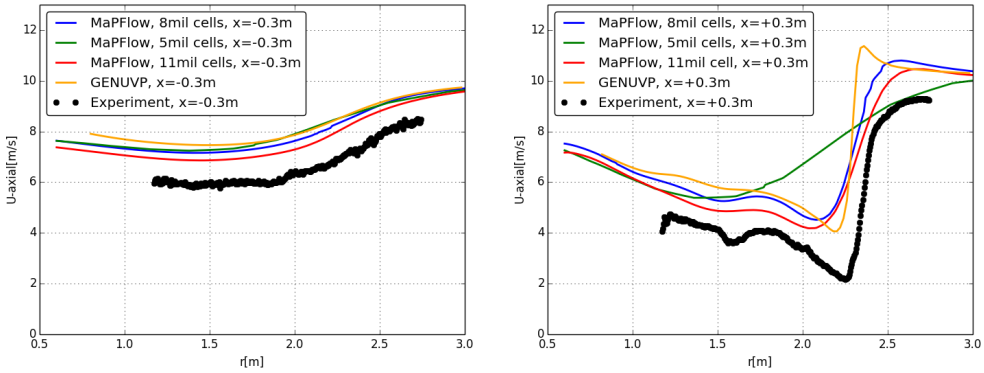


(d) Pressure Coefficient at r/R=82%



(e) Pressure Coefficient at r/R=92%

Figure 4.22: Mexico Rotor at $24m/s$. Chordwise pressure distribution comparison at different radial positions



(a) Axial velocity before (x=-0.3m) the WT

(b) Axial velocity after (x=+0.3m) the WT

Figure 4.23: Mexico Rotor at $10m/s$ inflow: Axial velocity radial distribution upstream and downstream of the rotor plane.

In order to better understand the origin of the deviations in Cp, the velocity data were examined. In Fig. 4.23a the upstream axial velocity at x=-0.3m is shown for the $10m/s$ case. There is a level difference of approximately $1m/s$ in all sets of predictions in comparison to the test data. This deviation in the axial velocity is in agreement with the pressure results. Similarly downstream of the rotor at x=+0.3m (Fig 4.23b), the same level shift is noted at $r > R$. At $r = R$ the presence of the tip vortex is depicted. Only GENUVP produces a sharp shear. This is expected for a Lagrangian inviscid method which is free of diffusion. On the contrary due to numerical diffusion, the CFD solver produces less pronounced shear. The coarser grid completely fails to capture the velocity deficit. The best resolution is produced with the locally refined grid of $8 \cdot 10^6$ points, which however still deviates from the measured data. Apparently an even more dense grid is required. Further inboard over the range $r < 2m$ (which corresponds to the part bellow 80% of the blade span) the predictions get closer. There is still a level shift which in the wake becomes $1.5m/s$.

4.1.7 NREL Phase VI Rotor

The next test concerns NREL Phase VI experiment. A two bladed 20kW wind turbine was tested in the large NASA Ames wind tunnel [129]. Pressure was recorded at various spanwise stations at different wind speeds covering the complete operational range. The specific data were the first detailed and of high quality wind tunnel measurements on a real wind turbine. Because of that, this experiment has been extensively used for CFD validation (see [130, 131, 132]). In this work a grid composed of $5 \cdot 10^6$ cells with y+ ≈ 1 was defined using ANSYS ICEM CFD grid generator and simulations are made with the $k - \omega$ SST turbulence model. Comparisons are made at three wind velocities as defined in Table 4.7.

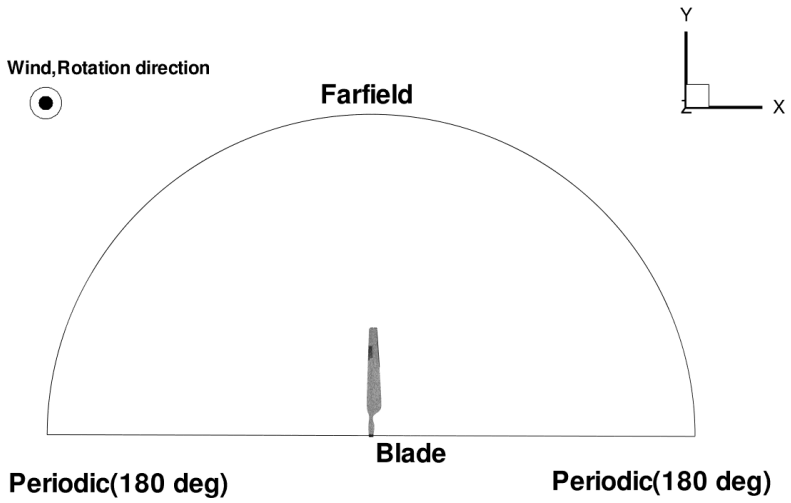


Figure 4.24: The Computational set-up

Table 4.7: NREL flow conditions

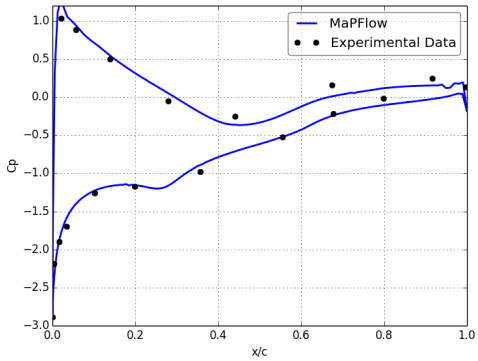
U_∞	Ω (RPM)	Pitch(^{o})
7m/s	72	2.98^o
10 m/s	72	2.98^o
15 m/s	72	2.98^o

At $7m/s$ (Fig. 4.25), the flow is mostly attached and predictions are in good agreement with the experimental data. At $10m/s$ (Fig. 4.26) while the flow at the largest part of the blade remains attached, at $r/R = 47\%$ the pressure distribution indicates the onset of separation. Finally, at $15m/s$ (Fig. 4.27) the flow is detached at least up to $r/R = 80\%$.

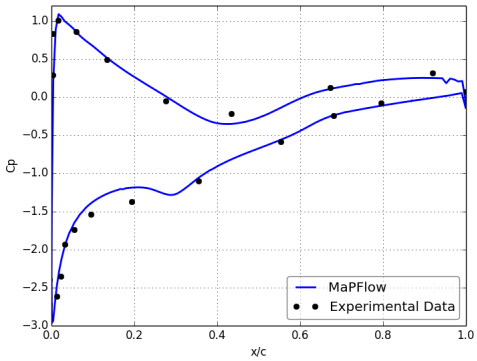
Overall predictions agree well with the measurements. There are deviations at the trailing edge region and over a part of the suction side (in between $x/c =$

$0.1 - 0.3$ and $r/R = 0.47 - 0.8$. They are related to the low accuracy of the surface geometry that was provided. A better geometrical description would probably improve the results. Clearly the differences between predictions and measurements are more pronounced in stalled conditions. For example at the $r/R = 0.30$ station in the $15m/s$ case. The flow under such conditions will be unsteady with probably large variations with respect to the averaged (mean) state. It is noted that both the experimental and numerical results that are presented have been time averaged.

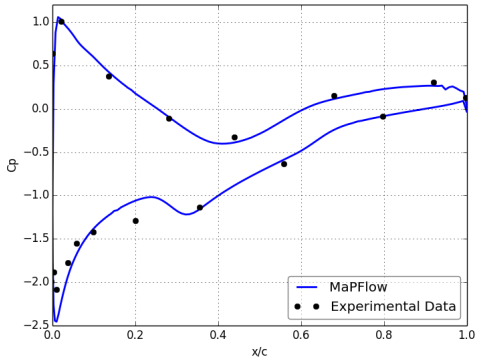
The $7m/s$ case



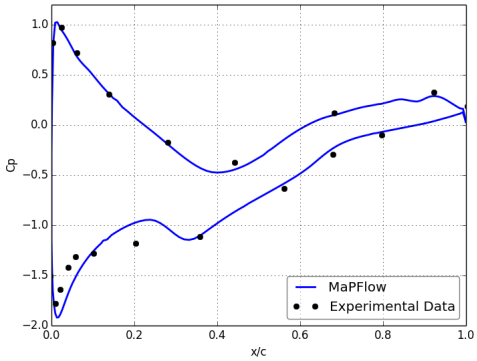
(a) Pressure Coefficient at r/R=30%



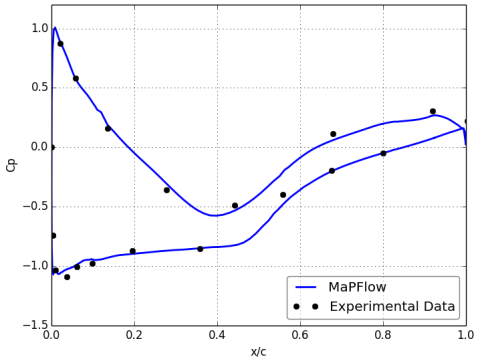
(b) Pressure Coefficient at r/R=47%



(c) Pressure Coefficient at r/R=63%



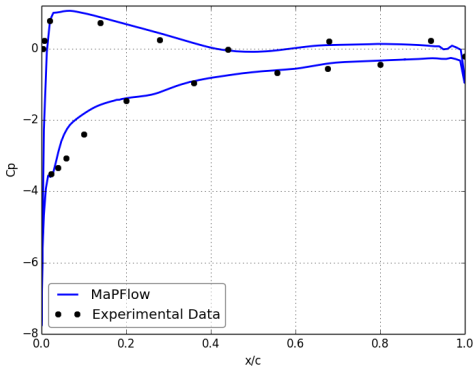
(d) Pressure Coefficient at r/R=80%



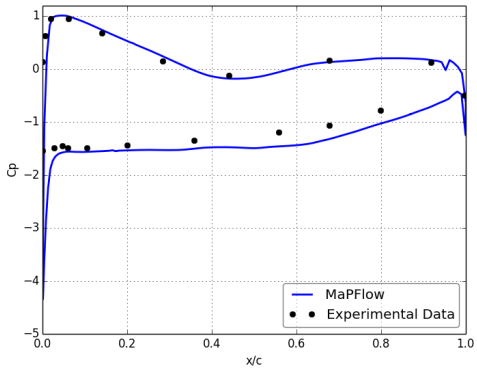
(e) Pressure Coefficient at r/R=95%

Figure 4.25: NREL Rotor at $7m/s$ inflow. Chordwise pressure distribution comparison at five radial positions

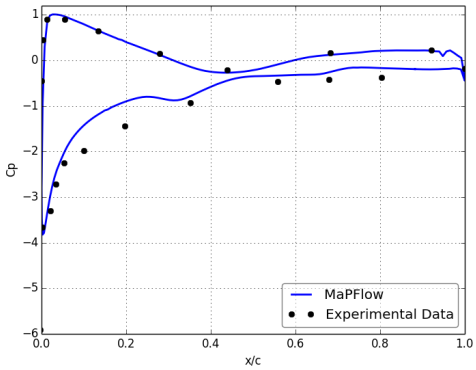
The $10m/s$ case



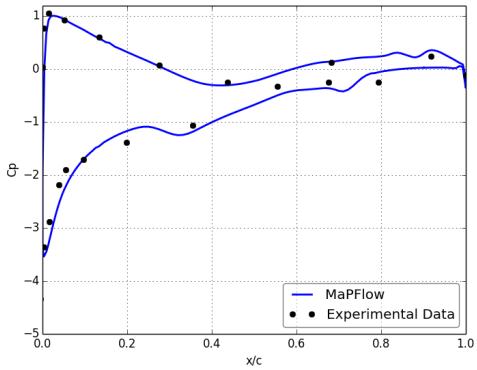
(a) Pressure Coefficient at r/R=30%



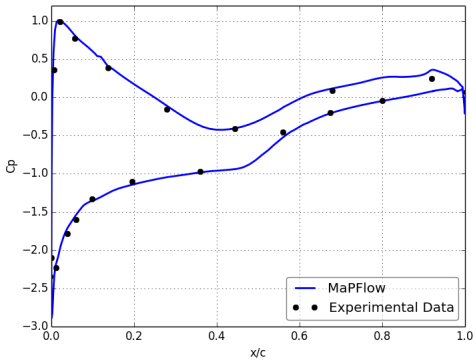
(b) Pressure Coefficient at r/R=47%



(c) Pressure Coefficient at r/R=63%



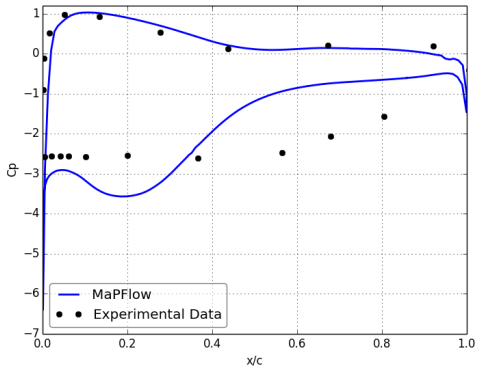
(d) Pressure Coefficient at r/R=80%



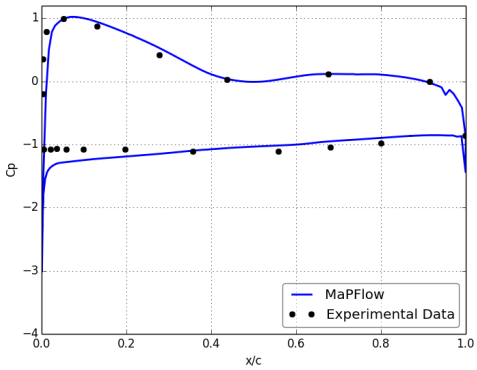
(e) Pressure Coefficient at r/R=95%

Figure 4.26: NREL Rotor at $10m/s$ inflow. Chordwise pressure distribution comparison at fiver radial positions

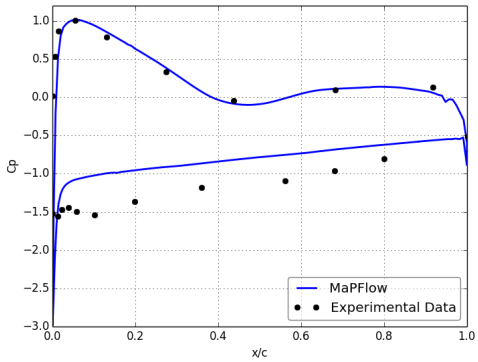
The $15m/s$ case



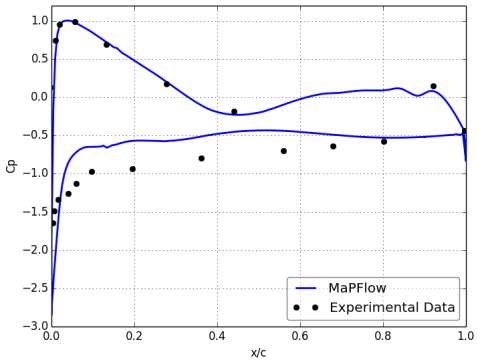
(a) Pressure Coefficient at r/R=30%



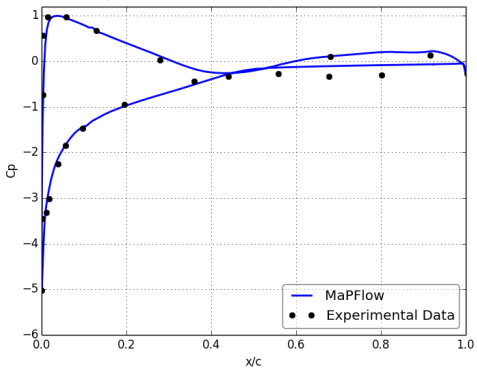
(b) Pressure Coefficient at r/R=47%



(c) Pressure Coefficient at r/R=63%



(d) Pressure Coefficient at r/R=80%



(e) Pressure Coefficient at r/R=95%

Figure 4.27: NREL Rotor at $15m/s$ inflow. Chordwise pressure distribution comparison at five radial positions

4.2 Validation and Verification of the Hybrid Solver (HoPFlow)

Before discussing the results obtained with the hybrid method, first the computational setup is explained followed by a parametric analysis with respect to the basic numerical parameters.

4.2.1 The Computational Set-up

The computational setup as presented in Fig. 4.28 consists of:

1. An Eulerian (CFD) grid,
2. The (Inner) PM grid, which is the domain of the Lagrangian solver,
3. The far-field (Outer) PM grid which provides the effect of the far wake on the inner PM grid.

The PM grids are characterized by their spacing $\Delta X_{PMout}, \Delta X_{PM}$ and their size N_{PMout}, N_{PM}. The CFD grid is also characterized by its size Nc_{CFD} and the number of CFD particles per cell NP_{CFD}.

Figure 4.28: The hybrid computational setup consists of a small CFD grid in which solid boundaries are included, an inner Particle Mesh (PM) region that contains the CFD grid and an outer PM grid that covers the full space.

The Outer PM grid grows in size as the far wake evolves and the particles in this region carry only vorticity. They have constant volume and are convected with U_∞. The Outer PM is coarser than the Inner one since it is only used to add the far wake contribution. Finally, in order to prevent it from growing infinitely a length threshold is applied.

Another important aspect of the computational setup is the need to have full coverage of the PM grid with particles. To alleviate the concentration or the scattering of particles, re-meshing is employed every N_{RM} time-steps. The frequency

of re-meshing is basically decided by the time-step and the resolution of the PM grid.

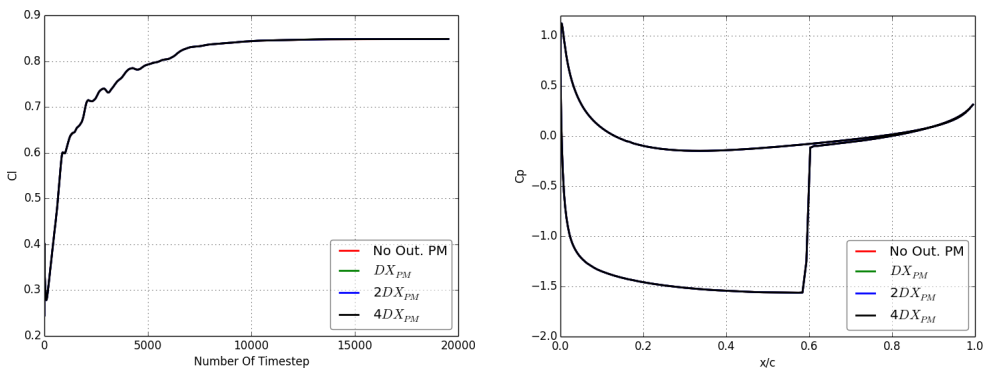
Finally of numerical significance is the time step. The Lagrangian Solver uses a 1st order explicit scheme while the CFD solver a 2nd order implicit. In this sense the time step is restricted by the Lagrangian solver.

4.2.2 Parametric study

For the parametric study, the case of a highly compressible inviscid flow around a NACA0012 airfoil is considered. The Mach Number is 0.755 and the Angle of Attack 4°.

Resolution of the Outer PM grid

The Outer PM domain is not fixed (as opposed for the Inner PM) and thus the domain grows together with the far wake. As previously stated, a maximum length criterion is applied to prevent the Outer PM grid from growing infinitely. In the present study the maximum length is chosen to be 30 chords. Four cases have been considered and results are compared in Fig. 4.29. All four sets of results were identical with respect to pressure distribution and convergence rate.



(a) Cl convergence history, $Ma = 0.755, \alpha = 4^o$

(b) Pressure coefficient distribution, $Ma = 0.755, \alpha = 4^o$

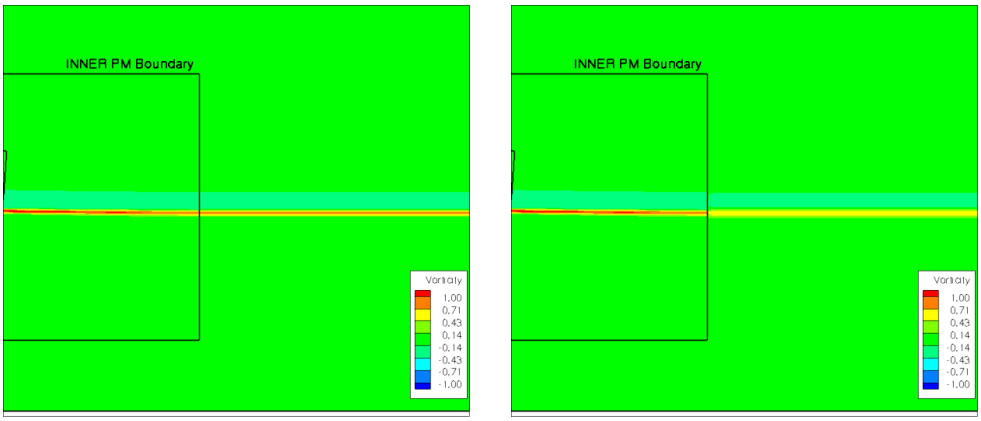
Figure 4.29: Comparison of different Outer PM grid resolutions. "No OUT PM": there is no Outer PM grid and the Inner grid extends to 30c; "DX_{PM}": $DX_{PMout} = DX_{PM}$; "$2DX_{PM}$": $DX_{PMout} = 2DX_{PM}$; "$4DX_{PM}$": $DX_{PMout} = 4DX_{PM}$

In Fig. 4.30 vorticity contours are displayed for $DX_{PMout} = DX_{PM}$ (Fig. 4.30a) and $DX_{PMout} = 4DX_{PM}$ (Fig. 4.30b). As expected, in the coarser PM grid, the projected vorticity is more diffused. However, since this part of the wake is away from the airfoil, any error on the loading would be negligible. This

is quantitatively supported by the coincidence of the cases with and without an Outer PM region. This actually clarifies that the use of an Outer PM does not degrade the accuracy of the solution.

The results indicate that a $DX_{PMout} = 4DX_{PM}$ is a logical decision. The option of a an even coarser Outer PM grid was considered, however, since the computational cost of the Outer PM is negligible the idea of an even coarser grid was abandoned.

It is noted that the specific flow is not the best suited for the Outer PM parametric study because the shock-induced vorticity is relatively small. A test case with strong vortices in the wake would be more suitable. In spite of this, the purpose of this study is to make some initial estimation of the basic computational setup.



(a) Far wake vorticity for $DX_{PMout} = D_{PM}$

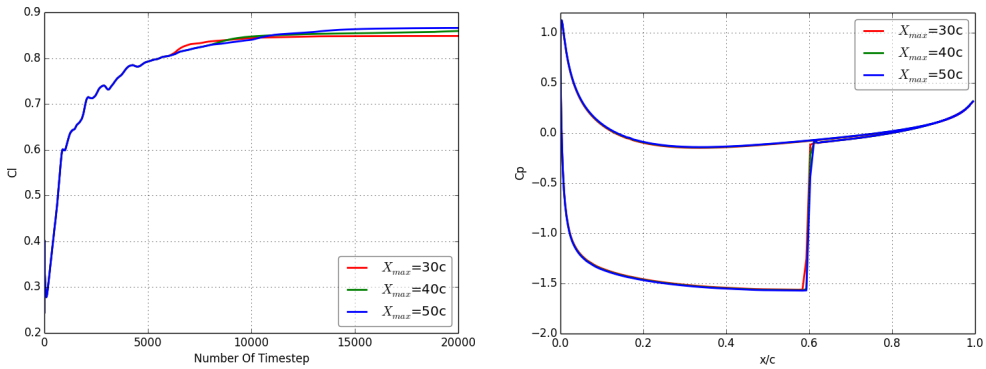
(b) Far wake vorticity for $DX_{PMout} = 4D_{PM}$

Figure 4.30: Comparison of projected far wake Vorticity for two grid resolutions

Length of the Outer PM grid

In order to fully define the Outer PM grid, the characteristics its maximum length allowed should be checked. As already mentioned, the Outer PM grid dynamically expands together with the far wake up to certain limit X_{max}. In order to check the effect this bound could have on the solution the following three cases were considered:

1. $X_{max} = 30c$ with $DX_{PMout} = 4DX_{PM}$
2. $X_{max} = 40c$ with $DX_{PMout} = 4DX_{PM}$
3. $X_{max} = 50c$ with $DX_{PMout} = 4DX_{PM}$



(a) Cl convergence history, $Ma = 0.755, \alpha = 4^o$

(b) Pressure coefficient distribution, $Ma = 0.755, \alpha = 4^o$

Figure 4.31: Comparison of the Outer PM grid length

Table 4.8: Cl Comparison

Cl $X_{max} = 30c$	Cl $X_{max} = 40c$	Cl $X_{max} = 50c$	$max(\Delta \text{Cl})$
8.48E-01	8.59E-01	8.65E-01	2.069 %

In Fig. 4.31 the Cl convergence histories and the pressure distributions on the airfoil are compared. In terms of Cl, a maximum difference of 2% was found. However, a closer look in the Cp curves (Fig. 4.31b) clarifies that these differences in Cl result from minor changes in the shock region. Small changes in the shock location have a much greater impact on Cl. Nevertheless $X_{max} = 30c$ was chosen for the cases presented later on by considering that the 2 % difference in Cl is acceptable.

Resolution in the Inner PM grid

After defining the parameters of the Outer PM grid the next step concerns the Inner PM grid characteristics. Contrary to the Outer PM, the Inner PM domain is fixed in time. For a $2L \times 2L$ PM grid the resolution is defined by ΔX_{PM}.

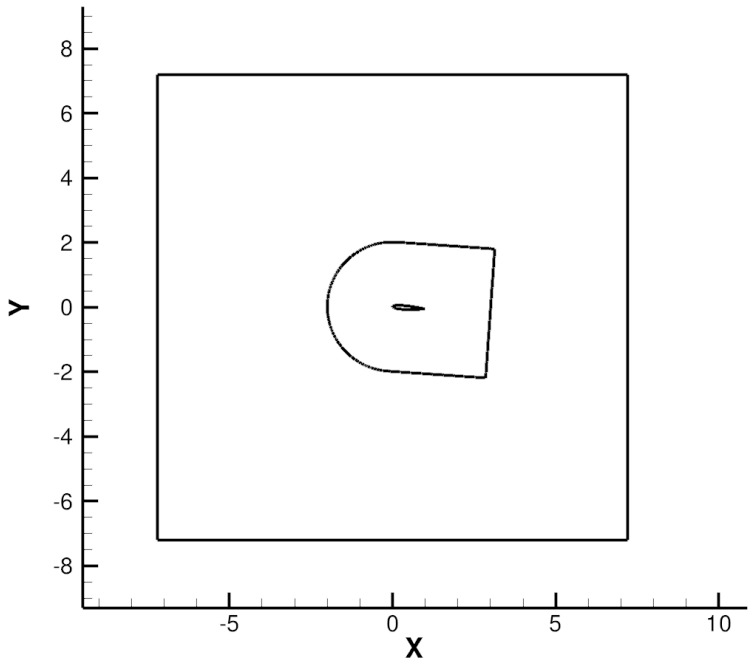
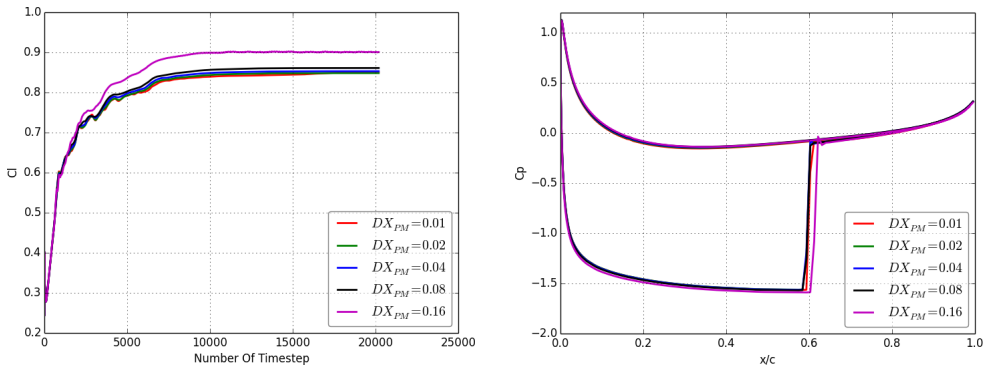


Figure 4.32: Inner PM Domain definition

For the set-up shown in Fig. 4.32, $L = 7.2$. Assuming the same uniform resolution in the x and y, a gradual increase in ΔX_{PM} from 0.01 up to 0.16 is considered. Except for the finest resolution, four particles per CFD cell proved to be sufficient. In the case of the finest grid, the number of particles per cell was raised to 9 in order to ensure full coverage of particles.

In Table 4.9 the effect of the ΔX_{PM} size on Cl is shown, while in Fig. 4.33 Cl convergence histories and pressure distributions are compared. With the exception of $DX_{PM} = 0.16$, which clearly stands out, all other are in good agreement. Although a $DX_{PM} = 0.08$ or even $DX_{PM} = 0.04$ can be safely used, in the validation test that follow, a $DX_{PM} = 0.02$ was chosen as a conservative choice since a lot of different test cases will be investigated in the following sections.



(a) Cl convergence history $Ma = 0.755, \alpha = 4^o$

(b) Cp $Ma = 0.755, \alpha = 4^o$

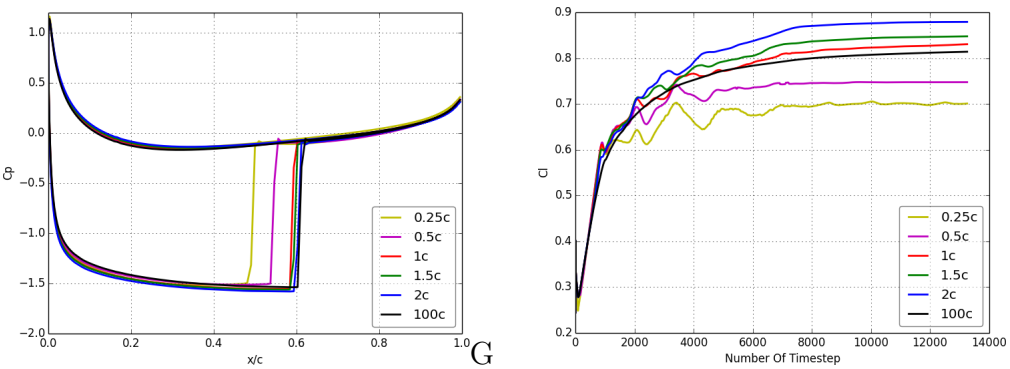
Figure 4.33: Comparison of the effect of PM grid resolution

Table 4.9: Cl relative error between various PM resolutions

DX_{PM}	Cl	(ΔCl)
0.01	8.50E-01	0.0 %
0.02	8.48E-01	-0.273 %
0.04	8.55E-01	0.583 %
0.08	8.61E-01	1.218 %
0.16	9.00E-01	5.905 %

Extent of the CFD grid

Another important parameter of the hybrid method proposed, is the extent of the CFD domain. This parameter has a crucial role, since the presence of solid boundaries in the Lagrangian solver is sensed through the CFD domain. Especially in cases with strong shocks and/or flow separation it is critical to know how the location of the far-field boundary affects the solution. In this connection, five different CFD grids are considered and results are compared to standard CFD simulations on a grid extending to 100 chord lengths. The extent of the CFD grid of the Hybrid solver was set equal to 2, 1.5, 1, 0.5 and 0.25 chord lengths away from the airfoil. All five grids were exact subsets of the grid used in the standard CFD simulations. The scope of this analysis is twofold. On one hand, validation and verification of the hybrid method is carried out and on the other the effect of the extent of the CFD domain on the results is investigated.



(a) Pressure distribution comparison, $Ma = 0.755, \alpha = 4^o$

(b) Cl convergence history, $Ma = 0.755, \alpha = 4^o$

Figure 4.34: Effect of different CFD grid extents

The comparison shown in Fig. 4.34 suggests that for extents greater than 1 chord length the Hybrid results almost collapse to those obtained with the standard CFD. If a smaller extent is used, the location of the shock is not well predicted and by that lower Cl values are obtained. In Table 4.10 the relative error with respect to the 100c case is given. It ranges from -13.8% to 8.00%. Even though the maximum error seems substantial, the pressure on the airfoil remains the same, except of the shock location. In fact, lift is very sensitive to even a small variance in the position of the shock wave or to minor differences in the pressure level over the suction side. Small variations can appear because of numerical errors or incompatibilities in the computational set-up. In particular, the apparent strange increase of the relative error when the CFD grid extends to 2 chord lengths is explained as follows. In the present verification, the CFD grid remains a subset of the 100c grid. Because of that, as the grid increases in extent, the grid cells will become larger and therefore the rule of having sufficient particle coverage is gradually no more satisfied. For the hybrid solver there is no real need to coarsen the grid, which would usually be approximately uniform. But if for any reason this is done, then the number of particles per CFD cell should be modified accordingly. Since this was not a necessity, the number of particles per cell was kept uniform in conformity to the fact that the PM grid is by default uniform.

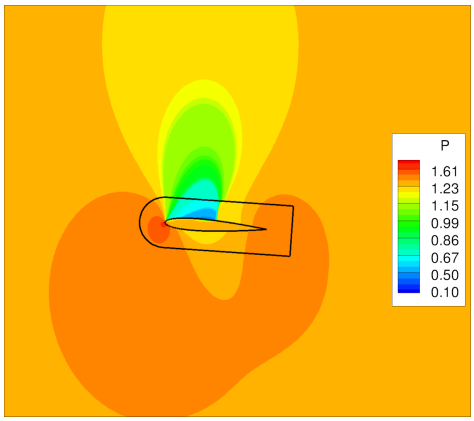
Good coverage of the PM with particles is a strict, so to say, numerical matter. Differences in Cl are also related to the diffusion characteristics of the numerically produced flow. For an inviscid flow, as the one considered here, only numerical diffusion is involved. However in viscous flows, it is combined with physical diffusion and by that affect the flow in an additive way. Discussion on this matter is given in the next subsections in relation to the physical problem that is examined each time.

It is worth noticing that in spite of the erroneous results obtained with grids of small extent, the solution across the outer boundary of the CFD grid is maintained.

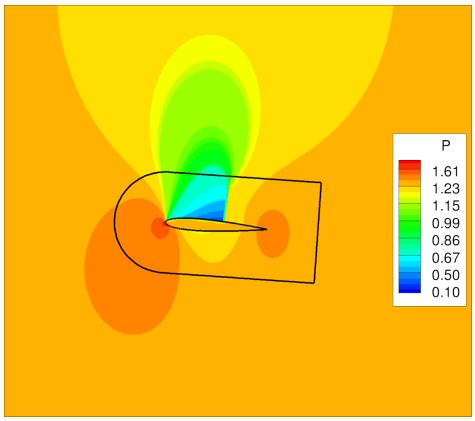
Table 4.10: Cl Comparison for various CFD grid sizes

Grid size	Grid Cells	Cl	ΔCl
0.25c	7968	7.00E-01	-1.38E+01%
0.5c	13563	7.47E-01	-8.16E+00 %
1.0c	18942	8.30E-01	2.01E+00 %
1.5c	24560	8.48E-01	4.12E+00 %
2.0c	32385	8.79E-01	8.00E+00 %
100c	42525	8.14E-01	0.00E+00%

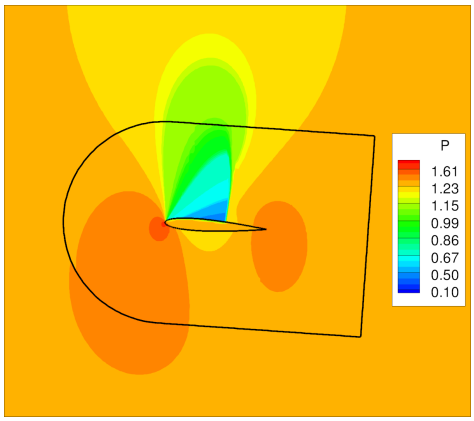
In Fig. 4.35 pressure contours for all six grids are displayed. The fact that the solution remains continuous is a proof that the coupling between the two solvers of the hybrid method is indeed strong. Furthermore, the comparison shows that the 1c grid (Fig. 4.35c) is sufficient to qualitatively capture the shock wave.



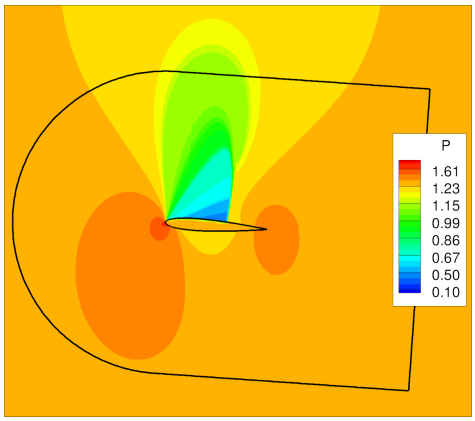
(a) 0.25 chords grid



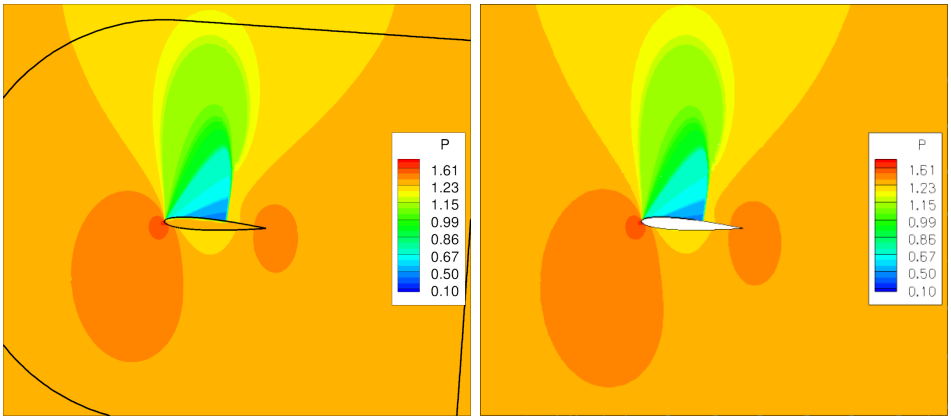
(b) 0.5 chords grid



(c) 1 chord grid



(d) 1.5 chords grid

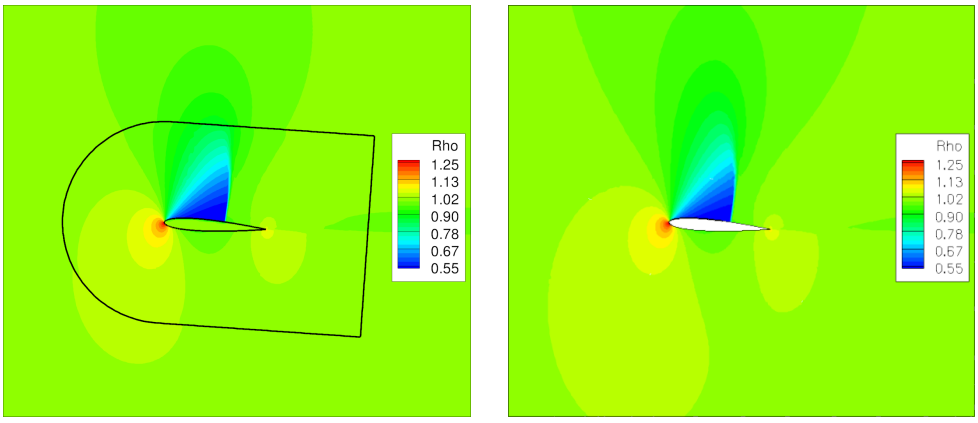


(e) 2 chords grid (f) 100 chords grid

Figure 4.35: Comparison of Pressure Contours for different CFD grid sizes (Ma=0.755, $\alpha = 4^o$)

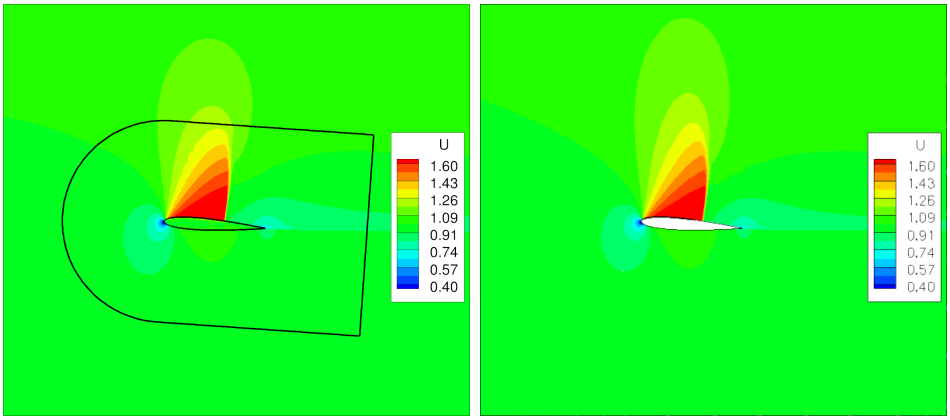
As previously discussed, dilatation (which affects pressure and density) is damped outside a circle of radius r_{max}. In all the 2D cases presented here, $r_{max} = 5, \sigma = 2$ (see Section 3.2.5) in order to prevent reflections on the boundary of the Inner PM grid located at $y = \pm 7.2$ (see Fig. 4.32).

In Fig. 4.36 and 4.37 density and velocity contours are compared. For both fields the Hybrid and standard CFD results are very close. As expected, there are minor differences especially in the purely Lagrangian region since the evolution equations are solved differently. However, the differences are considered insignificant.



(a) 1 chord grid (b) 100 chords grid

Figure 4.36: Comparison of Density Contours for two CFD grid sizes (Ma=0.755, $\alpha = 4^o$)

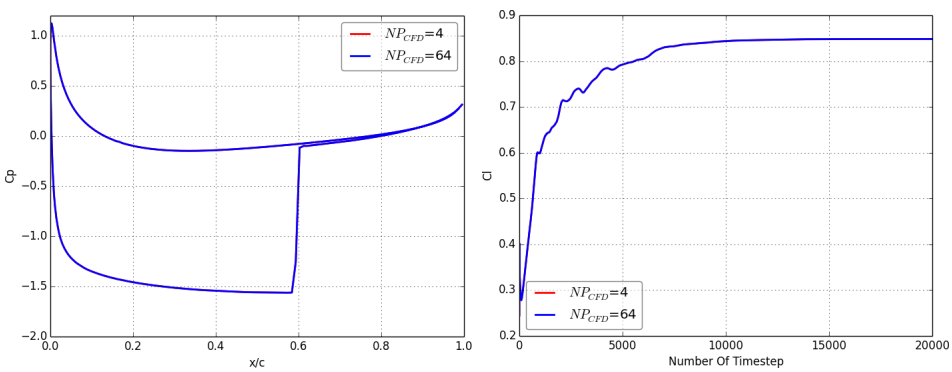


(a) 1 chords grid (b) 100 chords grid

Figure 4.37: Contour plots of the velocity component in the free-stream direction (NACA0012, Ma=0.755, $\alpha = 4^o$)

Density of particles in the CFD domain

The number of particles per CFD cell NP_{CFD} affects the coverage of the PM grid. In order to achieve the full coverage by particles of the volume occupied by the CFD particles should be at least comparable to the PM grid size. So it cannot be defined independently on the PM resolution. Out of the investigation done with respect to ΔX_{PM} it was found that NP_{CFD}=4 in 2D and NP_{CFD}=8 in 3D are adequate. By increasing the particles up to NP_{CFD}=64 in 2D, for the same PM grid, the results did not change.

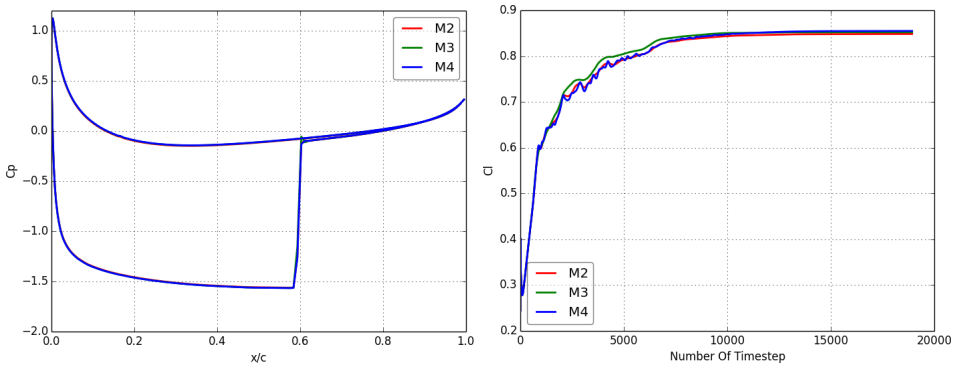


(a) Pressure coefficient distribution (b) Lift convergence history

Figure 4.38: Effect of the Number of CFD particles (NP_{CFD}) generated(NACA0012, $Ma = 0.755, \alpha = 4^o$)

Choice of the Projection Function

The $M2, M3$ and $M4$ projection function as described in section 3.2.4 are considered. As shown in Fig. 4.39 the choice of the projection function does not affect the loading on the airfoil. Since there is no significant reduction in computational cost by using a lower order function, $M4$ was finally adopted.



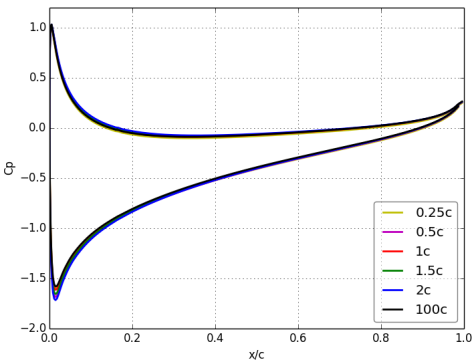
(a) Pressure Coefficient(Cp) (b) Lift Convergence History

Figure 4.39: Effect of the Projection Function (NACA0012, $Ma = 0.755, \alpha = 4^o$)

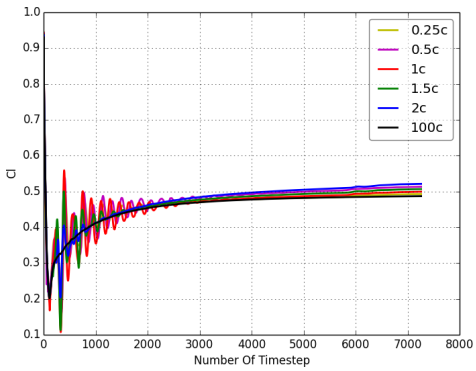
4.2.3 Inviscid flow around the NACA0012 airfoil in steady conditions

In this section the ability of HoPFlow to handle compressible flows at different Mach numbers is tested. The test cases were selected from [127] and include: the Ma =0.3 case without any shock, the Ma =0.6 case with a light shock wave and finally the Ma =0.755 where a strong shock is formed.

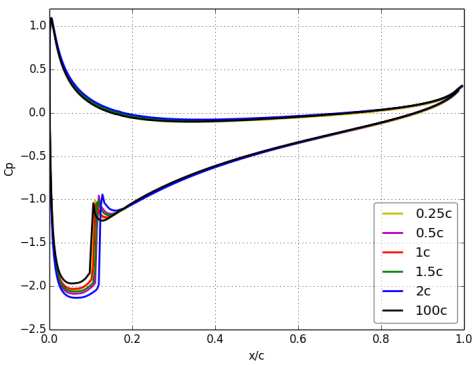
For the validation and verification part, the hybrid solver is compared to standard CFD, considered here as reference. The CFD grid extended to 100 chords. In the hybrid solver CFD grids of varying extent were used in order to examine how the Mach number affects different CFD grid size results. Even though the parametric analysis indicated that a grid size of extent in the order of 1c is sufficient, extents of 0.25 up to 2 chord lengths are also included as a supplement of the previous analysis.



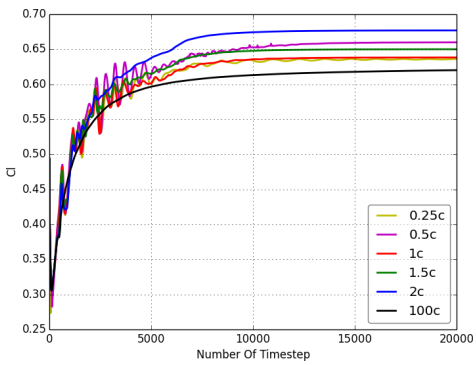
(a) Cp comparison, Ma=0.3, $\alpha = 4^o$



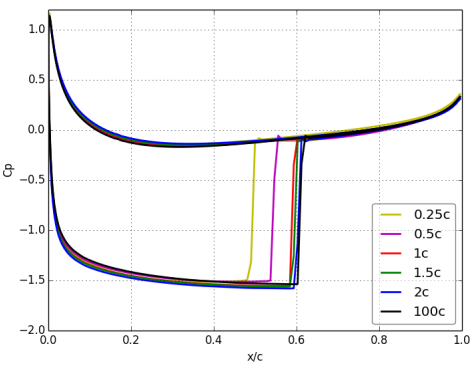
(b) Cl convergence history, Ma=0.3, $\alpha = 4^o$



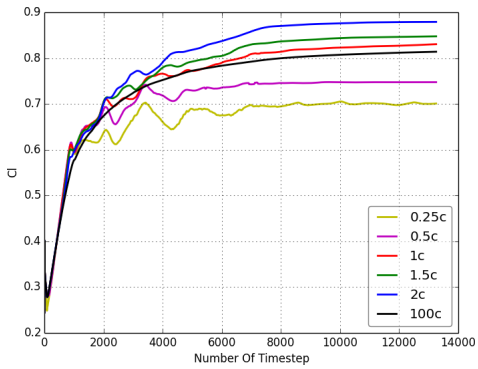
(c) Cp comparison, Ma=0.6, $\alpha = 4^o$



(d) Cl convergence history, Ma=0.6, $\alpha = 4^o$



(e) Cp comparison, Ma=0.755, $\alpha = 4^o$



(f) Cl convergence history, Ma=0.755, $\alpha = 4^o$

Figure 4.40: HoPFlow to MaPFlow comparison for the steady inviscid flow around a NACA 0012 airfoil at various Ma numbers

At Ma=0.3 all grids perform equally well as the comparisons in Fig. 4.40a and Fig. 4.40b indicate. With respect to the reference solution, the hybrid solver predicts slightly higher suction peak. In the Cl convergence histories the reference simulation proceeds smoothly until final convergence. On the contrary, in hybrid simulations significant oscillations appear that are gradually damped. These oscillations are connected to the evolution of the starting vortex which in the reference simulation fades out rapidly.

Moving on, at Ma=0.6 Fig. 4.40c and Fig. 4.40d lead to similar conclusions. On the pressure side the results are almost identical while on the suction side discrepancies appear in the position of the shock wave. At Ma=0.755 the differences in the position of the shock get even more pronounced, leading to different Cl values.

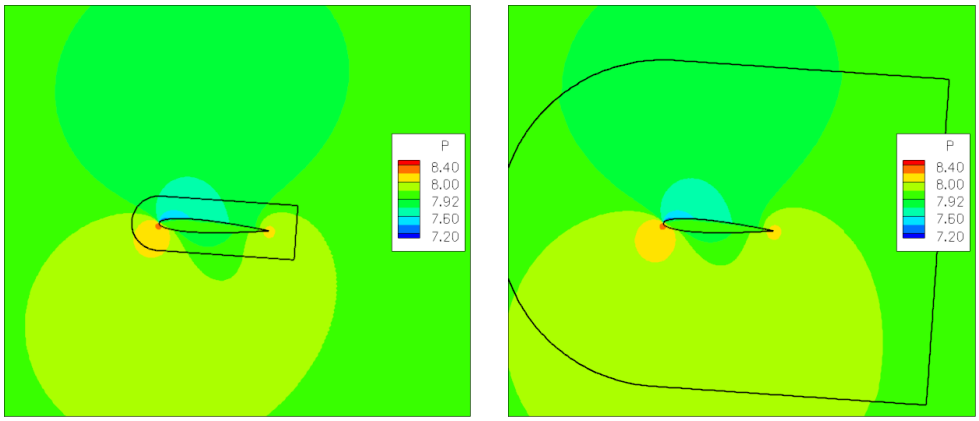
In this connection the Cl relative errors with respect to the reference solution(100c) are given in Table 4.11. For the three different Ma numbers (0.3, 0.6 and 0.755) the maximum difference is 7%, 9% and 8%, respectively. At the two lower Ma numbers the smallest hybrid CFD grid gives the best results in accordance to the previous discussion on how well the PM grid is covered with CFD particles and the connection to the diffusion of the starting vortex. Indeed, a very good agreement between the pressure contours obtained with the 0.25c and 1.5c hybrid grids at Ma=0.3 is observed in Fig. 4.41. On the contrary, at Ma=0.755 the smallest grid produces erroneous results, as the shock is suppressed by the small size of the CFD grid (see Fig.4.42). This indicates the need to include in the hybrid CFD grid any shock wave that could appear, in addition to the requirements set as regards the coverage of the PM grid with particles.

The greatest difference between the CFD and the Hybrid solver appears when the 2c grid is used. In fact, for Ma=0.3 and Ma=0.6 the hybrid solver consistently predicts higher Cl than CFD. From a physical stand point this can be also related

to the amount of diffusion added in the simulation. In theory particle methods have zero numerical diffusion while in practice it remains low. On the contrary standard 2nd order CFD introduces numerical diffusion which increases with the size of the grid cells. This means that for minimum diffusion, the CFD grid in the hybrid solver should have the minimum extent.

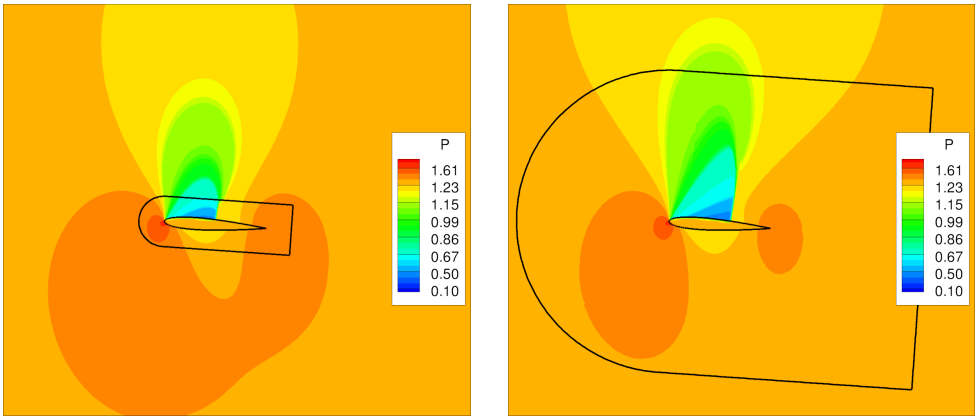
Table 4.11: Cl comparison for the various grids and Mach Numbers

	Ma=0.3		Ma=0.6		Ma=0.755	
Hybrid CFD grid size	Cl	ΔCl%	Cl	ΔCl%	Cl	ΔCl%
$0.25c$	4.92E-01	1.19E+00	6.35E-01	2.54E+00	7.00E-01	-1.38E+01
$0.5c$	5.13E-01	5.40E+00	6.59E-01	6.40E+00	7.47E-01	-8.16E+00
$1.0c$	4.99E-01	2.55E+00	6.38E-01	2.92E+00	8.30E-01	2.01E+00
$1.5c$	5.06E-01	4.05E+00	6.50E-01	4.82E+00	8.48E-01	4.12E+00
$2.0c$	5.21E-01	7.05E+00	6.76E-01	9.15E+00	8.79E-01	8.00E+00
$100c$	4.86E-01	0.00E+00	6.20E-01	0.00E+00	8.14E-01	0.00E+00



(a) Small grid 0.25 chords (b) Larger grid 1.5 chords

Figure 4.41: Pressure Contour Comparison at Ma=0.3, $\alpha = 4^o$



(a) Small grid 0.25 chords (b) Larger grid 1.5 chords

Figure 4.42: Pressure Contour Comparison at Ma=0.755, $\alpha = 4^o$

4.2.4 Inviscid Flow over a Pitching NACA0012 airfoil

Four cases of a pitching NACA0012 airfoil are considered. Three were obtained from [133] while the forth is a fast pitching case that was added in order to estimate the numerical diffusion. All cases concern oscillation around the quarter chord with the Angle of Attack (α) given by:

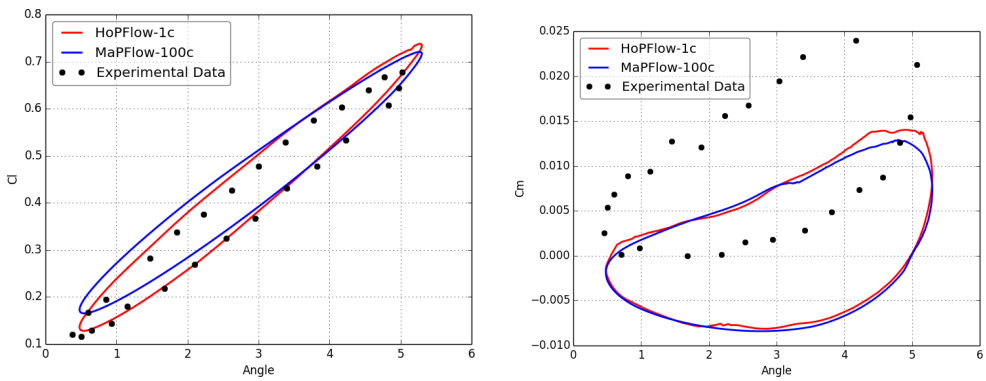
$$\alpha = \alpha_m + \alpha_s * sin(\omega t) \tag{4.1}$$

The flow conditions are given in Table 4.12. These cases have been included in order to check the behavior of the hybrid solver when the CFD grid is moving with respect to the PM grid. Additionally, comparison is made with experimental data from [133].

Table 4.12: NACA0012 Unsteady Flow conditions

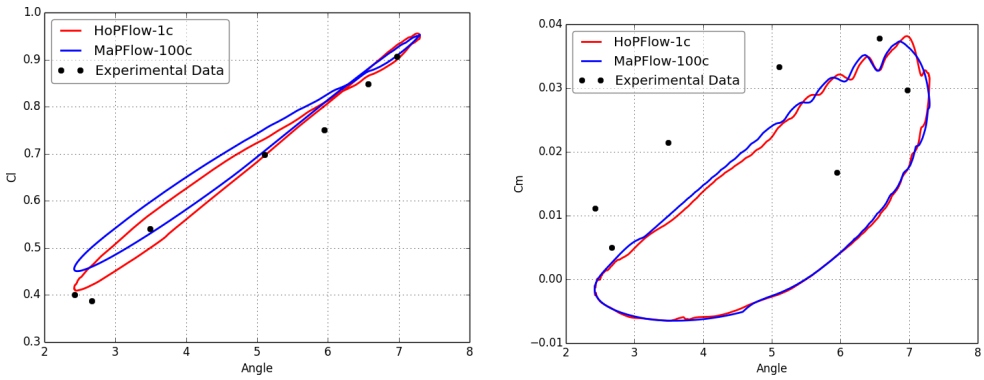
Case	Mach	α_m (deg)	α_s (deg)	ω
AGARD CT1	0.6	2.89^o	2.41^o	0.16
AGARD CT3	0.6	4.86^o	2.44^o	0.16
AGARD CT5	0.755	0.016^o	2.51^o	0.16
Fast Pitching	0.6	0°	2°	8

Hybrid results are again compared to those obtained with standard CFD for a grid extending to 100 chord lengths. The CFD grid for the hybrid solver was taken to extend 1 chord length as suggested by the parametric analysis.



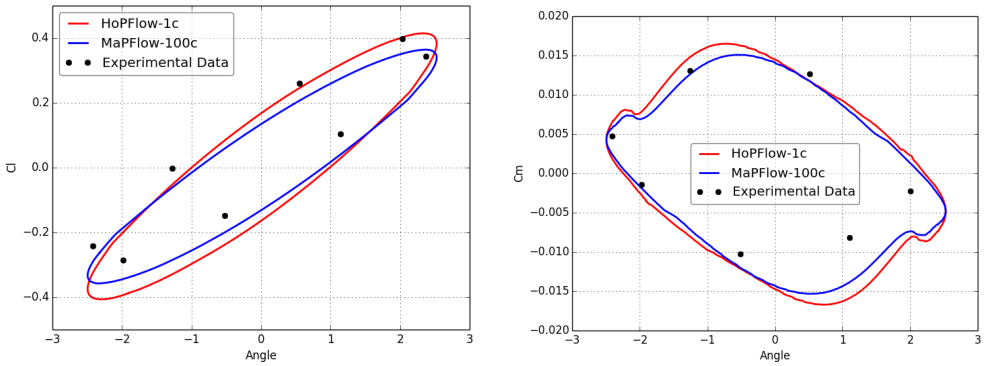
(a) Cl vs α (b) Cm vs α

Figure 4.43: Cl and Cm variation with Angle of Attack. Inviscid unsteady flow over a NACA0012 airfoil, AGARD CT1



(a) Cl vs α (b) Cm vs α

Figure 4.44: Cl and Cm variation with Angle of Attack. Inviscid unsteady flow over a NACA0012 airfoil, AGARD CT3

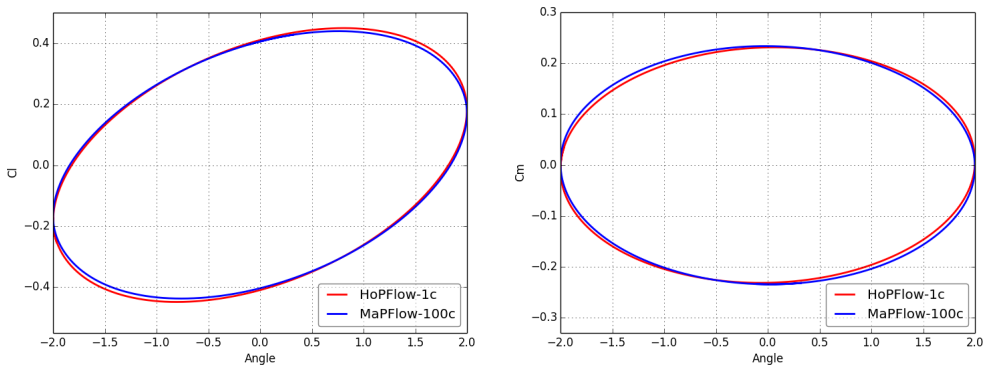


(a) Cl vs α (b) Cm vs α

Figure 4.45: Cl and Cm variation with Angle of Attack. Inviscid unsteady flow over a NACA0012 airfoil, AGARD CT5

By comparing the Cl and Cm loops in Fig. 4.43, 4.44 and 4.45 good agreement is found. Actually, in all hybrid results, the Cl loop has higher slope compared to the standard CFD predictions. This indicates lower diffusion in the hybrid solver. In order to examine further this point, a fast pitching case is considered.

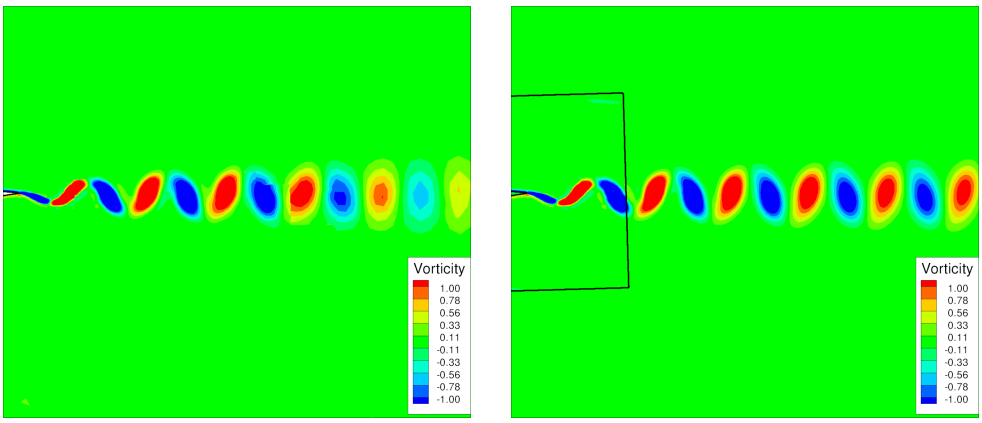
In Fig. 4.46 the Cl and Cm loops obtained with the two solvers are shown. The agreement is very good. However, the development of the wake is different (Fig. 4.47). In fact, the hybrid solver retains the shape and intensity of the shed vortices over a much longer distance as compared to standard CFD. This indicates the superiority of the hybrid solver in a case that in principle should have no diffusion. Finally, the hybrid results are in better agreement with the experimental data than the standard CFD results. The disagreement of both solvers with the experimental Cm is due to the inviscid assumption made in the simulations.



(a) Cl vs α (b) Cm vs α

Figure 4.46: Cl and Cm variation with Angle of Attack. Inviscid unsteady flow over a fast pitching NACA0012 airfoil

d



(a) MaPFlow Wake (b) HoPFlow Wake

Figure 4.47: Vorticity Contours of the wake behind the fast pitching NACA0012 airfoil

4.2.5 Viscous flow around a NACA0012 Airfoil

The next case concerns the flow around a NACA0012 airfoil. Hybrid predictions are compared to those obtained with standard CFD. The tests conditions are given in Table 4.1.2 and were taken from [114]. The grid used by the standard CFD extended to 100 chord lengths and amounted approximately 10^5 cells with a near wall resolution of y+ ≈ 1. The CFD grid for the Hybrid solver extended to 1.5 chord lengths, contained $0.6 \cdot 10^5$ cells and was an exact abstract of that used by the standard CFD. Note that the grid was extended to 1.5c so as to have within the CFD grid the separation bubble at the highest angle considered. This is in agreement with the remark made concerning shock waves. Also note that because the Hybrid solver can only perform time true simulations, the flow is considered unsteady even though it was actually steady in most of the tests.

Table 4.13: NACA0012 Viscous Flow Conditions

Mach	Reynolds	Angle of Attack	Δt
0.15	$6 \cdot 10^6$	$0^o, 10^o, 15^o, 20^o$	0.005

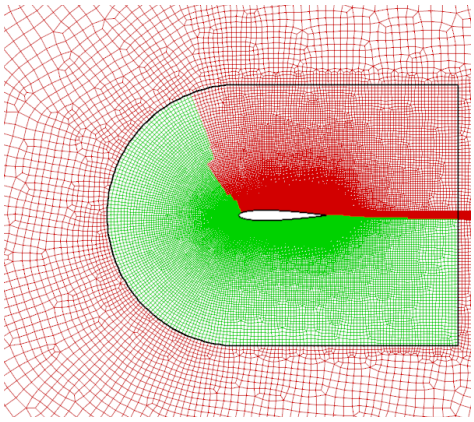


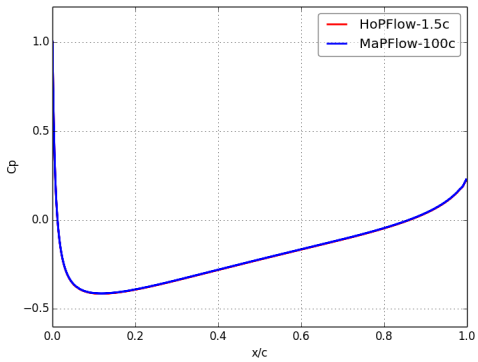
Figure 4.48: Detail of NACA0012 Viscous Computational grid.The continuous black line shows the boundary of the CFD grid used in the hybrid solver.

In Table 4.14 the relative lift and drag errors between the standard CFD and the Hybrid solvers are given. Results are provided for both the SA and the $k-\omega$ SST turbulence model.

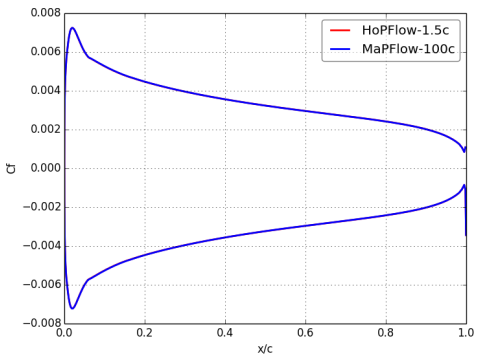
Table 4.14: Cl,Cd relative error of the Hybrid Results with respect to the pure CFD approach

	0°	10°	15°	20°
Cl Hybrid (SA)	8.11E-05	1.09E+00	1.49E+00	8.28E-01
Cl CFD (SA)	-1.56E-04	1.07E+00	1.48E+00	8.36E-01
Δ Cl (SA)	**-1.51E+02%**	**1.9%**	**0.89 %**	**-1.00 %**
Cd Hybrid (SA)	8.51E-03	1.14E-02	2.27E-02	2.29E-01
Cd CFD (SA)	8.51E-03	1.54E-02	2.98E-02	2.37E-01
Δ Cd (SA)	**0.023** %	**-26.27%**	**-23.10%**	**-3.27 %**
Cl Hybrid (SST)	-8.06E-05	1.08E+00	1.49E+00	1.19E+00
Cl CFD (SST)	-3.072E-04	1.07E+00	1.48E+00	1.16E+00
Δ Cl (SST)	**-7.37E+01%**	**1.74%**	**0.82 %**	**2.25 %**
Cd Hybrid (SST)	7.81E-03	9.56E-03	1.85E-02	4.42E-01
Cd CFD (SST)	7.81E-03	1.33E-02	2.48E-02	4.29E-01
Δ Cd (SST)	**-0.067 %**	**-28.59%**	**-25.28%**	**2.93 %**

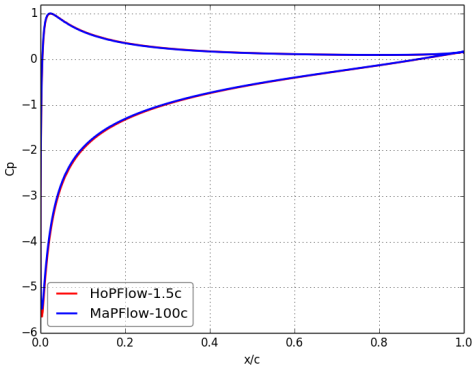
As already discussed, the Hybrid solver is less diffusive and thus results in lower drag and higher lift. The relative error in Cl is less than 2.25% while that in Cd is 28%. At 0° the large % errors in lift are due to the almost zero value of Cl. Although the relative error in Cd appears high, the agreement in Cp and Cf is very good as shown in Fig. 4.49 for the Spalart-Allmaras and in Fig. 4.50 for the $k-\omega$ SST modeling.



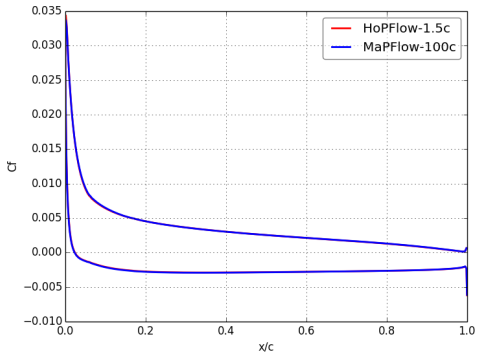
(a) Pressure Coefficient at 0 degrees



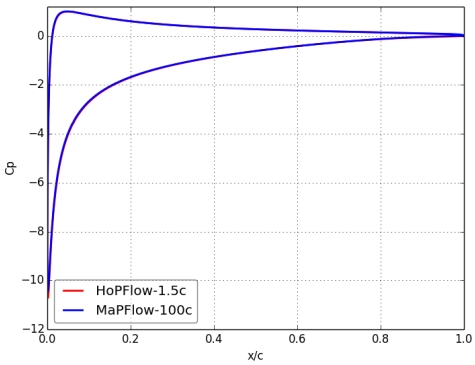
(b) Friction Coefficient at 0 degrees



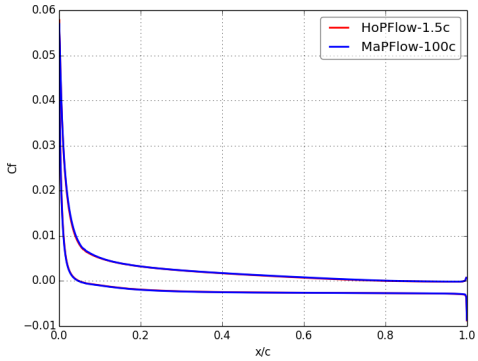
(c) Pressure Coefficient at 10 degrees



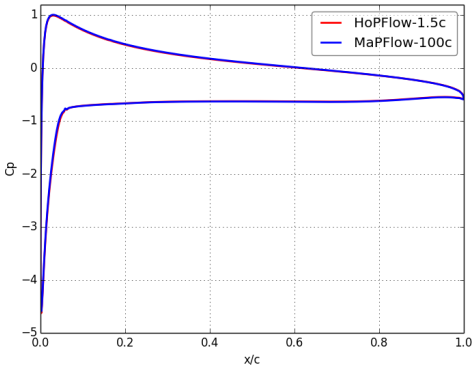
(d) Friction Coefficient at 10 degrees



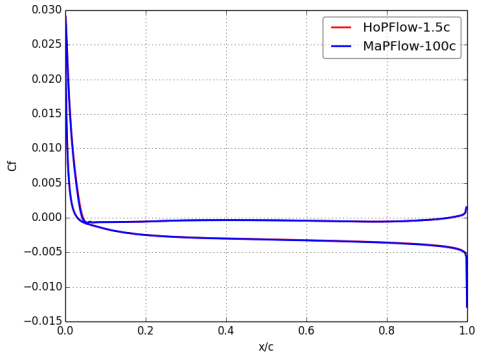
(e) Pressure Coefficient at 15 degrees



(f) Friction Coefficient at 15 degrees

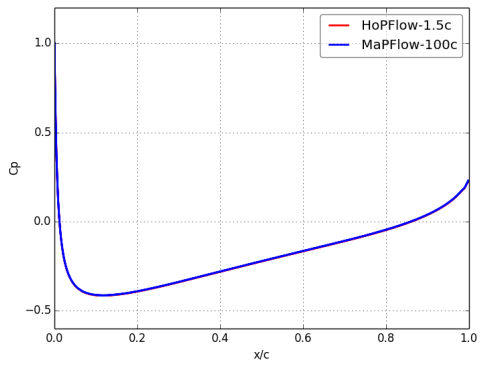


(g) Pressure Coefficient at 20 degrees

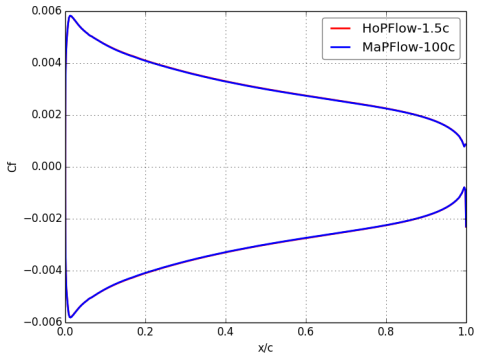


(h) Friction Coefficient at 20 degrees

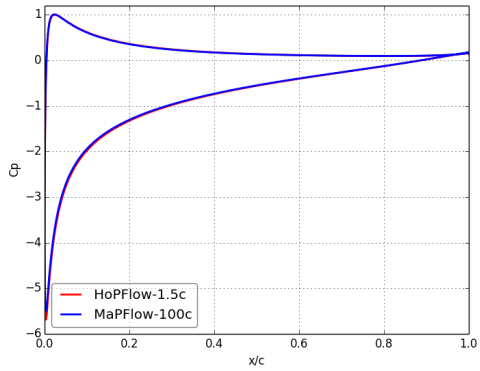
Figure 4.49: Pressure(left) and friction(right) coefficient variation on a NACA0012 airfoil, at $0^o, 10^o, 15^o$and20^o using SA model.



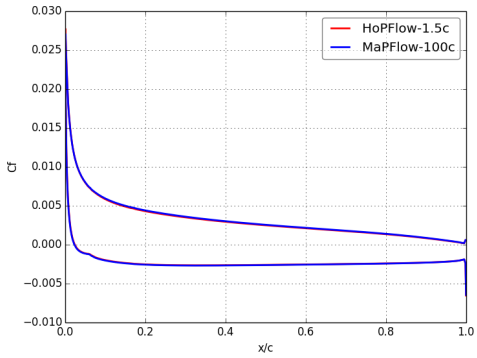
(a) Pressure Coefficient at 0 degrees



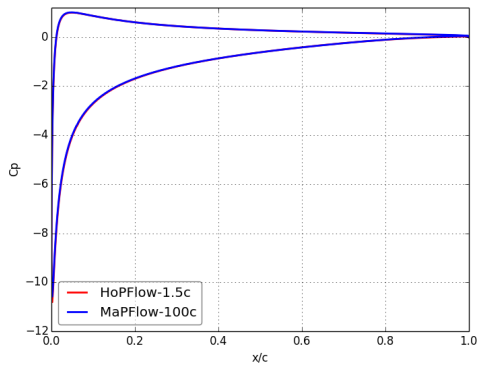
(b) Friction Coefficient at 0 degrees



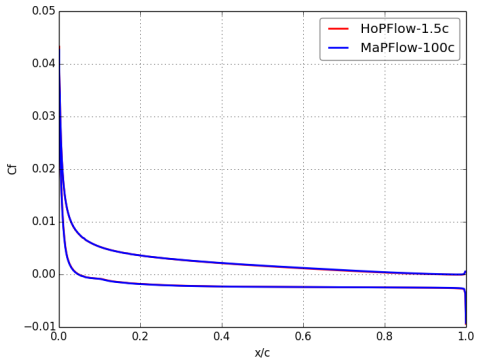
(c) Pressure Coefficient at 10 degrees



(d) Friction Coefficient at 10 degrees



(e) Pressure Coefficient at 15 degrees

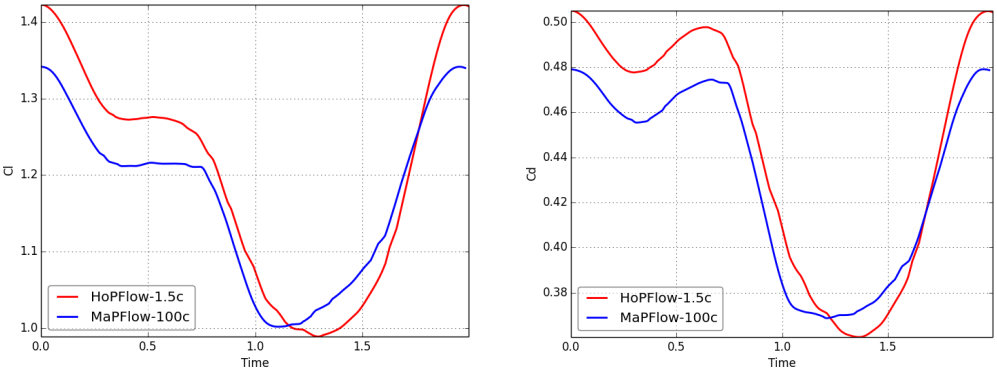


(f) Friction Coefficient at 15 degrees

Figure 4.50: Pressure(left) and friction(right) coefficient variation on a NACA0012 airfoil, at $0^o, 10^o, 15^o$and20^o using $k-\omega$ SST model.

Both turbulence models predict flow separation at $\alpha = 20^o$. However, in the k-ω SST results the point of separation is further upstream. Because of that the flow

becomes unsteady. The lift and drag variations over one period are displayed in Fig. 4.51. The shape of the curves and the frequencies are similar in both signals while in terms of range in the hybrid results is higher. This is also attributed to lower diffusion. In physical terms the range of the loading fluctuations are connected to the development of the wake which is differently predicted by the two solvers. This point is further discussed in the next two test cases.



(a) Lift Coefficient over a Period (b) Drag Coefficient over a Period

Figure 4.51: SST model $\alpha = 20^o$

The objective of the comparisons done for the present test case was to evaluate the performance of the hybrid solver in up to light stall flow conditions. The Hybrid method is found in good agreement with standard CFD. The same case was also presented in section 4.1.2 but there the CFD grid amounted double number of cells ($2 \cdot 10^5$) and extended over a distance four times longer (400 chord lengths). In fact, the hybrid results, using the 1.5c grid, are closer to the higher resolution CFD results than those given here.

4.2.6 Aerodynamic analysis of the FB-3750-1750 airfoil

The next case concerns the flow over a flat-back FB-3750-1750 airfoil. This is a thick airfoil with blunt trailing edge and is primarily designed for use over the inner part of wind turbine rotors. Flat-back airfoils have been investigated both computationally and experimentally in [134, 135, 136, 137].

Similarly to these works, a large grid extending to 50 chord lengths away from the airfoil was used for the standard CFD simulations with a wall resolution giving y+$\approx$ 1. For the Hybrid simulations an exact subset of this grid extending up to $\approx$ 1.5 chord lengths was used. In all simulations the SA turbulence model was used. The flow conditions of the tests are given in Table 4.15.

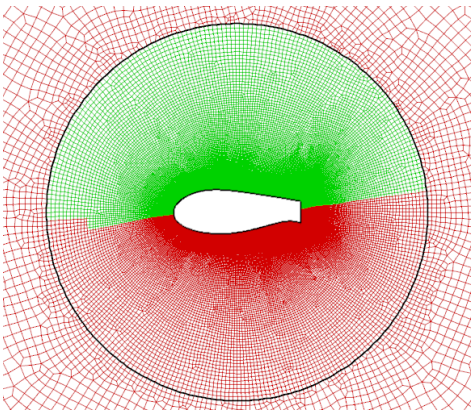


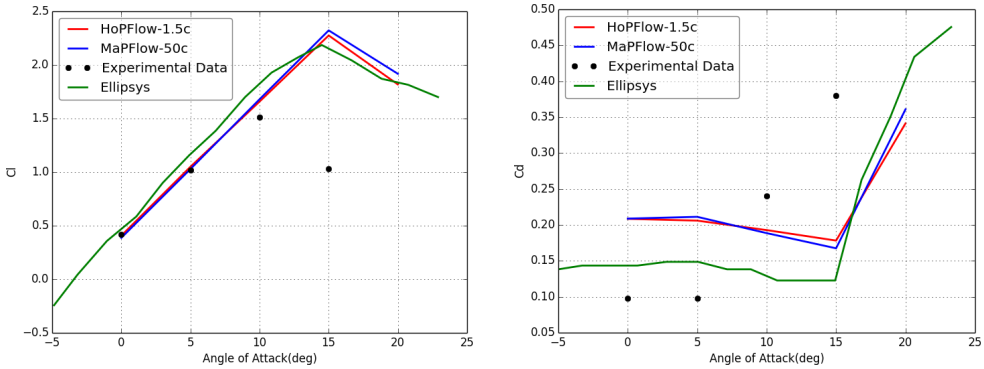
Figure 4.52: FB-3750-1750: The Computational Grid

The specific tests were selected in order to evaluate the performance of the Hybrid solver in massively separated flows. Separation on a flat-back airfoil is triggered because of the blunt trailing edge even at low Angles of Attack. A Von-Karman street like wake is generated which excite fluctuations of the flow that affect the loading on the airfoil.

Table 4.15: FB-3750-1750 flow conditions

Mach	Reynolds	Angle of Attack	Δt
0.15	$0.6 \cdot 10^6$	$0^o, 5^o, 10^o, 15^o, 20^o$	0.002

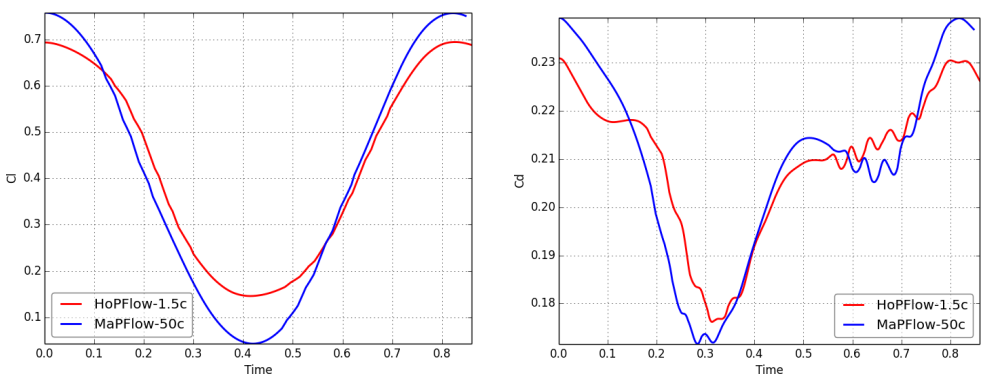
In Fig. 4.53 the predicted and measured mean lift and drag Coefficients are displayed. The experimental data were taken from [134]. Even though all solvers predict stall in delay as compared to tunnel data, they are in good agreement between them.



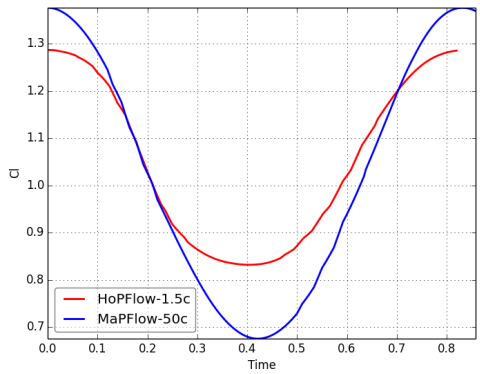
(a) Lift polar (b) Drag polar

Figure 4.53: Mean Cl, Cd polars for the FB-3750-1750 fat back airfoil at $Re = 0.6\,10^6$ using MaPFlow, HoPFlow and Ellipsys taken from [137], are compared to experimental data from [134]

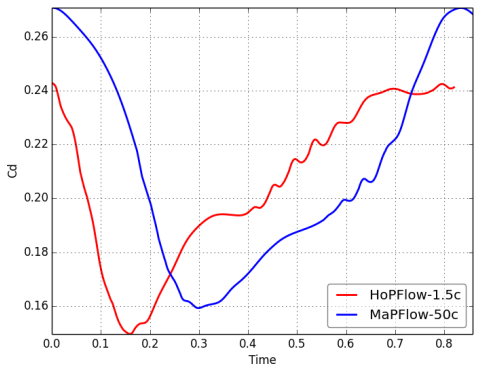
In Fig. 4.54 Cl and Cd variations over one period are presented. Both solvers predict the same basic frequency as shown in Fig. 4.55. Otherwise there are substantial differences. The Cl signals are similar in shape but not in range which are consistently smaller in the hybrid results. On the contrary, the differences in the Cd signals are definitely more pronounced. At 0° and 5°, the shape of the Cd variation is similar but substantially different at higher angles. Furthermore in the Hybrid results the second harmonic is excited in all angles while in the standard CFD results, it only appears at 20°.



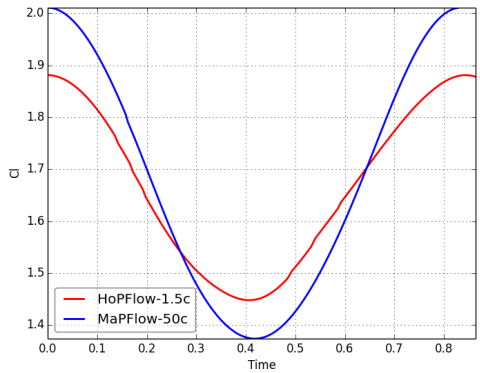
(a) Cl vs Time at 0° (b) Cd vs Time at 0°



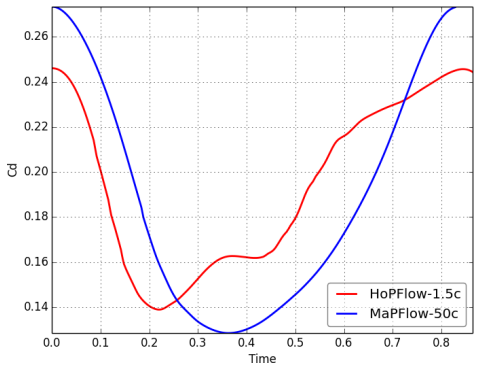
(c) Cl vs Time at 5°



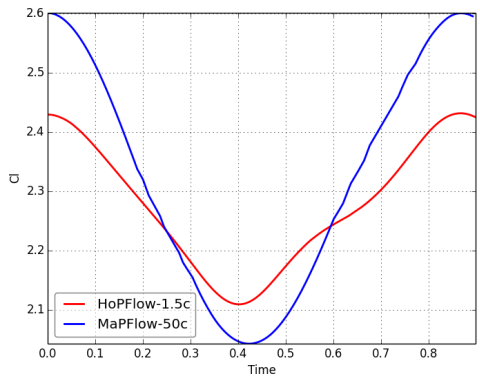
(d) Cd vs Time at 5°



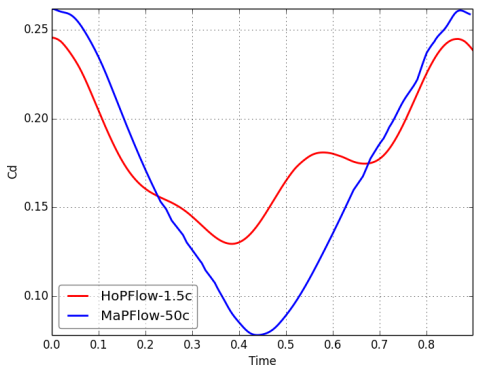
(e) Cl vs Time at 10°



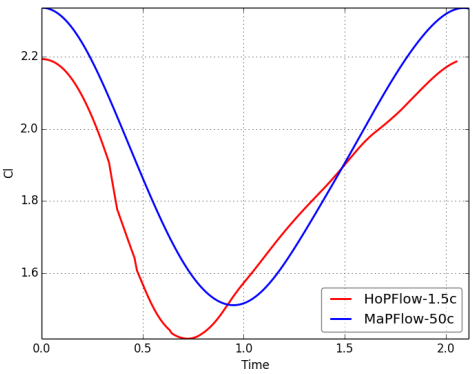
(f) Cd vs Time at 10°



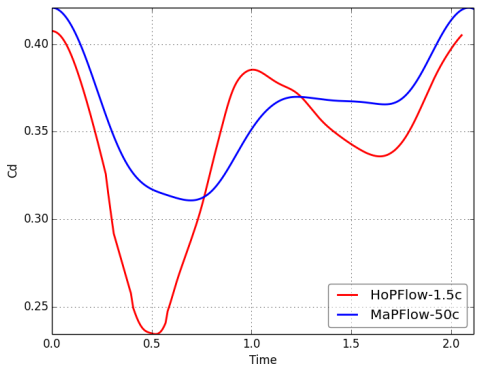
(g) Cl vs Time at 15°



(h) Cd vs Time at 15°



(i) Cl vs Time at 20°



(j) Cd vs Time at 20°

Figure 4.54: Cl (left) and Cd (right) signals over a period at $0^o, 5^o, 10^o, 15^o$ and 20°

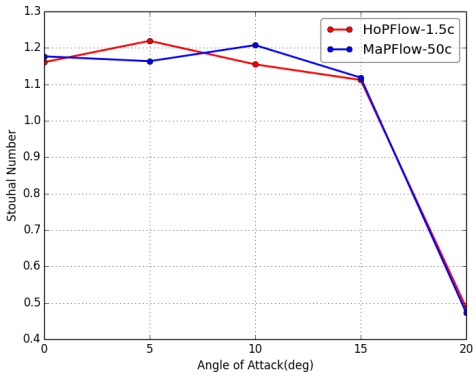
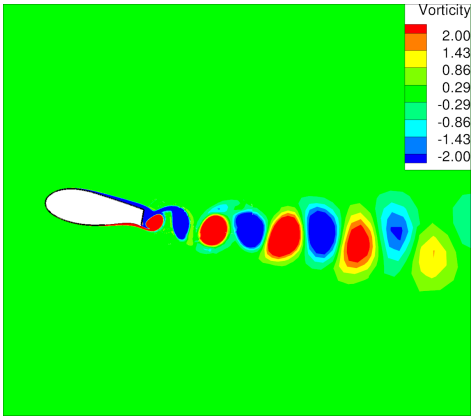
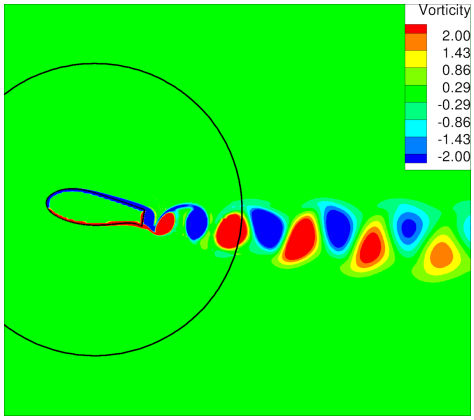


Figure 4.55: Strouhal Number ($Str = f \cdot U_\infty$)

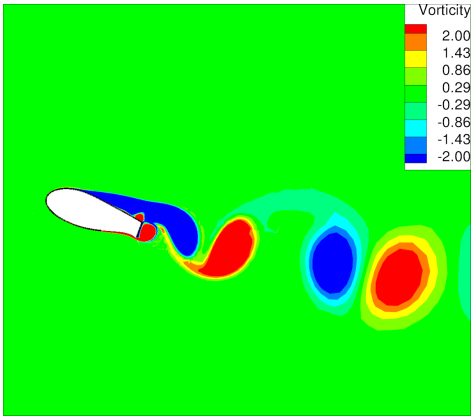
The difference in the Cl and Cd signals is attributed to the different wake dynamics the two solvers produce. In Fig. 4.56 vorticity contours are displayed as provided by the two solvers. It is clear that the wake dynamics is different. The Hybrid solver generates stronger and more slender vortex structures while keeping their connections. Due to numerical diffusion in the standard CFD results the vortices are less intense and have approximately circular shape. It is expected that differences of this kind appearing within a couple of chords downstream of the trailing edge will affect the aerodynamic loading. Nevertheless, there is qualitative consistency between the two solvers in terms of mean Cl and Cd as well as Strouhal number.



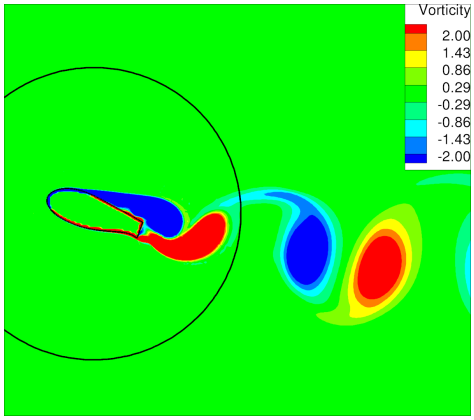
(a) Wake development at $\alpha = 10^o$ as obtained with standard CFD - MaPFlow



(b) Wake development at $\alpha = 10^o$ as obtained with the Hybrid solver - HoPFlow



(c) Wake development at $\alpha = 20^o$ as obtained with standard CFD - MaPFlow



(d) Wake development at $\alpha = 20^o$ as obtained with the Hybrid solver - HoPFlow

Figure 4.56: Wake development behind the FB-3750-1750 airfoil at 10° and 20° of incidence. Different dynamics in the wake leads to different loading variations on the airfoil.

Concluding the discussion, it is clear that this is a challenging test for CFD in general. The quality of the agreement obtained here is similar as compared to other CFD results (see references at the beginning of the section). Unfortunately, during the wind tunnel tests neither the loading dynamics nor the structure of the wake were recorded. This type of information could eventually help to better understand the origin of the deviations with respect to measurements.

4.2.7 Aerodynamic analysis of the FFA-W3-241 Airfoil

The last 2D case concerns the viscous flow around the FFA-W3-241 airfoil at post stall Angles of Attack ($\alpha > 20^o$). A Reynolds number of $12 \cdot 10^6$ is chosen which considerably exceeds the usual range. This choice was made in connection to current trends in wind turbine blade design. As in the previous section, the flow is expected to be unsteady but here the triggering mechanism is the pronounced adverse pressure gradient over the suction side of the airfoil.

The standard CFD used a grid of approximately $0.9 * 10^5$ cells that extended to 70 chord lengths. Out of this, a grid of $0.6 * 10^5$ cells was extracted and used in the hybrid solver. The grids were of low Reynolds type having a y+$\approx$ 1 and simulations were based on the $k - \omega$ SST turbulence model. The flow conditions are summarized in Table 4.16.

Table 4.16: FFA-W3-241 flow conditions

Mach	Reynolds	Angle of Attack
0.26	$12 \cdot 10^6$	$20^o, 24^o, 28^o, 32^o$

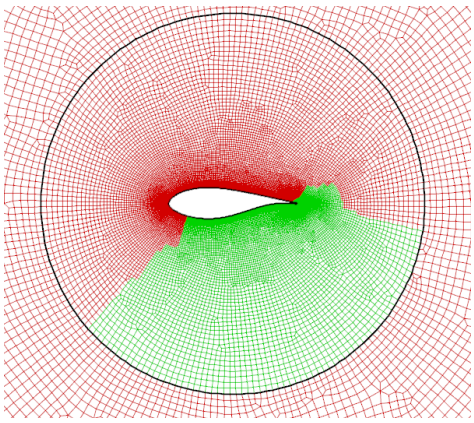
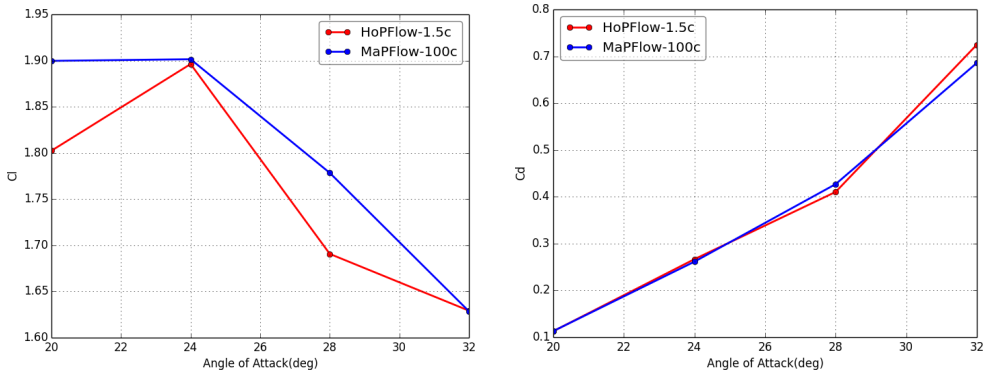


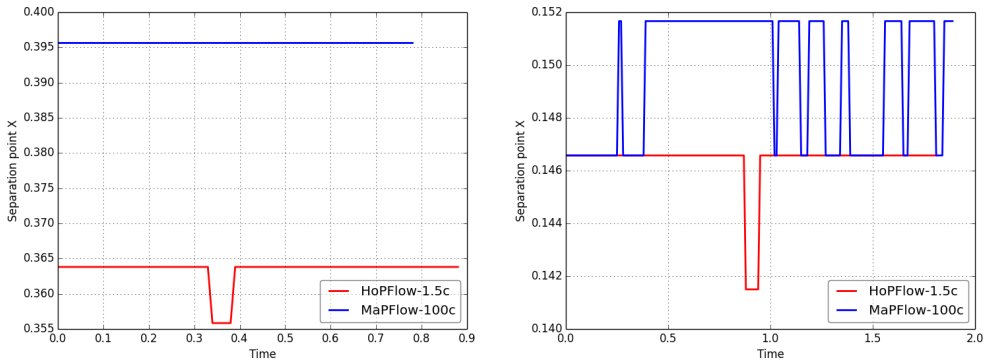
Figure 4.57: FFA-W3-241 Computational grid. The continuous black line shows the boundary of the CFD grid used in the Hybrid simulations.

With respect to mean lift and drag polars, significant differences in Cl were found at 20° and 28° (Fig. 4.58) when the drag predictions agree well. The discrepancies in Cl are due to disagreement in the point of separation. Even a small shift of the separation point (Fig. 4.59) can give deviations of the amount obtained. Conversely, the Drag Coefficient is primarily influenced by the pressure level in the separation bubble which is not so sensitive to the location of the separation.



(a) Cl vs angle of attack (b) Cd vs angle of attack

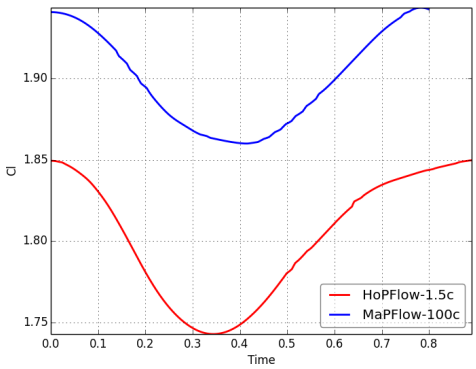
Figure 4.58: Mean Cl,Cd vs angle of attack



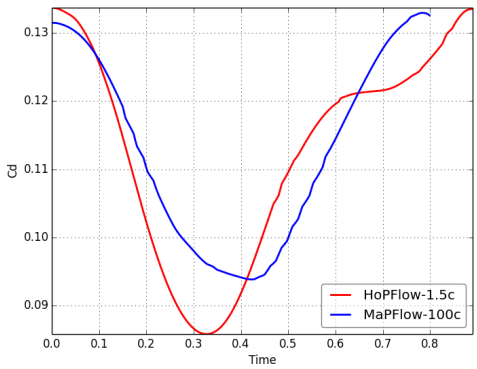
(a) Position of separation at $\alpha = 20^o$ (b) Position of separation at $\alpha = 28^o$

Figure 4.59: The variation of the location of separation over one period.

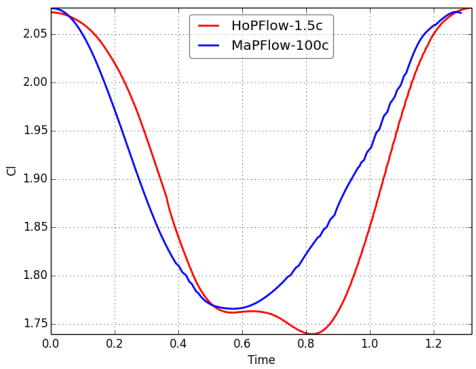
Next, in Fig. 4.60 the variation of Cl and Cd over a period is shown. Besides the level shift, similar shapes have been obtained by both solvers. Worth noticing is that as in the flat back case, a higher harmonic is also excited, indicative of lower diffusion.



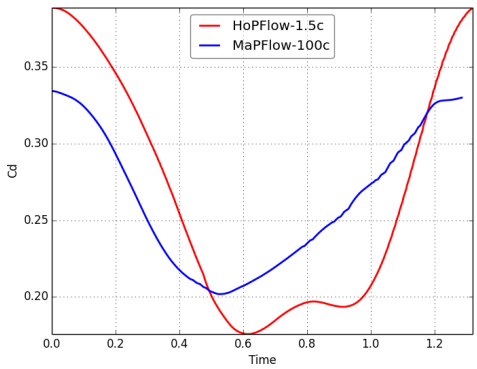
(a) Cl vs Time at 20°



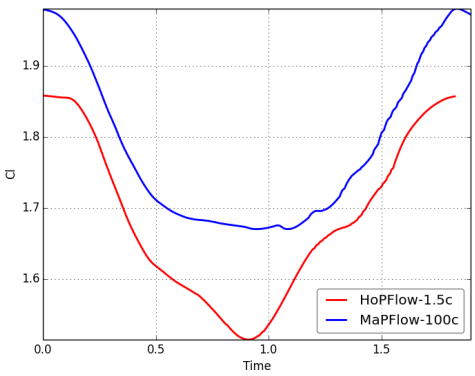
(b) Cd vs Time at 20°



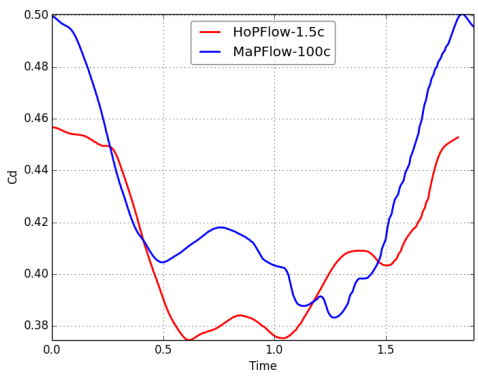
(c) Cl vs Time at 24°



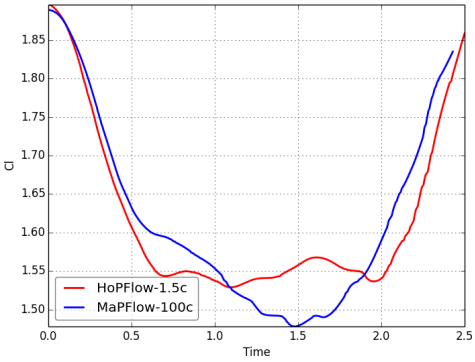
(d) Cd vs Time at 24°



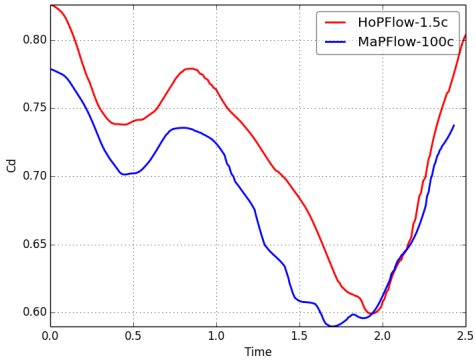
(e) l vs Time at 28°



(f) Cd vs Time at 28°



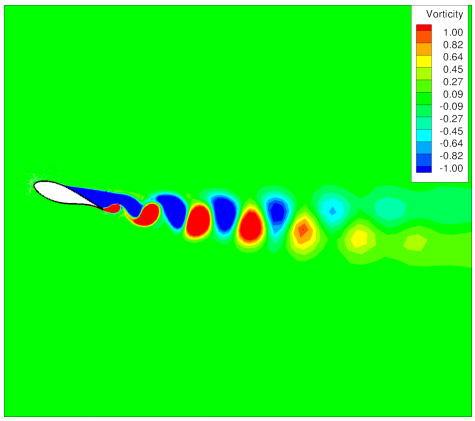
(g) Cl vs Time at 32°



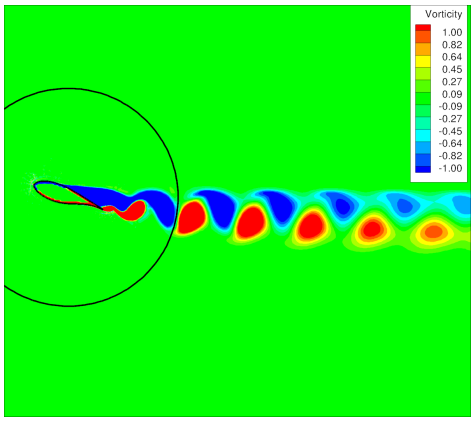
(h) Cd vs Time at 2°

Figure 4.60: Cl (left) and Cd (right) signal over a period at $20^o, 24^o, 28^o$ and 32°

The differences found in the lift and drag variations are again attributed to differences in the wake development. As shown in Fig. 4.61, standard CFD diffuses the wake vortices quite fast and consequently weakens the wake induced effect on the airfoil. In addition to increased diffusion, in the CFD results the vortex structures appear less sharp and less intense. This could explain why in the hybrid predictions the second harmonic is excited. However as regards the basic frequency of the flow, the two codes agree well (Fig. 4.62).



(a) CFD wake structure at 20°



(b) Hybrid wake structure at 20°

Figure 4.61: Vorticity contour plots as obtained by the two solvers.

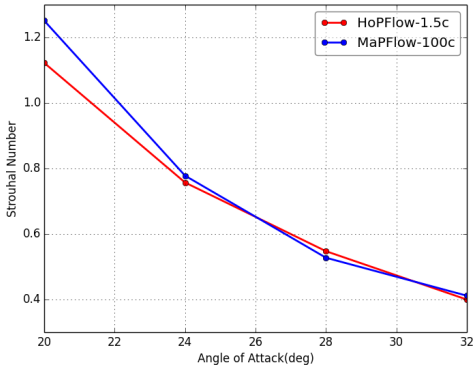
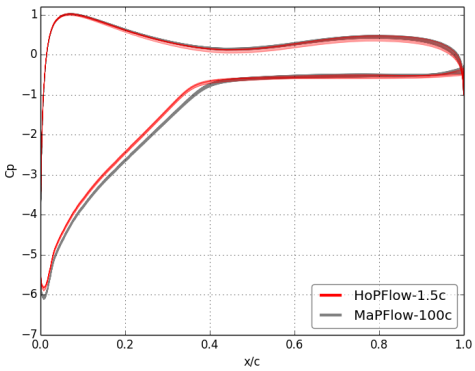
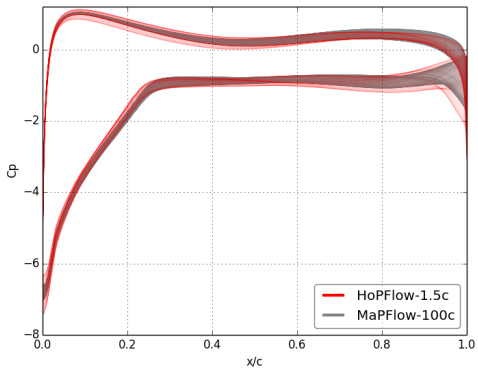


Figure 4.62: Strouhal Number vs Angle of Attack

The differences in the dynamic response of the airfoil can be also seen in the surface pressure histories. In Fig. 4.63 the Cp curves obtained over a period are plotted together. At $\alpha = 20^o$ (Fig. 4.63a) there is clear difference in the location of separation which explains the deviation in Cl. On the contrary at $\alpha = 24^o$ (Fig. 4.63b) the pressure time history significantly fluctuates around a more or less similar mean state. So in this case the mean Cl is similar but not the range of variation.



(a) Pressure Coefficient history at $\alpha = 20^o$



(b) Pressure Coefficient history at $\alpha = 24^o$

Figure 4.63: Pressure Coefficient history

Summarizing the last two cases, it can be concluded that in highly separated flows the hybrid solver behaves well and provides consistent results. The overall agreement between the hybrid and the standard CFD solver is good. Of course, highly separated flows are very sensitive to the structure of the wake that is formed. Small changes in the shape or strength of the wake vortices can alter the separation point which in turn could affect the predicted loads.

As a final remark it should be underlined that for highly separated flows dense grids are certainly needed. In this sense, the grid that was used in the standard

CFD solver is by no means adequate. This was done for two reasons: a) The specific analysis focuses on the performance assessment of the hybrid solver rather than on the rigorous analysis of the underlying technological problem (which can be found in [138]); b) The specific sectional grid size is typical in 3D flow simulations. Except for that, in highly separated flows like the ones presented here, eddy viscosity turbulence models can be inadequate and other methodologies such as the Detached Eddy Simulation (DES) or Large Eddy Simulation (LES) should be used [139].

4.2.8 The Caradonna-Tung Rotor in hover

This is a typical test case for validating helicopter simulations. The Caradonna-Tung rotor is a simple configuration extensively measured at NASA Ames [140]. The experimental setup is shown in Fig 4.64. The aim was to validate the hybrid code in a simple 3D case with moving (rotating) bodies. In this case $DX_{PM} = 0.1$ was used.

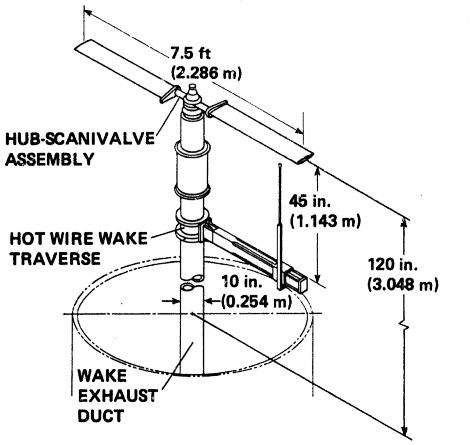


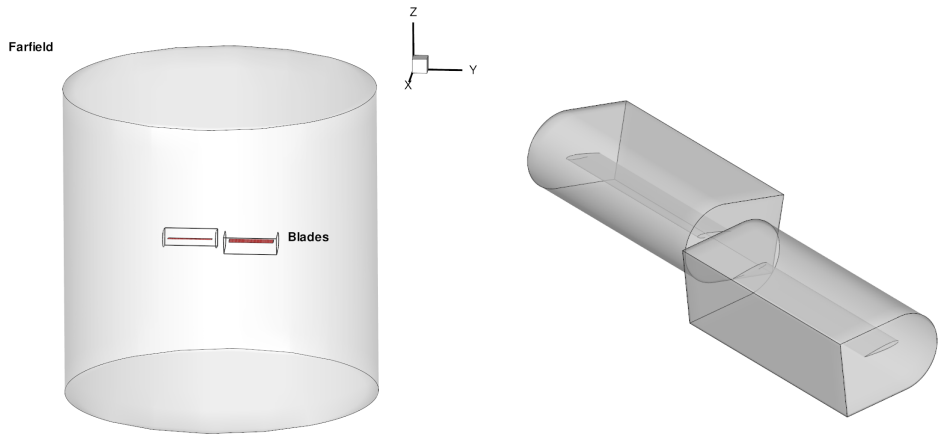
Figure 4.64: Experimental Setup

The blades are rectangular (constant chord) and untwisted. The NACA0012 profile is used throughout the entire blade that has an aspect ratio of $AR = 6$. The test case examined is an essentially compressible case at 8° pitch. The corresponding Mach Number at the tip of the blade is $Ma = 0.877$. Flow conditions are summarized in Table 4.17

Table 4.17: Flow conditions for the Caradonna-Tung rotor in hover

Mach Tip	Reynolds Number	Pitch	$\Omega(RPM)$	Δt
0.877	Inviscid	8°	2500	0.05 (720 steps per period)

The computational grid used for the standard CFD had 5 million cells and both blades were modeled. The reason to include both blades, was to allow a direct comparison with the hybrid method. The grid used by the hybrid solver had approximately 1.3 million cells. As in the 2D case the hybrid solver mesh is an exact subset of the larger computational domain. One particularity of the specific test case, is that the multi-body capabilities of the hybrid solver are presented. Two local and disconnected grids were used; one for each blade.



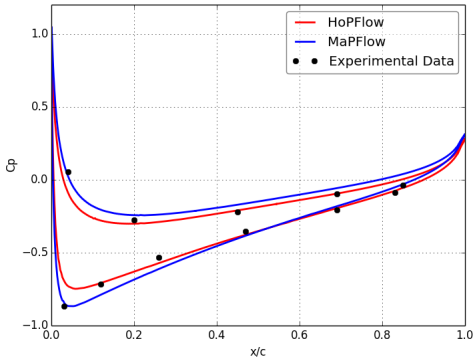
(a) The Computational Grid used by MaPFlow

(b) The Computational Grid used by HoPFlow

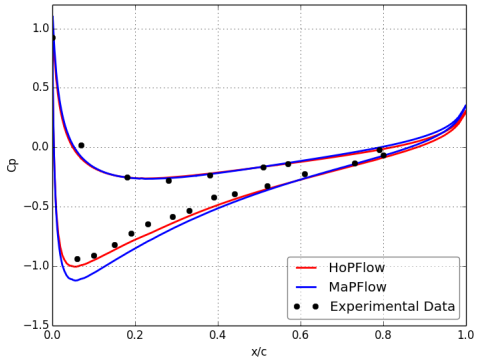
Figure 4.65: The Computational Grids

In Fig. 4.66 the pressure distributions at five spanwise locations are shown. Both codes provide comparable results, while in comparison to the measured data, HoPFlow predicts better the pressure variations at all stations except at r/R=50% (Fig.4.66). It is noted that in rotor applications Cp is defined with respect to the local inflow velocity. Therefore, deviations at inboard stations are amplified. Furthermore, due to this non-dimensionalization, deviations of this size will have minor effects on the blade loading.

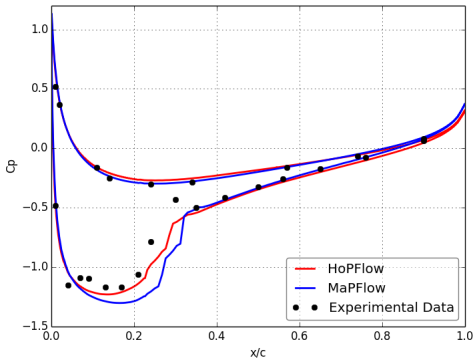
Over the outer part of the blade, the improvement brought by HoPFlow concerns the location of the shock wave. At high Ma numbers, small changes in the inflow sectional conditions can have a significant effect on the loading. Therefore it can be conceived that HoPFlow predicts more accurately the wake especially in the tip region. As further explained in the next section, diffusion in HoPFlow is minimum which means that the tip vortices in the present simulation will last longer than in MaPFlow. For helicopter applications such a feature is of particular importance. Tip vortices are responsible for Blade Vortex Interaction (BVI) encounters that generate noise in descent flight conditions. Besides that, small changes in the loading over the outer part of the blades will affect the trimming of the rotor which globally controls the flow development.



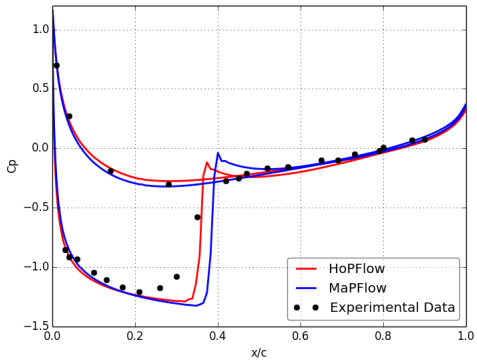
(a) Pressure distribution at r/R=50%



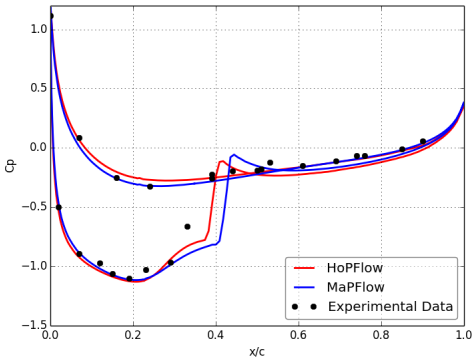
(b) Pressure distribution at r/R=68%



(c) Pressure distribution at r/R=80%



(d) Pressure distribution at r/R=89%



(e) Pressure distribution at r/R=96%

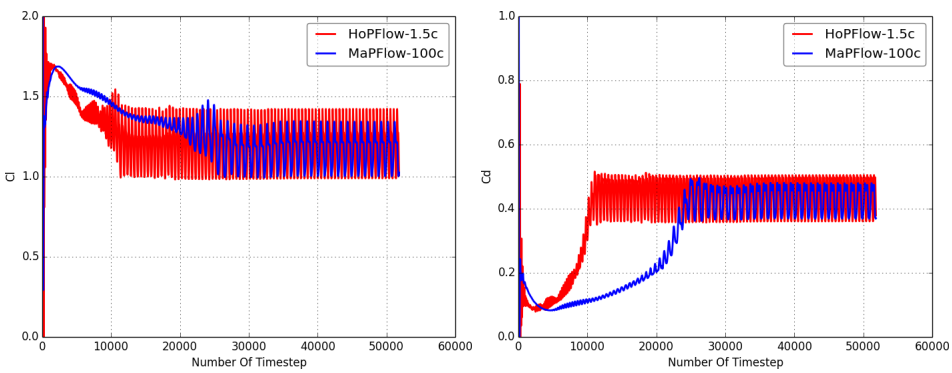
Figure 4.66: Caradonna-Tung Rotor: Surface pressure distributions at five radial positions.

4.3 Discussion on the Results of the Hybrid Method

In this section, an overall discussion on the validation of HoPFlow is given in an attempt to draw general conclusions on its behavior. The following three aspects are considered: the convergence properties, the amount of numerical diffusion and the computational cost.

4.3.1 Convergence Properties

Convergence is considered as a measure of cost and refers to the number of time steps required to reach a steady or periodic flow state. The first case examined is the viscous NACA0012 case presented in Section 4.2.5. In Fig. 4.67, the convergence rate in the lift and drag time histories is shown. HoPFlow clearly converges much faster than MaPFlow does. This is because in CFD, low frequency errors must travel up to the far-field boundary before they stop affecting the progress of the simulation. In the specific simulation the far-field in MaPFlow was set at 100 chord lengths while in HoPFlow the extent of the CFD grid was limited to 1.5 chord lengths.

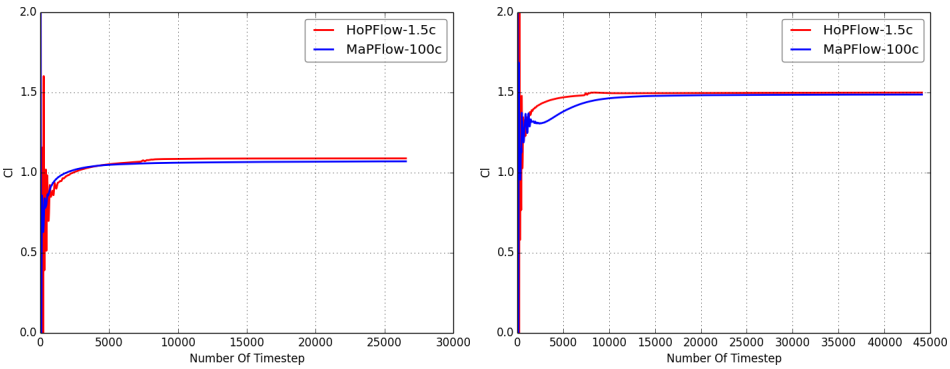


(a) Lift Coefficient Convergence History

(b) Drag Coefficient Convergence History

Figure 4.67: NACA0012: SST model convergence history at $\alpha = 20^o$. Results from the case presented in Section 4.2.5

The same behavior was also obtained in other Angles of Attack but not in all of them. Good convergence rate was obtained at $\alpha = 15^0$ (Fig. 4.68b), while at $\alpha = 10^o$ (Fig. 4.68a) HoPFlow converged more slowly.



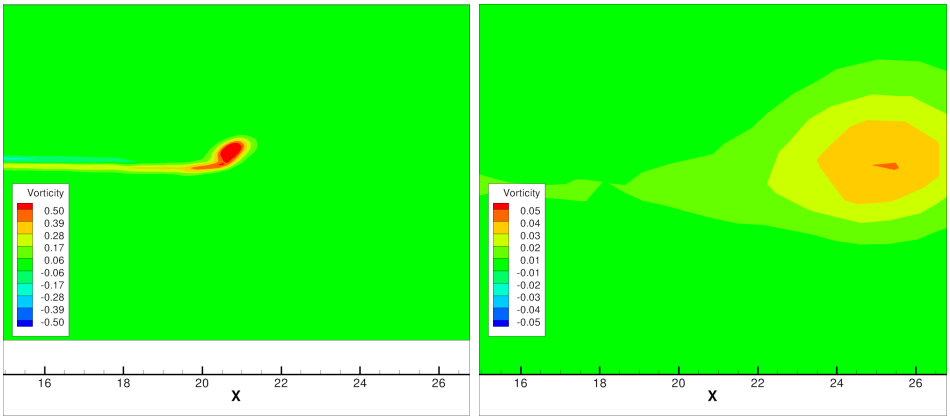
(a) Lift coefficient convergence history $\alpha = 10^o$

(b) Lift coefficient convergence history $\alpha = 15^o$

Figure 4.68: NACA0012: SST model Cl convergence histories at 10° and 15°.

This difference in the behavior of HoPFlow is related to the way the wake develops during the simulation. As already mentioned, the hybrid method is less diffusive compared to MaPFlow, as clearly shown in Fig. 4.69. This means that the starting vortex will keep affecting the flow much longer. If this is the only major excitation in the simulation, then HoPFlow will converge slower than MaPFlow, in agreement with the case at 10°. Moreover, standard CFD favors steady state simulations (even in an unsteady context such as this one) since the gradual grid coarsening towards the far-field adds artificial diffusion which damps the transient effects faster.

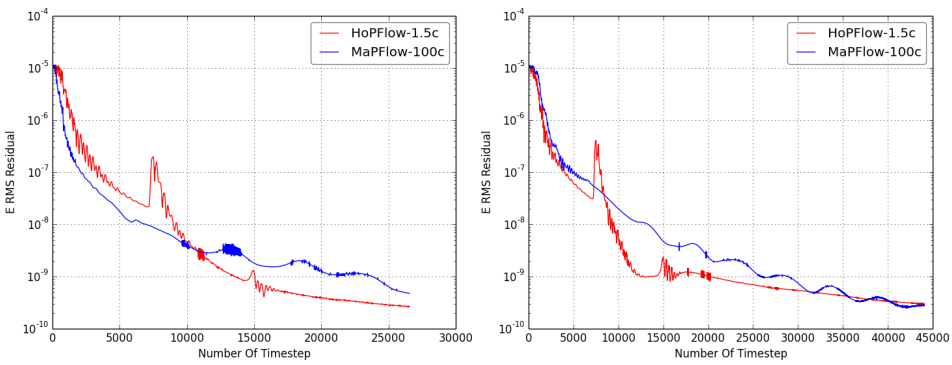
At higher angles and especially in stalled conditions the wake is continuously fed with vorticity; a process that controls the flow development. In MaPFlow, under such conditions convergence will be slower because numerical diffusion prevents fast convergence. This explains the high convergence rate at 20° as well as the lower rate at 15° for the NACA0012 case.



(a) HoPFlow starting vortex at t=20 (b) MaPFlow starting vortex at t=20

Figure 4.69: HoPFlow preserves the starting vortex longer

The effect of the starting vortex on the solution can be also seen in the convergence rate of the Energy residual. When the starting vortex is discarded as it exits from the outer PM grid and regardless the angle of attack, a spike appears at approximately the time of exit (Fig. 4.70).



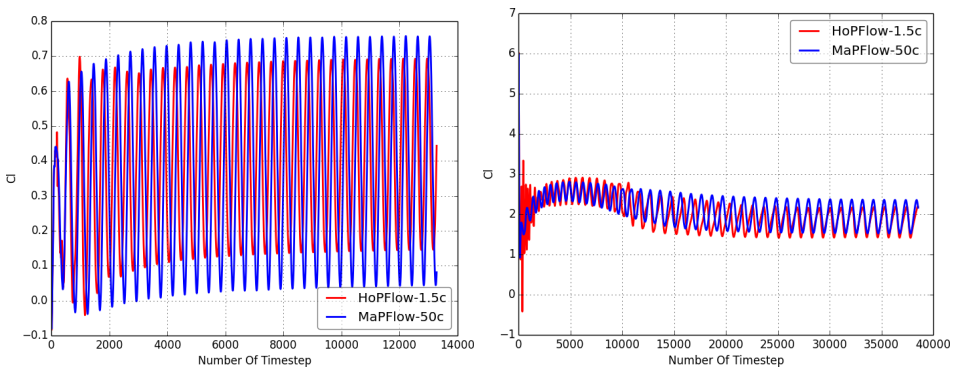
(a) Energy residual at $\alpha = 10^o$ (b) Energy residual at $\alpha = 15^o$

Figure 4.70: A spike appears in the Energy residual when the starting vortex leaves the Outer PM domain

When the flow is truly unsteady, the hybrid solver can converge faster because errors will leave the CFD domain faster. This is suggested by comparing the convergence behavior for a steady flow (Fig. 4.70) against an unsteady one (Fig. 4.67).

Finally, in Fig. 4.71 the lift coefficient convergence history is displayed for $\alpha = 0^o$ (Fig. 4.71a) and $\alpha = 20^o$ (Fig. 4.71b) for the flat-back FB-3750-1750

airfoil. In this case the convergence rate of the hybrid solver is approximately the same as for the CFD one. This is expected, because in the specific test case convergence heavily depends on the proper wake formation. Moreover, in Fig 4.71b at $\alpha = 20^o$, the transient effects at the initial time-steps are much more pronounced in the hybrid case. As already stressed, in the hybrid solver the transient solution is kept for longer time in the domain while in the CFD case transient effects are quickly diffused. So such a behavior, at least in the initial steps, is logical.



(a) FB-3750-1750: Cl Convergence History at $\alpha = 0^o$

(b) FB-3750-1750: Cl Convergence History at $\alpha = 20^o$

Figure 4.71: Lift coefficient convergence histories for the FB airfoil examined in Section 4.2.6. The wake must be well formed before the solution starts to converge towards a periodic state. This results the same convergence rate in MaPFlow and HoPFlow.

4.3.2 Diffusive Properties

In order to assess the diffusive properties of the Hybrid solver, a closer examination of the Fast Pitching NACA0012 case presented in Section 4.2.4 is made. Flow conditions are repeated in Table 4.18.

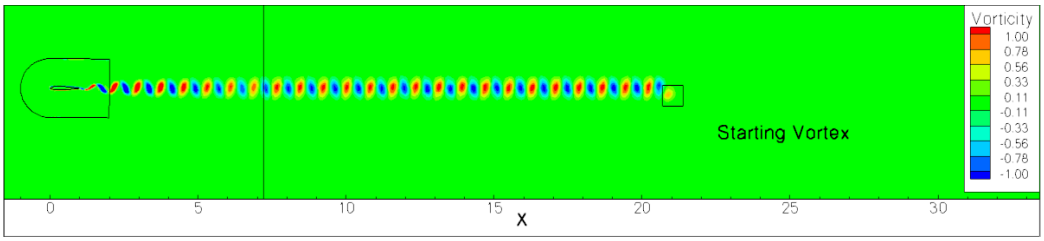
Table 4.18: NACA0012 Unsteady Flow conditions

Case	Mach	α_m (deg)	α_s (deg)	ω
Fast Pitching	0.6	0°	2°	8

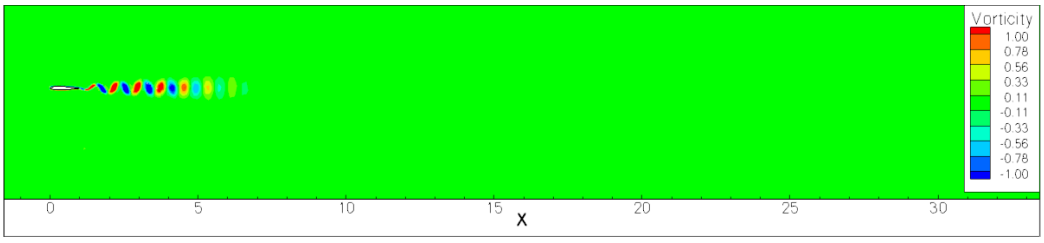
The specific test case was chosen since there is strong vortex shedding behind the airfoil due to the rapidly pitching motion. In addition, the flow can be considered inviscid which makes the grid requirements in the near wall region low. In Section 4.2.4 the 100c grid (CFD) had around 42000 cells while the 1c grid (Hybrid) has around 18000 cells. Even though in terms of lift and moment coefficient

the two Solvers predicted similar results, the flow-field in the wake region is very different.

Comparing the vorticity distribution in the wake shown in Fig 4.72 it is clear that the hybrid solver preserves vorticity much longer. In fact, even at a distance of 20 chord lengths, the vortices are well shaped and strong. On the contrary in standard CFD the vortices are quickly diffused and are hardy visible after 7 chord lengths.

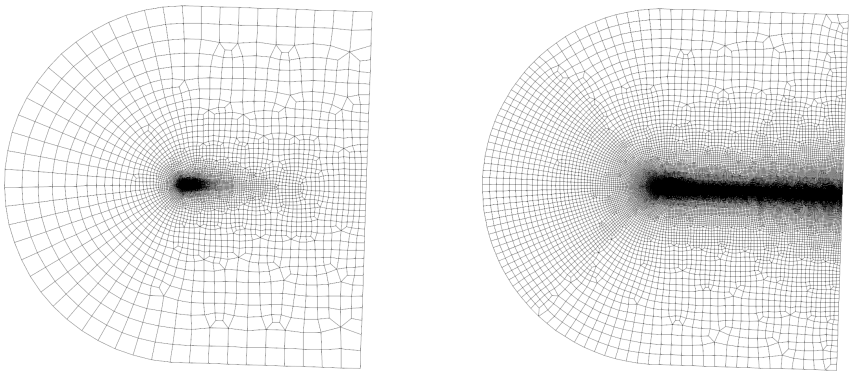


(a) Vorticity Contours in the Wake - HoPFlow



(b) Vorticity Contours in the Wake - MaPFlow

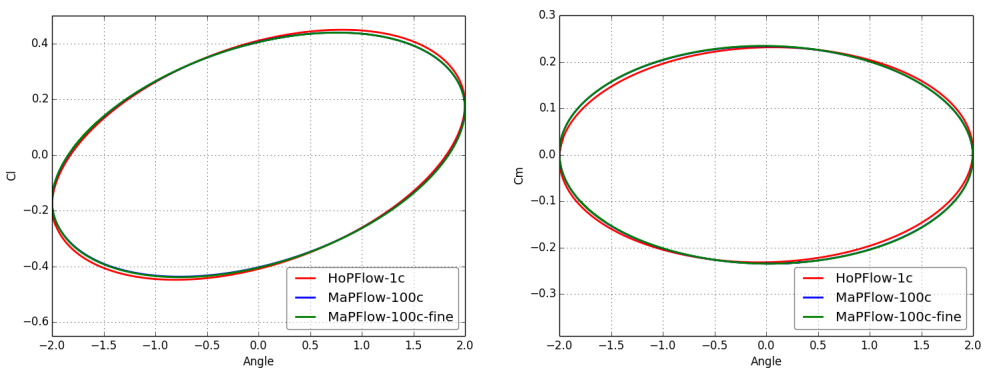
Figure 4.72: Comparison of the Wake characteristics between the two Solvers.



(a) Original CFD grid ≈ 42 K cells (b) Refined CFD grid ≈ 170 K cells

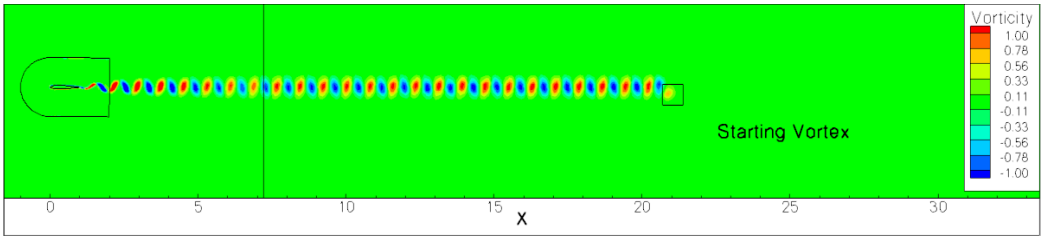
Figure 4.73: Mesh refinement in the wake

In order to examine whether a finer grid can substantially reduce numerical diffusion in standard CFD, a grid of 170000 cells was used. (Fig. 4.73). The lift and drag loops shown in Fig. 4.74 indicate that grid refinement had hardly any effect on the loads. However in the wake, there is clear improvement and the wake persists up to 15 chord lengths (Fig. 4.75).

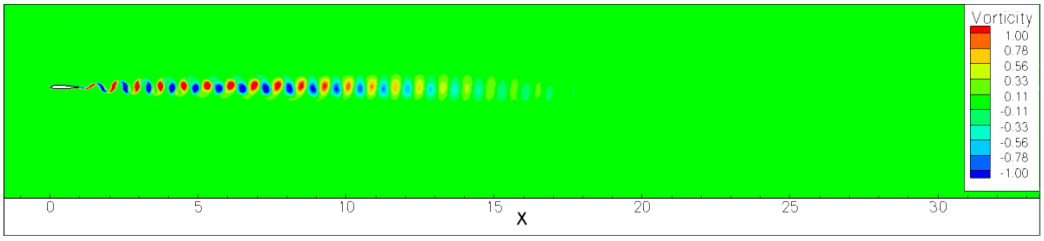


(a) Cl vs α (b) Cm vs α

Figure 4.74: Fast pitching NACA0012 results



(a) Vorticity Contours in the Wake - HoPFlow



(b) Vorticity Contours in the Wake region - MaPFlow

Figure 4.75: Comparison of the Wake characteristics between the two Solvers.

Despite this clear improvement, the performance of the Hybrid Solver is still superior. It is clear, that much denser grids would be required in order to achieve the same resolution in the wake. Of course, if the interest is on the airfoil loading, grid refinement is not needed. However, if the domain of interest is in the Wake region (as in the case where the wake impinges on another body that follows) then clearly the hybrid method outperforms.

4.3.3 Computational Cost

By restricting the CFD grid to only 1 chord length away from the solid boundary gives an advantage to the hybrid solver over the CFD one. However, this reduction goes along with the extra cost for solving the Lagrange equations. In all 2D simulations the same hybrid computational setup was used and thus the extra cost did not change regardless of the simulation.

For the computational cost assessment the fast pitching airfoil of the previous section is considered. The computational cost per time step is given in Table 4.19. In this case, the cost of the Lagrangian part dominates the overall cost per time step. In spite of that, the total cost is still smaller than that of the standard CFD on a fine grid of 170k cells. As discussed earlier, even this fine grid gives poorer results in the wake (Fig. 4.75). Nevertheless, the additional cost of the Lagrangian part is not negligible and thus the use of the hybrid solver proposed is not encouraged in such trivial cases.

Table 4.19: Time-step Runtime-4 cores Intel Xeon E5649

Standard CFD - 100c 170k cells	1.63 s
Standard CFD - 100c 42K cells	0.36 s
Standard CFD - 1c 19K cells	0.15 s
Hybrid Solver - 1c 19K cells	0.9s

However, in 3D simulations, the balance changes favorably for the hybrid solver. For the Caradonna-Tung rotor case the computational cost per time-step is displayed in Table 4.20. The reduction of the number of cells from 5.1 to 1.3 million compensates the Lagrangian solver cost. In fact, the hybrid solver is not only faster than the CFD one, but also compares better to measurements as shown in Fig 4.66 because the wake of rotor is better resolved.

Table 4.20: Time-step Runtime 8-cores Intel Xeon E5649

Standard CFD - 5.1mil. cells	1m 35s
Hybrid Solver - 1.3 mil.cells	1m 02 s

Strictly speaking, the comparison in the specific hover case is not fair. The standard CFD only considers one of the two blades and applies periodic conditions. On the contrary, the hybrid solver includes both blades. Furthermore, in standard CFD the specific problem can be solved in the rotating frame of reference as a steady state problem. So in this scenario, the standard CFD simulation runs faster than the hybrid one, but this does not guarantee equally good predictions as already noted. The situation will significantly change if instead of hover, a level or descent flight is considered. In this case standard CFD would need to extend the grid over the whole rotor and Chimera or Sliding grids should added (which will increase the complexity of the simulations). On the contrary the hybrid solver can handle all flight conditions directly.

Chapter 5

Conclusions & Recommendations

5.1 Conclusions

The aim of the present work was to develop a compressible hybrid CFD solver for external flow problems. To this end the following steps were taken:

- Development of a compressible URANS Solver (MaPFlow).
- Development of a compressible Lagrangian Solver.
- Coupling of the two solvers in a novel hybrid CFD solver (HoPFlow),

MaPFlow

MaPFlow solves the Unsteady Reynolds Averaged Navier-Stokes equations on unstructured grids. It is a parallel, multi block, all Mach, Finite Volume solver. In Chapter 4 a variety of test cases were considered and comparisons against measured data and other state of art CFD solvers were carried out. Out of these tests, MaPFlow was validated, proving that it performs at least similarly well to other existing codes and that it can successfully simulate a wide range of problems.

Lagrangian Solver

A compressible Lagrangian flow solver was developed. Particles carry mass, vorticity, dilatation and energy while their volume evolves in time. The specific formulation generalizes previous formulations that did not included energy as main flow quantity. Contrary to vortex methods, in the present formulation, particles must cover the complete flow field and this is because their volume is changing as a result of compressibility. In vortex methods the fluid is incompressible and therefore particles should only appear within the support of vorticity. This difference

leads to a vast amount of particles which renders the solver computationally inefficient. In order to mitigate the cost, the Particle Mesh method was adopted. By implementing the James-Lackner Algorithm the cost was successfully controlled.

HoPFlow

The final step in this work was to couple the Lagrangian and Eulerian solvers in one single package. The Eulerian solver was kept near the solid boundaries in order to accurately impose the solid wall boundary conditions while the Lagrangian solver was used to communicate accurate far-field conditions. The coupling of the two solvers was made strong and conservative in order to assure continuity of the flow properties throughout the whole flow-field. A novel hybrid CFD tool was thus produced featuring the following capabilities:

- **Easy handling of complex multi-body geometries:** In HoPFlow the CFD grid covers only a narrow zone around a solid boundary. Tests have shown that a width of 1 (or even less depending on the case) chord length is sufficient. This ability is a significant step. So in cases involving multi-body configurations, for every body a separate CFD grid is defined while the Lagrangian solver will take care to communicate the flow information in between. This also applies to geometries in which rotating and non-rotating bodies co-exist. This property of the hybrid solver resembles to the chimera technique in standard CFD, except that no grid topology search operations are needed. It also bypasses the restrictions the technique of sliding grids has.

- **High resolution in the near to the walls region:** The fact that the CFD grid covers a zone one chord wide, permits to have very fine grids in the near-wall region especially in 3D flow simulations. Usually standard 2D grids will have 70 % or more, of their cells within the near-wall region while 3D will have close to the solid boundaries 20-40 % of the grid. So limiting the extent of the CFD grid enables higher resolution in the boundary layer.

- **Better wake modeling:** In standard CFD by gradually coarsening the grid in the far field excess diffusion is added, which in certain cases is highly undesired. It will for example affect the way blade vortex interaction on helicopter rotors is predicted. Also it could affect the characteristics of the predicted inherent unsteadiness appearing under the onset of separation. In this respect, the hybrid solver provides a clear improvement. The underlying method has low numerical diffusion. In Chapter 4, wake modeling with Lagrangian particles was found superior as compared to standard CFD even when a locally refined grid is used. Combining this property with the usage of small CFD grids renders the hybrid solver most suitable in flow problems dominated by the dynamics of wakes.

In Chapter 4 the hybrid solver was validated in a wide variety of problems and flow conditions. It is important to stress, that most of the cases considered here, were selected in order to validate the proposed solver rather than to promote its capabilities. For example in Chapter 4, steady cases were treated as unsteady which unnecessarily increases the computational cost. The hybrid solver is by construction unsteady that can only produce time true simulations. However, this step was mandatory before considering more complex flows, where the capabilities of the proposed solver have crucial importance. In fact the hybrid method was developed to complement grid based CFD and by no means compete with it in standard cases. In this sense the proposed hybrid method can be considered as an enhancement to the current state of the art of CFD methods.

5.2 Recommendations

The recommendations discussed next concern at first short term improvements of the solver and then some long term goals for future research.

Short-term improvements:

Time integration: The current version uses a 1st order scheme in the Lagrangian part while the CFD part is kept 2nd order. This order mismatch restricts the allowable time step. Since in the CFD part the time scheme is already implicit, it is of interest to be able to use larger time steps. To this end the time scheme in the Lagrangian part should be upgraded to higher order. This will increase the size of allocated memory. So careful programming and management of the data storage is required.

Matching Conditions: In the current implementation, a buffer zone was introduced along the outer boundary of the CFD grid. This was added in order to eliminate the reflections of non-physical entropy waves. The research for truly transparent far-field boundary conditions is very active. In particular the LODI schemes formulated in [17] seem very promising.

Low Mach region: It has been observed, that for low Mach numbers (Ma<0.15), the Lagrangian solver required a very small time step to converge. It is a well known problem in CFD that led to preconditioning in order to improve stability. Likewise, in the Lagrangian part preconditioning should be added.

Diffusion: The clustering or spreading of particles deteriorates the Lagrangian algorithm. Re-meshing is a way to counteract the distortion of particles during the course of the simulation. However, re-meshing adds diffusion and thus ways of re-meshing with better diffusive properties should be investigated. One option in this respect is to add anti-diffusion. The other is to multi scale analysis and

control the spreading of vorticity.

Viscous Effects: In particle methods diffusion has been successfully treated by the Particle Strength Exchange (PSE) method [46]. In PM methods the implementation of PSE is straightforward as long as viscosity remains constant. However in turbulent flows the situation is different. Fortunately, far from solid boundaries ν_T can be considered constant and equal to the physical viscosity. Viscosity will also vary across regions of high shear which in flows with massive separation is connected to the detached vortex structures. In this case Lagrangian modeling could be adapted by formulating an LES like approach incorporating multi-scale analysis [39]. A consistent approach would be to introduce a so called turbulent velocity that would "correct" the spatial distribution of particles.

Cost: In the present work, the main goal was to formulate and check the foundations of the hybrid method rather than minimize the cost. The current implementation has not been optimized in terms of cost nor full use of parallel programming was made. Therefore, a thorough cost assessment of the hybrid solver should be conducted at programming level.

Additional CFD features: The present version of the solver does not contain certain valuable CFD techniques such Mesh Adaptation and Multigrid solution strategy. They would only affect the Eulerian part and in reality they will upgrade MaPFlow which is used in the Eulerian part of the hybrid solver.

Long terms goals

As stated in the introduction of this work, the development of the hybrid method was motivated by certain engineering problems that constitute major challenges for standard CFD. A list of such problems are given bellow:

1. Analysis of the flow around complex configurations that involve independently moving bodies. In this respect the following problems could be considered: the flow around a Vertical Axis Wind Turbine, the interaction between the rotor and the tower of a downwind wind turbine, the interaction between the main and tail rotor of a helicopter, the full helicopter configuration, the propeller wing interaction problem etc.

2. Analysis of flows that are dominated by large scale vortex structures. For example, the interaction of wind turbines in a wind farm, the blade vortex interaction problem in helicopter configurations, the effect of incoming atmospheric turbulence on structures etc.

3. Generation of synthetic turbulence using particles and by that LES flow modeling.

4. Analysis of Fluid Structure Interaction problems as in the case of wind turbines and helicopters.

5. Aeracoustic analysis of rotors as in the case of helicopters.

5.3 Publications

- G. Papadakis and S.G. Voutsinas. Particle Methods and Coupled Problems in Rotorcraft Aeromechanics. In *7th GRACM International Congress on Computational Mechanics,2011*

- M. Manolesos, G. Papadakis, and Spyros G. Voutsinas. A combined investigation on the formation of stall-cells on airfoils. The Science of Making Torque from Wind, 2012

- M. Manolesos, G. Papadakis, and S. G. Voutsinas. Experimental and computational analysis of stall cells on rectangular wings. *Wind Energy*, 17(6):939–955, 2014

- M. Manolesos, G. Papadakis, and S. G. Voutsinas. Assessment of the CFD capabilities to predict aerodynamic flows in presence of VG arrays. In *Journal of Physics: Conference Series*, volume 524, pages 012–029. IOP Publishing, 2014

- G. Papadakis, S.G. Voutsinas, G. Sieros, and T. Chaviaropoulos. CFD Aerodynamic Analysis of Non-Conventional Airfoil Sections for Very Large Rotor Blades. The Science of Making Torque from Wind, 2012

- J. Prospathopoulos, G. Papadakis, Giorgos Sieros, S.G. Voutsinas, Takis Chaviaropoulos, and Kostas Diakakis. Assessment of the aerodynamic characteristics of thick airfoils in high reynolds and moderate ma numbers using cfd modeling. In *Journal of Physics: Conference Series*, volume 524, page 012015. IOP Publishing, 2014

This page intentionally left blank.

Appendix A

Derivation of the Flow equations in Lagrangian (material) Description

A.1 Continuity equation

Let $\theta = \nabla \cdot \vec{u}$. Then:

$$\frac{\partial \rho}{\partial t} + \nabla\left(\rho \vec{u}\right) = 0 \Rightarrow \frac{\partial \rho}{\partial t} + (\vec{u} \cdot \nabla)\rho \; = \; -\rho\left(\nabla \cdot \vec{u}\right) \Rightarrow \frac{D\rho}{Dt} = -\rho(\nabla \cdot \vec{u}) \; = \; -\rho\theta$$

while,

$$\frac{D}{Dt}\left(\frac{1}{\rho}\right) \; = \; -\frac{1}{\rho^2}\frac{D\rho}{Dt} \; = \; \frac{\theta}{\rho}$$

Since, $M = \rho\, V$ and $\frac{DM}{Dt} = 0$,

$$\frac{D}{Dt}\left(\frac{1}{\rho}\right) \; = \; \frac{D}{Dt}\left(\frac{V}{M}\right) \Rightarrow \frac{DV}{Dt} = \theta V \tag{A.1}$$

A.2 Momentum

By considering the momentum equation in conservation form:

$$\frac{\partial(\rho\vec{u})}{\partial t} + \nabla\left(\vec{u} \otimes (\rho\vec{u})\right) = \nabla\left(-p + \sigma\right)$$

expanding the convection term and using the continuity equation,

$$\rho\frac{\partial \vec{u}}{\partial t} + \vec{u}\frac{\partial \rho}{\partial t} + (\vec{u} \cdot \nabla)\rho\vec{u} + (\rho\vec{u} \cdot \nabla)\vec{u} = \nabla\left(-p + \sigma\right) \Rightarrow$$

$$\rho\left(\frac{\partial \vec{u}}{\partial t} + (\vec{u} \cdot \nabla)\;\vec{u}\right) + \vec{u}\;\underbrace{\left(\frac{\partial \rho}{\partial t} + \nabla\left(\rho\vec{u}\right)\right)}_{\textbf{Continuity equation}} = \nabla\left(-p + \sigma\right)$$

and therefore,

$$\frac{\partial \vec{u}}{\partial t} + (\vec{u} \cdot \nabla)\vec{u} = \frac{\nabla\left(-p+\sigma\right)}{\rho} = 0 \Rightarrow \frac{D\vec{u}}{Dt} = \frac{\nabla\left(-p+\sigma\right)}{\rho} \tag{A.2}$$

Since,

$$(\vec{u} \cdot \nabla)\vec{u} = \nabla\frac{u^2}{2} - \vec{u} \times \vec{\omega}$$

then

$$\frac{\partial \vec{u}}{\partial t} + \nabla\frac{u^2}{2} - \vec{u} \times \vec{\omega} = \frac{\nabla\left(-p+\sigma\right)}{\rho} \tag{A.3}$$

Vorticity Transport Equation

Taking the curl ($\nabla\times$) of A.3 and introducing vorticity $\vec{\omega} = \nabla \times \vec{u}$:

$$\frac{\partial \nabla \times \vec{u}}{\partial t} + \nabla \times \left\{\nabla\frac{u^2}{2} - \vec{u} \times \vec{\omega}\right\} = \nabla \times \frac{\nabla\left(-p+\sigma\right)}{\rho} \Leftrightarrow$$

$$\frac{\partial \vec{\omega}}{\partial t} - \nabla \times (\vec{u} \times \vec{\omega}) = \nabla \times \frac{\nabla\left(-p+\sigma\right)}{\rho} \Leftrightarrow$$

$$\frac{\partial \vec{\omega}}{\partial t} - \{\vec{u}(\nabla \cdot \vec{\omega}) - \vec{\omega}(\nabla \cdot \vec{u}) + (\vec{\omega} \cdot \nabla)\vec{u} - (\vec{u} \cdot \nabla)\vec{\omega})\} = \nabla \times \frac{\nabla\left(-p+\sigma\right)}{\rho} = 0$$

Since $\nabla \cdot \vec{\omega} = 0$ and $\nabla \cdot \vec{u} = \theta$:

$$\frac{\partial \vec{\omega}}{\partial t} + \vec{\omega}\theta - (\vec{\omega} \cdot \nabla)\vec{u} + (\vec{u} \cdot \nabla)\vec{\omega} = \nabla \times \frac{\nabla\left(-p+\sigma\right)}{\rho} \Leftrightarrow$$

$$\underbrace{\frac{\partial \vec{\omega}}{\partial t} + (\vec{u} \cdot \nabla)\vec{\omega}}_{\textbf{Material Derivative}} + \vec{\omega}\theta - (\vec{\omega} \cdot \nabla)\vec{u} = \nabla \times \frac{\nabla\left(-p+\sigma\right)}{\rho} \Leftrightarrow$$

$$\frac{D\vec{\omega}}{Dt} = -\vec{\omega}\theta + (\vec{\omega} \cdot \nabla)\vec{u} + \nabla \times \frac{\nabla\left(-p+\sigma\right)}{\rho}$$

while

$$\nabla \times \frac{\nabla\left(-p+\sigma\right)}{\rho} = \frac{1}{\rho}\nabla \times \nabla\left(-p+\sigma\right) + \nabla\frac{1}{\rho} \times \nabla\left(-p+\sigma\right) \Leftrightarrow$$

$$= -\frac{1}{\rho^2}\nabla\rho \times \nabla\left(-p+\sigma\right)$$

Thus the vorticity equation becomes :

$$\frac{D\vec{\omega}}{Dt} = -\vec{\omega}\theta + (\vec{\omega} \cdot \nabla)\vec{u} - \frac{1}{\rho^2}\nabla\rho \times \nabla\left(-p+\sigma\right) \tag{A.4}$$

By introducing $\vec{\Omega} = \vec{\omega}\, V$,

$$\frac{D\vec{\Omega}}{Dt} = \frac{D(\vec{\omega}V)}{Dt} = \vec{\omega}\frac{DV}{Dt} + V\frac{D\vec{\omega}}{Dt} \Leftrightarrow$$

$$\frac{D\vec{\Omega}}{Dt} = \vec{\omega}\theta V + V\left(-\vec{\omega}\theta + (\vec{\omega} \cdot \nabla)\vec{u} - \frac{1}{\rho^2}\nabla\rho \times \nabla\left(-p+\sigma\right)\right)$$

which finally gives,

$$\frac{D\vec{\Omega}}{Dt} = V\left((\vec{\omega}\cdot\nabla)\vec{u} - \frac{1}{\rho^2}\nabla\rho\times\nabla\left(-p+\sigma\right)\right) \tag{A.5}$$

Dilatation Transport Equation

By taking the divergence ($\nabla\cdot$) of the momentum equation:

$$\begin{aligned}
&\nabla\cdot\left\{\frac{\partial\vec{u}}{\partial t} + \nabla\cdot\left(\vec{u}\cdot\nabla\right)\vec{u} + \frac{\nabla\left(p-\sigma\right)}{\rho}\right\} = 0 \Leftrightarrow \\
&\frac{\partial\nabla\cdot\vec{u}}{\partial t} + \nabla\cdot\left(\vec{u}\cdot\nabla\right)\vec{u} + \nabla\cdot\frac{\nabla\left(p-\sigma\right)}{\rho} = 0 \\
&\frac{\partial\theta}{\partial t} + \nabla\cdot\left(\vec{u}\cdot\nabla\right)\vec{u} + \nabla\cdot\frac{\nabla\left(p-\sigma\right)}{\rho} = 0
\end{aligned}$$

Expanding the divergence of the advection term $\left(\nabla\cdot\left(\vec{u}\cdot\nabla\right)\vec{u}\right)$:

$$\begin{aligned}
\nabla\cdot\left(\vec{u}\cdot\nabla\right)\vec{u} &= (\vec{u}\cdot\nabla)(\nabla\cdot\vec{u}) + (\nabla\cdot\vec{u})^2 - 2\|\nabla\vec{u}\| \\
&= (\vec{u}\cdot\nabla)\theta + \theta^2 - 2\|\nabla\vec{u}\|
\end{aligned}$$

where:

$$\nabla\vec{u} = \begin{bmatrix} \frac{\partial u}{\partial x} & \frac{\partial v}{\partial x} & \frac{\partial w}{\partial x} \\ \frac{\partial u}{\partial y} & \frac{\partial v}{\partial y} & \frac{\partial w}{\partial y} \\ \frac{\partial u}{\partial z} & \frac{\partial v}{\partial z} & \frac{\partial w}{\partial z} \end{bmatrix}$$

it follows that,

$$\begin{aligned}
&\underbrace{\frac{\partial\theta}{\partial t} + (\vec{u}\cdot\nabla)\theta}_{\textbf{Material Derivative}} \quad +\theta^2 - 2\|\nabla\vec{u}\| + \nabla\cdot\frac{\nabla\left(p-\sigma\right)}{\rho} = 0 \Leftrightarrow \\
&\frac{D\theta}{Dt} = -\theta^2 + 2\|\nabla\vec{u}\| - \nabla\cdot\frac{\nabla\left(-p+\sigma\right)}{\rho}
\end{aligned} \tag{A.6}$$

So for $\Theta = \theta\, V$, we obtain,

$$\begin{aligned}
&\frac{D\Theta}{Dt} = \frac{D\theta V}{Dt} = \theta\frac{DV}{Dt} + V\frac{D\theta}{Dt} \Leftrightarrow \\
&\frac{D\Theta}{Dt} = \theta^2 V + V\left(-\theta^2 + 2\|\nabla\vec{u}\| + \nabla\cdot\frac{\nabla\left(-p+\sigma\right)}{\rho}\right) \Leftrightarrow \\
&\frac{D\Theta}{Dt} = V\left(2\|\nabla\vec{u}\| + \nabla\cdot\frac{\nabla\left(-p+\sigma\right)}{\rho}\right)
\end{aligned} \tag{A.7}$$

A.3 Energy Equation

Starting from the energy equation in conservative form,

$$\frac{\partial \rho E}{\partial t} + \nabla(\rho \vec{u} E) = \nabla\left(\vec{u} \cdot (-pI + \sigma)\right) = 0$$

it follows that,

$$\begin{aligned}
&\rho \frac{\partial E}{\partial t} + E \frac{\partial \rho}{\partial t} + (\rho \vec{u} \cdot \nabla) E + E \nabla(\rho \vec{u}) = \nabla\left(\vec{u} \cdot (-pI + \sigma)\right) \Leftrightarrow \\
&\rho \left(\frac{\partial E}{\partial t} + (\vec{u} \cdot \nabla) E \right) + E \underbrace{\left(\frac{\partial \rho}{\partial t} + \nabla \rho \vec{u}) \right)}_{\textbf{Continuity Equation}} = \nabla\left(\vec{u} \cdot (-pI + \sigma)\right) \Leftrightarrow \\
&\rho \left(\frac{\partial E}{\partial t} + (\vec{u} \cdot \nabla) E \right) = \nabla\left(\vec{u} \cdot (-pI + \sigma)\right) \Leftrightarrow \\
&\rho \frac{DE}{Dt} = \nabla\left(\vec{u} \cdot (-pI + \sigma)\right)
\end{aligned}$$

By introducing $\Pi = \rho V E$,

$$\begin{aligned}
&\frac{D\Pi}{Dt} = \frac{D\rho V E}{Dt} = \rho V \frac{DE}{Dt} + E \frac{D\rho V}{Dt} \Leftrightarrow \\
&\frac{D\Pi}{Dt} = E \frac{DM}{Dt} + V \nabla\left(\vec{u} \cdot (-pI + \sigma)\right) \Leftrightarrow \\
&\frac{D\Pi}{Dt} = V \nabla\left(\vec{u} \cdot (-pI + \sigma)\right)
\end{aligned} \tag{A.8}$$

Bibliography

[1] Forrester T Johnson, Edward N Tinoco, and N Jong Yu. Thirty years of development and application of cfd at boeing commercial airplanes, seattle. *Computers & Fluids*, 34(10):1115–1151, 2005.

[2] P. Jawahar and H. Kemath. A high-resolution procedure for Euler and Navier-Stokes computations on unstructured grids. *J. Computational Physics*, 164:165–203, 2000.

[3] Charles Hirsch. *Numerical Computation of Internal and External Flows.* Wiley, 1990.

[4] Randall J LeVeque. *Finite volume methods for hyperbolic problems*, volume 31. Cambridge university press, 2002.

[5] Bernardo Cockburn and Chi-Wang Shu. Runge–kutta discontinuous galerkin methods for convection-dominated problems. *Journal of scientific computing*, 16(3):173–261, 2001.

[6] Thomas A Zang, Craig L Streett, and M Yousuff Hussaini. Spectral methods for cfd. 1989.

[7] A. Quarteroni and A. Valli. *Domain decomposition methods for partial differential equations.* Oxford Science Publications, 1999.

[8] G. Carey. *Computational Grids: Generation, Adaptation, and Solution Stategies.* Taylor & Francis, 1997.

[9] D.J. Mavriplis. Accurate multigrid solution of the Euler equations on unstructured and adaptive meshes. *AIAA Journal*, 28:213–221, 1990.

[10] A. Haselbacher and J. Blazek. Accurate and efficient discretization of Navier-Stokes equations on mixed grids. *AIAA Journal*, 38:2094–2102, 2000.

[11] ZJ Wang. A fully conservative interface algorithm for overlapped grids. *Journal of Computational Physics*, 122(1):96–106, 1995.

[12] R Steijl and G Barakos. Sliding mesh algorithm for cfd analysis of helicopter rotor–fuselage aerodynamics. *International journal for numerical methods in fluids*, 58(5):527–549, 2008.

[13] David C Wilcox et al. *Turbulence modeling for CFD*, volume 2. DCW industries La Canada, CA, 1998.

[14] W Rodi, JH Ferziger, M Breuer, and M Pourquie. Status of large eddy simulation: results of a workshop. *Transactions-American Society of Mechanical Engineers Journal of Fluids Engineering*, 119:248–262, 1997.

[15] Myoungkyu Lee, Nicholas Malaya, and Robert D Moser. Petascale direct numerical simulation of turbulent channel flow on up to 786k cores. In *Proceedings of SC13: International Conference for High Performance Computing, Networking, Storage and Analysis*, page 61. ACM, 2013.

[16] Jochen Fröhlich and Dominic von Terzi. Hybrid les/rans methods for the simulation of turbulent flows. *Progress in Aerospace Sciences*, 44(5):349–377, 2008.

[17] T J&amp Poinsot and SK Lelef. Boundary conditions for direct simulations of compressible viscous flows. *Journal of computational physics*, 101(1):104–129, 1992.

[18] Sandip Ghosal. An analysis of numerical errors in large-eddy simulations of turbulence. *Journal of Computational Physics*, 125(1):187–206, 1996.

[19] LJ Vermeer, Jens Nørkær Sørensen, and A Crespo. Wind turbine wake aerodynamics. *Progress in aerospace sciences*, 39(6):467–510, 2003.

[20] John Steinhoff, Y Wenren, Lesong Wang, M Fan, Min Xiao, and Carsten Braun. The computation of flow over helicopter rotors and complex bodies using vorticity confinement. In *Proceedings of the 8th International Symposium of Computational Fluid Dynamics' 99*, 2001.

[21] Michel Costes and Gunawan Kowani. An automatic anti-diffusion method for vortical flows based on vorticity confinement. *Aerospace science and technology*, 7(1):11–21, 2003.

[22] G-H. Cottet and P.D. Koumoutsakos. *Vortex methods: Theory and Practice.* Cambridge University Press, 2000.

[23] Joe J Monaghan. Smoothed particle hydrodynamics. *Reports on progress in physics*, 68(8):1703, 2005.

[24] R. Gingold and J. Monaghan. Shock simulation by the particle method SPH. *J. Computational Physics*, 52:374–389, 1983.

[25] L. Barba, A. Leonard, and C. Allen. Advances in viscous vortex methods - meshless spatial adaptation based on radial basis function interpolation. *Int. J. Numerical Methods in Fluids*, 47:387–421, 2005.

[26] P.D. Koumoutsakos and A. Leonard. High resolution simulations of the flow around an impulsively started cylinder using vortex methods. *Fluid Mechanics*, 296:1–38, 1995.

[27] SG Voutsinas, MA Belessis, and KG Rados. Investigation of the yawed operation of wind turbines by means of a vortex particle method. In *AGARD CONFERENCE PROCEEDINGS AGARD CP*, pages 11–11. AGARD, 1995.

[28] Spyros G Voutsinas and Dimitris G Triantos. Aeroacoustics of full helicopter configurations using vortex particle flow approximation. In *Proceedings of CEAS Forum on Aeracoustics of Rotors and Propellers, Rome, Italy*, pages 175–91. DTIC Document, 1999.

[29] Philippe Chatelain, Alessandro Curioni, Michael Bergdorf, Diego Rossinelli, Wanda Andreoni, and Petros Koumoutsakos. Billion vortex particle direct numerical simulations of aircraft wakes. *Computer Methods in Applied Mechanics and Engineering*, 197(13):1296–1304, 2008.

[30] S.G. Voutsinas. Vortex Methods in Aeronautics: How to make things work. *Int. Journal of Computational Fluid Dynamics*, 20, 2006.

[31] G.K. Batchelor. *An Introduction to Fluid Mechanics*. Cambridge University Press, UK, 1967.

[32] J. Katz and A. Plotkin. *Low speed aerodynamics*. Cambridge University Press, UK, 2001.

[33] S. Huberson and S.G. Voutsinas. Particles and grids. *Computers and Fluids*, 31:607–625, 2002.

[34] R. W. Hockney and J. W. Eastwood. *Computer Simulation Using Particles*. McGraw-Hill, 1981.

[35] B. Couet, O. Buneman, and A. Leonard. Simulation of three-dimensional incompressible flows with a vortex in cell method. *J. Computational Physics*, 39:305–328, 1981.

[36] L. Greengard and V. Rokhlin. A Fast Algorithm for Particle Simulations. *J. Computational Physics*, 73, 1987.

[37] L. Greengard and V. Rokhlin. A new version of the fast multipole method for the Laplace equation in three dimensions. *Acta Numerica*, 7:229–269, 1997.

[38] J.T. Beale and A. Majda. Vortex methods II: Higher order accuracy in 2 and 3 dimensions. *Math. Comput.*, 32:29–52, 1982.

[39] M. Bergdorf, G.-H. Cottet, and P.D. Koumoutsakos. Multilevel adaptive particle methods fr convection-diffusion equations. *SIAM Multiscale Modeling and Simulation*, 4:328–357, 2005.

[40] G-H. Cottet and A. Magni. TVD remeshing formulas for particle methods. *C.R. Math*, 347:1367–1372, 2009.

[41] I.F. Sbalzarini, J.H. Walther, M. Bedrgdorf, S.E. Hieber, and P.D. Koumoutsakos. PPM - A highly efficient parallel particle-mesh library for the simulation of continuum systems. *J. Computational Physics*, 215:566–588, 2006.

[42] Alexandre Joel Chorin. Numerical study of slightly viscous flow. *Journal of Fluid Mechanics*, 57(04):785–796, 1973.

[43] E Rivoalen and S Huberson. The particle strength exchange method applied to axisymmetric viscous flows. *Journal of Computational Physics*, 168(2):519–526, 2001.

[44] J.D. Eldredge, T. Colonius, and A. Leonard. A vortex particle method for two-dimensional compressible flow. *J. Computational Physics*, 179:371–399, 2002.

[45] Monika Nitsche and James H Strickland. Extension of the gridless vortex method into the compressible flow regime*. *Journal of Turbulence*, 3:40–40, 2002.

[46] S Mas-Gallic, M Louaked, and O Pironneau. A particle in cell method for the 2-d compressible euler equations. In *Vortex flows and related numerical methods*, pages 373–387. Springer, 1993.

[47] R Meske. *The vortex-in-cell method for compressible flow.* PhD thesis, PhD thesis, Imperial College of Science and Technology, London, UK, 1994.

[48] Adrin Gharakhani and Ahmed F Ghoniem. Massively parallel implementation of a 3d vortex-boundary element method. In *ESAIM: Proceedings*, volume 1, pages 213–223. EDP Sciences, 1996.

[49] M. Drela. XFOIL: An Analysis and Design System for Low Reynolds Number Airfoils. *Conference on Low Reynolds Number Aerodynamics, University Notre Dame*, 1989.

[50] V. Riziotis and S.G. Voutsinas. A Viscous-Inviscid Interaction Model for Dynamic Stall Simulations on Airfoils. *Int. J. Num. Methods in Fluids*, 56:185–208, 2008.

[51] P. K. Chaviaropoulos, I. G. Nikolaou, K. A. Aggelis, M. N. N. Soerensen andJ. Johansen, O. L. Hansen, Mac Gaunaa, T. Hambraus, Heiko Frhr. von Geyr, Ch. Hirsch, Kang Shun, S. G. Voutsinas, G. Tzabiras, Y. Perivolaris, and S. Z. Dyrmose. Viscous and Aeroelastic Effects on Wind Turbine Blades. The VISCEL project. Part I: 3D Navier-Stokes Rotor simulations. *Wind Energy*, 6:365–385, 2003.

[52] Sven Schmitz and Jean-Jacques Chattot. A coupled Navier-Stokes/VortexPanel solver for the numerical analysis of wind turbines. *Computers & Fluids*, 35:742–745, 2006.

[53] J. Guermond, S. Huberson, and W. Shen. Simulation of 2D external viscous flows by means of a domain decomposition method. *J. Computational Physics*, 108:343–352, 1993.

[54] P. Plouhmans, G.S. Winckelmans, J.K. Salmon, A. Leonard, and M.S.Warren. Vortex methods for direct numerical simulation of three-dimensional bluff body flows: Application to the sphere at Re=300, 500, and 1000. *J. Computational Physics*, 178:427–436, 2002.

[55] G-H. Cottet. Particle-grid domain decomposition methods for the Navier-Stokes equations in exterior domains. *Lectures in Applied Mathematics Series,*, 28:100–118, 1991.

[56] Anusonti-Inthra, Phuriwat, and Matt Floros. Coupled CFD and Particle Vortex Transport Method: Wing Performance and Wake Validations. *38th AIAA Fluid Dynamics Conference and Exhibit.*, 2008.

[57] Christopher P. Stone, Earl P.N. Duque, Chr/Coistopher C. Hennes, and Adrin Gharakhani. Rotor wake modeling with a coupled Eulerian and Vortex Particle Method. *AIAA Paper 312*, 2010.

[58] G. S. Oxley. *A 2-D Hybrid Euler-Compressible Vortex Particle Method For Transonic Rotorcraft Flows.* PhD thesis, Dept. Mech. & Aero. Engineering, Carleton University, Ottawa, Canada, 2009.

[59] F.R. Menter. Zonal Two Equation k-omega Turbulence Models for Aerodynamic Flows. *AIAA Paper 93-2906*, 1993.

[60] P.R. Spalart and S.R. Allmaras. One-Equation Turbulence Model for Aerodynamic Flows. *Recherche Aerospatiale*, pages 5–21, 1994.

[61] Chi-Wang Shu and Stanley Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes. *Journal of Computational Physics*, 77(2):439–471, 1988.

[62] Chi-Wang Shu and Stanley Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes, ii. *Journal of Computational Physics*, 83(1):32–78, 1989.

[63] Ami Harten, Bjorn Engquist, Stanley Osher, and Sukumar R Chakravarthy. Uniformly high order accurate essentially non-oscillatory schemes, iii. *Journal of computational physics*, 71(2):231–303, 1987.

[64] Chi-Wang Shu. *Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws*. Springer, 1998.

[65] RA James. The solution of poisson's equation for isolated source distributions. *Journal of Computational Physics*, 25(2):71–93, 1977.

[66] Karl Lackner. Computation of ideal mhd equilibria. *Computer Physics Communications*, 12(1):33–44, 1976.

[67] Christopher R Anderson. A method of local corrections for computing the velocity field due to a distribution of vortex blobs. *Journal of Computational Physics*, 62(1):111–123, 1986.

[68] Gregory T. Balls and Phillip Colella. A Finite Difference Domain Decomposition Method Using Local Corrections for the Solution of Poissons Equation. *J. Computational Physics*, 180:25–53, 2002.

[69] François Thirifay, Grégoire Winckelmans, et al. Development of a lagrangian method for combustion and application to the planar methane-air jet diffusion flame*. *Journal of Turbulence*, 3(59):1–7, 2002.

[70] Y-H Choi and Charles L Merkle. The application of preconditioning in viscous flows. *Journal of Computational Physics*, 105(2):207–223, 1993.

[71] Eli Turkel and Veer N Vatsa. Local preconditioners for steady and unsteady flow applications. *ESAIM: Mathematical Modelling and Numerical Analysis*, 39(03):515–535, 2005.

[72] L-E Eriksson. A preconditioned navier-stokes solver for low mach number flows. In *ECCOMAS computational fluid dynamics conference*, pages 199–205, 1996.

[73] V.G. Asouti, D.I. Papadimitriou, D.G. Koubogiannis, and K.C. Giannakoglou. Low Mach Number Preconditioning For 2d and 3d Upwind Flow Solution Schemes On Unstructured Grids. *5th GRACM International Congress on Computational Mechanics,Limassol*, 2005.

[74] Y. Colin, H. Deniau, and J.-F. Boussuge. A robust low speed preconditioning formulation for viscous flow computations. *Computers and Fluids*, 47(1):1 – 15, 2011.

[75] Biedron Robert and Thomas James. Recent Enhancements to the FUN3D Flow Solver for Moving-Mesh Applications. *47th AIAA Aerospace Sciences Meeting including The New Horizons Forum and Aerospace Exposition*, pages 1–19, jan 2009.

[76] R.K. Agarwal and J.E. Deese. Euler Calculations for Flowfield of a Helicopter Rotor in Hover. *Journal of Aircraft*, 24(4):231–238, 1987.

[77] Aftosmis M., Gaitonde D., and Tavares T.S. Behavior of Linear Reconstruction Techniques on Unstructured Meshes. *AIAA Journal*, 33:2038–2049, nov 1995.

[78] V. Venkatakrishnan. On the Accuracy of Limiters and Convergence to Steady State Solutions. *AIAA paper 93-0880*, 1993.

[79] V. Venkatakrishnan. Convergence to Steady State Solutions of the Euler Equations on Unstructured Grids with Limiters. *Journal of Computational Physics*, 118(1):120–130, 1995.

[80] Antony Jameson, Wolfgang Schmidt, Eli Turkel, et al. Numerical solutions of the euler equations by finite volume methods using runge-kutta time-stepping schemes. *AIAA paper*, 1259, 1981.

[81] Bram Van Leer. Flux-vector splitting for the euler equations. In *Eighth international conference on numerical methods in fluid dynamics*, pages 507–512. Springer, 1982.

[82] Joseph L Steger and RF Warming. Flux vector splitting of the inviscid gasdynamic equations with application to finite-difference methods. *Journal of computational physics*, 40(2):263–293, 1981.

[83] Philip L Roe. Approximate riemann solvers, parameter vectors, and difference schemes. *Journal of computational physics*, 43(2):357–372, 1981.

[84] V.G. Asouti. *Aerodynamic Analysis and design methods at high and low speed flows, on multiprocessor platforms.* PhD thesis, Laboratory of Thermal Turbomachines, Fluids Section, Department of Mechanical Engineering, NTUA, 2009.

[85] Jiri Blazek. *Computational Fluid Dynamics: Principles and Applications.* Elsevier Science, 2001.

[86] Biedron Robert, Vatsa Veer, and Atkins Harold. Simulation of Unsteady Flows Using an Unstructured Navier-Stokes Solver on Moving and Stationary Grids. *23rd AIAA Applied Aerodynamics Conference*, pages 1–17, June 2005.

[87] Veer N Vatsa, Mark H Carpenter, and David P Lockard. Re-evaluation of an optimized second order backward difference (bdf2opt) scheme for unsteady flow applications. *AIAA Paper*, 122:2010, 2010.

[88] D.J. Mavriplis and A. Jameson. Multigrid solution of the Navier-Stokes equations on triangular meshes. *AIAA Journal*, 28(8):1415–1425, Aug 1990.

[89] Antony Jameson. Time dependent calculations using multigrid, with applications to unsteady flows past airfoils and wings. *AIAA paper*, 1596:1991, 1991.

[90] Kazem Hejranfar and Ramin Kamali-Moghadam. Preconditioned characteristic boundary conditions for solution of the preconditioned Euler equations at low Mach number flows. *Journal of Computational Physics*, 231(2):4384 – 4402, 2012.

[91] F.R. Menter. Two-Equation Eddy-Viscosity Turbulence Models for Engineering Applications. *AIAA Journal*, 32:1598–1605., 1994.

[92] Dimitrios Koubogiannis. *Numerical Solution of the Navier-Stokes Equations on Unstructured Grids in a Parallel Processing Environment.* PhD thesis, Laboratory of Thermal Turbomachines, Fluids Section, Department of Mechanical Engineering, NTUA, 1998.

[93] Y. Saad. *Iterative Methods for Sparse Linear Systems.* Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2nd edition, 2003.

[94] Aleksios Theofilopoulos. Numerical analysis of the dynamic behavior of airfoils with deformable and articulated trailing edge flap, 2013.

[95] Y Zhao, J Tai, and F Ahmed. Simulation of micro flows with moving boundaries using high-order upwind fv method on unstructured grids. *Computational mechanics*, 28(1):66–75, 2002.

[96] PD Thomas and CK Lombard. Geometric conservation law and its application to flow computations on moving grids. *AIAA journal*, 17(10):1030–1037, 1979.

[97] Dimitri J Mavriplis and Zhi Yang. Construction of the discrete geometric conservation law for high-order time-accurate simulations on dynamic meshes. *Journal of Computational Physics*, 213(2):557–573, 2006.

[98] Hyung Taek Ahn and Yannis Kallinderis. Strongly coupled flow/structure interactions with a geometrically conservative ale scheme on general hybrid meshes. *Journal of Computational Physics*, 219(2):671–696, 2006.

[99] Adam Jirasek. Vortex-generator model and its application to flow control. *Journal of Aircraft*, 42(6):1486–1491, 2005.

[100] Julianne C Dudek. *Modeling Vortex Generators in the Wind-US Code.* National Aeronautics and Space Administration, Glenn Research Center, 2010.

[101] Marinos Manolesos. *Experimental and computational study of three-dimensional separation and its control using passive vortex generators.* PhD thesis, Laboratory of Aerodynamics, Fluids Section, Department of Mechanical Engineering, NTUA, 2013.

[102] J Thomas Beale and Andrew Majda. Vortex methods. i. convergence in three dimensions. *Mathematics of Computation*, 39(159):1–27, 1982.

[103] J Thomas Beale and Andrew Majda. Vortex methods. ii. higher order accuracy in two and three dimensions. *Mathematics of Computation*, 39(159):29–52, 1982.

[104] L Rosenhead. The formation of vortices from a surface of discontinuity. *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character*, pages 170–192, 1931.

[105] J Thomas Beale and Andrew Majda. High order accurate vortex methods with explicit velocity kernels. *Journal of Computational Physics*, 58(2):188–208, 1985.

[106] Robert Krasny. Desingularization of periodic vortex sheet roll-up. *Journal of Computational Physics*, 65(2):292–313, 1986.

[107] S.M. Richardson and A.R.H. Cornish. Solution of three dimensional incompressible flow problems. *J. of Fluid Mechanics*, 82:309–319, 1977.

[108] R. Boisvert. A fourth order accurate fast direct method for the Helmholtz equation. *Academic Press*, pages 35–44, 1984.

[109] Tim Colonius and Kunihiko Taira. A fast immersed boundary method using a nullspace approach and multi-domain far-field boundary conditions. *Comput. Methods Appl. Mech. Engrg.*, 197:2131–2146, 2008.

[110] M. Lai and C.S. Peskin. An immersed boundary method with formal second-order accuracy and reduced numerical viscosity. *J. Computational Physics*, 160:705–719, 2000.

[111] Kevin W Thompson. Time dependent boundary conditions for hyperbolic systems. *Journal of computational physics*, 68(1):1–24, 1987.

[112] SK Richards, X Zhang, XX Chen, and PA Nelson. The evaluation of non-reflecting boundary conditions for duct acoustic computation. *Journal of Sound and Vibration*, 270(3):539–557, 2004.

[113] M Berkman, L Sankar, C Berezin, and M Torok. A navier-stokes/full potential/free wake method for advancing multi-bladed rotors. In *ANNUAL FORUM PROCEEDINGS-AMERICAN HELICOPTER SOCIETY*, volume 53, pages 305–314. AMERICAN HELICOPTER SOCIETY, 1997.

[114] NASA Turbulence Modeling Resource. http://turbmodels.larc.nasa.gov/.

[115] Christopher Lockwood Rumsey, Robert T Biedron, and James Lee Thomas. *CFL3D, Its History and Some Recent Applications*, volume 112861. National Aeronautics and Space Administration, Langley Research Center, 1997.

[116] Robert T Biedron, Joseph M Derlaga, Peter A Gnoffo, Dana P Hammond, William T Jones, Bil Kleb, Elizabeth M Lee-Rausch, Eric J Nielsen, Michael A Park, Christopher L Rumsey, et al. Fun3d manual: 12.4. 2014.

[117] N. Gregory and C.L. O'Reilly. Low-Speed Aerodynamic Characteristics of NACA 0012 Aerofoil Sections, including the Effects of Upper-Surface Roughness Simulation Hoar Frost. *NASA R&M 3726*, Jan 1970.

[118] D. D. Mourikis, V. Riziotis, and S.G. Voutsinas. Aerodynamic design using genetic algorithms and application to rotor blades. In *Paper presented at the Proceedings of the International Conference on Computational and ICCES*, 2004.

[119] M. Manolesos, G. Papadakis, and S. G. Voutsinas. Experimental and computational analysis of stall cells on rectangular wings. *Wind Energy*, 17(6):939–955, 2014.

[120] M. Manolesos and S. G. Voutsinas. Study of a stall cell using stereo particle image velocimetry. *Physics of Fluids (1994-present)*, 26(4):045101, 2014.

[121] M. Manolesos and S. G. Voutsinas. Geometrical characterization of stall cells on rectangular wings. *Wind Energy*, 2013.

[122] Julianne C. Dudek. Modeling Vortex Generators in the WIND-US Code. *Nasa Technical Report-NASA / TM2010-216744*, 2010.

[123] Chung-Sheng Yao, John C Lin, and Brian G Allan. Flow-field measurement of device-induced embedded streamwise vortex on a flat plate. *NASA STI/Recon Technical Report N*, 3:12931, 2002.

[124] M. Manolesos, G. Papadakis, and S. G. Voutsinas. Assessment of the CFD capabilities to predict aerodynamic flows in presence of VG arrays. In *Journal of Physics: Conference Series*, volume 524, pages 012–029. IOP Publishing, 2014.

[125] NPARC Alliance Validation Archive. http://www.grc.nasa.gov/WWW/wind/valid/homepage.html.

[126] CFL3D Navier Stokes Code. http://cfl3d.larc.nasa.gov/.

[127] *AGARD Advisory Report No. 138, Experimental data base for computer program assessment.* 1979.

[128] Mexnext(Phase 1): Analysis of Mexico wind tunnel measurements,Final report of IEA Task 29.

[129] M Maureen Hand, DA Simms, L Fingersh, D Jager, J Cotrell, S Schreck, and S Larwood. *Unsteady aerodynamics experiment phase V: test configuration and available data campaigns.* National Renewable Energy Laboratory, 2001.

[130] Niels N Sørensen, JA Michelsen, and S Schreck. Navier–stokes predictions of the nrel phase vi rotor in the nasa ames 80 ft× 120 ft wind tunnel. *Wind Energy*, 5(2-3):151–169, 2002.

[131] Mukesh Marutrao Yelmule and EswaraRao Anjuri VSJ. Cfd predictions of nrel phase vi rotor experiments in nasa/ames wind tunnel. *International Journal of Renewable Energy Research (IJRER)*, 3(2):261–269, 2013.

[132] David A Simms, S Schreck, M Hand, and LJ Fingersh. *NREL unsteady aerodynamics experiment in the NASA-Ames wind tunnel: a comparison of predictions to measurements.* National Renewable Energy Laboratory Colorado, USA, 2001.

[133] R. Landon. NACA0012, oscillatory and transient pitching. *AGARD Report No.707*, pages 3.1–3.25, 1982.

[134] Jonathon P Baker, CP Van Dam, and Benson L Gilbert. Flatback airfoil wind tunnel experiment. *Report, Sandia National Laboratories, Albuquerque, NM and Livermore, CA, USA*, 2008.

[135] JP Baker and CP van Dam. Drag Reduction of Blunt Trailing-Edge Airfoils, BBAA VI International Colloquium on: Bluff Bodies Aerodynamics & Applications. Technical report, Milano, Italy, 2008.

[136] Edward A Mayda, CP van Dam, David D Chao, and Dale E Berg. Computational design and analysis of flatback airfoil wind tunnel experiment. Technical report, Sandia National Laboratories, 2008.

[137] Niels N Sørensen. A Small Study of Flatback Airfoils. In *Aeroelastic Workshop*, pages 153–195.

[138] J. Prospathopoulos, G. Papadakis, Giorgos Sieros, S.G. Voutsinas, Takis Chaviaropoulos, and Kostas Diakakis. Assessment of the aerodynamic characteristics of thick airfoils in high reynolds and moderate ma numbers using cfd modeling. In *Journal of Physics: Conference Series*, volume 524, page 012015. IOP Publishing, 2014.

[139] R Rubinstein. Rumsey, cl, salas, md, thomas, jl (2001): Turbulence modelling workshop. Technical report, NASA-CR-2001-210841.

[140] F.X. Caradonna and C. Tung. Experimental and Analytical Studies of a Model Helicopter Rotor in Hover. *NASA Technical Memorandum 81232, National Aeronautics and Space Administration*, 1981.

[141] G. Papadakis and S.G. Voutsinas. Particle Methods and Coupled Problems in Rotorcraft Aeromechanics. In *7th GRACM International Congress on Computational Mechanics,2011.*

[142] M. Manolesos, G. Papadakis, and Spyros G. Voutsinas. A combined investigation on the formation of stall-cells on airfoils. The Science of Making Torque from Wind, 2012.

[143] G. Papadakis, S.G. Voutsinas, G. Sieros, and T. Chaviaropoulos. CFD Aerodynamic Analysis of Non-Conventional Airfoil Sections for Very Large Rotor Blades. The Science of Making Torque from Wind, 2012.

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΕΡΓΑΣΤΗΡΙΟ ΑΕΡΟΔΥΝΑΜΙΚΗΣ

Διδακτορική Διατριβή

Ανάπτυξη υβριδικής μεθόδου στοιχείων στροβιλότητας και εφαρμογή σε εξωτερικές ροές συμπεριλαμβανομένης αυτής γύρω από ελικόπτερο

Γιώργος Παπαδάκης

Δεκέμβριος 2014

Ανάπτυξη υβριδικής μεθόδου στοιχείων στροβιλότητας και εφαρμογή σε εξωτερικές ροές συμπεριλαμβανομένης αυτης γύρω από ελικόπτερο

Διδακτορική Διατριβή

Γιώργος Παπαδάκης

Εξεταστική Επιτροπή

1. Γ. Μπεργελές *, Καθηγητής ΕΜΠ, Σχολής Μηχανολόγων Μηχανικών
2. Α. Μπουντουβής, Καθηγητής ΕΜΠ, Σχολής Χημικών Μηχανικών
3. Δ. Μπούρης, Επ. Καθηγητής ΕΜΠ, Σχολής Μηχανολόγων Μηχανικών
4. Κ. Γιαννάκογλου *, Καθηγητής ΕΜΠ, Σχολής Μηχανολόγων Μηχανικών
5. Γ.Τζαμπίρας, Καθηγητής ΕΜΠ, Σχολής Ναυπηγών Μηχανικών
6. Σ.Τσαγγάρης, Καθηγητής ΕΜΠ, Σχολής Μηχανολόγων Μηχανικών
7. Σ. Βουτσινάς, (Επιβλέπων) *, Αν. Καθηγητής ΕΜΠ, Σχολής Μηχανολόγων Μηχανικών

* Μέλος της Συμβουλευτικής (Τριμελούς) Επιτροπής.

This page intentionally left blank.

Περιεχόμενα

Κατάλογος σχημάτων

Κατάλογος πινάκων

This page intentionally left blank.

Περίληψη

Σκοπός της διδακτορικής διατριβής ήταν η ανάπτυξη μιας νέας υβριδικής μεθοδολογίας CFD για την επίλυση εξωτερικών αεροδυναμικών ροών. Η ιδέα πίσω από την εργασία ήταν η ανάγκη για προσομοιώσεις σύνθετων προβλημάτων στα οποία κυριαρχούν ισχυρές δομές στροβιλότητας και που υπάρχουν σώματα τα οποία κινούνται ανεξάρτητα μεταξύ τους. Για το λόγο αυτό αναπτύχθηκαν δύο υπολογιστικά εργαλεία τα οποία συνενώθηκαν σε ένα υβριδικό επιλυτή. Πιο συγκεκριμένα:

Ο Eulerian CFD επιλυτής (MaPFlow): Αναπτύχθηκε ένας συμπιεστός URANS επιλυτής που λύνει πάνω σε μή δομημένα πλέγματα. Ο συγκεκριμένος επιλυτής είναι εφοδιασμένος με προσταθεροποιητή για χαμηλούς αριθμούς Mach για την προσομοίωση ασυμπίεστων ροών. Η μοντελοποίηση της τύρβης γίνεται είτε με το μοντέλο μίας εξίσωσης του Spalart-Almaras είτε με το μοντέλο δύο εξισώσεων k-ω SST του Menter. Ακόμη, ο επιλυτής μπορεί να χειριστεί κινούμενα ή παραμορφώσιμα πλέγματα ενώ έχει παραλληλοποιηθεί με τη χρήση του πρωτοκόλλου MPI.

Ο Lagrangian επιλυτής: Διατυπώθηκε και αναπτύχθηκε ένας συμπιεστός Lagrangian επιλυτής που χρησιμοποιεί στοιχεία στροβιλότητας. Η συγκεκριμένη διατύπωση χρησιμοποιεί στοιχεία ρευστού που μεταφέρουν μάζα, μεταβολή του όγκου, στροβιλότητα, ενέργεια και όγκο για να μπορεί να διαχειριστεί συμπιεστές ροές. Για να μειωθεί το υπολογιστικό κόστος του επιλυτή χρησιμοποιήθηκε η μέθοδος Particle Mesh (PM) η οποία παραλληλοποιήθηκε χρησιμοποιώντας τον αλγόριθμο του James-Lackner.

Σύυζευξη των δύο επιλυτών σε ένα υπολογιστικό εργαλειό (HoPFlow): Υλοποιήθηκε ισχυρή σύζευξη των Eulerian και Lagrangian επιλυτών σε μία υβριδική μεθοδολογία. Η σύζευξη έγινε με τέτοιο τρόπο ώστε να διασφαλίζει συνέχεια και συνέπεια της λύσης ανάμεσα στους δύο επιλυτές.

Τα αποτελέσματα που παρουσιάζονται στην παρούσα εργασία έχουν σκοπό την πιστοποίηση των εργαλείων που υλοποιήθηκαν. Αρχικά, παρουσιάζονται αποτελέσματα που αφορούν την πιστοποίηση του Eulerian URANS επιλυτή. Η πιστοποίηση περιλαμβάνει συγκρίσεις με πειραματικά αλλά και υπολογιστικά δεδομένα σε διάφορες διδιάστατες και τριδιάστατες ροές. Στη συνέχεια, ακολουθεί η πιστοποίηση του υβριδικού επιλυτή όπου γίνεται σύγκριση με τα αντίστοιχα Eulerian αποτελέσματα αλλά και με πειραματικά δεδομένα.

Οι περιπτώσεις πιστοποίησης που εξετάστηκαν περιλαμβάνουν διδιάστατες ροές γύρω από σταθερές και κινούμενες αεροτομές σε πληθώρα αριθμών Reynolds και Mach. Οι τρισδιάστατες περιπτώσεις που παρουσιάζονται αφορούν ροές γύρω από σταθερά και περιστρεφόμενα πτερύγια (Δρομείς Ανεμογεννητριών και Ελικοπτέρου).

Η χρήση των υπολογιστικών εργαλείων σε πληθώρα περιπτώσεων έδειξαν ότι και ο Eulerian CFD επιλυτής (MaPFlow) όπως και ο υβριδικός επιλυτής (HoPFlow) παράγουν ικανοποιητικά αποτελέσματα. Συγκεκριμένα, ο υβριδικός επιλυτής έχει λιγότερη διάχυση από τον Eulerian και για αυτό στις περιπτώσεις όπου κυριαρχούν ισχυροί στρόβιλοι (όπως ο δρομέας ελικοπτέρου σε αιώρηση) τα αποτελέσματα που παράγονται είναι καλύτερα. Θα πρέπει να τονιστεί ότι οι περισσότερες περιπτώσεις που εξετάστηκαν, είναι απλούστερες από αυτές για τις οποίες αναπτύχθηκε η υβριδική μέθοδος. Παρόλα αυτά, η επιλογή τους έγινε με σκοπό την πιστοποίηση της καινούργιας μεθόδου που προηγείται της χρήσης της σε πιο σύνθετες ροές.

Κεφάλαιο 1

Εισαγωγή

1.1 Εξέλιξη Υπολογιστικών μεθόδων στη Ρευστομηχανική

Η Υπολογιστική Ρευστομηχανική είναι ένας ενεργός τομέας έρευνας που τις τελευταίες δεκαετίες χρησιμοποιείται όλο και περισσότερο στην ανάλυση προβλημάτων από απλά αεροδυναμικά προβλήματα μέχρι αερο-ελαστικά προβλήματα πλήρους κλίμακας.

Σε συνδυασμό με τη μεγάλη πρόοδο στην τεχνολογία των υπολογιστών μπορούν πλέον να γίνουν προσομοιώσεις ροών γύρω από πολύπλοκες διατάξεις με χρήση πλεγμάτων αποτελούμενα από πολλά εκατομμύρια κελιά [1, 2].

Οι υπολογιστικές μέθοδοι CFD μπορούν να διαχωριστούν σε δύο μεγάλες κατηγορίες: οι μέθοδοι που βασίζονται στη χρήση υπολογιστικών πλεγμάτων και οι μέθοδοι που δεν χρησιμοποιούν πλέγματα. Στην τελευταία κατηγορία χρησιμοποιούνται στοιχεία που μεταφέρουν ποσότητες της ροής σε μια Lagrangian εκδοχή. Στις Eulerian μεθόδους, το υπολογιστικό πλέγμα αποτελεί το χωρίο επίλυσης ενώ το πεδίο ροής περιγράφεται από μεγέθη ορισμένα πάνω στο υπολογιστικό πλέγμα.

Eulerian Επιλυτές

Οι Eulerian μέθοδοι υπολογιστικής ρευστομηχανικής χρησιμοποιούνται κατά κόρον για την προσομοίωση αεροδυναμικών ροών. Οι συνθήκες στερεού ορίου ικανοποιούνται ακριβώς ενώ οι εξισώσεις που περιγράφουν τη ροή λύνονται στο υπολογιστικό πλέγμα. Τα πλέγματα που χρησιμοποιούνται μπορεί να είναι δομημένα ή μηδομημένα, ενώ ανάλογα με την διατύπωση που χρησιμοποιείται για την διακριτοποίηση των εξισώσεων χωρίζονται σε: μεθόδους Πεπερασμένων Διαφορών [3], μεθόδους Πεπερασμένων Όγκων [4], ασυνεχείς μεθόδους Galerkin [5] και Φασματικές μεθόδους [6].

Στα σύγχρονα υπολογιστικά εργαλεία CFD έχουν γίνει σημαντικές προσπάθειες στην αύξηση της ακρίβειας και στην ταυτόχρονη μείωση του κόστους. Αυτές περι-

λαμβάνουν: παραλληλοποίηση της μεθόδου χωρίζοντας το υπολογιστικό πλέγμα σε επιμέρους κομμάτια [7]; τοπική πύκνωση του πλέγματος κατά τη διάρκεια της επίλυσης στις περιοχές μεγάλων κλίσεων [8]; τεχνικές πολλαπλών πλεγμάτων για γρηγορότερη σύγκλιση [9]; χρήση μεικτών πλεγμάτων για καλύτερη προσομοίωση του οριακού στρώματος [10]. Επιπλέον για την προσομοίωση πολύπλοκων γεωμετρίων με σώματα να κινούνται ελεύθερα μεταξύ τους αναπτύχθηκαν οι μέθοδοι Chimera [11] και Sliding Grids [12].

Αν και η έρευνα στον τομέα των Eulerian υπολογιστικών εργαλείων έχει κάνει πολύ σημαντικά βήματα υπάρχουν ακόμη αρκετά περιθώρια βελτίωσης. Στην πράξη τα περισσότερα εργαλεία που χρησιμοποιούνται είναι δεύτερης τάξης ακριβείας στο χώρο και στο χρόνο. Ακόμη στην περίπτωση που εξετάζονται εξωτερικές αεροδυναμικές ροές, το υπολογιστικό πλέγμα θα πρέπει να έχει μεγάλη έκταση για να ικανοποιούνται σωστά οι επ' άπειρο οριακές συνθήκες. Αν και η τεχνολογία των υπολογιστών εξελίσσεται συνεχώς, η σταδιακή αραίωση των πλεγμάτων καθώς προσεγγίζεται το εξωτερικό όριο του πλέγματος είναι απαραίτητη. Αυτό έχει ως αποτέλεσμα τη μεγάλη αριθμητική διάχυση της μεθόδου στην περιοχή του ομόρου. Επιπλέον, στην περίπτωση που η υπολογιστική διάταξη περιέχει ελεύθερα κινούμενα σώματα όπως στην περίπτωση των Α/Γ καθέτου άξονα ή στην αλληλεπίδραση δρομέα-πύργου στις Α/Γ οριζοντίου άξονα οι υπολογισμοί είναι πολύ ακριβοί καθώς τα πλέγματα που θα χρησιμοποιηθούν είναι πολύ πυκνά ενώ η σχετική κίνηση μεταξύ των σωμάτων βάζει επιπλέον δυσκολίες στους υπολογισμούς.

Lagrangian Επιλυτές

Η άλλη κατηγορία επιλυτών υπολογιστικής ρευστομηχανικής αφορά τις Lagrangian μεθόδους. Σε αυτή την κατηγορία μεθόδων δεν χρησιμοποιείται υπολογιστικό πλέγμα και συνεπώς έχουν μικρότερη διάχυση από τις Eulerian μεθόδους. Επίσης, οι επ' άπειρο συνθήκες ικανοποιούνται ακριβώς καθώς εμπεριέχονται μέσα στις εξισώσεις [13].

Στις Lagrangian μεθόδους υπάρχουν δύο κύριες επιλογές: οι μέθοδοι στοιχείων στροβιλότητας [13] και οι μέθοδοι Smooth Particle Hydrodynamics (SPH) [14, 15]. Η βασική διαφορά των δύο έγκειται στις ποσότητες που μεταφέρονται από τα στοιχεία που προσομοιώνουν τη ροή. Ανεξάρτητα από την επιλογή οι οριακές συνθήκες στερεού ορίου μοντελοποιούνται δύσκολα καθώς χρειάζεται μεγάλος αριθμός στοιχείων στροβιλότητας [16, 17].

Οι μέθοδοι στοιχείων στροβιλότητας έχουν χρησιμοποιηθεί πολύ στη μοντελοποίηση του ομόρου πίσω από περιστρεφόμενα σώματα όπως οι δρομείς Α/Γ [18], οι δρομείς ελικοπτέρων [19] αλλά και στην περίπτωση των αεροσκαφών [20]. Συνήθως συνδυάζονται με μεθόδους Συνοριακών Στοιχείων [21] για την μοντελοποίηση του στερεού ορίου. Οι μέθοδοι στοιχείων στροβιλότητας είναι ιδανικοί για την προσομοίωση ομόρου λόγω της μικρής αριθμητικής διάχυσης, όμως ταυτόχρονα έχουν μεγάλο υπολογιστικό κόστος καθώς ο υπολογισμός των ταχυτήτων μεταφοράς δίνεται από ολοκληρωτικές αναπαραστάσεις [22, 23]. Στην κατεύθυνση της μείωσης του κόστους του υπολογισμού των ταχυτήτων, τεχνικές επιτάχυνσης των υπολογισμών

έχουν αναπτυχθεί. Αυτές περιλαμβάνουν, τις μεθόδους Particle Mesh [24, 25, 26] και της μεθόδους multi-pole expansions [27, 28].

Σημαντική παράμετρος στην ακρίβεια των μεθόδων στοιχείων στροβιλότητας είναι η κατανομή των στοιχείων. Η τοπική συγκέντρωση/αραίωση των στοιχείων στροβιλότητας μειώνει την ακρίβεια της μεθόδου [13, 29]. Για αυτό το λόγο υιοθετήθηκε η τεχνική του remeshing, κατά την οποία γίνεται επαναδιανομή των στοιχείων στροβιλότητας σε διατεταγμένες θέσεις [13, 30, 31, 32].

Οι μέθοδοι στοιχείων στροβιλότητας αρχικά αναπτύχθηκαν για την προσομοίωση ατριβών, ασυμπίεστων ροών. Οσο αναφορά τη συνεκτικότητα έχουν γίνει προσπάθειες να συμπεριληφθεί στην επίλυση των εξισώσεων σε στρωτές ροές [33, 34] χωρίς και με την ύπαρξη στερεού ορίου [16, 17]. Προσομοιώσεις σε υψηλούς αριθμούς Reynolds δεν έχουν γίνει ακόμη καθώς αυτές θα απαιτούσαν πολύ μεγάλο αριθμό στοιχείων στροβιλότητας.

Όσο αναφορά την επέκταση της μεθόδου στοιχείων στροβιλότητας σε συμπιεστές ροές έχουν γίνει προσπάθειες με από τους [35, 36] με χρήση μόνο στοιχείων στροβιλότητας και από τους [37, 38] όπου χρησιμοποιήθηκε ένα καρτεσιανό πλέγμα για την επίλυση των συμπιεστών χαρακτηριστικών της ροής.

Eulerian vs Lagrangian Methods

Λαμβάνοντας υπόψη τα υπέρ (+) και τα κατά (-) των δύο κυρίων κατηγοριών των μεθόδων υπολογιστικής ρευστομηχανικής γίνεται η παρακάτω σύγκριση:

Eulerian Επιλυτές

+ Οι οριακές συνθήκες τοίχου ικανοποιούνται ακριβώς.
+ Έχουν χρησιμοποιηθεί σε πληθώρα προβλημάτων.
+ Χρησιμοποιώντας πυκνό υπολογιστικό πλέγμα μπορούν να γίνουν ακριβείς υπολογισμοί.
- Η ικανοποίηση των επ' άπειρο συνθηκών απαιτούν πλέγματα μεγάλης έκτασης.
- Η σταδιακή αραίωση του πλέγματος εισάγει αριθμητική διάχυση.
- Η προσομοίωση πολλαπλών σωμάτων είναι δύσκολη και απαιτεί ειδικές τεχνικές.

Lagrangian Επιλυτές

+ Δεν απαιτείται χρήση πλέγματος.
+ Έχουν μικρή αριθμητική διάχυση.
+ Ικανοποιούν ακριβώς τις επ' άπειρο συνθήκες.
+ Χρησιμοποιούνται με επιτυχία στην μοντελοποίηση του ομόρου .
+ Οι υπολογισμοί γύρω από πολλαπλά σώματα είναι απλοί.
- Οι ολοκληρωτικές αναπαραστάσεις αυξάνουν δραματικά το κόστος.
- Οι ικανοποίηση των οριακών συνθηκών τοίχου είναι πολύ δύσκολη.

Υβριδικοί Επιλυτές

Εξετάζοντας τα χαρακτηριστικά των Eulerian και Lagrangian μεθόδων παρατηρείται ότι οι δύο μέθοδοι αλληλοσυμπληρώνονται. Οι Eulerian μέθοδοι προσομοιώνουν με ακρίβεια την περιοχή κοντά στο στερεό σύνορο, ενώ οι Lagrangian ικανοποιούν ακριβώς τις επ' άπειρο οριακές συνθήκες. Συνεπώς, είναι λογική η προσπάθεια σύζευξης των δύο σε ένα υβριδικό υπολογιστικό εργαλείο. Υβριδικές μέθοδοι που συνδυάζουν Eulerian και Lagrangian επιλυτές έχουν αναπτυχθεί στο παρελθόν [39, 40] αλλά έχουν εφαρμοστεί κυρίως σε ασυμπίεστες ροές. Η πρώτη προσπάθεια υβριδικής μεθόδου για συμπιεστές ροές έγινε στο [41], όπου αναπτύχθηκε ένας υβριδικός επιλυτής Eulerian-Lagrangian για διδιάστατες ροές στην ασθενή συμπιεστή περιοχή.

1.2 Σκοπός της Εργασίας

1.2.1 Η Βασική Ιδέα

Το κίνητρο πίσω από την παρούσα εργασία είναι η ανάπτυξη μια υβριδικής μεθόδου για δυδιάστατες και τριδιάστατες εξωτερικές ροές, μέσω της ισχυρής σύζευξης ενός Eulerian και ενός Lagrangian επιλυτή, με τα εξής χαρακτηριστικά:

- Προσομοίωση ροών σε όλους τους αριθμούς Mach.
- Να είναι τουλάχιστον ίδιας ακρίβειας με τον Eulerian επιλυτή.
- Να έχει μειωμένη αριθμητική διάχυση.
- Να χειρίζεται εύκολα διατάξεις πολλαπλών σωμάτων.
- Να επιτρέπει αερο-ακουστικούς και αερο-ελαστικούς υπολογισμούς.

Η βασική ιδέα του υβριδικού επιλυτή συνίσταται στα εξής:

1. Αντικατάσταση των μεγάλης έκτασης CFD πλεγμάτων με υπολογιστικά πλέγματα μικρότερης έκτασης. Με αυτόν τον τρόπο ικανοποιείται ακριβώς η οριακή συνθήκη τοίχου, ενώ αποφεύγουμε την αριθμητική διάχυση λόγω της σταδιακής αραίωσης του πλέγματος. Επίσης με τον περιορισμό του πλέγματος στην περιοχή κοντά στο σώμα γίνονται εύκολη η προσομοίωση πολλαπλών σωμάτων.

2. Για να χρησιμοποιηθούν πλέγματα μικρότερης έκτασης θα πρέπει να εξασφαλίζεται ότι οι συνθήκες στο εξωτερικό όριο του CFD πλέγματος παρέχονται σωστά. Για αυτό το λόγο:

   - Ο Υπολογιστικός χώρος καλύπτεται από στοιχεία στροβιλότητας όπου οι εξισώσεις της ροής λύνονται σε Lagrangian μορφή.
   - Για να είναι καθορισμένο πλήρως το πεδίο ροής τα στοιχεία στροβιλότητας πρέπει να μεταφέρουν μάζα, μεταβολή του όγκου, στροβιλότητα και ενέργεια.

3. Να είναι εύκολη η χρήση τεχνικών όπως οι τεχνικές αυτόματης πύκνωσης του πλέγματος και multigrid για το CFD κομμάτι.

Θα πρέπει να τονιστεί, ότι ο υβριδικός επιλυτής δεν αποτελεί αντικαταστάτη των Eulerian επιλυτών αλλά αντίθετα η ιδέα είναι να χρησιμοποιηθούν τα στοιχεία στροβιλότητας για να απλοποιηθεί ο υπολογισμός γύρω από πολύπλοκες διατάξεις με χρήση των CFD εργαλείων.

1.2.2 Επι μέρους περιγραφή των Εργαλείων του Υβριδικού Επιλυτή

Για την κατασκευή του υβριδικού επιλυτή χρειάστηκε να αναπτυχθούν τα εξής υπολογιστικά εργαλεία:

Ο Eulerian CFD επιλυτής

Για το CFD μέρος του υβριδικού επιλυτή αναπτύχθηκε ο συμπιεστός κώδικας πεπερασμένων όγκων MaPFlow. Ο CFD επιλυτής λύνει τις μη-μόνιμες συμπιεστές Reynolds-Averaged εξισώσεις Navier-Stokes. Η διακριτοποίηση γίνεται με την μέθοδο πεπερασμένων όγκων πάνω σε μη-δομημένα πλέγματα. Επίσης είναι εφοδιασμένος με μητρώο προσταθεροποίησης για την επίλυση στην ασυμπίεστη περιοχή. Όσο αναφορά τα μοντέλα τύρβης υπάρχουν δύο επιλογές: το μοντέλο τύρβης δύο εξισώσεων $k-\omega$ SST του Menter [42] και το μοντέλο μίας εξίσωσης των Spalart Allmaras [43].

Ο Lagrangian Επιλυτής

Όσο αναφορά τον Lagrangian επιλυτή, προκειμένου να μπορεί να προσομοιώσει ροές σε όλους τους αριθμούς Mach, σε γενίκευση παλαιότερων διατυπώσεων [41, 37] τα στοιχεία στροβιλότητας μεταφέρουν επιπλέον ενέργεια.

Ο Particle Mesh Poisson επιλυτής

Όπως προαναφέρθηκε το κόστος των Lagrangian υπολογισμών είναι μεγάλο καθώς περιγράφεται από ολοκληρωτικές αναπαραστάσεις. Για αυτό το λόγο αναπτύχθηκε ο επιλυτής Particle Mesh για επιτάχυνση των υπολογισμών της Lagrangian μεθόδου. Ο αλγόριθμος που υλοποιήθηκε ήταν αυτός των James-Lackner [44, 45].

Σύζευξη

Ο τρόπος που γίνεται ή σύζευξη των μεθόδων παίζει κυρίαρχο ρόλο στους υβριδικούς επιλυτές. Ο τρόπος που γίνεται η σύζευξη είναι ο εξής:

- Προκειμένου να γίνει η ολοκλήρωση στο χρόνο, ο Eulerian επιλυτής χρειάζεται την λύση στο παρόν χρονικό βήμα και τις σωστές οριακές συνθήκες. Οι οριακές συνθήκες τοίχου ικανοποιούνται ακριβώς, ενώ ο Lagrangian επιλυτής παρέχει τις κατάλληλες συνθήκες στο εξωτερικό όριο του CFD πλέγματος.

- Τα στερεά όρια δεν μοντελοποιούνται στον Lagrangian επιλυτή αλλά μόνο από τον Eulerian. Για αυτόν το λόγο στη διάρκεια της σύζευξης παράγονται από τον CFD επιλυτή στοιχεία στροβιλότητας που αντικαθιστούν τα Lagrangian στο εσωτερικό του υπολογιστικού πλέγματος. Με αυτόν τον τρόπο λαμβάνονται υπόψη οι συνθήκες τοίχου στο Lagrangian κομμάτι.

1.3 Διάρθρωση της Εργασίας

Η παρούσα εργασία χωρίζεται σε πέντε κεφάλαια:

- Στο Κεφάλαιο 1 παρουσιάζεται ο σκοπός της εργασίας και μία σύντομη βιβλιογραφική επισκόπηση

- Στο Κεφάλαιο 2 γίνεται περιγραφή του CFD επιλυτή
- Στο Κεφάλαιο 3 γίνεται περιγραφή του υβριδικού επιλυτή καθώς και των Lagrangian εργαλείων που χρησιμοποιούνται.
- Στο Κεφάλαιο 4 γίνεται πιστοποίηση του CFD επιλυτή καθώς και του υβριδικού επιλυτή με σύγκριση με αριθμητικούς υπολογισμούς αλλά και με πειραματικά δεδομένα.
- Στο Κεφάλαιο 5 παρουσιάζονται τα συμπεράσματα και οι προτάσεις για μελλοντική εργασία

This page intentionally left blank.

Κεφάλαιο 2

MaPFlow

2.1 Εξισώσεις Navier-Stokes

Το σύστημα εξισώσεων ολοκληρωμένο σε όγκο Ω με σύνορο $\partial\Omega$ έχει τη μορφή :

$$\frac{\partial}{\partial t}\int_{\Omega}\vec{U}d\Omega + \oint_{\partial\Omega}(\vec{F_c}dS - \vec{F_v})dS = \int_{\Omega}\vec{Q}d\Omega \tag{2.1}$$

όπου $\vec{U}$,

$$\vec{U} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho E \end{pmatrix} \tag{2.2}$$

το διάνυσμα των συντηρητικών μεταβλητών,
$\vec{F_c}$,

$$\vec{F_c} = \begin{pmatrix} \rho V \\ \rho u V + n_x p \\ \rho v V + n_y p \\ \rho w V + n_z p \\ \rho(E + \frac{p}{\rho})V \end{pmatrix} \tag{2.3}$$

το διάνυσμα των συντηρητικών γενικευμένων παροχών (Convective Fluxes) και $V = \vec{u}\cdot\vec{n}$,
$\vec{F_v}$,

$$\vec{F_v} = \begin{pmatrix} 0 \\ n_x\tau_x x + n_y\tau_x y + n_z\tau_x z \\ n_x\tau_y x + n_y\tau_y y + n_z\tau_y z \\ n_x\tau_z x + n_y\tau_z y + n_z\tau_z z \\ n_x\Theta_x + n_y\Theta_y + n_z\theta_z \end{pmatrix} \tag{2.4}$$

το διάνυσμα των συνεκτικών παροχών (Viscous fluxes) και :

$$\Theta_x = u\tau_{xx} + v\tau_{xy} + w\tau_{xz} + k\frac{\partial T}{\partial x}$$
$$\Theta_y = u\tau_{yx} + v\tau_{yy} + w\tau_{yz} + k\frac{\partial T}{\partial y}$$
$$\Theta_z = u\tau_{zx} + v\tau_{zy} + w\tau_{zz} + k\frac{\partial T}{\partial z}$$

Το σύστημα των εξισώσεων κλείνει με την καταστατική εξίσωση για ιδανικό αέριο:

$$p = (\gamma - 1)\rho[E - \frac{u^2 + v^2 + w^2}{2}] \tag{2.5}$$

2.2 Χωρική διακριτοποίηση

Η χωρική διακριτοποίηση αφορά στην αριθμητική προσέγγιση των συντηρητικών και των συνεκτικών παροχών καθώς και των πηγαίων όρων. Πολλές διαφορετικές μεθοδολογίες έχουν αναπτυχθεί για αυτό το σκοπό και άλλες ακόμα εξελίσσονται. Οι μέθοδοι αυτοί ανάλογα με τον τρόπο που γίνεται η διακριτοποίηση χωρίζονται σε:

- Μεθόδους πεπερασμένων διαφορών (Finite Difference).
- Μεθόδους πεπερασμένων όγκων (Finite Volume).
- Μεθόδους πεπερασμένων στοιχείων (Finite Element).

Επίσης ανάλογα με το πλέγμα στο οποίο γίνεται η επίλυση χωρίζονται σε:

- Μεθόδους δομημένων πλεγμάτων (Stuctured Grid).
- Μεθόδους μη-δομημένων πλεγμάτων (Unstructured Grid).

Για την επίλυση των παραπάνω εξισώσεων επιλέχτηκε η μέθοδος πεπερασμένων όγκων γιατί:

- Η χωρική διακριτοποίηση γίνεται στο φυσικό υπολογιστικό χωρίο.
- Μπορεί να εφαρμοστεί σε δομημένα αλλά και σε μη-δομημένα πλέγματα.

Τέλος οι Μεθόδιο Πεπερασμένων Όγκων χωρίζονται σε :

- Κεντροκυψελικές (Cell-centered), στις οποίες οι μεταβλητές των εξισώσεων υπολογίζονται στα κέντρα των κελιών του πλέγματος. Στην περίπτωση αυτή τα υπολογιστικά κελιά ταυτίζονται με αυτά του πλέγματος.
- Κεντροκομβικές (Vertex-centered), στις οποίες οι μεταβλητές των εξισώσεων υπολογίζονται στους κόμβους του πλέγματος. Σε αυτή την περίπτωση το υπολογιστικό κελί ορίζεται από κάποιον όγκο γύρω απο τον κόμβο του πλέγματος.

Στη παρούσα εργασία υιοθετείται η κεντροκομβική διάταξη όπου τα κελιά ορίζονται από τους κόμβους του πλέγματος ενώ, οι μεταβλητές της ροής υπολογίζονται στα κέντρα των κελιών. Θεωρώντας ότι ο όγκος των κελιών δεν μεταβάλλεται με το χρόνο :

$$\frac{\partial}{\partial t}\int_{\Omega}\vec{U}d\Omega = \Omega\frac{\partial\vec{U}}{\partial t} \tag{2.6}$$

όπου :

$$\vec{U} = \frac{1}{\Omega}\int_{\Omega}\vec{U}_{exact}d\Omega \tag{2.7}$$

Συνεπώς η εξίσωση 2.1 γίνεται :

$$\frac{\partial\vec{U}}{\partial t} = -\frac{1}{\Omega}[\oint_{\partial\Omega}(\vec{F_c}-\vec{F_v})dS - \int_{\Omega}\vec{Q}d\Omega] \tag{2.8}$$

Το επιφανειακό ολοκλήρωμα στην παραπάνω εξίσωση προσεγγίζεται από το άθροισμα των παροχών στις επιφάνειες (faces) που περιγράφουν το εκάστοτε κελί. Συνήθως θεωρείται ότι η παροχή παραμένει σταθερή πάνω σε μιά επιφάνεια και υπολογίζεται στο κέντρο της.

Η εξίσωση 2.8 για ένα κελί I γράφεται :

$$\frac{d\vec{U}_I}{dt} = -\frac{1}{\Omega_I}\underbrace{[\sum_{m=1}^{N_f}(\vec{F_c}-\vec{F_v})_m\Delta S_m) - (\vec{Q}\Omega)_I]}_{R_I} \tag{2.9}$$

όπου N_f είναι ο αριθμός των επιφανείων που περιγράφουν το κελί και ΔS_m είναι η επιφάνεια της πλευράς "m". Ο όρος R_I ονομάζεται υπόλοιπο (residual) με αποτέλεσμα η τελική διακριτοποιημένη εξίσωση να γράφεται :

$$\frac{d\vec{U}_I}{dt} = -\frac{1}{\Omega_I}\vec{R}_I \tag{2.10}$$

2.3 Προσέγγιση των μεταβλητών στα σύνορα των κελιών

Η διακριτοποίηση των συντηρητικών παροχών (convective fluxes) αφορά στην διαδικασία εύρεσης των παροχών στα σύνορα των κελιών. Όπως αναφέρθηκε στα Cell-centered σχήματα οι τιμές των συντηρητικών μεταβλητών $(\rho, \rho\vec{U}, \rho E)$, είναι γνωστές στα κέντρα των κελιών. Για να υπολογιστούν οι παροχές στα σύνορα των κελιών χρειάζεται να προσεγγίσουμε είτε τις τιμές των συντηρητικών μεταβλητών εκεί, είτε τις τιμές των παροχών απευθείας.

Στη μέθοδο που αναπτύχθηκε υπολογίζονται οι τιμές των μεταβλητών (πρωτογενών) στα σύνορα και στη συνέχεια υπολογίζονται οι παροχές. Αυτή η διαδικασία ονομάζεται ανακατασκευή των μεταβλητών (reconstruction).

Για να υπολογιστεί η τιμή στις επιφάνειες των κελιών χρησιμοποιούνται τα λεγόμενα left και right states. Η χρήση αυτών έγκειται στο γεγονός ότι η παρεμβολή των μεταβλητών πάνω σε μία συγκεκριμένη επιφάνεια του κελιού γίνεται 2 φορές: μία από τα αριστερά και μία από τα δεξιά της επιφάνειας και στη συνέχεια υπολογίζεται η παροχή διαμέσου της επιφάνειας.

Στη μέθοδο αυτή γίνεται η υπόθεση ότι η λύση είναι τμηματικά, γραμμικά κατανεμημένη στο πεπερασμένο όγκο. Τα left και right states υπολογίζονται ως εξής:

$$\vec{W}_L = \vec{W}_I + \Psi_I(\nabla\vec{W}_I \cdot \ \vec{r}_L) \tag{2.11}$$

$$\vec{W}_R = \vec{W}_J - \Psi_J(\nabla\vec{W}_J \cdot \ \vec{r}_R) \tag{2.12}$$

οπού $\vec{W}$ είναι το διάνυσμα των πρωτογενών μεταβλητών:

$$\vec{W} = \begin{pmatrix} \rho \\ u \\ v \\ w \\ E \end{pmatrix} \tag{2.13}$$

και $\vec{r}_L, \vec{r}_R$ είναι οι αποστάσεις των κέντρων των κελιών από το κέντρο της κοινής επιφάνειας.

Σημαντικός παράγοντας στο παραπάνω σχήμα είναι ο υπολογισμός του $\nabla\vec{W}_I$ στις εξισώσεις 2.11,2.12. Ο υπολογισμός της παραγώγου γίνεται με την προσέγγιση Green-Gauss. Η παράγωγος προσεγγίζεται από ένα επιφανειακό ολοκλήρωμα :

$$\nabla\vec{W} \approx \frac{1}{\Omega}\int_{\partial\Omega} \vec{W}\vec{n}dS \tag{2.14}$$

Στα Cell-centered σχήματα η παραπάνω εξίσωση διακριτοποιείται ως εξής:

$$\nabla\vec{W}_I \approx \frac{1}{\Omega}\sum_{J=1}^{N_f}\frac{1}{2}(\vec{W}_I + \vec{W}_J)\vec{n}_{IJ}\Delta S_{IJ} \tag{2.15}$$

Η συνάρτηση Ψ είναι μια συνάρτηση που αποτρέπει τις μεταβλητές να πάρουν ακραίες τιμές εκεί που εμφανίζονται ασυνέχειες. Αυτές οι συναρτήσεις ονομάζονται "περιοριστές" (limiters).Στη συγκεκριμένη περίπτωση εφαρμόστηκε ο limiter του Venkatakrishnan [46], ο οποίος εφαρμόζεται στο ∇U και υλοποιείται ως εξής :

$$\Psi_i = min_j \begin{cases} \frac{1}{\Delta_2}[\frac{\Delta_{1,max}^2+\epsilon^2)\Delta_2+2\Delta_2^2\Delta_{1,max}}{\Delta_{1,max}^2+2\Delta_2^2+\Delta_{1,max}\Delta_2+\epsilon^2}] & \text{αν } \Delta_2 > 0 \\ \frac{1}{\Delta_2}[\frac{\Delta_{1,min}^2+\epsilon^2)\Delta_2+2\Delta_2^2\Delta_{1,min}}{\Delta_{1,min}^2+2\Delta_2^2+\Delta_{1,min}\Delta_2+\epsilon^2}] & \text{αν } \Delta_2 < 0 \\ 1 & \text{αν } \Delta_2 = 0 \end{cases} \tag{2.16}$$

όπου

$$\Delta_{1,max} = U_{max} - U_i \tag{2.17}$$

$$\Delta_{1,min} = U_{min} - U_i \tag{2.18}$$

Η παράμετρος ϵ^2 έχει στόχο να καθορίσει την αυστηρότητα του limiter. Στην περίπτωση που τεθεί 0, ο limiter γίνεται πολύ αυστηρός, ενώ αν πάρει μεγάλες τιμές ο limiter επιστρέφει τιμές κοντά στη μονάδα (οπότε και δεν αλλάζει πολύ την τιμή των μεταβλήτων). Στην πράξη το ϵ ειναι ανάλογο της κλίμακας μήκους του πλέγματος.

$$\epsilon^2 = (K\Delta h)^3 \tag{2.19}$$

όπου το είναι ελεύθερη παράμετρος.

Δοκιμάστηκε και ο limiter των Barth και Jespersen [47], αλλά προτιμήθηκε ο πρώτος αφού ο δεύτερος παρουσιάζει μεγαλύτερη αριθμητική διάχυση.

2.4 Υπολογισμός των συντηρητικών παροχών

Ο υπολογισμός των συντηρητικών παροχών γίνεται σύμφωνα με το σχήμα του Roe.
Καταρχάς υπολογίζονται οι κατά Roe μεταβλητές:

$$\begin{gathered}
\tilde{\rho} = \sqrt{\rho_L \rho_R} \\
\tilde{u} = \frac{u_L\sqrt{\rho_L}+u_R\sqrt{\rho_R}}{\sqrt{\rho_L}+\sqrt{\rho_R}} \\
\tilde{v} = \frac{v_L\sqrt{\rho_L}+v_R\sqrt{\rho_R}}{\sqrt{\rho_L}+\sqrt{\rho_R}} \\
\tilde{w} = \frac{w_L\sqrt{\rho_L}+w_R\sqrt{\rho_R}}{\sqrt{\rho_L}+\sqrt{\rho_R}} \\
\tilde{H} = \frac{H_L\sqrt{\rho_L}+H_R\sqrt{\rho_R}}{\sqrt{\rho_L}+\sqrt{\rho_R}} \\
\tilde{c} = \sqrt{(\gamma-1)(\tilde{H}-\tilde{q}^2/2)} \\
\tilde{q} = \tilde{u}^2 + \tilde{v}^2 + \tilde{w}^2
\end{gathered}$$

Η παροχή στην επιφάνεια $I + \frac{1}{2}$ ορίζεται ως:

$$(\vec{F_c})_{I+\frac{1}{2}} = \frac{1}{2}[\vec{F_c}(\vec{U_R}) + \vec{F_c}(\vec{U_L}) - |A_{Roe}|_{I+\frac{1}{2}}(\vec{U_R} - \vec{U_L})] \tag{2.20}$$

όπου:

$$|A_{Roe}|_{I+\frac{1}{2}}(\vec{U_R} - \vec{U_L}) = |\Delta\vec{F_1}| + |\Delta\vec{F}_{2,3,4}| + |\Delta\vec{F_5}| \tag{2.21}$$

$$|\Delta\vec{F_1}| = |\tilde{V} - \tilde{c}|(\frac{\Delta p - \tilde{\rho}\tilde{c}\Delta V}{2\tilde{c}^2}) \begin{pmatrix} 1 \\ \tilde{u} - \tilde{c}n_x \\ \tilde{v} - \tilde{c}n_y \\ \tilde{w} - \tilde{c}n_z \\ \tilde{H} - \tilde{c}\tilde{V} \end{pmatrix} \tag{2.22}$$

$$|\Delta\vec{F}_{2,3,4}| = |\tilde{V}| \left\{ (\Delta\rho - \frac{\Delta p}{\tilde{c}^2}) \begin{pmatrix} 1 \\ \tilde{u} \\ \tilde{v} \\ \tilde{w} \\ \tilde{q}^2/2 \end{pmatrix} + \tilde{\rho} \begin{pmatrix} 0 \\ \Delta u - \Delta V n_x \\ \Delta v - \Delta V n_y \\ \Delta w - \Delta V n_z \\ \tilde{u}\Delta u + \tilde{v}\Delta v + \tilde{w}\Delta w - \tilde{V}\Delta V \end{pmatrix} \right\} \tag{2.23}$$

$$|\Delta\vec{F}_5| = |\tilde{V} + \tilde{c}|(\frac{\Delta p + \tilde{\rho}\tilde{c}\Delta V}{2\tilde{c}^2}) \begin{pmatrix} 1 \\ \tilde{u} + \tilde{c}n_x \\ \tilde{v} + \tilde{c}n_y \\ \tilde{w} + \tilde{c}n_z \\ \tilde{H} + \tilde{c}\tilde{V} \end{pmatrix} \tag{2.24}$$

$$\Delta(\cdot) = (\cdot)_R - (\cdot)_L \tag{2.25}$$

Η παραπάνω διατύπωση επιτρέπει λύσεις που δεν είναι φυσικά σωστές και γι'αυτό εισάγεται η διόρθωση εντροπίας (entropy correction) του Harten [47] στις ιδιοτιμές του συστήματος:

$$|\Lambda_c| = \begin{cases} |\Lambda_c| & \text{αν } |\Lambda_c| > \delta \\ \frac{\Lambda_c^2+\delta^2}{2\delta} & \text{αν } |\Lambda_c| \leq \delta \end{cases} \tag{2.26}$$

όπου το $\delta = \frac{1}{10}c$

2.5 Συνοριακές συνθήκες

Οι Συνοριακές συνθήκες χωρίζονται σε:

- Συνοριακές συνθήκες τοίχου (Solid Wall)
- Συνοριακές συνθήκες εισόδου/εξόδου (Farfield inflow/outflow)
- Συνοριακές συνθήκες συμμετρίας (Farfield inflow/outflow)
- Συνοριακές συνθήκες ανάμεσα σε blocks (Multiblock boundary conditions)

Για την εφαρμογή των συνοριακών συνθηκών χρησιμοποιούνται φανταστικά κελιά (dummy cells) τα οποία μεγαλώνουν το υπολογιστικό χωρίο, ώστε να γίνεται πιο εύκολα ο υπολογισμός των μεγεθών στα σύνορα.

2.5.1 Συνοριακές συνθήκες τοίχου(Solid Wall)

Οι συνοριακές συνθήκες τοίχου έχουν διαφορετική έκφραση στην περίπτωση του συνεκτικού και του μη-συνεκτικού ρευστού.

Στην περίπτωση του μή συνεκτικού ρευστού οι συνοριακές συνθήκες τοίχου εκφράζονται από τη συνθήκη μη εισχώρησης:

$$V = \vec{u} \cdot \vec{n} = 0 \text{ στον τοίχο} \tag{2.27}$$

Συνεπώς το διάνυσμα των συντηρητικών παροχών στον τοίχο γίνεται:

$$(\vec{F_c})_w = \begin{pmatrix} 0 \\ n_x p_w \\ n_y p_w \\ n_z p_w \\ 0 \end{pmatrix} \tag{2.28}$$

όπου p_w είναι η πίεση στον τοίχο.

Στην περίπτωση του συνεκτικού ρευστού η ταχύτητα στον τοίχο είναι μηδενική (συνοριακή συνθήκη μη-ολίσθησης). Συνεπώς η συνοριακή συνθήκη στον τοίχο είναι:

$$u = v = w = 0 \text{ στον τοίχο} \tag{2.29}$$

2.5.2 Συνοριακές συνθήκες εισόδου/εξόδου

Ανάλογα με το πρόσημο των ιδιοτιμών των συντηρητικών παροχών, η πληροφορία μεταφέρεται από ή προς τα έξω απο το υπολογιστικό χωρίο πάνω στις χαρακτηριστικές. Ο αριθμός των επιβαλλόμενων συνθηκών στο σύνορο πρέπει να είναι ίδιος με τον αριθμό των χαρακτηριστικών που εισέρχονται στο υπολογιστικό χωρίο. Οι υπόλοιπες συνθήκες υπολογίζονται απο την υπάρχουσα λύση στο χωρίο.

Η ροή μπορεί είτε να εισέρχεται είτε να εξέρχεται απο το υπολογιστικό χωρίο. Συνεπώς, ανάλογα με τον τοπικό αριθμό Mach, προκύπτουν 4 τύποι συνοριακών συνθηκών:

- Υπερηχητική είσοδος(Supersonic inflow)
- Υπερηχητική έξοδος(Supersonic ouflow)
- Υποηχητική είσοδος(Subsonic inflow)
- Υποηχητική έξοδος (Subsonic ouflow)

Υπερηχητική είσοδος : Σε αυτήν την περίπτωση όλες οι ιδιοτιμές είναι θετικές και άρα όλη η πληροφορία εισέρχονται στο χωρίο. Συνεπώς οι συντηρητικές μεταβλητές στο σύνορο υπολογίζονται μόνο από την επερχόμενη ροή.

$$\vec{U}_{boundary} = \vec{U}_{freestream}$$

Υπερηχητική έξοδος : Σε αυτήν την περίπτωση όλες οι ιδιοτιμές έχουν το ίδιο πρόσημο και εξέρχονται απο το χωρίο. Συνεπώς οι συντηρητικές μεταβλητές στο σύνορο υπολογίζονται προεκβάλλοντας την υπάρχουσα λύση στο χωρίο.

$$\vec{U}_{boundary} = \vec{U}_{computational}$$

Υποηχητική είσοδος : Σε αυτήν την περίπτωση, τέσσερις χαρακτηριστικές εισέρχονται στο χωρίο και μία εξέρχεται. Συνεπώς μια χαρακτηριστική μεταβλητή υπολογίζεται στο σύνορο από το εσωτερικό του χωρίου.

$$\begin{aligned}
p_b &= \frac{1}{2}\{p_a + p_d - \rho_0 c_0[n_x(u_a - u_d) + n_y(v_a - v_d) + n_z(w_a - w_d)\} \\
\rho_b &= \rho_a + (p_b - p_a)/c_0^2 \\
u_b &= u_a - n_x(p_a - p_b)/(\rho_0 c_0) \\
v_b &= v_a - n_y(p_a - p_b)/(\rho_0 c_0) \\
w_b &= w_a - n_z(p_a - p_b)/(\rho_0 c_0)
\end{aligned}$$

όπου το ρ_0, c_0 αναφέρονται στο εσωτερικό του χωρίου.

Υποηχητική έξοδος : Σε αυτην την περίπτωση, τέσσερις χαρακτηριστικές εξέρχονται από το χωρίο και μία εισέρχεται σε αυτο. Συνεπώς τέσσερις μεταβλητές υπολογίζονται από το εσωτερικό του χωρίου και μία (συνήθως η πίεση) επιβάλλεται εξωτερικά:

$$\begin{aligned}
p_b &= p_a \\
\rho_b &= \rho_d + (p_b - p_d)/c_0^2 \\
u_b &= u_d + n_x(p_d - p_b)/(\rho_0 c_0) \\
v_b &= v_d + n_y(p_d - p_b)/(\rho_0 c_0) \\
w_b &= w_d + n_z(p_d - p_b)/(\rho_0 c_0)
\end{aligned}$$

Οι παραπάνω συνοριακές συνθήκες υποθέτουν μηδενική κυκλοφορία το οποίο είναι λάθος για ένα ανωστικό σώμα. Γι αυτό το λόγο το εξωτερικό όριο πρέπει να βρίσκεται μακριά απο το σώμα. Η απόσταση του εξωτερικού ορίου μπορεί να μειωθεί σημαντικά, αν η επερχόμενη ροή περιστραφεί αναλογικά με την κυκλοφορία. Αυτή η διόρθωση ονομάζεται vortex correction.

2.6 Χρονική διακριτοποίηση

Για τη χρονική διακριτοποίηση εφαρμόζουμε τη μέθοδο των γραμμών (method of lines). Αυτό σημαίνει ότι ξεχωρίζουμε τη χωρική από τη χρονική διακριτοποίση

οπότε καταλήγουμε σ'ένα σύστημα πεπλεγμένων διαφορικών εξισώσεων στο χρόνο για κάθε κελί :

$$\frac{d(\Omega \bar{M} \vec{U})_i}{dt} = -\vec{R}_i \tag{2.30}$$

Το παραπάνω σύστημα διαφορικών εξισώσεων πρέπει να ολοκληρωθεί στο χρόνο είτε για να πάρουμε τη λύση (στα μόνιμα προβλήματα) είτε για να αναπαράγουμε τη χρονική ιστορία στα μή μόνιμα προβλήματα. Για μη μεταβαλλόμενα πλέγματα η παραπάνω εξίσωση γράφεται:

$$\frac{(\Omega \bar{M})_I}{\Delta t_I} \Delta \vec{U}_I^n = \frac{\beta}{1+\omega} \vec{R}_I^{n+1} - \frac{1-\beta}{1+\omega} \vec{R}_I^n + \frac{\omega}{1+\omega} \frac{(\Omega \bar{M})_i}{\Delta t_I} \Delta \vec{U}_i^{n-1} \tag{2.31}$$

όπου:

$$\Delta \vec{U}_I^n = \vec{U}_I^{n+1} - \vec{U}_I^n \tag{2.32}$$

είναι η διόρθωση της λύσης. Το μητρώο $\bar{M}$ ειναι το λεγομένο μητρώο μάζας(mass matrix), το οποίο για τα προβλήματα που εξετάζονται είναι μοναδιαίο.

Ο καθορισμός του χρονικού βήματος είναι σημαντικός παράγοντας για την ευστάθεια του προβλήματος. Το μέγιστο χρονικό βήμα για κάθε κελί καθορίζεται από την παρακάτω σχέση :

$$\Delta t_I = \sigma \frac{\Omega}{(\hat{\Lambda}_c + 4 \cdot \hat{\Lambda}_v)_I} \tag{2.33}$$

όπου τα $\hat{\Lambda}_c, \hat{\Lambda}_v$ είναι ένα άθροισμα των συντηρητικών και των συνεκτικών ιδιοτιμών των επιφανειών που αποτελούν το κελί. Αυτά ορίζονται σαν:

$$(\hat{\Lambda}_c)_I = \sum_{J=1}^{N_f} (|\vec{u}_{IJ} \cdot \vec{n}_{IJ}| + c_i j) \Delta S_{IJ} \tag{2.34}$$

$$(\hat{\Lambda}_v)_I = \frac{1}{\Omega_I} \sum_{J=1}^{N_f} [max(\frac{3}{3\rho_{IJ}}, \frac{\gamma_{IJ}}{\rho_{IJ}})(\frac{\mu_L}{Pr_L} + \frac{\mu_T}{Pr_T})_{IJ} (\Delta S_{IJ})^2] \tag{2.35}$$

2.6.1 Άρρητη Ολοκλήρωση στο χρόνο (Implicit Time-Stepping)

Υπάρχουν αρκετά σχήματα άρρητης ολοκλήρωσης στο χρόνο. Ένα άρρητο σχήμα προκύπτει βάζοντας $\Omega = 0$ στην εξίσωση (2.31) και $\beta = 1$:

$$\frac{(\Omega \bar{M})_I}{\Delta t)_I} \Delta \vec{U}_I^n = -\vec{R}_I^{n+1} - \vec{R}_I^n \tag{2.36}$$

Όπως φαίνεται και παραπάνω χρειάζονται οι τιμές των μεταβλητών στη χρονική στιγμή $(n+1)$. Για να υπολογιστεί το υπόλοιπο σε αυτή τη χρονική στιγμή, γραμμικοποιείται γύρω από την τρέχουσα χρονική στιγμή:

$$\vec{R}_I^{n+1} \approx \vec{R}_I^n + (\frac{\partial \vec{R}}{\partial \vec{U}})_I \Delta \vec{U}^n \tag{2.37}$$

όπου ο όρος $(\frac{\partial \vec{R}}{\partial \vec{U}})_I$ είναι η ιακωβιανή των παροχών.

Αν εισάγουμε την ιακωβιανή των παροχών στην (2.36) και για $\bar{M} = 1$ παίρνουμε το ακόλουθο σχήμα :

$$\underbrace{[\frac{(\Omega)_I}{\Delta t_I} + \left(\frac{\partial \vec{R}}{\partial \vec{U}}\right)_I]}_{\text{implicit operator}} \Delta \vec{U}^n = -\vec{R}_I^n \tag{2.38}$$

2.6.2 Άρρητος τελεστής (Implicit Operator)

Η ιακωβιανή των παροχών γράφεται ώς:

$$\frac{\partial \vec{R}}{\partial \vec{U}} = \sum_{m=1}^{N_f} \frac{\partial (\vec{F}_c)_m}{\partial \vec{U}} \Delta S_m - \sum_{m=1}^{N_f} \frac{\partial (\vec{F}_v)_m}{\partial \vec{U}} \Delta S_m - \frac{\partial (\Omega \vec{Q})}{\partial \vec{U}} \tag{2.39}$$

Είναι χαρακτηριστικό ότι οι παροχές που αναφέρονται στην παραπάνω εξίσωση δεν είναι αναγκαστικά ίδιες με αυτές που χρησιμοποιήθηκαν στη χωρική διακριτοποίηση αφού η ιακωβιανή επηρεάζει μόνο τη διόρθωση της λύσης.

Χρησιμοποιώντας τις παροχές κατά Roe και βάζοντας την ιακωβιανή των συνεκτικών παροχών μηδέν προκύπτει:

$$\begin{aligned} \frac{\partial \vec{R}}{\partial \vec{U}} \Delta \vec{U}^n = & \sum_{m=1}^{N_f} \Delta S_m \{ (\bar{A}_c)_{L,m} \Delta \vec{W}_{L,m}^n + (\bar{A}_c)_{R,m} \Delta \vec{W}_{R,m}^n \\ & - \frac{\partial}{\partial \vec{W}_{L,m}} [|\tilde{A}_{Roe}|_m (\vec{W}_{R,m}^n - \vec{W}_{L,m}^n)] \Delta \vec{W}_{L,m}^n \\ & - \frac{\partial}{\partial \vec{W}_{R,m}} [|\tilde{A}_{Roe}|_m (\vec{W}_{R,m}^n - \vec{W}_{L,m}^n)] \Delta \vec{W}_{R,m}^n \} \end{aligned} \tag{2.40}$$

αν υποθέσουμε οτι τοπικά οι ιακωβιανές κατα Roe είναι σταθερές, τότε :

$$\begin{aligned} \frac{\partial \vec{R}}{\partial \vec{U}} \Delta \vec{U}^n \approx & \sum_{m=1}^{N_f} \Delta S_m \{ (\bar{A}_c)_{L,m} \Delta \vec{W}_{L,m}^n + (\bar{A}_c)_{R,m} \Delta \vec{W}_{R,m}^n \\ & - |\tilde{A}_{Roe}|_m (\Delta \vec{W}_{R,m}^n - \Delta \vec{W}_{L,m}^n)] \} \end{aligned} \tag{2.41}$$

2.6.3 Μη-μόνιμες Ροές

Για τον υπολογισμό μη-μόνιμων ροών χρησιμοποιείται η τεχνική dual time-stepping. Για $\beta = 1$ και $\omega = 1/2$ η εξίσωση (2.31) γίνεται:

$$\frac{3\Omega_I^{n+1}\vec{U}_I^{n+1} - 4\Omega_I^n\vec{U}_I^n + \Omega_I^{n-1}\vec{U}_I^{n-1}}{2\Delta t} = -\vec{R}_I^{n+1} \tag{2.42}$$

όπου το Δt είναι το φυσικό χρονικό βήμα. Για να λύσουμε το παραπάνω σύστημα εξισώσεων εισάγεται η έννοια του ψευδοχρόνου. Στην πραγματικότητα το μη-μόνιμο πρόβλημα στον πραγματικό χρόνο μετατρέπεται σε ένα μόνιμο πρόβλημα στο ψευδοχρόνο. Κάνοντας την υπόθεση ότι:

$$\frac{\partial}{\partial t^*}(\Omega_I^{n+1}\vec{U}^*) = -\vec{R}_I^*(\vec{U}^*) \tag{2.43}$$

όπου $\vec{U}^*$ είναι η προσέγγιση του $\vec{U}^{n+1}$ και t^* είναι η μεταβλητή του ψευδοχρόνου. Το μη-μόνιμο υπόλοιπο ορίζεται ώς:

$$\vec{R}_I^*(\vec{U}^*) = \vec{R}_I^*(\vec{U}^*) + \frac{3}{2\Delta t}\Omega_I^{n+1}\vec{W}_I^* - \vec{Q}_I^* \tag{2.44}$$

Όλοι οι όροι που παραμένουν σταθεροί κατά τη διάρκεια των ψευδοχρονικών βημάτων τοποθετούνται στον όρο πηγών:

$$\vec{Q}_I^* = \frac{2}{\Delta t}\Omega_I^n\vec{U}_I^n - \frac{1}{2\Delta t}\Omega_I^{n-1}\vec{U}_I^{n-1} \tag{2.45}$$

Όταν η λύση στον ψευδοχρόνο συγκλίνει, τότε το υπόλοιπο $\vec{R}_I^*$ θα είναι μηδέν και $\vec{U}_I^* = \vec{U}_I^{n+1}$. Συνεπώς θα ικανοποιείται η αρχική εξίσωση.

Πιο συγκεκριμένα στην περίπτωση των άρρητων μεθόδων ολοκλήρωσης στο χρόνο η παραπάνω μέθοδος εφαρμόζεται ως εξής:

$$\frac{\partial}{\partial t^*}(\Omega_I^{n+1}\vec{U}^*) = -(\vec{R}_I^*)^{l+1} \tag{2.46}$$

όπου $(l+1)$ είναι το νέο ψευδοχρονικό βήμα. Ύστερα γίνεται γραμμικοποίηση του μη-μόνιμου υπόλοιπου στο ψευδοχρόνο:

$$(\vec{R}^*)^{l+1} \approx (\vec{R}^*)^l + \frac{\partial\vec{R}^*}{\partial\vec{U}^*}\Delta\vec{U}^* \tag{2.47}$$

με $\Delta\vec{U}^* = (\vec{U}^*)^{l+1} - (\vec{U}^*)^l$, με την ιακωβιανή των παροχών να ορίζεται ώς:

$$\frac{\partial\vec{R}^*}{\partial\vec{U}^*} = \frac{\partial\vec{R}}{\partial\vec{U}} + \frac{3}{2\Delta t}\Omega^{n+1} \tag{2.48}$$

Με την εισαγωγή της γραμμικοποίησης στην αρχική εξίσωση παίρνουμε το άρρητο αριθμητικό σχήμα :

$$\left[(\frac{1}{\Delta t_I^*} + \frac{3}{2\Delta t})\Omega_I^{n+1} + \frac{\partial \vec{R}}{\partial \vec{U}}\right] \Delta \vec{U}^* == -(R_I^*)^l \tag{2.49}$$

όπου το Δt_I^* αναφέρεται στον ψευδοχρόνο (το οποίο διαφέρει από κελί σε κελί) και το Δt αναφέρεται στο πραγματικό χρονικό βήμα.

1

Κεφάλαιο 3

Ο Lagrangian και Ο Υβριδικός Επιλυτής

3.1 Ο Lagrangian επιλυτής

3.1.1 Στοιχεία στροβιλότητας

Ο Lagrangian επιλυτής χρησιμοποιεί στοιχεία στροβιλότητας για να προσομοιώσει τη ροή. Τα στοιχεία στροβιλότητας έχουν μάζα M_p, καταλαμβάνουν όγκο $V_p(t)$ και χαρακτηρίζονται από τη θέση $\vec{Z}_p(t)$. Εκτός από τη μάζα τα στοιχεία στροβιλότητας μεταφέρουν ορμή και ενέργεια $\Pi_p(t)$. Χρησιμοποιώντας το διαχωρισμό ταχύτητας του Helmholtz , το πεδίο ταχύτητας και συνεπώς η ορμή εκφράζεται μέσω της μεταβολής του όγκου $\Theta_p(t)$ και της στροβιλότητας $\vec{\Omega}_p(t)$. Τα μεγέθη που μεταφέρουν τα στοιχεία στροβιλότητας ορίζονται σαν χωρικά ολοκληρώματα. Συνεπώς:

$$M_p = \int_{V_p(t)} \rho(\vec{x},t) dV_p(t) \; = \rho_p(t)\, V_p(t) \tag{3.1}$$

$$\Theta_p(t) = \int_{V_p(t)} \theta(\vec{x},t) dV_p(t) \; = \theta_p(t)\, V_p(t) \; = (\nabla \vec{u})_p\, V_p \tag{3.2}$$

$$\vec{\Omega}_p(t) = \int_{V_p(t)} \vec{\omega}(\vec{x},t) dV_p(t) = \vec{\omega}_p(t) V_p(t) \; = (\nabla \times \vec{u})_p\; V_p(t) \tag{3.3}$$

$$\Pi_p(t) = \int_{V_p(t)} \varepsilon(\vec{x},t) dV_p(t) \; = \varepsilon_p(t)\, V_p(t) \; = (\rho E)_p\,(t) V_p(t) \tag{3.4}$$

στα οποία, $\rho(\vec{x},t)$, $\theta(\vec{x},t)$, $\omega(\vec{x},t)$, $\varepsilon(\vec{x},t)$, στο συνεχές πεδίο της ροής.

Η πληροφορία μπορεί να μεταφερθεί από τα στοιχεία στροβιλότητας στο συνεχές

πεδίο χρησιμοποιώντας συνελίξεις[13]:

$$\rho(\vec{x},t) = \sum_p \rho_p(t)V_p(t)\zeta_\epsilon(\vec{x}-\vec{Z_p}(t)) \tag{3.5}$$

$$\theta(\vec{x},t) = \sum_p \theta_p(t)V_p(t)\zeta_\epsilon(\vec{x}-\vec{Z_p}(t)) \tag{3.6}$$

$$\vec{\omega}(\vec{x},t) = \sum_p \vec{\omega_p}(t)V_p(t)\zeta_\epsilon(\vec{x}-\vec{Z_p}(t)) \tag{3.7}$$

$$\varepsilon(\vec{x},t) = \sum_p \varepsilon_p(t)V_p(t)\zeta_\epsilon(\vec{x}-\vec{Z_p}(t)) \tag{3.8}$$

όπου ή ζ_ϵ είναι μία συνάρτηση κατανομής.

3.1.2 Lagrangian Διατύπωση των Εξισώσεων ροής

Στην Lagrangian εκδοχή οι ατριβείς εξισώσεις ροής παίρνουν την εξής μορφή:

$$\text{Διατήρηση Μάζας: } \frac{DM_p}{Dt} = 0 \tag{3.9}$$

$$\text{Διατήρηση Ορμής: } \frac{D\vec{\Omega}_p}{Dt} = V_p\left((\vec{\omega}\cdot\nabla)\vec{u} + \frac{1}{\rho^2}\nabla\rho\times\nabla p\right)_p \tag{3.10}$$

$$\frac{D\Theta_p}{Dt} = V_p\left(2\|\nabla\vec{u}\| - \nabla\cdot\frac{\nabla p}{\rho}\right)_p \tag{3.11}$$

$$\text{Διατήρηση Ενέργειας: } \frac{D\Pi_p}{Dt} = -V_p\nabla\left(\vec{u}p\right) \tag{3.12}$$

όπου $(\cdot)_p$ αντιστοιχεί σε υπολογισμό στη θέση του στοιχείου στροβιλότητας $\vec{Z_p}$.

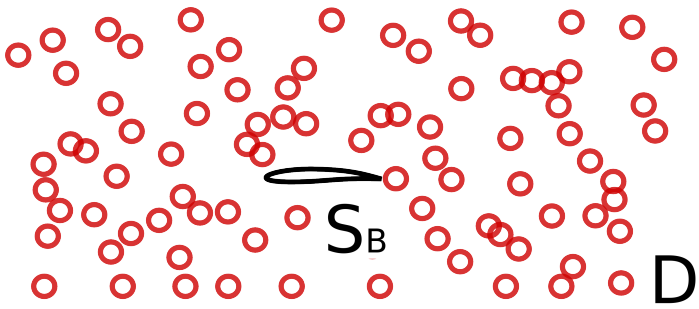
Το παραπάνω σύστημα κλείνει με τον διαχωρισμό ταχυτήτων του Helmholtz:

$$\vec{u} = \vec{U}_\infty + \vec{u}_\phi + \vec{u}_\omega, \quad \vec{u}_\phi = \nabla\phi, \quad \vec{u}_\omega = \nabla\times\vec{\psi} \tag{3.13}$$

Σύμφωνα με τον διαχωρισμό, το πεδίο ταχύτητας αποτελείται από ένα αστρόβιλο και ένα στροβιλό πεδίο. Το αστρόβιλο πεδίο αναπαρίσταται από το βαθμωτό δυναμικό ϕ ενώ το στροβιλό από το διανυσματικό δυναμικό $\vec{psi}$.

Χρησιμοποιώντας το στροβιλισμό ($\nabla\times$) και την απόκλιση της (3.13), ο υπολογισμός των ταχυτήτων καταλήγει στην επίλυση δύο εξισώσεων Poisson. Η επίλυση των εξισώσεων αυτών γίνεται με τη χρήση της τεχνικής Particle Mesh (PM)[48, 49].

$$\begin{aligned}\nabla^2\phi &= \theta\\ \nabla^2\vec{\psi} &= -\vec{\omega}\end{aligned} \tag{3.14}$$



Σχήμα 3.1: Το πεδίο ροής στην περίπτωση μιας εξωτερικής ροής γύρω από αεροτομή. Οι κόκκινοι κύκλοι αντιπροσωπεύουν τα στοιχεία στροβιλότητας.

Με S συμβολίζονται τα σύνορα του πεδίου ροής D. Χρησιμοποιώντας τις συναρτήσεις Green, μπορούμε να ανακτήσουμε τις ολοκληρωτικές αναπαραστάσεις για τον υπολογισμό του πεδίου ταχύτητας $\vec{u}$.

$$\vec{u}(\vec{x}) = \vec{U}_\infty + \int_D (\theta(\vec{y})\,\vec{K}(\vec{r}) + \vec{\omega}(\vec{y}) \times \vec{K}(\vec{r}))\,dD(\vec{y}) + \int_S (u_n(\vec{y})\,\vec{K}(\vec{r}) + \vec{u_\tau}(\vec{y}) \times \vec{K}(\vec{r}))\,dS(\vec{y}) \tag{3.15}$$

Στην παραπάνω έκφραση , u_n, $\vec{u}_\tau$ συμβολίζουν στην κάθετη και εφαπτομενική συνιστώσα της ταχύτητας διαταραχής πάνω στο σύνορο S και με $\vec{K}(\vec{r}) = \nabla G(\vec{r})$ η παράγωγος της συνάρτησης Green ενώ με $\vec{r} = \vec{x} - \vec{y}$ συμβολίζεται η διανυσματική απόσταση.

Στην περίπτωση που δεν υπάρχουν στερεά σύνορα, ο υπολογισμός των ταχυτήτων γίνεται με βάση την ολοκληρωτική αναπαράσταση (3.15) και στη συνέχεια οι εξισώσεις(3.9) ολοκληρώνονται στο χρόνο. Στην περίπτωση που υπάρχουν στερεά σύνορα, το πήδημα της ταχύτητας στο σύνορο θα πρέπει να ληφθεί υπόψη στον επιφανειακό όρο της αναπαράστασης (3.15)[50, 51].

Το υπολογιστικό κόστος της μεθόδου κυριαρχείται από τον υπολογισμό των ολοκληρωτικών αναπαραστάσεων. Για N στοιχεία στροβιλότητας το υπολογιστικό κόστος είναι ανάλογο του N^2. Στην περίπτωση που υπάρχουν στερεά σώματα το κόστος μεγαλώνει ακόμα περισσότερο καθώς θα πρέπει να υπολογιστεί επιπλέον και ο επιφανειακός όρος. Προκειμένου να μειωθεί το υπολογιστικό κόστος υιοθετήθηκε η τεχνική Particle Mesh (PM).

Η βασική ιδέα πίσω από το Particle Mesh είναι η επίλυση των Poisson εξισώσεων πάνω σε ένα Καρτεσιανό πλέγμα με τη χρήση Fast Poisson Solvers [49]. Με αυτόν τον τρόπο το υπολογιστικό κόστος μειώνεται από N^2 σε $logN$.

3.1.3 Συναρτήσεις Παρεμβολής & Προβολής

Όπως προαναφέρθηκε η τεχνική Particle Mesh επιλύει τις εξισώσεις Poisson πάνω σε ένα Καρτεσιανό πλέγμα. Αυτό προϋποθέτει τη μεταφορά της πληροφορίας της ροής

από τα στοιχεία στροβιλότητας στο πλέγμα. Ακόμη, αφού επιλυθούν οι εξισώσεις Poisson πάνω στο πλέγμα η λύση πρέπει να μεταφερθεί από το πλέγμα πίσω στα στοιχεία στροβιλότητας.

Αν με $Q_p = q_p \cdot V_p$ συμβολίζεται το μέγεθος που μεταφέρεται από τα στοιχεία στροβιλότητας, χρησιμοποιώντας τις συναρτήσεις προβολής:

$$q_{i,j,k} \equiv \mathrm{Proj}_{\mathrm{PM}}(q_p; V_p) = \frac{\sum_p q_p V_p W(\vec{x}_{i,j,k} - \vec{Z}_p)}{\sum_p V_p W(\vec{x}_{i,j,k} - \vec{Z}_p)} \tag{3.16}$$

όπου $\vec{x}_{i,j,k}$ συμβολίζεται το διάνυσμα θέσης των i, j, k κόμβων του πλέγματος. Η συνάρτηση παρεμβολής ορίζεται ως:

$$W(\vec{r}) = W_1(r_x/h)W_1(r_y/h)W_1(r_z/h)$$

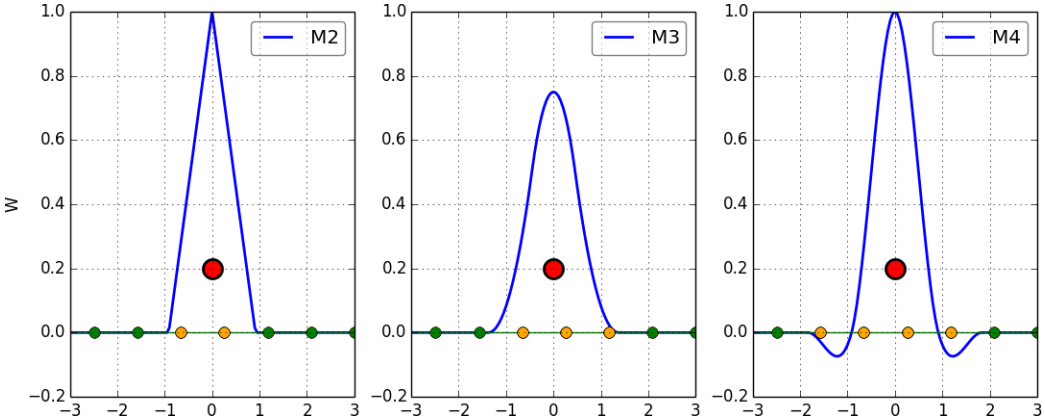
όπου h συμβολίζει την απόσταση μεταξύ δύο κόμβων του πλέγματος. Η συνάρτηση W_1 είναι η μονοδιάστατη συνάρτηση παρεμβολής. Στην παρούσα εργασία χρησιμοποιήθηκε η συνάρτηση $M_4{}'$ [13]) .

Η πληροφορία μεταφέρεται από το πλέγμα στα στοιχεία στροβιλότητας χρησιμοποιώντας τις ίδιες συναρτήσεις παρεμβολής:

$$q_p \equiv \mathrm{Interp}(q_{i,j}) = \sum_{i,j,k} q_{i,j,k} W(\vec{x}_{i,j,k} - \vec{Z}_p) \tag{3.17}$$

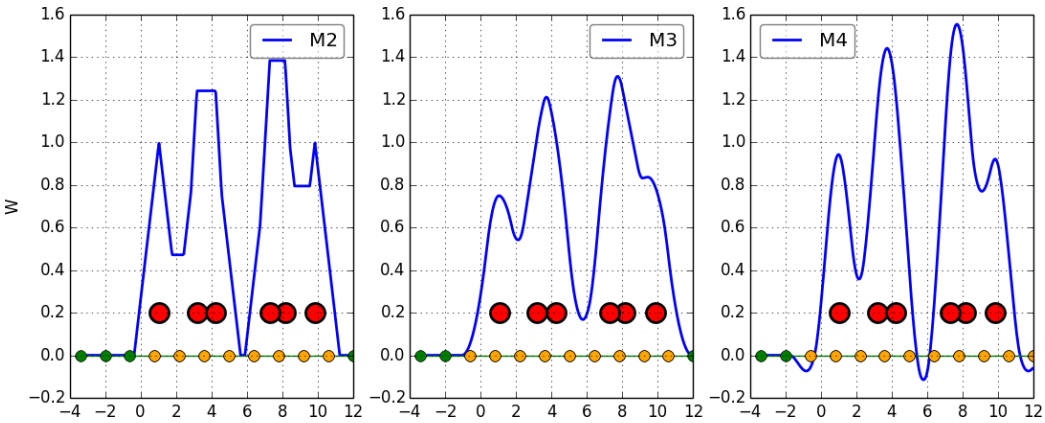
Στην περίπτωση των μεθόδων στοιχείων στροβιλότητας είναι γνωστό ότι η χωρική κατανομή των στοιχείων μπορεί να επηρεάσει την ακρίβεια της μεθόδου [13]. Για αυτό το λόγο ανά τακτά χρονικά βήματα χρειάζεται η ανακατασκευή των ροικών δεδομένων γνωστή ως re-meshing. Το re-meshing συνίσταται στην επαναδιανομή των στοιχείων στροβιλότητας σε δομημένες θέσεις ώστε να διατηρηθεί η ομοιόμορφη κατανομή των στοιχείων.

Για δεδομένο PM πλέγμα η διαδικασία προβολής μεταφέρει την πληροφορία από ένα στοιχείο στροβιλότητας σε δύο ή παραπάνω πλεγματικούς κόμβους. Στο Σχήμα 3.2α′ φαίνονται οι πλεγματικοί κόμβοι που ενεργοποιούνται (κίτρινοι κόμβοι) στη διαδικασία προβολής ανάλογα με την συνάρτηση που θα επιλεγεί.

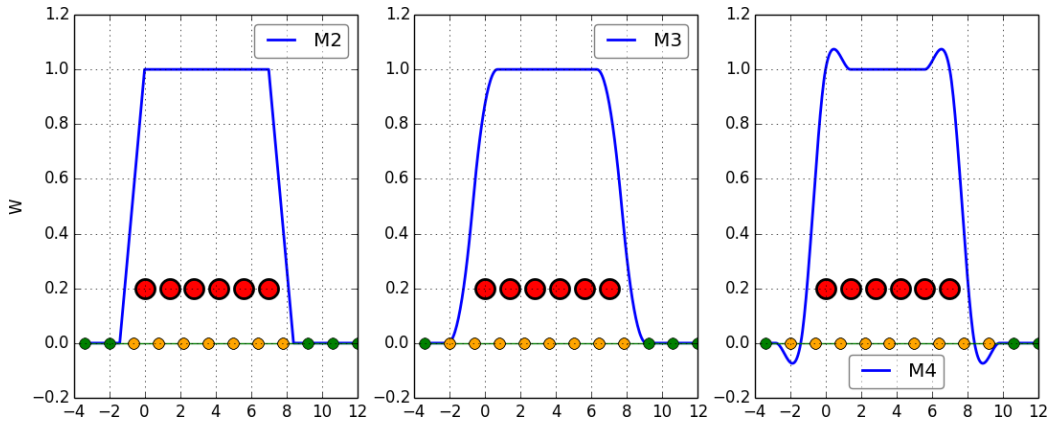


(α′) Προβολή στο πλέγμα χρησιμοποιώντας $1, 2$ και 3 τάξης συνάρτηση παρεμβολής. Με κόκκινο παρουσιάζεται το στοιχείο στροβιλότητας ενώ οι κόμβοι που ενεργοποιούνται με κίτρινο.

Η επιρροή της κατανομής των στοιχείων στροβιλότητας στις συναρτήσεις παρεμβολής φαίνεται στο Σχήμα 3.4α′. Στην περίπτωση που τα στοιχεία στροβιλότητας κατανέμονται ομοιόμορφα η ποιότητα της προβεβλημένης πληροφορίας είναι πολύ καλύτερη από την περίπτωση που η κατανομή των στοιχείων είναι ανομοιόμορφη (Σχήμα 3.3α′). Η σημασία της κατανομής των στοιχείων στροβιλότητας γίνεται ακόμα μεγαλύτερη στην περίπτωση που μεταφέρεται μάζα και όγκος, όπου μπορούν να δημιουργηθούν εστίες με αριθμητικής προέλευσης ελλείμματα μάζας.



(α′) Προβεβλημένη πληροφορία στην περίπτωση ανομοιόμορφης κατανομής στοιχείων στροβιλότητας



(α′) Προβεβλημένη πληροφορία στην περίπτωση ομοιόμορφης κατανομής στοιχείων στροβιλότητας

3.1.4 Αλγόριθμος Lagrangian επιλυτή

Τα βήματα του Lagrangian επιλυτή μπορούν να συνοψιστούν στα εξής:

Για δεδομένα στοιχεία στροβιλότητας: $\{\vec{Z}_p^n,\, m_p^n,\, V_p^n,\, \vec{\Omega}_p^n,\, \Theta_p^n,\, \Pi_p^n\}$ at $t = n\Delta t$:

Βήμα 1: Προβολή των $\{m_p^n,\, \Theta_p^n, \vec{Omega}_p^n,\, \Pi_p^n\}$ στο PM πλέγμα: $\rho_{i,j}^n$, $\theta_{i,j}^n$, $\vec{\omega}_{i,j}^n$, $\varepsilon_{i,j}^n$.
Βήμα 2: Επίλυση $\nabla^2\phi = \theta, \nabla^2\vec{\psi} = -\vec{\omega}$ και υπολογισμός των $\phi_{i,j}^n$, $\psi_{i,j}^n$, $\vec{u}_{i,j}^n$, $\vec{w}_{i,j}^n$ με χρήση πεπερασμένων διαφορών
Βήμα 3: Υπολογισμός στο PM πλέγμα των όρων των εξισώσεων εξέλιξης (3.9), π.χ. $\nabla\rho_{ij}^n$, ∇p_{ij}^n, $\nabla\vec{u}_{ij}^n$
Βήμα 4: Παρεμβολή των ποσοτήτων στο πλέγμα q_{ij}^n στις θέσεις των στοιχείων στροβιλότητας: $q_p^n = \sum_{ij} q_{ij}^n W(\vec{x}_{i,j} - \vec{Z}_p)$.
Βήμα 5: Ανανέωση των ποσοτήτων που μεταφέρουν τα στοιχεία στροβιλότητας (ολοκλήρωση των (3.9)
Βήμα 6: Re-meshing εφόσον χρειάζεται.

Απόσβεση μεταβολής όγκου

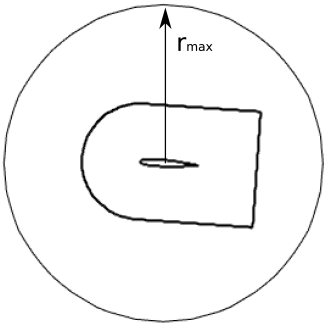
Η μεταβολή όγκου των στοιχείων στροβιλότητας συνδέεται άμεσα με ακουστικά κύματα. Επειδή χρησιμοποιείται πεπερασμένο υπολογιστικό χωρίο, τα κύματα μεταβολής όγκου ανακλώνται στα εξωτερικά σύνορα. Για αυτό το λόγο υιοθετήθηκε μια Gaussian συνάρτηση απόσβεσης:

$$\zeta(r) = \left\{ \begin{array}{ll} e^{-\sigma(r-r_{max})^2} & , r > r_{max} \\ 1 & , \text{else} \end{array} \right\} \tag{3.18}$$

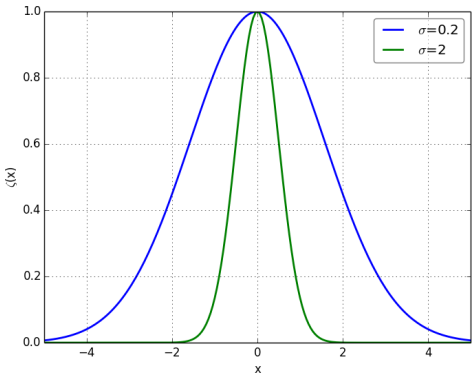
η οποία εφαρμόζεται απευθείας στην εξίσωση εξέλιξης της μεταβολής του όγκου.

$$\theta = \theta \cdot \zeta(r) \tag{3.19}$$

Η συνάρτηση απόσβεσης εφαρμόζεται έξω από ένα κύκλο ακτίνας r_{max} ενώ η παράμετρος σ καθορίζει το ρυθμό της απόσβεσης. Οι παράμετροι σ, r_{max} επιλέγονται ώστε $\theta = 0$ έξω από το PM χωρίο. Ένα παράδειγμα φαίνεται στο Σχήμα 3.6.

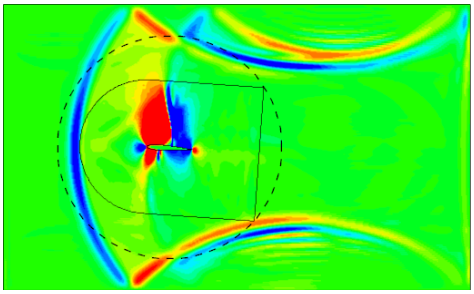


(α′) Έξω από κύκλο ακτίναςr_{max} ενεργοποιείται ή συνάρτηση απόσβεσης.

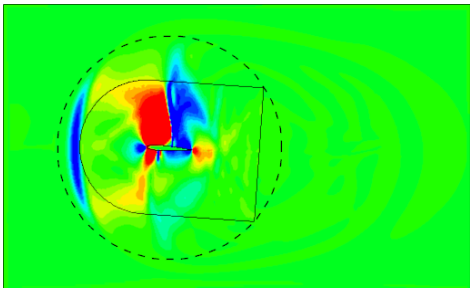


(β′) Ο ρόλος της παραμέτρου σ στην ταχύτητα απόσβεσης.

Σχήμα 3.5: Ορισμός της συνάρτησης απόσβεσης της μεταβολής του όγκου.



(α′) Ανακλάσεις κυμάτων μεταβολής του όγκου στο εξωτερικό σύνορο του PM.

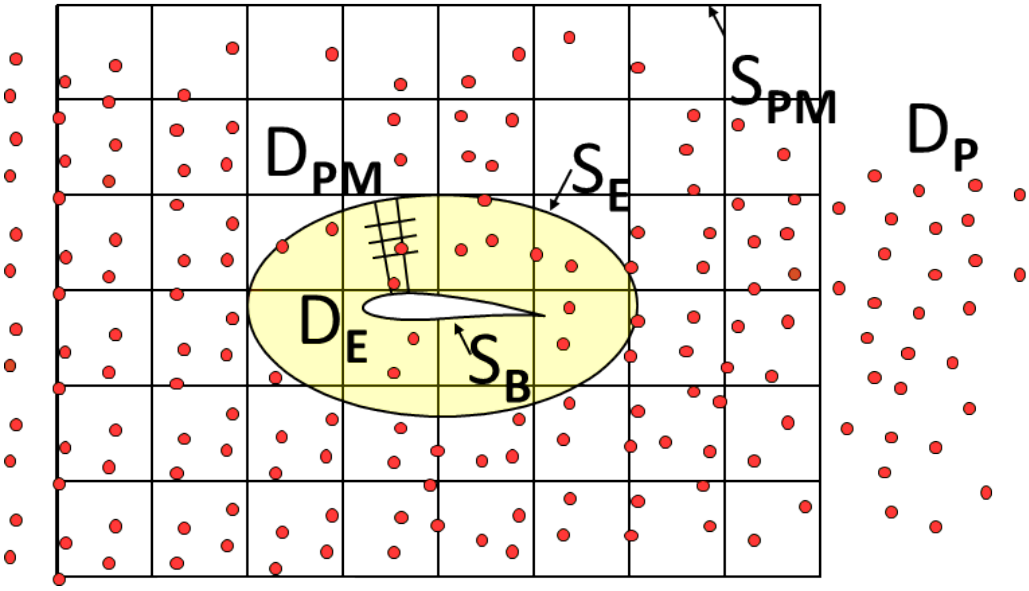


(β′) Η συνάρτηση απόσβεσης ακυρώνει τις ανακλάσεις στο σύνορο S_o.

Σχήμα 3.6: Επίδραση της συνάρτησης απόσβεσης στις ανακλάσεις κυμάτων στα σύνορα του υπολογιστικού χωρίου. Με διακεκομμένη γραμμή ορίζεται ο κύκλος έξω απο τον οποίο εφαρμόζεται η συνάρτηση απόσβεσης.

3.2 Ο Υβριδικός επιλυτής

Στον υβριδικό επιλυτή γίνεται η σύζευξη του Eulerian και του Lagrangian επιλυτή. Ο Eulerian (ή CFD) επιλυτής χρησιμοποιείται για την ακριβή αναπαράσταση των στερεών συνόρων, ενώ ο Lagrangian για την ακριβή αναπαράσταση των επ' άπειρων συνθηκών. Ο Lagrangian (ή PM) επιλυτής καλύπτει όλο το υπολογιστικό χωρίο ενώ ο CFD επιλυτής περιλαμβάνει το στερεό σύνορο και μια μικρή περιοχή γύρω από αυτό.



Σχήμα 3.7: Καθορισμός των υπολογιστικών χωρίων.

Όπως φαίνεται και απο το Σχήμα 3.7 τα χωρία των δύο επιλυτών αλληλεπικαλύπτονται. Στο χωρίο D_E εφαρμόζεται ο CFD επιλυτής ενώ το σύνορο S_E αποτελεί την διεπιφάνεια μεταξύ των δύο επιλυτών. Πρέπει να τονιστεί ότι τα στερεά σύνορα μοντελοποιούνται μόνο στο CFD επιλυτή και όχι στον Lagrangian, έτσι ώστε να αποφευχθεί ή χρήση ολοκληρωτικών αναπαραστάσεων.

3.2.1 Η Σύζευξη

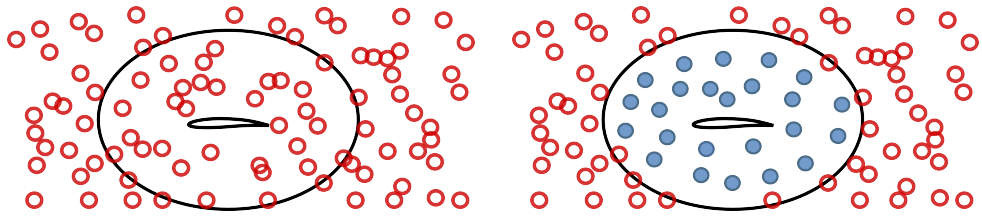
Ο τρόπος που γίνεται η σύζευξη των μεθόδων είναι καθοριστικός στις υβριδικές μεθοδολογίες. Στην προκειμένη περίπτωση η σύζευξη γίνεται με δύο τρόπους.

- Γίνεται χρήση της CFD λύσης U_E για ανανέωση της PM λύσης Q_{PM} στο χωρίο D_E.
- Η PM λύση χρησιμοποιείται για τον καθορισμό των συνοριακών συνθηκών για το CFD επιλυτή πάνω στο S_E.

Η βασική ιδέα πίσω από την σύζευξη είναι στο τέλος κάθε χρονικού βήματος η PM λύση να είναι συνεχής επέκταση της CFD λύσης. Για αυτό το λόγο χρησιμοποιείται μια επαναληπτική μέθοδος κατά την οποία τα στοιχεία στροβιλότητας στο D_E αντικαθιστώνται από στοιχεία στροβιλότητας που παράγονται από τον CFD επιλυτή. Με αυτόν τον τρόπο, η CFD λύση κοντά στο στερεό όριο διορθώνει την Lagrangian. Από την άλλη πλευρά, κάθε φορά που ανανεώνονται τα στοιχεία στροβιλότητας, ανανεώνονται και οι συνοριακές συνθήκες που δίνει το PM στο σύνορο S_E του CFD επιλυτή.

3.2.2 Διόρθωση της Lagrangian λύσης

Όπως προαναφέρθηκε, η Lagrangian λύση διορθώνεται χρησιμοποιώντας την πληροφορία από την CFD λύση. Αυτό γίνεται με την αντικατάσταση των Lagrangian στοιχείων στροβιλότητας με Eulerian (Σχήμα 3.8).



(α′) Τα PM στοιχεία στροβιλότητας.

(β′) Μέσα στο χωρίο D_E τα PM στοιχεία στροβιλότητας αντικαθιστώνται από τα CFD στοιχεία στροβιλότητας (απεικονίζονται με μπλε χρώμα).

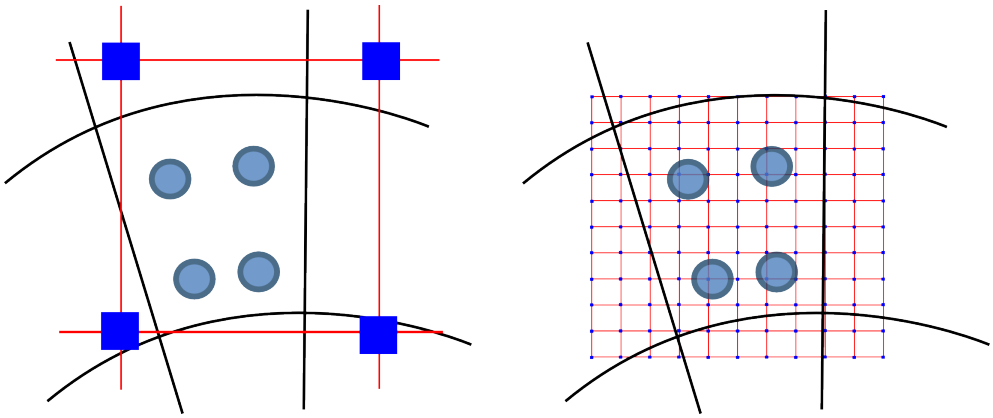
Σχήμα 3.8: Διόρθωση των Q_{PM} στο χωρίοD_E.

Ορισμός των CFD στοιχείων στροβιλότητας

Τα CFD στοιχειά στροβιλότητας ορίζονται σε καθορισμένες θέσεις του CFD πλέγματος. Αρχικά, ή CFD λύση U_E μετατρέπεται στις μεταβλητές που χρησιμοποιεί ο Lagrangian επιλυτής ($q_E = \{\rho, \vec{\omega}, \theta, \Pi\}_E$ στις θέσεις P_E) και στη συνέχεια προβάλλεται η λύση στο PM πλέγμα. Η λύση U_E καθώς και οι παράγωγοι της είναι γνωστές στα κέντρα των πλεγματικών κελιών. Συνεπώς, όλη η απαραίτητη πληροφορία είναι γνωστή στα κέντρα των CFD κελιών.

Από κάθε Eulerian πλεγματικό κελί μπορούν να παραχθούν ένα ή περισσότερα στοιχεία στροβιλότητας. Ο αριθμός των στοιχείων που πρέπει να γεννηθεί εξαρτάται από το μέγεθος του πλεγματικού κελιού σε συνδυασμό με το μέγεθος κελιού που χρησιμοποιείται στο PM πλέγμα. Όπως φαίνεται και στο Σχήμα 3.9 ανάλογα με τη διακριτοποίηση του PM πλέγματος ο αριθμός στοιχείων στροβιλότητας μπορεί να είναι είτε επαρκής είτε όχι. Στην παρούσα εργασία στις περιπτώσεις που εξετάστηκαν τέσσερα ή οκτώ στοιχεία στροβιλότητας ανά Eulerian πλεγματικό κελί χρησιμοποιήθηκαν ανάλογα αν το πρόβλημα ήταν διδιάστατο ή τριδιάστατο. Σημαντικό είναι να

τονιστεί ότι μέσω αυτής της διαδικασίας γίνεται αντιληπτό το στερεό όριο από τον Lagrangian επιλυτή



(α′) Τα CFD στοιχεία στροβιλότητας καλύπτουν επαρκώς το PM πλέγμα.

(β′) Σε αυτήν την περίπτωση η επαρκής κάλυψη του PM πλέγματος απαιτεί περισσότερα από τέσσερα στοιχεία στροβιλότητας ανά CFD κελί.

Σχήμα 3.9: Παραδείγματα επαρκούς (αριστερά) και ανεπαρκούς (δεξιά) κάλυψης του PM πλέγματος από CFD στοιχεία στροβιλότητας.

Ορισμός του σχήματος διόρθωσης

Αρχικά, μέσω της αντικατάστασης των στοιχείων στροβιλότητας στο χωρίο D_E, η παρουσία του στερεού συνόρου γίνεται αντιληπτή στον PM επιλυτή και η Lagrangian λύση στο χωρίο D_{PM} διορθώνεται. Προσοχή πρέπει να δοθεί στην περιοχή κοντά στο εξωτερικό όριο του CFD πλέγματος. Σε αυτήν την περιοχή υπάρχουν στοιχεία στροβιλότητας μέσα στο χωρίο D_E που επηρεάζουν την περιοχή έξω από αυτό, ενώ αντίθετα υπάρχουν Lagrangian στοιχεία στροβιλότητας έξω από το χωρίο D_E που επηρεάζουν τη λύση μέσα στο χωρίο αυτό. Ακόμη το γεγονός ότι τα CFD στοιχεία στροβιλότητας βρίσκονται σε διαφορετικές θέσεις απο τα αντίστοιχα Lagrangian εισάγει ένα ακόμη σφάλμα στη διαδικασία προβολής

Προκειμένου να διορθωθούν τα παραπάνω σφάλματα χρησιμοποιήθηκε η ακόλουθη μεθοδολογία(Σχήμα 3.10):

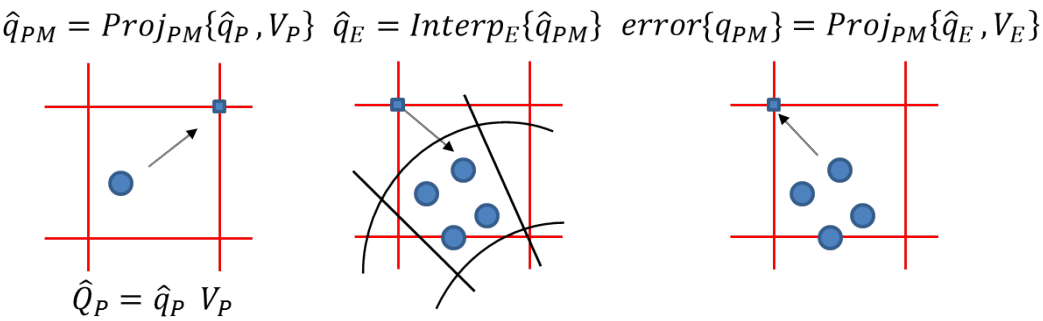
- Αρχικά, τα Lagrangian στοιχεία στροβιλότητας προβάλλονται στο PM πλέγμα ($\hat{q}_{\mathrm{PM}}$).
- Στην συνέχεια, παρεμβάλλεται η πληροφορία από το PM πλέγμα στις θέσεις των CFD στοιχείων στροβιλότητας. ($\hat{Q}_E$)
- Έπειτα, προβάλλονται ξανά στο PM πλέγμα($\hat{q}_E$)

$$error\{q_{PM}\} = \mathrm{Proj}_{\mathrm{PM}}\{\mathrm{Interp}_{P_E}(\hat{q}_{\mathrm{PM}})\} \tag{3.20}$$

Η διορθωμένη PM λύση ανακτάται αφαιρώντας το $error\{q_{PM}\}$ από $\hat{q}_{PM}$ και αθροίζοντας την $\hat{q}_E$:

$$correct\{q_{\mathrm{PM}}\} = \hat{q}_{\mathrm{PM}} - error\{q_{\mathrm{PM}}\} + q_E \tag{3.21}$$

$$q_E = \mathrm{Proj}_{\mathrm{PM}}(Q_E; V_E) \tag{3.22}$$



Σχήμα 3.10: Ορισμός του σφάλματος $error\{q_{PM}\}$.

Το σφάλμα όπως ορίστηκε προηγουμένως αναμένεται να έχει μεγάλες τιμές κοντά στο εξωτερικό σύνορο S_E και συνεπώς:

$$\begin{aligned} q_{\mathrm{PM}} &= \hat{q}_{\mathrm{PM}} && \text{έξω από το Eulerian χωρίο} \\ q_{\mathrm{PM}} &= q_E + \underbrace{\hat{q}_{\mathrm{PM}} - error\{q_{\mathrm{PM}}\}}_{\text{orrection}} && \text{μέσα στο Eulerian χωρίο} \end{aligned} \tag{3.23}$$

3.2.3 Ο αλγόριθμος του υβριδικού επιλυτή HoPFlow

Για δεδομένα στοιχεία στροβιλότητας: $\{\vec{Z}_p^n,\, m_p^n,\, V_p^n,\, \vec{\Omega}_p^n,\, \Theta_p^n,\, \Pi_p^n\}$ στο χρόνο $t = n\Delta t$,

Βήμα 1: Προβολή των $\{m_p^n,\, \Theta_p^n,\, \vec{\Omega}_p^n,\, \Pi_p^n\}$ στο PM πλέγμα: ρ_{ijk}^n, θ_{ijk}^n, $\vec{\omega}_{ijk}^n$, ε_{ijk}^n.

Βήμα 2: Επίλυση των εξισώσεων Poisson $\nabla^2\phi = \theta, \nabla^2\vec{\psi} = -\vec{\omega}$ και μέσω πεπερασμένων διαφορών α

Βήμα 3a: Καθορισμός των συνοριακών συνθηκών στο εξωτερικό όριο του CFD πλέγματος.

Βήμα 3b: Ανανέωση της CFD λύσης.

Βήμα 3c: Δημιουργία CFD στοιχείων στροβιλότητας.

Βήμα 3d: Αν δεν έχει επέλθει σύγκλιση επιστροφή στο Βήμα 2

Βήμα 4: Υπολογισμός στο PM πλέγμα των όρων για την ολοκλήρωση στο χρόνο των εξισώσεων εξ

Βήμα 5: Παρεμβολή της πληροφορίας από το PM πλέγμα q_{ij}^n στις θέσεις των στοιχείων στροβιλότητ $q_p^n = \sum_{ijk} q_{ijk}^n W(\vec{x}_{ijk} - \vec{Z}_p)$.

Βήμα 7a: Ολοκλήρωση του συστήματος (3.9) στο χρόνο: $\{\vec{Z}_p^{n+1},\, m_p^{n+1},\, V_p^{n+1},\, \vec{\Omega}_p^{n+1},\, \Theta_p^{n+1},\, \Pi_p^{n+1}\}$

Βήμα 7b: Re-meshing εφόσον χρειάζεται.

This page intentionally left blank.

Κεφάλαιο 4

Πιστοποίηση

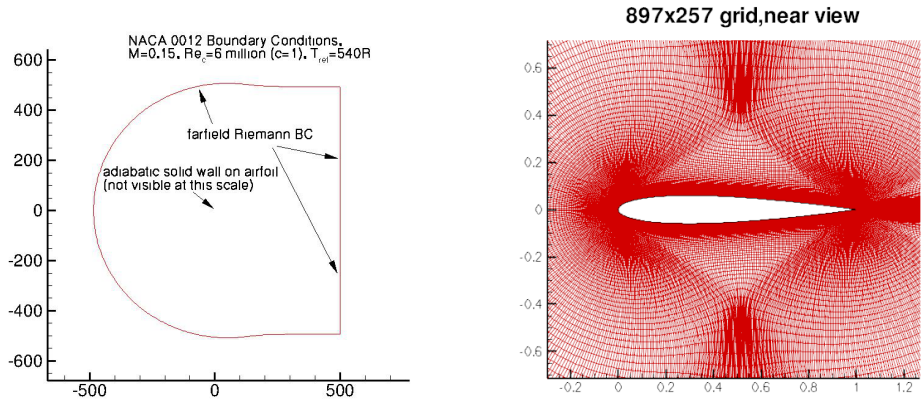
4.1 Πιστοποίηση του CFD επιλυτή(MaPFlow)

4.1.1 Ροή γύρω από αεροτομήNACA0012

Η παρούσα περίπτωση πιστοποίησης αφορά διδιάστατη ροή γύρω από αεροτομή NACA0012 και η οποία παρουσιάζεται στο [52]. Οι γωνίες πρόσπτωσης που εξετάζονται αφορούν προσκολλημένη ροή ενώ οι συνθήκες ροής συνοψίζονται στον Πίνακα 4.1. Στα αποτελέσματα που ακολουθούν γίνεται σύγκριση με αριθμητικά αποτελέσματα από τον υπολογιστικό κώδικα CFL3D [53] άλλα και με πειραματικά δεδομένα από [54]. Το πλέγμα που χρησιμοποιήθηκε αποτελείται από 897x257 κόμβους με 514 σημεία στην αεροτομή το οποίο είναι διαθέσιμο στο [52]. Παρουσιάζονται αποτελέσματα χρησιμοποιώντας τα μοντέλα τύρβης SA καί $k-\omega$ SST.

Πίνακας 4.1: Συνθήκες ροής - NACA0012

Mach	Reynolds	Γωνία Πρόσπτωσης
0.15	$6 \cdot 10^6$	$0^o, 10^o, 15^o$



Σχήμα 4.1: Το υπολογιστικό πλέγμα (διαθέσιμο στο [52])

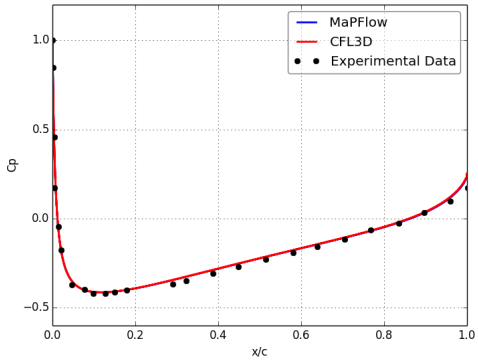
Παρουσιάζονται αποτελέσματα για τρεις γωνίες πρόσπτωσης: $0^o, 10^o$ καί 15°. Οι συγκρίσεις αφορούν στους συντελεστές άνωσης (Cl) και αντίστασης (Cd) ανάμεσα στον κώδικα MaPFlow και τον CFL3D ενώ γίνεται και σύγκριση σε επίπεδο συντελεστών πίεσης (Cp) και επιφανειακής τριβής (Cf).

Στον Πίνακα 4.2 παρουσιάζεται το σχετικό σφάλμα για το Cl και το Cd ανάμεσα στους κώδικες όπου φαίνεται ότι το μέγιστο σφάλμα στο Cl είναι 0.4 % ενώ για το Cd το μέγιστο σφάλμα είναι της τάξης του 5.5 %. Το μεγάλο σφάλμα του Cl στη μηδενική γωνία πρόσπτωσης οφείλεται στο ότι η άνωση είναι σχεδόν μηδενική σε αυτήν την περίπτωση.

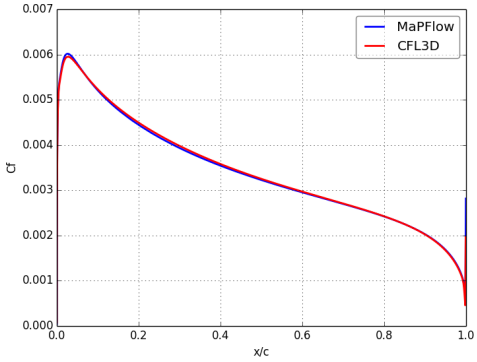
Πίνακας 4.2: Σύγκριση Cl,Cd

	0°		10°		15°	
Κώδικας	Cl	Cd	Cl	Cd	Cl	Cd
MaPFlow(SA)	-6.85E-06	8.19E-03	1.090E+00	1.23E-02	1.546E+00	2.12E-02
CFL3D (SA)	-5.68E-06	8.34E-03	1.091E+00	1.25E-02	1.548E+00	2.21E-02
Δ %(SA)	**1.70E+01**	**-1.90E+00**	**-5.878E-02**	**-1.76E+00**	**-1.455E-01**	**-4.13E+00**
MaPFlow(SST)	-7.62E-06	8.09E-03	1.077E+00	1.236E-02	1.50E+00	2.21E-02
CFL3D (SST)	-5.15E-06	8.07E-03	1.078E+00	1.217E-02	1.51E+00	2.09E-02
Δ %(SST)	**3.24E+01**	**2.17E-01**	**-3.984E-02**	**1.535E+00**	**-4.09E-01**	**5.51E+00**

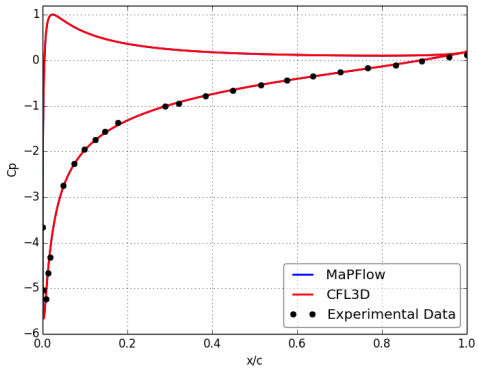
Όπως φαίνεται και στα Σχήματα 4.2 για το μοντέλο τύρβης SA και 4.3 για το μοντέλο τύρβης $k-\omega$ SST ή σύγκριση είναι ικανοποιητική σε επίπεδο Cp και Cf. Μικρές διαφορές στο Cf είναι αναμενόμενες αφού ο συντελεστής τριβής είναι πολύ ευαίσθητος στις αριθμητικές λεπτομέρειες.



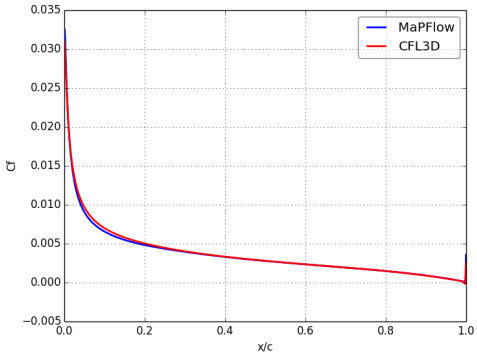
(α′) Συντελεστής Πίεσης $\alpha = 0^o$



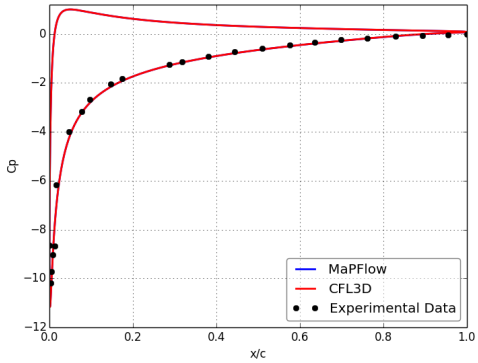
(β′) Συντελεστής Αντίστασης $\alpha = 0^o$



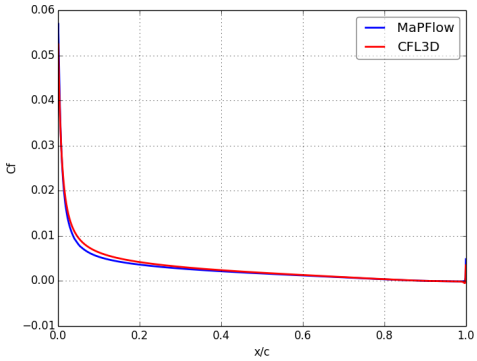
(γ′) Συντελεστής Πίεσης $\alpha = 10^o$



(δ′) Συντελεστής Αντίστασης $\alpha = 10^o$

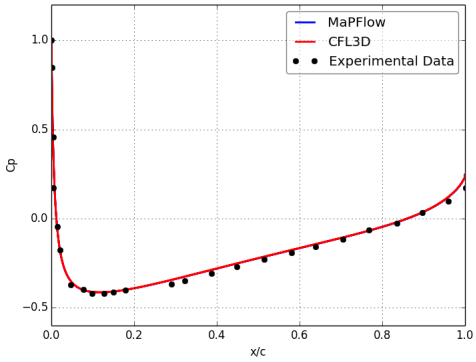


(ε′) Συντελεστής Πίεσης $\alpha = 15^o$

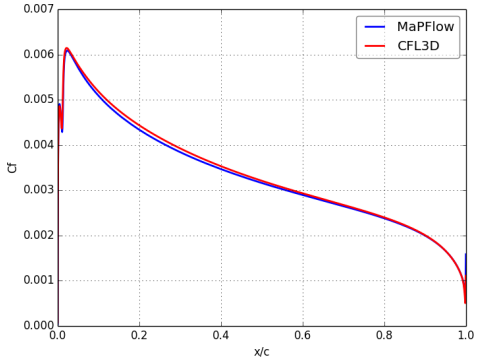


(στ′) Συντελεστής Αντίστασης $\alpha = 15^o$

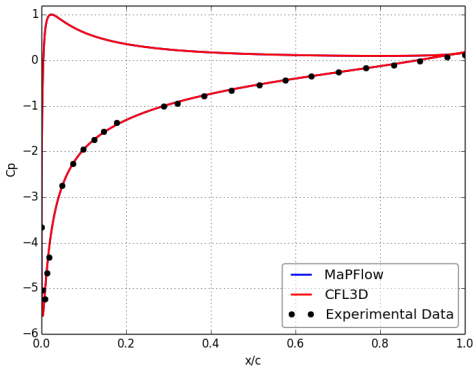
Σχήμα 4.2: Αποτελέσματα μοντέλου τύρβης SA. Ικανοποιητική συμφωνία με αριθμητικούς υπολογισμούς και πειραματικά δεδομένα.



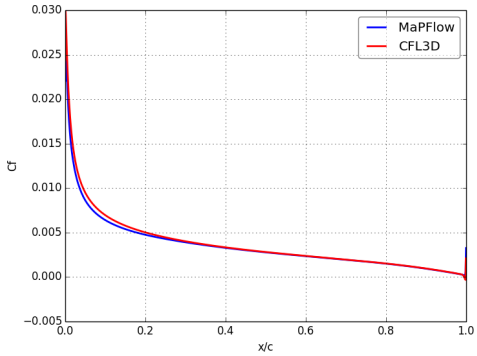
(α′) Συντελεστής Πίεσης $\alpha = 0^o$



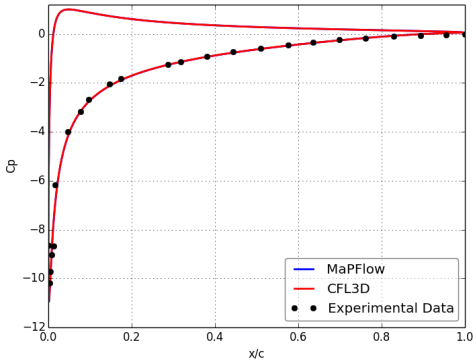
(β′) Συντελεστής Αντίστασης $\alpha = 0^o$



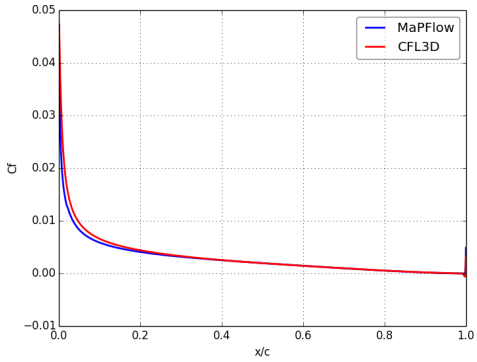
(γ′) Συντελεστής Πίεσης $\alpha 10^o$



(δ′) Συντελεστής Αντίστασης $\alpha = 10^o$



(ε′) Συντελεστής Πίεσης $\alpha = 15^o$



(στ′) Συντελεστής Αντίστασης $\alpha = 15^o$

Σχήμα 4.3: Αποτελέσματα μοντέλου τύρβης $k - \omega$ SST. Ικανοποιητική συμφωνία με αριθμητικούς υπολογισμούς και πειραματικά δεδομένα.

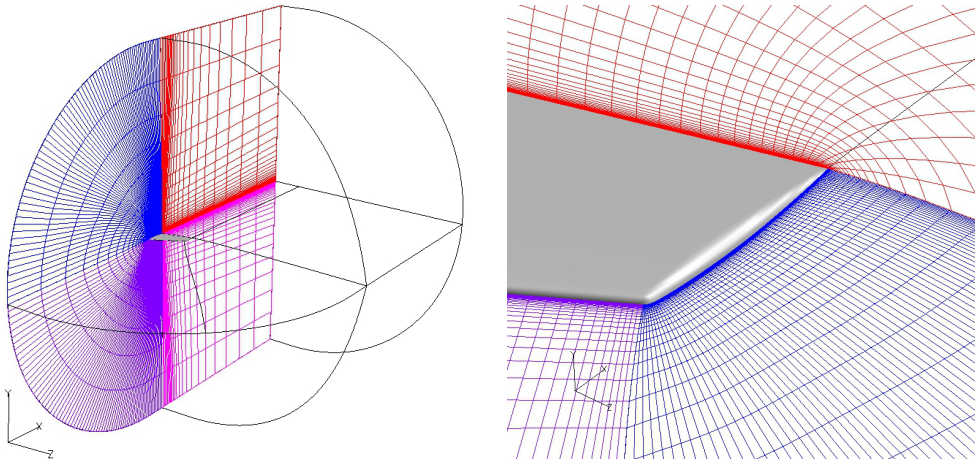
4.1.2 Ροή γύρω από την πτέρυγα ONERA M6

Αυτή η περίπτωση πιστοποίησης επιλέχθηκε από το [55] καθώς είναι μια τριδιάστατη ροή με έντονα συμπιεστά φαινόμενα. Οι συνθήκες ροής παρουσιάζονται στον Πίνακα 4.3.

Πίνακας 4.3: Συνθήκες Ροής - ONERA M6

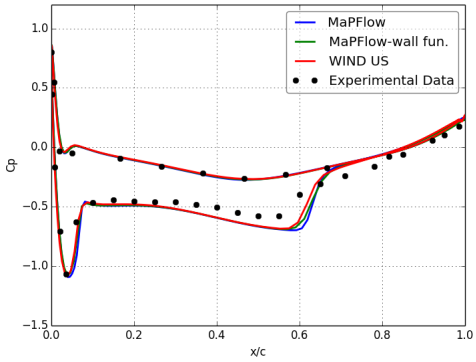
Mach	Reynolds	Γωνία Πρόσπτωσης
0.8395	$11.72 \cdot 10^6$	3.06^o

Στην παρούσα ανάλυση χρησιμοποιήθηκαν δυο διαφορετικά πλέγματα. Ένα αραιό με $y+ \approx 30$ και ένα πυκνό με $y+ \approx 1$. Το αραιό πλέγμα είναι διαθέσιμο από [55] (Σχήμα 4.4) ενώ το πυκνό από [53]. Το αραιό πλέγμα αποτελείται από $3 \cdot 10^5$ κελιά ενώ χρησιμοποιήθηκαν συναρτήσεις τοίχου για μοντελοποίηση του οριακού στρώματος και είναι το ίδιο πλέγμα με το οποίο παρήχθησαν τα αποτελέσματα του κώδικα WINDUS. Τέλος, τα πειραματικά δεδομένα βρίσκονται στο [56].

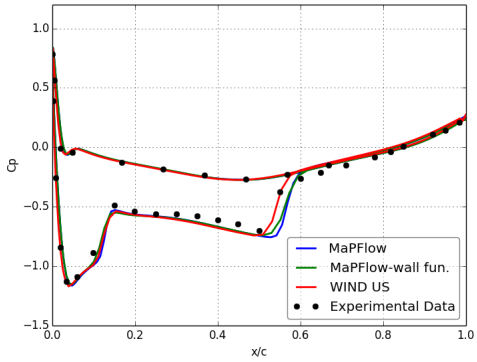


Σχήμα 4.4: Το υπολογιστικό πλέγμα (διαθέσιμο στο [55])

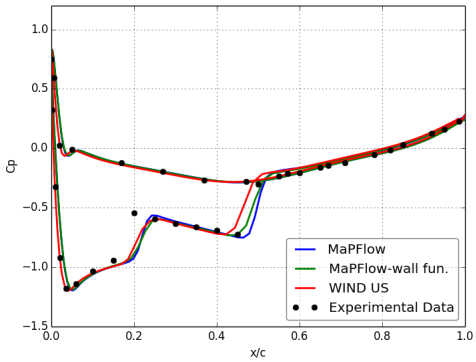
Στο Σχήμα 4.5 γίνεται σύγκριση του συντελεστή πίεσης σε έξι ακτινικές θέσεις της πτέρυγας. Παρουσιάζεται σύγκριση με αριθμητικά αποτελέσματα του κώδικα WIND-US αλλα και με πειραματικά δεδομένα. Σημειώνεται, ότι στην περίπτωση του MaPFlow παρουσιάζονται αποτελέσματα με και χωρίς συναρτήσεις τοίχου. Όπως φαίνεται και στο Σχήμα 4.5 και σε αυτήν την περίπτωση η σύγκριση είναι ικανοποιητική.



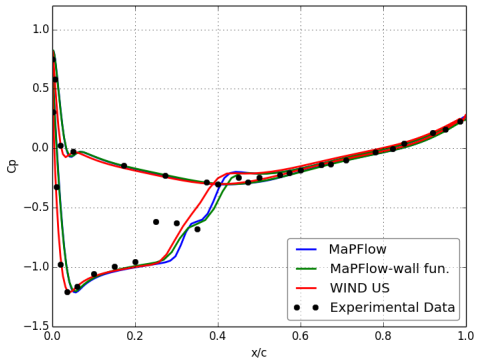
(α′) Συντελεστής Πίεσης $y/(b/2)$=20%



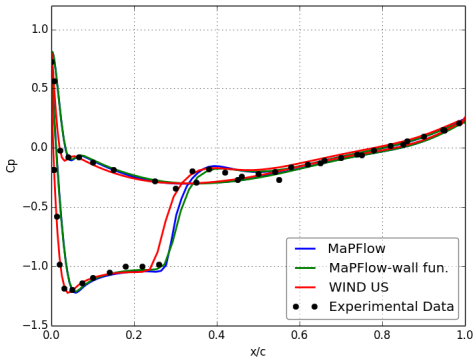
(β′) Συντελεστής Πίεσης $y/(b/2)$=44%



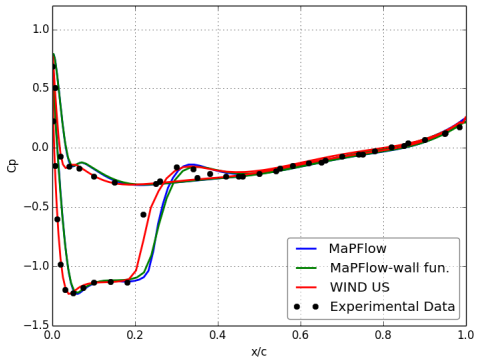
(γ′) Συντελεστής Πίεσης $y/(b/2)$=65%



(δ′) Συντελεστής Πίεσης $y/(b/2)$=80%



(ε′) Συντελεστής Πίεσης $y/(b/2)$=90%



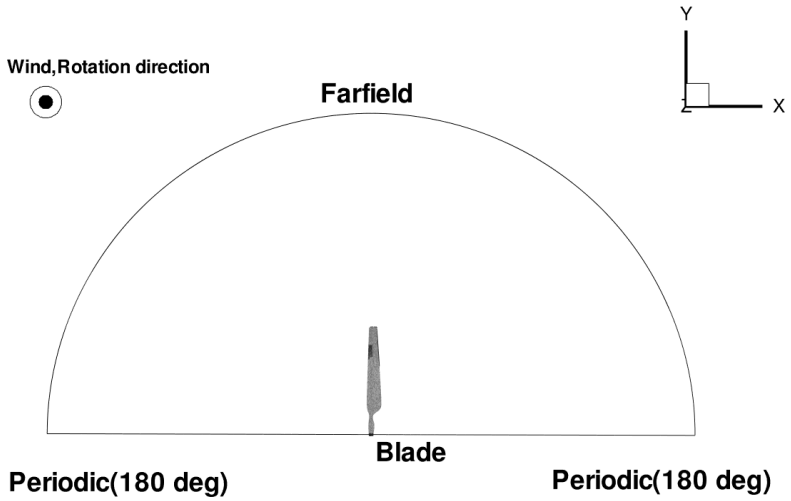
(στ′) Συντελεστής Πίεσης $y/(b/2)$=95%

Σχήμα 4.5: ONERA M6: Κατανομή Πίεσης σε έξι ακτινικές θέσεις (Ma=0.84, $\alpha = 3.06^o$).

4.1.3 Πείραμα NREL Phase VI

Αυτή η περίπτωση αφορά στο πείραμα NREL Phase VI. Η ανάλυση γίνεται για ένα δίπτερο δρομέα ανεμογεννήτριας ονομαστικής ισχύς 20kW. Το πείραμα έγινε στην αεροσύραγγα NASA Ames [57] και λόγω της καλής ποιότητας των πειραμάτων η περίπτωση αυτή χρησιμοποιείται συχνά στην πιστοποίηση υπολογιστικών κωδίκων [58, 59, 60].

Το πλέγμα που χρησιμοποιήθηκε στη συγκεκριμένη περίπτωση αποτελείται από $5 \cdot 10^6$ κελιά, με y+ ≈ 1 και με χρήση περιοδικών συνθηκών (Σχήμα 4.6). Για την κατασκευή του πλέγματος χρησιμοποιήθηκε το εμπορικό πακέτο ANSYS ICEM CFD. Για τη μοντελοποίησης της τύρβης χρησιμοποιήθηκε το μοντέλο $k - \omega$ SST. Οι συνθήκες ροής συνοψίζονται στον Πίνακα 4.4.



Σχήμα 4.6: Το υπολογιστικό πλέγμα

Πίνακας 4.4: Συνθήκες ροής - NREL Phase VI

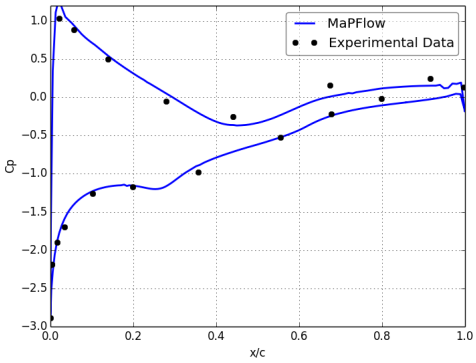
U_∞	Ω (RPM)	Γωνία Βήματος(^{o})
7m/s	72	2.98^o
10 m/s	72	2.98^o
15 m/s	72	2.98^o

Όπως φαίνεται στο Σχήμα 4.7 για ταχύτητα $7m/s$ η ροή είναι προσκολλημένη και οι αριθμητικές προλέξεις είναι σε καλή συμφωνία με τα πειραματικά δεδομένα. Στα $10m/s$ (Σχήμα 4.8) η ροή αποκολλάται τοπικά στη θέση $r/R = 47\%$. Τέλος στην ταχύτητα των $15m/s$(Σχήμα 4.9) η ροή είναι έντονα αποκολλημένη στο μεγαλύτερο κομμάτι του πτερυγίου.

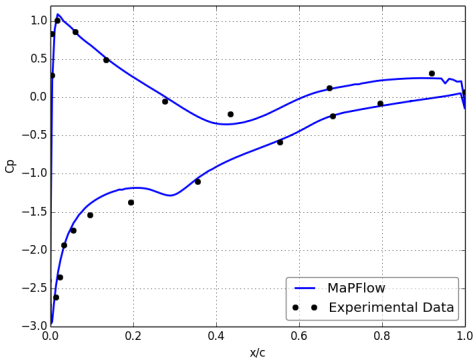
Συνολικά οι αριθμητικές προλέξεις συγκρίνονται ικανοποιητικά με τα πειραματικά δεδομένα. Παρόλα αυτά, υπάρχουν μικρές διαφορές στον συντελεστή πίεσης στην

πάνω πλευρά του πτερυγίου (στη περιοχή $x/c = 0.1 - 0.3$ και στις ακτινικές θέσεις $r/R = 0.47 - 0.8$). Αυτές οι διαφορές σχετίζονται με την χαμηλή ποιότητα της γεωμετρίας που διατέθηκε για την κατασκευή του πλέγματος. Αυτές οι διαφορές αποκτούν πιο έντονο χαρακτήρα στα $15m/s$, όπου η ροή είναι μη-μόνιμη και έντονα αποκολλημένη.

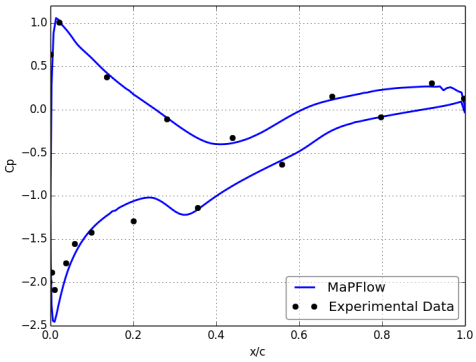
Η Περίπτωση των $7m/s$



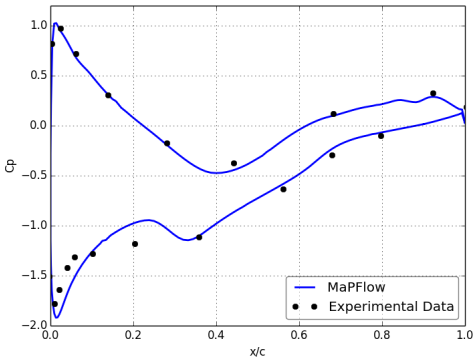
(α′) Συντελεστής Πίεσης r/R=30%



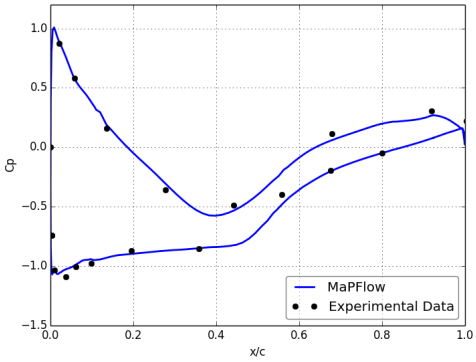
(β′) Συντελεστής Πίεσης r/R=47%



(γ′) Συντελεστής Πίεσης r/R=63%



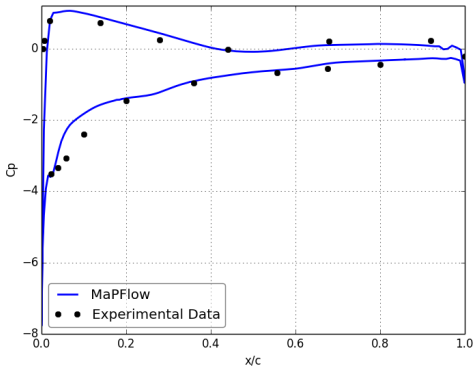
(δ′) Συντελεστής Πίεσης r/R=80%



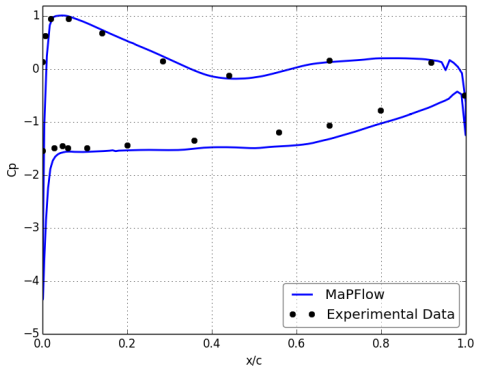
(ε′) Συντελεστής Πίεσης r/R=95%

Σχήμα 4.7: Δρομέας NREL σε ταχύτητα ανέμου $7m/s$. Κατανομή του συντελεστή πίεσης σε πέντε ακτινικές θέσεις.

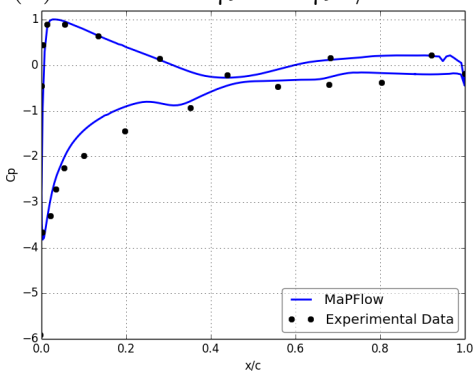
Η Περίπτωση των $10m/s$



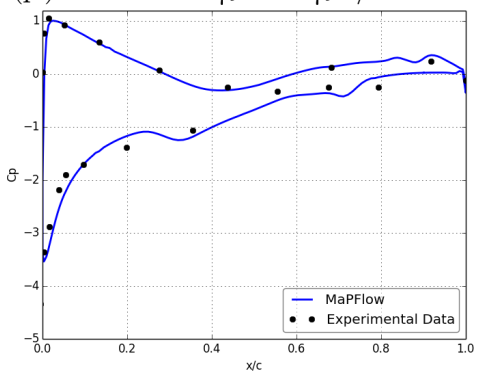
(α′) Συντελεστής Πίεσης r/R=30%



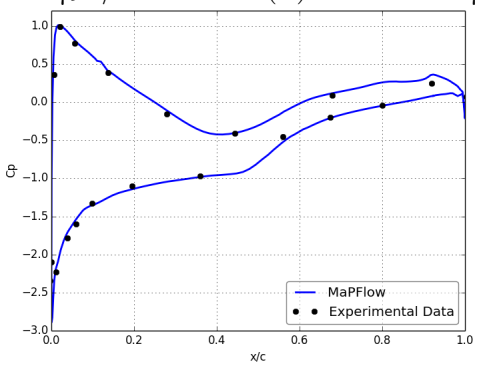
(β′) Συντελεστής Πίεσης r/R=47%



(γ′) Συντελεστής Πίεσης r/R=63%



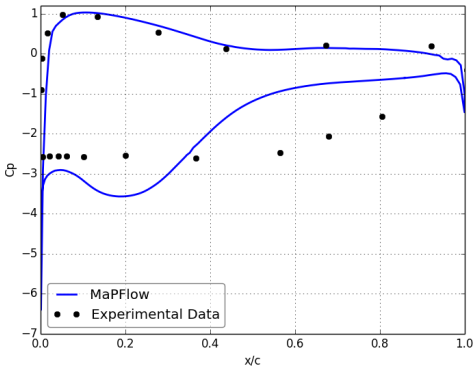
(δ′) Συντελεστής Πίεσης r/R=80%



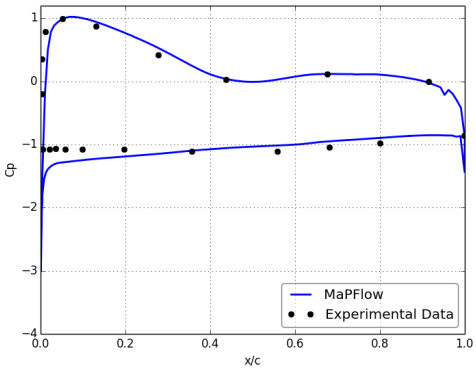
(ε′) Συντελεστής Πίεσης r/R=95%

Σχήμα 4.8: Δρομέας NREL σε ταχύτητα ανέμου $10m/s$. Κατανομή του συντελεστή πίεσης σε πέντε ακτινικές θέσεις.

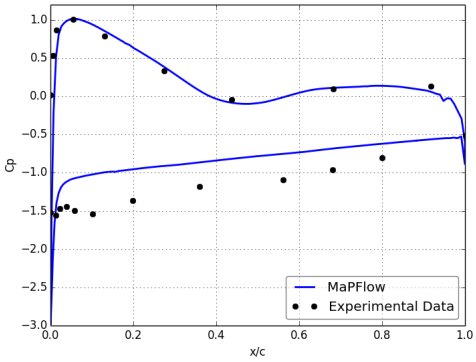
Η Περίπτωση των $15m/s$



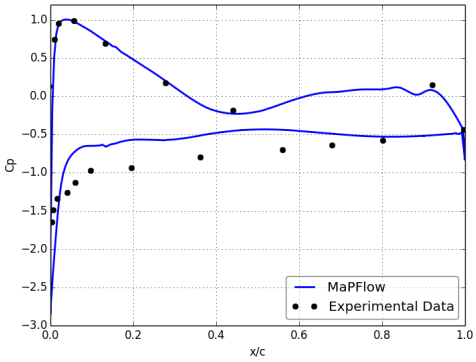
(α′) Συντελεστής Πίεσης r/R=30%



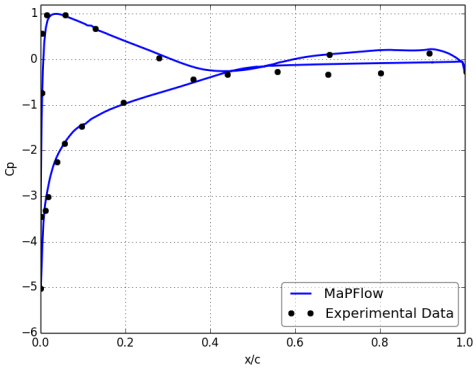
(β′) Συντελεστής Πίεσης r/R=47%



(γ′) Συντελεστής Πίεσης r/R=63%



(δ′) Συντελεστής Πίεσης r/R=80%



(ε′) Συντελεστής Πίεσης r/R=95%

Σχήμα 4.9: Δρομέας NREL σε ταχύτητα ανέμου $15m/s$. Κατανομή του συντελεστή πίεσης σε πέντε ακτινικές θέσεις.

4.2 Πιστοποίηση του Υβριδικού Επιλυτή(HoPFlow)

4.2.1 Η Υπολογιστική Διάταξη

Η βασική υπολογιστική διάταξη του υβριδικού επιλυτή (Σχήμα 4.10) αποτελείται από:

1. Το Eulerian (CFD) πλέγμα,
2. Ένα εσωτερικό PM πλέγμα που είναι το χωρίο που ορίζεται ο Lagrangian επιλυτής.
3. Ένα εξωτερικό PM πλέγμα που χρησιμοποιείται για τη μοντελοποίηση του μακρινού ομόρρου.

Τα δύο PM πλέγματα χαρακτηρίζονται από το μέγεθος των κελιών τους ΔX_{PMout}, ΔX_{PM} και την έκταση του N_{PMout}, N_{PM}. Το CFD πλέγμα χαρακτηρίζεται από την έκταση του Nc_{CFD} και των αριθμών των στοιχείων στροβιλότητας που παράγονται ανά υπολογιστικό κελί NP_{CFD}.

Σχήμα 4.10: Η υπολογιστική διάταξη του υβριδικού επιλυτή. Αποτελείται από το CFD πλέγμα στο οποίο μοντελοποιούνται τα στερεά όρια, ένα εσωτερικό PM πλέγμα και ένα μεγαλύτερο για τη μοντελοποίηση του μακρινού ομόρρου.

Το εξωτερικό PM πλέγμα μεγαλώνει σε έκταση καθώς ο μακρινός ομόρρους εξελίσσεται. Στην περιοχή του εξωτερικού PM τα στοιχεία στροβιλότητας μεταφέρουν μόνο στροβιλότητα, έχουν σταθερό όγκο ενώ μεταφέρονται με την επ' άπειρο ταχύτητα U_∞.

Σε όλη τη διάρκεια των υπολογισμών είναι σημαντικό να υπάρχει πλήρης κάλυψη του PM πλέγματος από στοιχεία στροβιλότητας. Για αυτό το λόγο, η τεχνική του re-meshing εφαρμόζεται κάθε N_{RM} βήματα οπότε αναδιατάσσονται τα στοιχεία στροβιλότητας σε δομημένες θέσεις.

Τέλος, όσο αναφορά τη χρονική ολοκλήρωση ο Lagrangian επιλυτής χρησιμοποιεί ένα άμεσο σχήμα πρώτης τάξης σε αντίθεση με τον CFD επιλυτή που χρησιμοποιεί ένα έμμεσο σχήμα δεύτερης τάξης. Συνεπώς, είναι λογικό το μέγιστο επιτρεπόμενο χρονικό βήμα να περιορίζεται από τον Lagrangian επιλυτή.

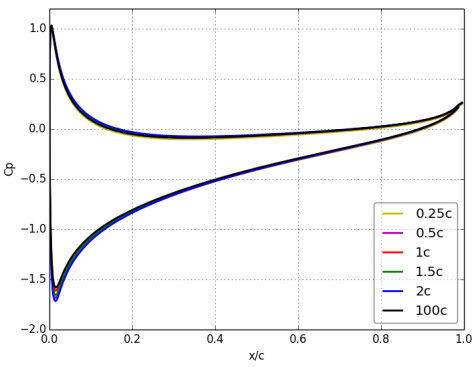
Έπειτα από παραμετρική ανάλυση έγιναν οι εξής επιλογές:

- $\Delta X_{PM} = 0.02$, $\Delta X_{PMout} = 4\Delta X_{PM}$
- N_{PMout} =30c,N_{PM} = 7c
- N_{RM} = 2-4 βήματα
- $NP_{CFD} = 4$
- Nc_{CFD} = 1-1.5c

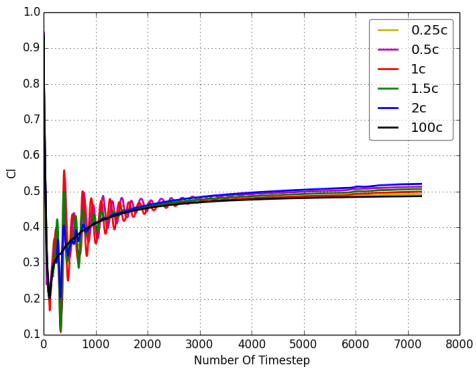
4.2.2 Ατριβής ροή γύρω απο αεροτομή NACA0012 σε μόνιμες συνθήκες

Στη περίπτωση αυτή γίνεται αξιολόγηση του υβριδικού επιλυτή σε διάφορους αριθμούς Mach. Οι περιπτώσεις που εξετάζονται επιλέχθηκαν από [56] και αφορούν ροή σε γωνία πρόσπτωσης $\alpha = 4$ και σε αριθμούς Mach=0.3, 0.6, 0.755.

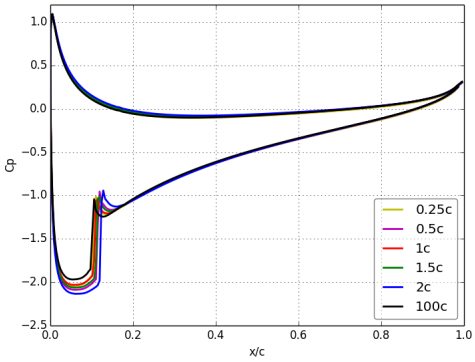
Τα αποτελέσματα του υβριδικού επιλυτή (HoPFlow) συγκρίνονται με αυτά του CFD επιλυτή (MaPFlow). Το υπολογιστικό πλέγμα στην περίπτωση του CFD είχε έκταση 100 χορδές. Στην περίπτωση του υβριδικού επιλυτή εξετάστηκαν πλέγματα διαφορετικής έκτασης από 0.25 χορδές έως 2 χορδές, για να αξιολογηθεί και η επίδραση της έκτασης του πλέγματος στους υπολογισμούς.



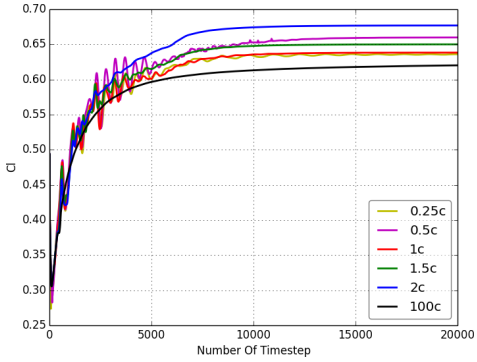
(α′) Σύγκριση Cp, Ma=0.3, $\alpha = 4^o$



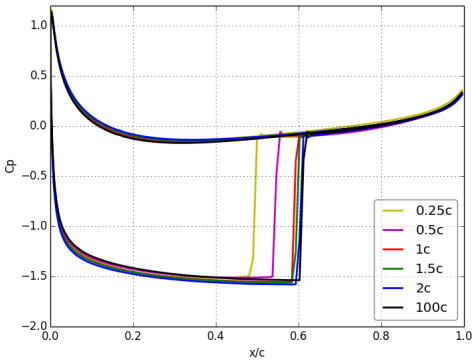
(β′) Ιστορία σύγκλισης Cl, Ma=0.3, $\alpha = 4^o$



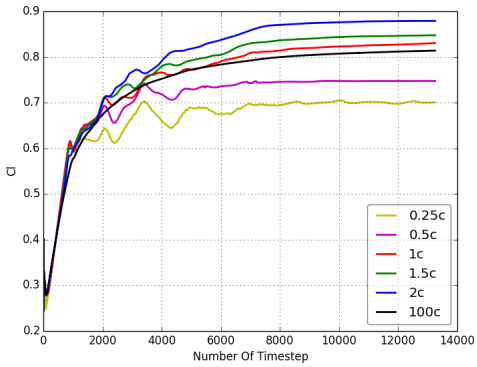
(γ′) Σύγκριση Cp, Ma=0.6, $\alpha = 4^o$



(δ′) Ιστορία σύγκλισης Cl, Ma=0.6, $\alpha = 4^o$



(ε′) Σύγκριση Cp, Ma=0.755, $\alpha = 4^o$



(στ′) Ιστορία σύγκλισης Cl, Ma=0.755, $\alpha = 4^o$

Σχήμα 4.11: Σύγκριση HoPFlow με MaPFlow για μόνιμη ατριβής ροή γύρω από NACA0012 σε διάφορους αριθμούς Mach

Στην περίπτωση του Ma=0.3 όλα τα πλέγματα που χρησιμοποιήθηκαν στον υβριδικό επιλυτή παρήγαγαν ικανοποιητικά αποτελέσματα όπως φαίνεται στο Σχήμα 4.11α′) και στο Σχήμα 4.11β′. Αν και στην τελική λύση τα αποτελέσματα είναι αρκετά κοντά, στην ιστορία σύγκλισης του Cl (Σχήμα 4.11β′) παρατηρείται ότι στον υβριδικό επιλυτή παράγονται ταλαντώσεις στη διαδικασία της σύγκλισης σε αντίθεση με την περίπτωση του CFD. Αυτές οι ταλαντώσεις σχετίζονται με τον αρχικό στρόβιλο που στην περίπτωση του CFD διαχέεται γρήγορα.

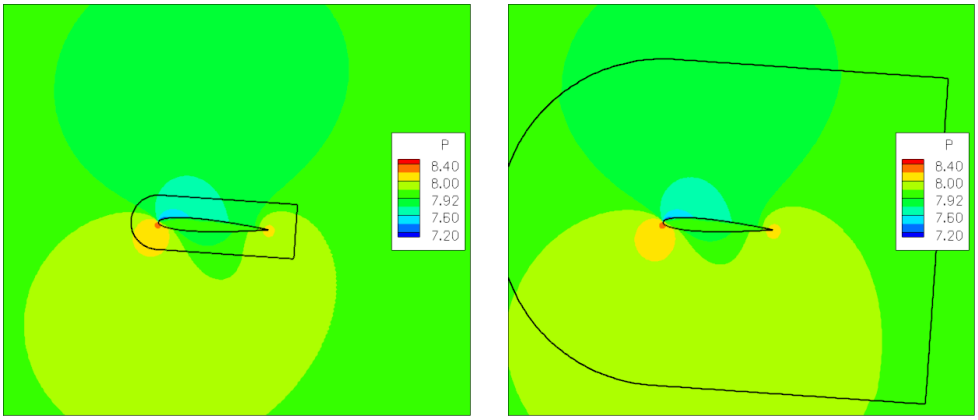
Στην περίπτωση του Ma =0.6 όπως προκύπτει από το Σχήμα 4.11γ′ και τό Σχήμα 4.11δ′ τα συμπεράσματα είναι ίδια. Τα αποτελέσματα του υβριδικού επιλυτή είναι κοντά με αυτά του CFD και το μικρό κύμα κρούσης που προκύπτει προλέγεται σε παρόμοια θέση σε όλες τις περιπτώσεις. Η μεγάλη διαφορά που παρατηρείται στον συντελεστή άνωσης (Cl) στο Σχήμα 4.11δ′ οφείλεται σε μικρές διαφορές στη θέση και το μέγεθος του κύματος κρούσης που αποτυπώνονται πιο έντονα στο Cl.

Τέλος, στην περίπτωση του Ma=0.755 όπου παρουσιάζεται ένα ισχυρό κύμα κρούσης, φαίνεται ότι τα μικρότερα πλέγματα (0.25c και 0.5c) αδυνατούν να προλέξουν σωστά τη θέση του κύματος κρούσης (Σχήμα 4.11ε΄ και 4.11στ΄). Αντίθετα, τα πλέγματα έκτασης από μία χορδή και πάνω προλέγουν ικανοποιητικά τη θέση του κύματος κρούσης. Όπως και προηγουμένως, μικρές διαφορές στη θέση και το μέγεθος του κύματος κρούσης αποτυπώνονται πολύ έντονα στην ιστορία του Cl.

Στον πίνακα 4.5 παρουσιάζεται το σχετικό σφάλμα του Cl για τις διάφορες περιπτώσεις σε σχέση με τις προλέξεις του CFD επιλυτή. Για τις τρεις περιπτώσεις που παρουσιάστηκαν για αριθμούς Mach=0.3,0.6, 0.755 το μέγιστο σχετικό σφάλμα είναι 7%, 9% και 8% αντίστοιχα. Στους δύο χαμηλότερους αριθμούς Mach τα καλύτερα αποτελέσματα παράγονται από το μικρότερο πλέγμα. Αντίθετα, στην περίπτωση με το ισχυρό κύμα κρούσης τα μικρότερα πλέγματα αποτυγχάνουν να προλέξουν σωστά την θέση του κύματος κρούσης. Ανεξάρτητα από τις διαφορές που παρουσιάζονται η ισχυρή σύζευξη φαίνεται στις ισουψείς πίεσης στο Σχήμα 4.12 για Ma=0.3 και στο Σχήμα 4.13 για Ma=0.755 όπου φαίνεται ότι η λύση είναι συνεχής ακόμα και στην περίπτωση που το κύμα κρούσης προλέγεται εσφαλμένα.

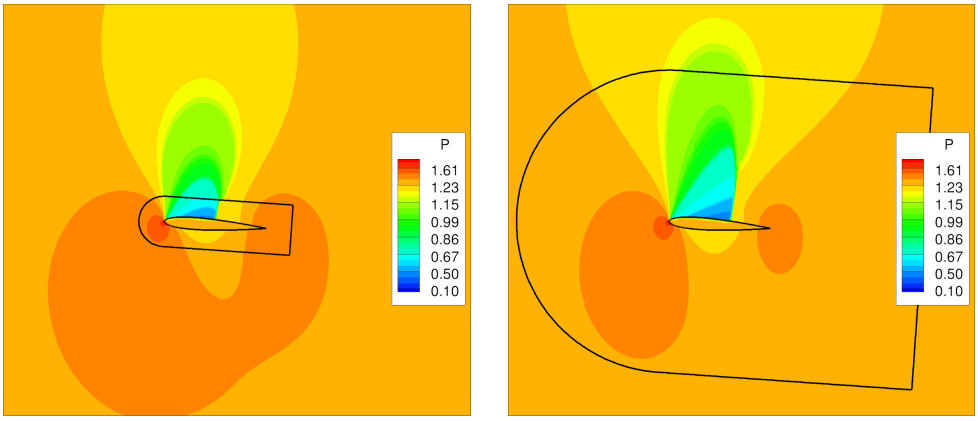
Πίνακας 4.5: Σύγκριση του Cl στα διαφορετικά πλέγματα σε

	Ma=0.3		Ma=0.6		Ma=0.755	
Έκταση πλέγματος HoPFlow	Cl	ΔCl%	Cl	ΔCl%	Cl	ΔCl%
$0.25c$	4.92E-01	1.19E+00	6.35E-01	2.54E+00	7.00E-01	-1.38E+01
$0.5c$	5.13E-01	5.40E+00	6.59E-01	6.40E+00	7.47E-01	-8.16E+00
$1.0c$	4.99E-01	2.55E+00	6.38E-01	2.92E+00	8.30E-01	2.01E+00
$1.5c$	5.06E-01	4.05E+00	6.50E-01	4.82E+00	8.48E-01	4.12E+00
$2.0c$	5.21E-01	7.05E+00	6.76E-01	9.15E+00	8.79E-01	8.00E+00
$100c$	4.86E-01	0.00E+00	6.20E-01	0.00E+00	8.14E-01	0.00E+00



(α′) Πλέγμα έκτασης 0.25 χορδών (β′) Πλέγμα έκτασης 1.5 χορδών

Σχήμα 4.12: Σύγκριση κατανομών πίεσης για Ma=0.3, $\alpha = 4^o$



(α′) Πλέγμα έκτασης 0.25 χορδών (β′) Πλέγμα έκτασης 1.5 χορδών

Σχήμα 4.13: Σύγκριση κατανομών πίεσης για Ma=0.755, $\alpha = 4^o$.

4.2.3 Ατριβής ροή γύρω από κινούμενη αεροτομή NACA0012

Σε αυτήν την περίπτωση εξετάζεται η ικανότητα του υβριδικού επιλυτή να υπολογίζει ροές γύρω από κινούμενες γεωμετρίες. Για αυτό το λόγο, παρουσιάζονται τέσσερις περιπτώσεις κινούμενης αεροτομής. Οι τρεις πρώτες περιπτώσεις προήλθαν από το [61] ενώ η τέταρτη αφορά κίνηση σε πολύ μεγάλη συχνότητα. Σε όλες τις περιπτώσεις η ταλάντωση έγινε γύρω από το 0.25 της χορδής ενώ η εξίσωση κίνησης του σώματος έχει την παρακάτω μορφή:

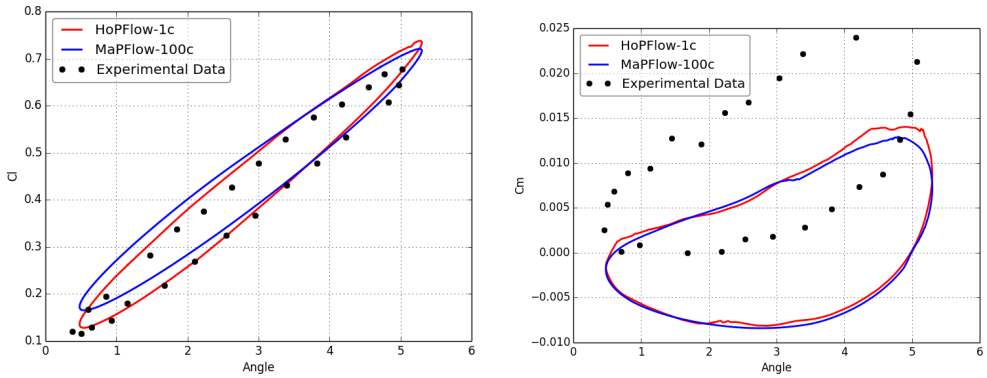
$$\alpha = \alpha_m + \alpha_s * sin(\omega t) \tag{4.1}$$

Οι περιπτώσεις συνοψίζονται στον Πίνακα 4.6.

Πίνακας 4.6: Μη-μονιμη ροή γύρω απο NACA0012.

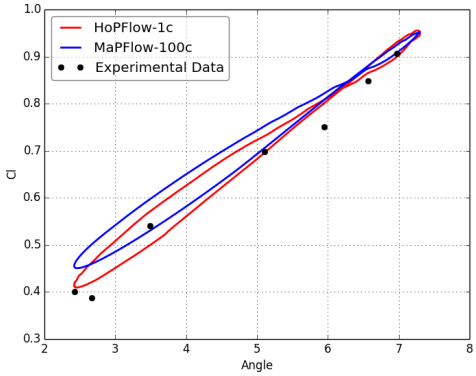
Case	Mach	α_m (deg)	α_s (deg)	ω
AGARD CT1	0.6	2.89^o	2.41^o	0.16
AGARD CT3	0.6	4.86^o	2.44^o	0.16
AGARD CT5	0.755	0.016^o	2.51^o	0.16
Fast Pitching	0.6	0°	2°	8

Τα υβριδικά αποτελέσματα (HoPFlow) συγκρίνονται με τα αυτά του CFD (MaPFlow) επιλυτή αλλά και με πειραματικά δεδομένα. Όπως και στην προηγούμενη περίπτωση το πλέγμα που χρησιμοποιήθηκε για τον CFD επιλυτή είχε έκταση 100 χορδές ενώ, στο πλέγμα που χρησιμοποιήθηκε για τον υβριδικό επιλυτή είχε έκταση 1 χορδή.

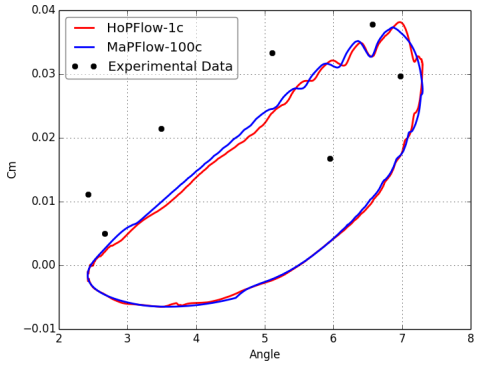


(α′) Cl vs α (β′) Cm vs α

Σχήμα 4.14: Cl και Cm κατανομές. Ατριβής ροή γύρω από κινούμενη NACA0012, AGARD CT1.

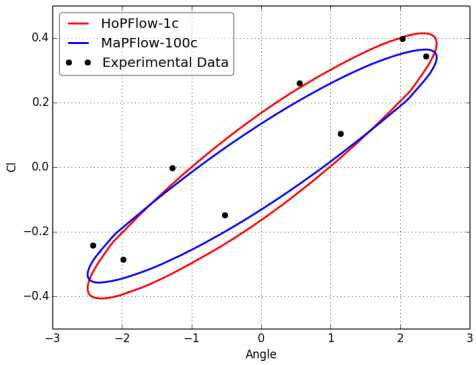


(α′) Cl vs α

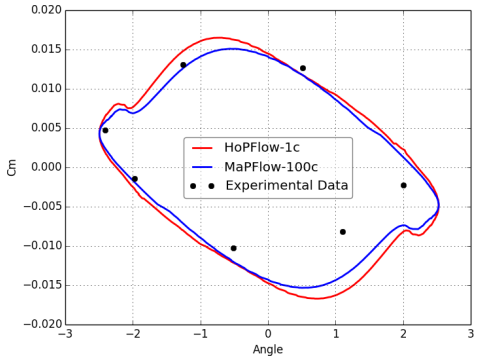


(β′) Cm vs α

Σχήμα 4.15: Cl και Cm κατανομές. Ατριβής ροή γύρω από κινούμενη NACA0012, AGARD CT3.



(α′) Cl vs α

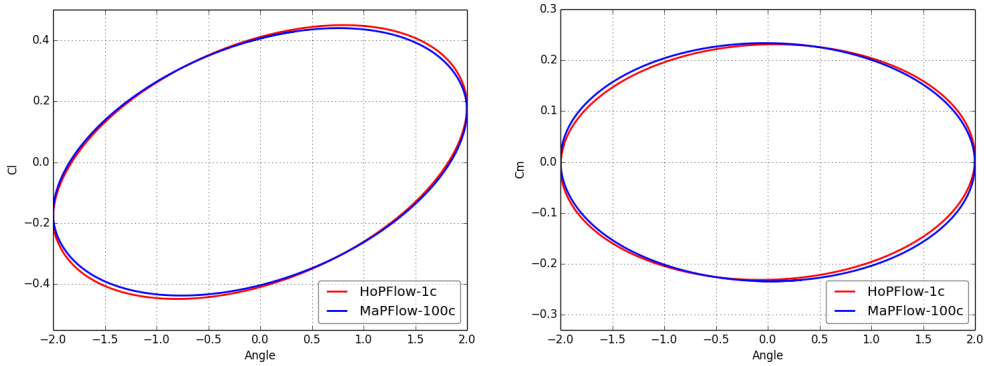


(β′) Cm vs α

Σχήμα 4.16: Cl και Cm κατανομές. Ατριβής ροή γύρω από κινούμενη NACA0012, AGARD CT5.

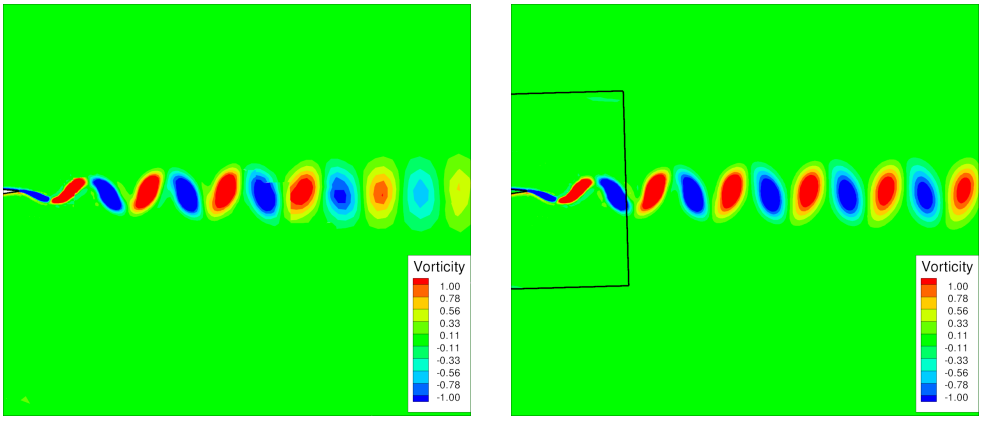
Η σύγκριση των βρόχων Cl και Cm στα Σχήματα 4.14, 4.15 και 4.16 δείχνει ικανοποιητική συμφωνία των αποτελεσμάτων. Πιο συγκεκριμένα σε όλα τα αποτελέσματα του HoPFlow η κλίση του Cl βρόχου είναι μεγαλύτερη σε σχέση με αυτή του CFD, κάτι που υποδεικνύει μικρότερη διάχυση στον υβριδικό επιλυτή. Για καλύτερη διερεύνηση των ιδιοτήτων διάχυσης του υβριδικού επιλυτή εξετάζεται η περίπτωση της κινούμενης αεροτομής σε μεγάλη συχνότητα.

Στο Σχήμα 4.17 παρουσιάζονται οι βρόχοι του Cl και του Cm στην περίπτωση των δύο επιλυτών όπου φαίνεται ότι οι δύο επιλυτές προλέγουν αποτελέσματα σε πολύ καλή συμφωνία. Παρόλα αυτά, εξετάζοντας τον ομόρρου στις δύο περιπτώσεις (Σχήμα 4.18) φαίνεται ότι στην περίπτωση του υβριδικού επιλυτή (HoPFlow) η αριθμητική διάχυση είναι πολύ μικρότερη από αυτήν του CFD επιλυτή.



(α′) Cl vs α (β′) Cm vs α

Σχήμα 4.17: Cl και Cm κατανομές. Ατριβής ροή γύρω από κινούμενη NACA0012. Περίπτωση μεγάλης συχνότητας



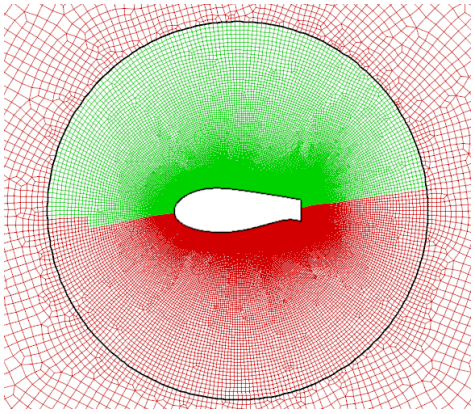
(α′) Ομόρους MaPFlow (β′) Ομόρους HoPFlow

Σχήμα 4.18: Ισουψείς στροβιλότητας πίσω από την κινούμενη αεροτομή NACA0012

4.2.4 Συνεκτική ροή γύρω από την αεροτομή FB-3750-1750

Σε αυτήν την περίπτωση εξετάζεται συνεκτική ροή γύρω από την αεροτομή FB-3750-1750. Η συγκεκριμένη αεροτομή έχει την ιδιαιτερότητα ότι έχει αποκομμένη ακμή εκφυγής. Τέτοιου τύπου αεροτομές χρησιμοποιούνται στους δρομείς ανεμογεννητριών κοντά στη ρίζα του πτερυγίου και έχουν διερευνηθεί υπολογιστικά και πειραματικά στα [62, 63, 64, 65].

Όπως και προηγουμένως τα αποτελέσματα του υβριδικού επιλυτή συγκρίνονται με τα αποτελέσματα του CFD επιλυτή αλλά και με πειραματικά δεδομένα όπου υπάρχουν. Για τους CFD υπολογισμούς χρησιμοποιήθηκε πλέγμα έκτασης 50 χορδών με y+≈ 1. Στην περίπτωση των υβριδικών υπολογισμών το πλέγμα που χρησιμοποιήθηκε ήταν υποσύνολο του μεγαλύτερου έκτασης ≈ 1.5 χορδής. Το μοντέλο τύρβης που χρησιμοποιήθηκε σε όλες της περιπτώσεις είναι το SA. Οι συνθήκες ροής συνοψίζονται στον Πίνακα 4.7.



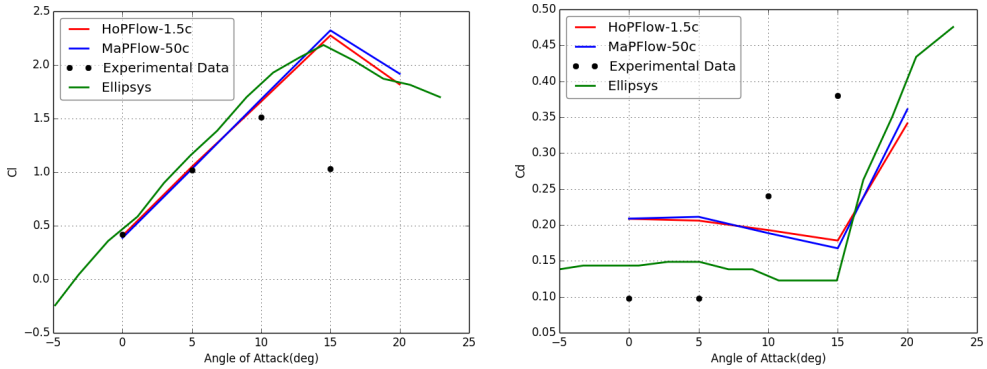
Σχήμα 4.19: FB-3750-1750: Το Υπολογιστικό πλέγμα.

Αυτή η περίπτωση επιλέχθηκε με σκοπό να αξιολογηθεί η ικανότητα πρόλεξης του υβριδικού επιλυτή σε αποκολλημένες ροές. Η αποκόλληση προκύπτει από την αποκομμένη ακμή εκφυγής από την οποία δημιουργούνται ισχυροί στρόβιλοι στον ομόρρου.

Πίνακας 4.7: Συνθήκες Ροής - Αεροτομή FB-3750-1750

Mach	Reynolds	Angle of Attack	Δt
0.15	$0.6 \cdot 10^6$	$0^o, 5^o, 10^o, 15^o, 20^o$	0.002

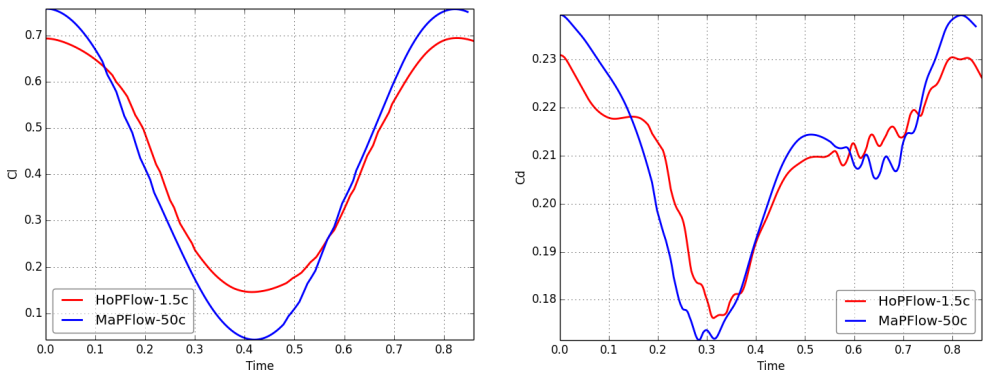
Στο Σχήμα 4.20 απεικονίζονται οι μέσες τιμές των Cl και Cd στις γωνίες προσβολής που εξετάστηκαν. Αν και όλοι οι επιλυτές προλέγουν το σημείο αποκόλλησης αργότερα από το πείραμα η μεταξύ τους σύγκριση θεωρείται ικανοποιητική.



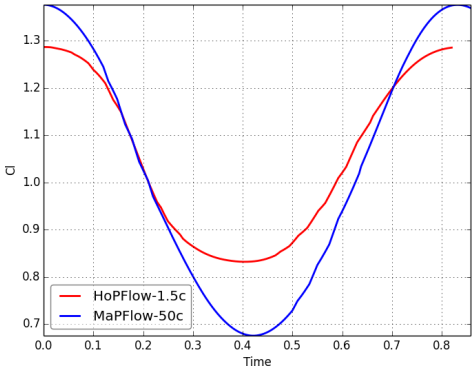
(α′) Μέσο Cl vs α. (β′) Μέσο Cd vs α.

Σχήμα 4.20: Κατανομές των μέσων Cl και Cd για την αεροτομή FB-3750-175 σε$Re = 0.6\,10^6$.Αποτελέσματα παρουσιάζονται από τους κώδικες MaPFlow, HoPFlow και Ellipsys [65] αλλά και με πειραματικά δεδομένα από [62].

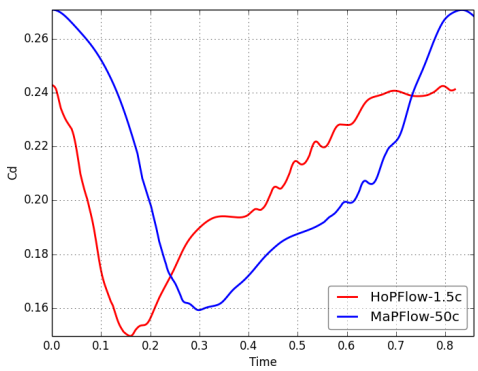
Στα Σχήματα 4.21 παρουσιάζονται οι χρονοσειρές του Cl και του Cd σε μία περίοδο. Ο υβριδικός επιλυτής (HoPFlow) όπως και ο CFD (MaPFlow) προλέγουν την ίδια βασική συχνότητα σε όλες τις γωνίες πρόσπτωσης όπως φαίνεται και από τον αριθμό Strouhal (Σχήμα 4.22). Όσο αναφορά το Cl, αν και υπάρχει συμφωνία στην βασική συχνότητα τα αποτελέσματα διαφέρουν ως προς τα εύρη του σήματος με τον υβριδικό επιλυτή να προλέγει μικρότερα εύρη. Αντίθετα, στις χρονοσειρές του Cd οι διαφορές μεταξύ των επιλυτών είναι μεγαλύτερες με τον υβριδικό επιλυτή να προλέγει και διέγερση σε υψηλώτερες αρμονικές.



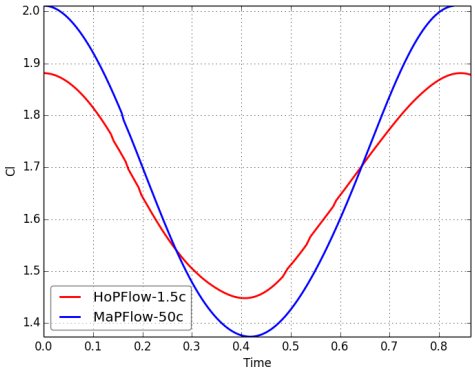
(α′) Cl σε μία περίοδο στις 0° (β′) Cd σε μία περίοδο στις 0°



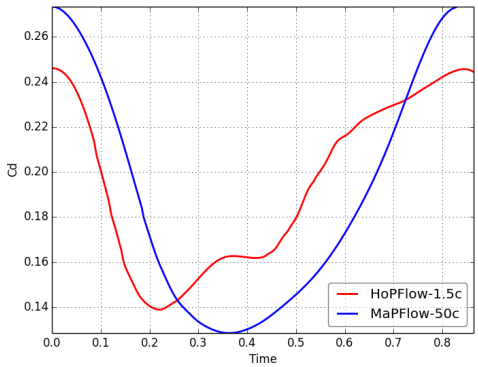
(γ′) Cl σε μία περίοδο στις 5°



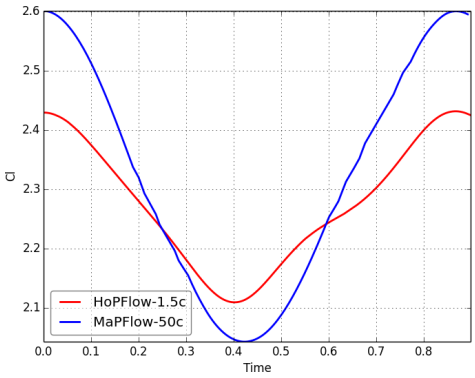
(δ′) Cd σε μία περίοδο στις 5°



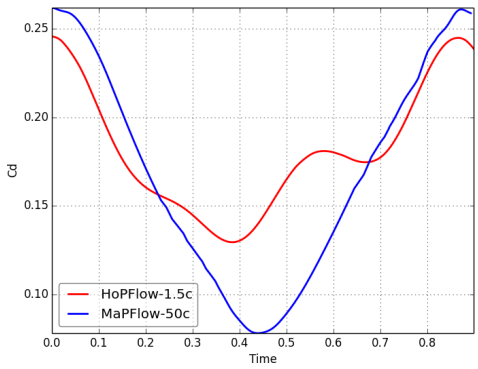
(ε′) Cl σε μία περίοδο στις 10°



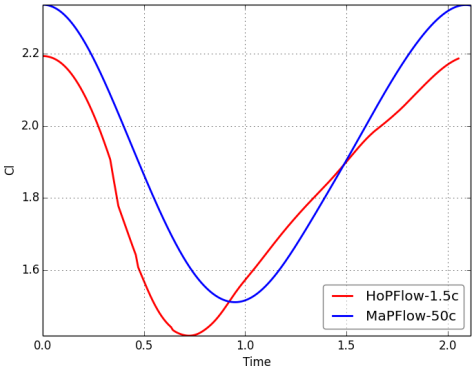
(στ′) Cd σε μία περίοδο στις 10°



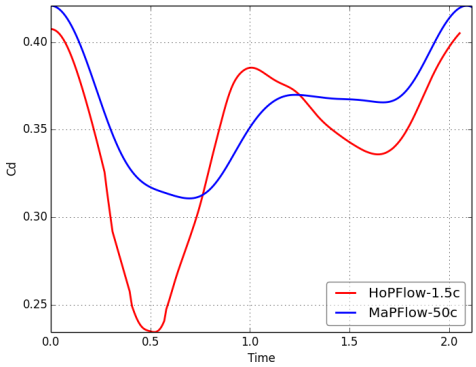
(ζ′) Cl σε μία περίοδο στις 15°



(η′) Cd σε μία περίοδο στις 15°

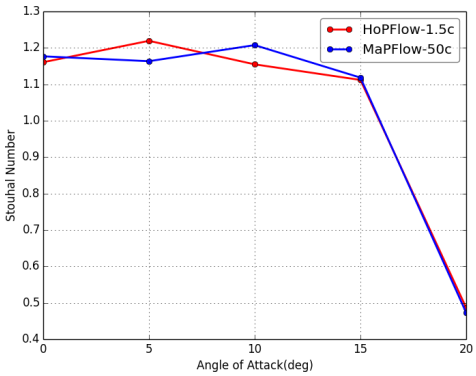


(θ′) Cl σε μία περίοδο στις 20°



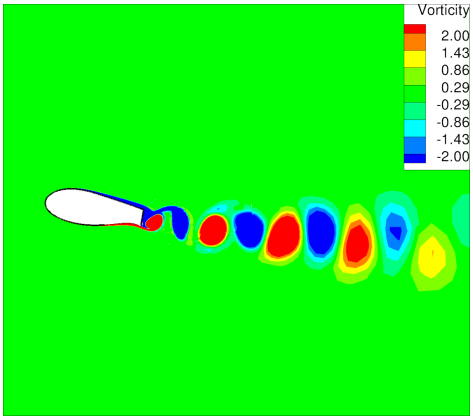
(ι′) Cd σε μία περίοδο στις 20°

Σχήμα 4.21: Σήματα Cl (αριστερά) και Cd (δεξιά) σε μία περίοδο στις $0^o, 5^o, 10^o, 15^o$ και 20°

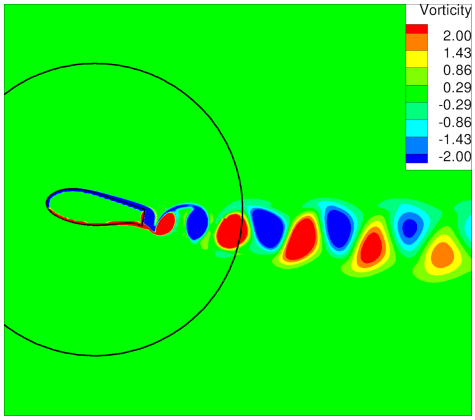


Σχήμα 4.22: Κατανομή αριθμού Strouhal ($Str = f \cdot U_\infty$)

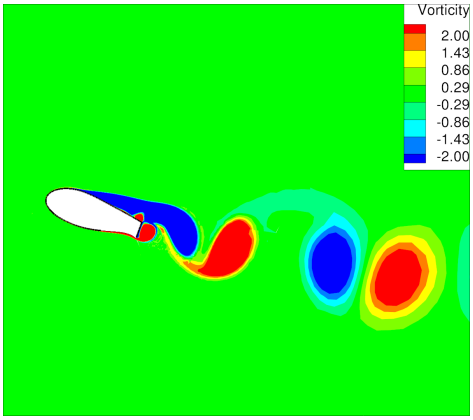
Η διαφορά στα σήματα του Cl και του Cd μπορεί να αποδοθεί στα διαφορετικά χαρακτηριστικά του ομόρρου που προλέγουν οι δύο επιλυτές. Στο Σχήμα 4.23 απεικονίζονται ισουψείς στροβιλότητας στην περιοχή του κοντινού ομόρρου για τους δύο επιλυτές. Ο υβριδικός επιλυτής προλέγει πιο ισχυρές δομές στροβιλότητας, ενώ στον CFD επιλυτή η διάχυση αποδυναμώνει τους στροβίλους. Αφού ο όμορρους παίζει καθοριστικό ρόλο στην δυναμική των φορτιών είναι φυσιολογικό η διαφορετική εξέλιξη του ομόρρου να οδηγεί σε διαφορές στις χρονοσειρές των φορτίων. Αν και υπάρχουν διαφορές στη λεπτομέρεια των αποτελεσμάτων, ποιοτικά οι δύο επιλυτές προλέγουν παρόμοια αποτελέσματα.



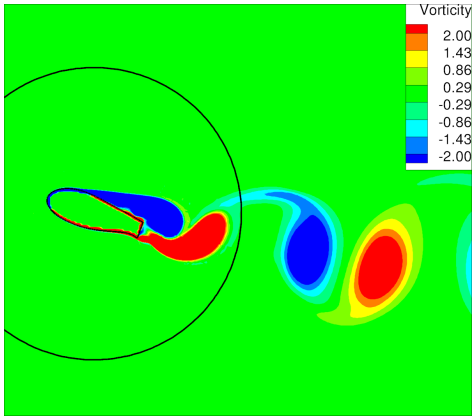
(α′) Ανάπτυξη του ομόρρου για $\alpha = 10^o$ στην περίπτωση του CFD - MaPFlow



(β′) Ανάπτυξη του ομόρρου για $\alpha = 10^o$ στην περίπτωση του Υβριδικού επιλυτή - HoPFlow



(γ′) Ανάπτυξη του ομόρρου για $\alpha = 20^o$ στην περίπτωση του CFD - MaPFlow

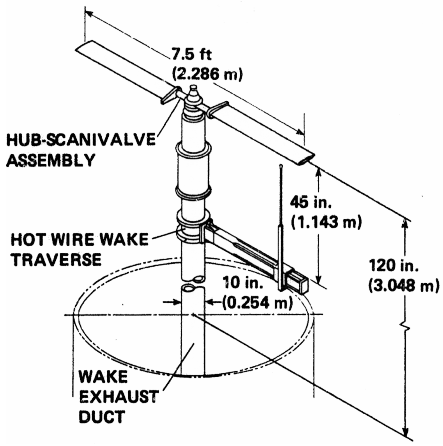


(δ′) Ανάπτυξη του ομόρρου για $\alpha = 20^o$ στην περίπτωση του Υβριδικού επιλυτή - HoPFlow

Σχήμα 4.23: Ανάπτυξη του ομόρρου πίσω απο την αεροτομή FB-3750-1750 στίς 10° καί 20°. Η διαφορετική δομή του ομόρου οδηγεί σε διαφορετικά φορτία πάνω στην αεροτομή

4.2.5 Δρομέας Ελικοπτέρου Caradonna-Tung σε αιώρηση

Σε αυτήν την περίπτωση εξετάζεται η τριδιάστατη ροή γύρω από το δρομέα ελικοπτέρου Caraddonna-Tung σε αιώρηση. Πειραματικά δεδομένα για αυτόν το δρομέα μπορούν να βρεθούν στο [66] ενώ η πειραματική διάταξη που χρησιμοποιήθηκε φαίνεται στο Σχήμα 4.24. Σκοπός σε αυτή την περίπτωση είναι να πιστοποιηθεί ο υβριδικός επιλυτής σε μια σχετικά απλή τριδιάστατη περίπτωση με κινούμενα σώματα. Σε αυτή την περίπτωση χρησιμοποιήθηκε $DX_{PM} = 0.1$.



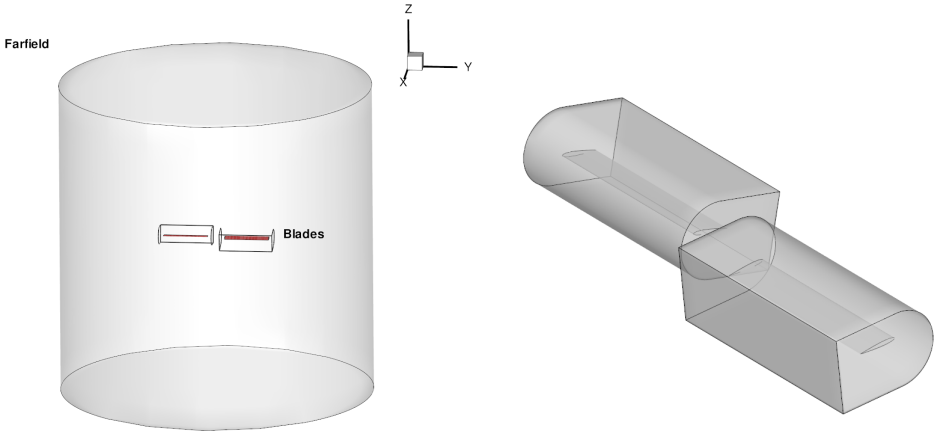
Σχήμα 4.24: Πειραματική Διάταξη

Η περίπτωση που εξετάζεται αφορά συμπιεστή ροή αφού ο αριθμός Mach στο ακροπτερύγιο είναι Ma=0.877. Οι συνθήκες ροής συνοψίζονται στον Πίνακα 4.8.

Πίνακας 4.8: Συνθήκες ροής για τον δρομέα Caradonna-Tung σε αιώρηση

Mach Tip	Αριθμός Reynolds	Γωνία βήματος	$\Omega(RPM)$	Δt
0.877	Inviscid	8°	2500	0.05 (720 βήματα στην περίοδο)

Το υπολογιστικό πλέγμα που χρησιμοποιήθηκε στην περίπτωση των CFD αποτελεσμάτων αποτελείται από $5 \cdot 10^6$ κελιά. Το πλέγμα που χρησιμοποιήθηκε στον υβριδικό επιλυτή ήταν $1.3 \cdot 10^6$ κελιά. Μοντελοποιήθηκαν και τα δύο πτερύγια του δρομέα ώστε να υπάρχει απευθείας σύγκριση με τον υβριδικό επιλυτή. Όπως φαίνεται και στο Σχήμα 4.25β΄ στην περίπτωση του υβριδικού επιλυτή τα δύο πλέγματα, ένα σε κάθε πτερύγιο, είναι ανεξάρτητα μεταξύ τους.

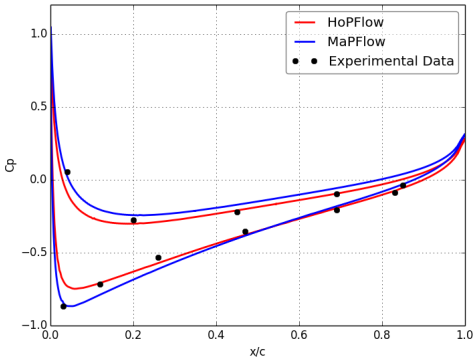


(α′) Το υπολογιστικό πλέγμα που χρησιμοποιήθηκε στο MaPFlow

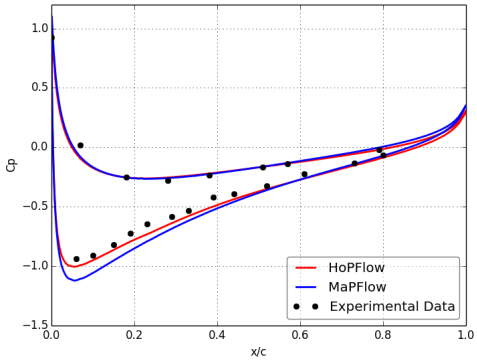
(β′) Το υπολογιστικό πλέγμα που χρησιμοποιήθηκε στο HoPFlow

Σχήμα 4.25: Τα υπολογιστικά πλέγματα

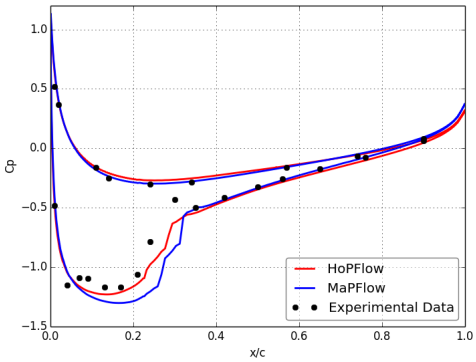
Στο Σχήμα 4.26 φαίνονται οι κατανομές πίεσης σε 5 ακτινικές θέσεις. Οι δύο επιλυτές δίνουν συγκρίσιμα αποτελέσματα που είναι επίσης κοντά στα πειραματικά δεδομένα. Ο υβριδικός επιλυτής συγκρίνεται καλύτερα με τα πειραματικά δεδομένα εκτός από τη θέση r/R=50% (Σχήμα 4.26). Αντίθετα πλησιάζοντας στο ακροπτερύγιο, οι προλέξεις του υβριδικού επιλυτή είναι πιο κοντά στα πειραματικά δεδομένα σε σύγκριση με την κατανομή πιέσεων του CFD. Αυτή η διαφορά αποδίδεται στο ότι ο υβριδικός επιλυτής μοντελοποιεί τον ομόρρου με μικρότερη αριθμητική διάχυση. Διαφορές στον ομόρρου οδηγούν σε διαφορές στην φαινόμενη γωνία πρόσπτωσης και έτσι δικαιολογούνται οι διαφορές μεταξύ των δύο επιλυτών.



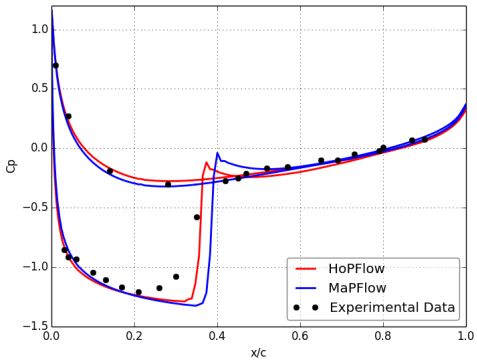
(α′) Κατανομή Cp στη θέση r/R=50%



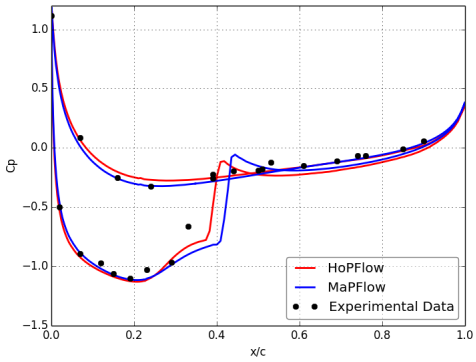
(β′) Κατανομή Cp στη θέση r/R=68%



(γ′) Κατανομή Cp στη θέση r/R=80%



(δ′) Κατανομή Cp στη θέση r/R=89%



(ε′) Κατανομή Cp στη θέση r/R=96%

Σχήμα 4.26: Δρομέας Caradonna-Tung: Κατανομές πιέσεων σε πέντε ακτινικές θέσεις.

This page intentionally left blank.

Κεφάλαιο 5

Συμπεράσματα & Μελλοντική Έρευνα

5.1 Συμπεράσματα

Σκοπός τής εργασίας ήταν η ανάπτυξη ενός συμπιεστού υβριδικού CFD επιλυτή για την προσομοίωση εξωτερικών αεροδυναμικών ροών. Σε αυτήν τη κατεύθυνση:

- Αναπτύχθηκε ένας συμπιεστός URANS επιλυτής (MaPFlow).
- Αναπτύχθηκε ένας συμπιεστός Lagrangian επιλυτής .
- Έγινε η σύζευξη των δύο σε ένα υβριδικό CFD επιλυτή (HoPFlow).

MaPFlow

Ο κώδικας MaPFlow επιλύει τις μη-μόνιμες εξισώσεις Navier-Stokes σε μη δομημένα πλέγματα. Η επίλυση γίνεται με τη χρήση της μεθόδου πεπερασμένων όγκων, ενώ μπορεί να προσομοιώσει ροές σε όλους τους αριθμού Mach. Τα αποτελέσματα που παρουσιαστήκαν, στα οποία έγινε σύγκριση με πειραματικά δεδομένα αλλά και με αριθμητικά αποτελέσματα, πιστοποιήθηκε η ικανότητα του CFD επιλυτή να προλέγει με ακρίβεια εξωτερικές ροές σε διδιάστατα και τριδιάστατα προβλήματα.

Ο Lagrangian Επιλυτής

Αναπτύχθηκε ένας συμπιεστός Lagrangian Επιλυτής. Τα στοιχεία στροβιλότητας που χρησιμοποιούνται μεταφέρουν μάζα, στροβιλότητα, μεταβολή όγκου και ενέργεια, ενώ ο όγκος τους μεταβάλλεται με το χρόνο. Η συγκεκριμένη διατύπωση συμπεριλαμβάνει και την ενέργεια σαν μεταβλητή μεταφοράς, ώστε να μπορούν να προσομοιωθούν ισχυρά συμπιεστές ροές. Επειδή η συγκεκριμένη διατύπωση απαιτεί ο

υπολογιστικός χώρος να καλυφθεί με στοιχεία στροβιλότητας, το υπολογιστικό κόστος είναι μεγάλο. Για αυτόν το λόγο χρησιμοποιήθηκε η τεχνική Particle Mesh για την επιτάχυνση των υπολογισμών.

HoPFlow

Στη συνέχεια έγινε σύζευξη των δύο προηγούμενων μεθόδων σε ένα υβριδικό υπολογιστικό εργαλείο. Προκειμένου να ικανοποιηθούν με ακρίβεια οι συνθήκες τοίχου, ο CFD επιλυτής χρησιμοποιήθηκε κοντά στα στερεά όρια ενώ ο Lagrangian επιλυτής χρησιμοποιήθηκε για να παρέχει τις εξωτερικές συνοριακές συνθήκες στα CFD πλέγματα. Με την ισχυρή σύζευξη των δύο μεθόδων δημιουργήθηκε μια υβριδική μέθοδος με τα εξής χαρακτηριστικά:

- **Εύκολη διαχείριση διατάξεων πολλαπλών σωμάτων:** Στον υβριδικό επιλυτή τα υπολογιστικά πλέγματα είναι μικρά σε έκταση. Στην περίπτωση πολλαπλών σωμάτων χρησιμοποιείται ένα ξεχωριστό πλέγμα για κάθε σώμα και συνεπώς κάθε σώμα μπορεί να κινείται ανεξάρτητα από τα υπόλοιπα. Οι αλληλεπιδράσεις μεταξύ των σωμάτων μεταφέρονται μέσω του Lagrangian επιλυτή.

- **Υψηλή ακρίβεια στην περιοχή κοντά στα στερεά όρια:** Το γεγονός ότι η έκταση των CFD πλεγμάτων περιορίζεται στη μία χορδή μας επιτρέπει να χρησιμοποιήσουμε περισσότερα κελιά στην περιοχή κοντά στα στερεά όρια. Στους διδιάστατους υπολογισμούς το κέρδος σε υπολογιστικά κελιά λόγω της μικρότερης έκτασης δεν είναι μεγάλο, καθώς το 70% του πλέγματος ή και περισσότερο βρίσκεται στην περιοχή κοντά στο σύνορο. Στις τριδιάστατες περιπτώσεις όμως το κέρδος είναι αρκετά μεγαλύτερο καθώς η περιοχή κοντά στο όριο κυμαίνεται στο 20-40% του πλέγματος.

- **Καλύτερη μοντελοποίηση του ομόρρου:** Στην περίπτωση του CFD επιλυτή το υπολογιστικό πλέγμα αραιώνει σταδιακά καθώς προσεγγίζεται το εξωτερικό όριο του πλέγματος. Αυτή η σταδιακή αραίωση του πλέγματος εισάγει αριθμητική διάχυση που αλλοιώνει γρήγορα τα χαρακτηριστικά του ομόρρου. Αντίθετα όπως φάνηκε και στο κεφάλαιο των αποτελεσμάτων ο υβριδικός επιλυτής διατηρεί τον ομόρρου αναλλοίωτο για πολύ μεγάλη έκταση.

Ο υβριδικός επιλυτής εξετάστηκε σε διδιάστατες και τριδιάστατες ροές σε πληθώρα αριθμών Mach και Reynolds. Αν και οι περιπτώσεις που εξετάστηκαν προσομοιώνονται εξίσου ικανοποιητικά και από τον CFD επιλυτή , επιλέχθηκαν σαν μια αρχική διερεύνηση των χαρακτηριστικών του υβριδικού επιλυτή. Τα αποτελέσματα που παρουσιάστηκαν δείχνουν ότι η υβριδική μέθοδος που προτείνεται μπορεί να προσομοιώσει με ακρίβεια μεγάλη πληθώρα εξωτερικών αεροδυναμικών ροών.

5.2 Προτάσεις για Μελλοντική Έρευνα

Οι προτάσεις για μελλοντική έρευνα μπορούν να κατηγοριοποιηθούν ανάλογα με το αν αφορούν τη μεθοδολογία ή εφαρμογές.

Μεθοδολογικές προτάσεις:

Αριθμητική Ολοκλήρωση: Στην παρούσα εργασία η αριθμητική ολοκλήρωση στον Lagrangian επιλυτή ήταν πρώτης τάξης. Αυτό είχε ως συνέπεια τον περιορισμό του χρονικού βήματος της υβριδικής μεθόδου. Αύξηση του επιτρεπόμενου χρονικού βήματος και άρα μείωση του κόστους απαιτεί τη χρήση αριθμητικής ολοκλήρωσης ανώτερης τάξης στο Lagrangian κομμάτι.

Περιοχή χαμηλών Αριθμών Mach: Παρατηρήθηκε ότι καθώς ο αριθμός Mach πλησίαζε την ασυμπίεστη περιοχή ο περιορισμός του χρονικού βήματος γινόταν όλο και μεγαλύτερος. Είναι γνωστό από τις Navier-Stokes εξισώσεις ότι καθώς μικραίνει ο αριθμός Mach η ευστάθεια του συστήματος μειώνεται και για αυτόν το λόγο χρησιμοποιούνται τα μητρώα προσταθεροποίησης. Παρόμοιες τεχνικές θα πρέπει να διερευνηθούν και για τον Lagrangian επιλυτή.

Αριθμητική Διάχυση: Όπως προαναφέρθηκε, η κατανομή των στοιχείων στροβιλότητας μπορεί να μειώσει την ακρίβεια του Lagrangian επιλυτή. Για αυτό το λόγο ανά τακτά χρονικά βήματα γίνεται remeshing. Όμως η αναδιανομή των στοιχείων στροβιλότητας εισάγει διάχυση στα όρια της κατανομής στροβιλότητας. Σε αυτήν την κατεύθυνση θα πρέπει να διερευνηθούν σχήματα remeshing με την λιγότερη δυνατή αριθμητική διάχυση.

Επίδραση Συνεκτικότητας: Στην περίπτωση των μεθόδων στοιχείων στροβιλότητας η επίδραση της συνεκτικότητας λαμβάνεται υπόψη με τις μεθόδους Particle Strength Exchange (PSE). Αυτές οι μέθοδοι έχουν εφαρμοστεί επιτυχώς σε ροές με σταθερή συνεκτικότητα αλλά στην περίπτωση τυρβωδών ροών είτε θα πρέπει να συμπληρωθεί με τη μεταβλητή στο χώρο τυρβώδη συνεκτικότητα είτε να προσομοιωθούν οι κλίμακες της τύρβης με μεθόδους τύπου LES. Σημειώνεται σχετικά ότι μακρυά από στερεά σύνορα η τυρβώδης συνεκτικότητα παραμένει μικρή οπότε δεν αναμένονται μεγάλες αποκλίσεις. Στη περίπτωση που στοχεύουμε στην προσομοίωση των κλιμάκων της τύρβης προτείνεται η χρήση μεθόδων πολλαπλών κλιμάκων με τις οποίες οι στοιχειακές μέθοδοι ταιριάζουν καλά. .

Υπολογιστικό κόστος: Στην παρούσα εργασία σκοπός ήταν η διατύπωση και η πιστοποίηση της υβριδικής μεθόδου και όχι η απομείωση του κόστους. Παρόλα αυτά, το υπολογιστικό κόστος είναι μια πολύ σημαντική παράμετρος και συνεπώς θα πρέπει να γίνουν βήματα προς την ποσοτικοποίηση και την απομείωση του κόστους του υβριδικού επιλυτή.

Επιπλεόν CFD τεχνικές: Ο CFD επιλυτής που αναπτύχθηκε στα πλαίσια της διδακτορικής διατριβής δεν χρησιμοποιεί σημαντικές υπολογιστικές τεχνικές όπως η αυτόματη πύκνωση του πλέγματος σε περιοχές μεγάλων κλίσεων και η τεχνική Multigrid. Τέτοιες τεχνικές θα μπορούσαν να ενσωματωθούν στον υβριδικό επιλυτή.

Προτάσεις εφαρμογής

Όπως αναφέρεται και στην εισαγωγή ο σκοπός του υβριδικού επιλυτή είναι η προσομοίωση πολύπλοκων προβλημάτων που αφορούν γεωμετρίες πολλαπλών σωμάτων και αλληλεπίδραση ομόρρου στερεού. Σε αυτήν τη κατεύθυνση προτείνεται να εξεταστούν:

1. Ροές γύρω από σώματα που κινούνται ελεύθερα μεταξύ τους, όπως η ροή γύρω από Α/Γ καθέτου άξονα και η αλληλεπίδραση κυρίως δρομέα και ουραίου δρομέα σε ελικόπτερα.

2. Ροές όπου η ανάπτυξη και εξέλιξη του ομόρρου είναι καθοριστικής σημασίας, όπως η προσομοίωση ροών σε αιολικά πάρκα, η αλληλεπίδραση στροβιλότητας στερεού συνόρου.

3. Η αερο-ελαστική συμπεριφορά Α/Γ και ελικοπτέρων.

4. Η παραγωγή θορύβου σε ελικόπτερα.

5. Η Δημιουργία συνθετικής τύρβης χρησιμοποιώντας στοιχεία στροβιλότητας σε συνδυασμό με μεθόδους LES και η ανάλυση της επίδρασης της τύρβης σε κατασκευές όπως για παράδειγμα η πραγματική φόρτιση Α/Γ..

Βιβλιογραφία

[1] P. Jawahar and H. Kemath. A high-resolution procedure for Euler and Navier-Stokes computations on unstructured grids. *J. Computational Physics*, 164:165–203, 2000.

[2] Forrester T Johnson, Edward N Tinoco, and N Jong Yu. Thirty years of development and application of cfd at boeing commercial airplanes, seattle. *Computers & Fluids*, 34(10):1115–1151, 2005.

[3] Charles Hirsch. *Numerical Computation of Internal and External Flows.* Wiley, 1990.

[4] Randall J LeVeque. *Finite volume methods for hyperbolic problems*, volume 31. Cambridge university press, 2002.

[5] Bernardo Cockburn and Chi-Wang Shu. Runge–kutta discontinuous galerkin methods for convection-dominated problems. *Journal of scientific computing*, 16(3):173–261, 2001.

[6] Thomas A Zang, Craig L Streett, and M Yousuff Hussaini. Spectral methods for cfd. 1989.

[7] A. Quarteroni and A. Valli. *Domain decomposition methods for partial differential equations.* Oxford Science Publications, 1999.

[8] G. Carey. *Computational Grids: Generation, Adaptation, and Solution Stategies.* Taylor & Francis, 1997.

[9] D.J. Mavriplis. Accurate multigrid solution of the Euler equations on unstructured and adaptive meshes. *AIAA Journal*, 28:213–221, 1990.

[10] A. Haselbacher and J. Blazek. Accurate and efficient discretization of Navier-Stokes equations on mixed grids. *AIAA Journal*, 38:2094–2102, 2000.

[11] ZJ Wang. A fully conservative interface algorithm for overlapped grids. *Journal of Computational Physics*, 122(1):96–106, 1995.

[12] R Steijl and G Barakos. Sliding mesh algorithm for cfd analysis of helicopter rotor–fuselage aerodynamics. *International journal for numerical methods in fluids*, 58(5):527–549, 2008.

[13] G-H. Cottet and P.D. Koumoutsakos. *Vortex methods: Theory and Practice.* Cambridge University Press, 2000.

[14] Joe J Monaghan. Smoothed particle hydrodynamics. *Reports on progress in physics*, 68(8):1703, 2005.

[15] R. Gingold and J. Monaghan. Shock simulation by the particle method SPH. *J. Computational Physics*, 52:374–389, 1983.

[16] L. Barba, A. Leonard, and C. Allen. Advances in viscous vortex methods - meshless spatial adaptation based on radial basis function interpolation. *Int. J. Numerical Methods in Fluids*, 47:387–421, 2005.

[17] P.D. Koumoutsakos and A. Leonard. High resolution simulations of the flow around an impulsively started cylinder using vortex methods. *Fluid Mechanics*, 296:1–38, 1995.

[18] SG Voutsinas, MA Belessis, and KG Rados. Investigation of the yawed operation of wind turbines by means of a vortex particle method. In *AGARD CONFERENCE PROCEEDINGS AGARD CP*, pages 11–11. AGARD, 1995.

[19] Spyros G Voutsinas and Dimitris G Triantos. Aeroacoustics of full helicopter configurations using vortex particle flow approximation. In *Proceedings of CEAS Forum on Aeracoustics of Rotors and Propellers, Rome, Italy*, pages 175–91. DTIC Document, 1999.

[20] Philippe Chatelain, Alessandro Curioni, Michael Bergdorf, Diego Rossinelli, Wanda Andreoni, and Petros Koumoutsakos. Billion vortex particle direct numerical simulations of aircraft wakes. *Computer Methods in Applied Mechanics and Engineering*, 197(13):1296–1304, 2008.

[21] S.G. Voutsinas. Vortex Methods in Aeronautics: How to make things work. *Int. Journal of Computational Fluid Dynamics*, 20, 2006.

[22] G.K. Batchelor. *An Introduction to Fluid Mechanics.* Cambridge University Press, UK, 1967.

[23] J. Katz and A. Plotkin. *Low speed aerodynamics.* Cambridge University Press, UK, 2001.

[24] S. Huberson and S.G. Voutsinas. Particles and grids. *Computers and Fluids*, 31:607–625, 2002.

[25] R. W. Hockney and J. W. Eastwood. *Computer Simulation Using Particles.* McGraw-Hill, 1981.

[26] B. Couet, O. Buneman, and A. Leonard. Simulation of three-dimensional incompressible flows with a vortex in cell method. *J. Computational Physics*, 39:305–328, 1981.

[27] L. Greengard and V. Rokhlin. A Fast Algorithm for Particle Simulations. *J. Computational Physics*, 73, 1987.

[28] L. Greengard and V. Rokhlin. A new version of the fast multipole method for the Laplace equation in three dimensions. *Acta Numerica*, 7:229–269, 1997.

[29] J.T. Beale and A. Majda. Vortex methods II: Higher order accuracy in 2 and 3 dimensions. *Math. Comput.*, 32:29–52, 1982.

[30] M. Bergdorf, G.-H. Cottet, and P.D. Koumoutsakos. Multilevel adaptive particle methods fr convection-diffusion equations. *SIAM Multiscale Modeling and Simulation*, 4:328–357, 2005.

[31] G-H. Cottet and A. Magni. TVD remeshing formulas for particle methods. *C.R. Math*, 347:1367–1372, 2009.

[32] I.F. Sbalzarini, J.H. Walther, M. Bedrgdorf, S.E. Hieber, and P.D. Koumoutsakos. PPM - A highly efficient parallel particle-mesh library for the simulation of continuum systems. *J. Computational Physics*, 215:566–588, 2006.

[33] Alexandre Joel Chorin. Numerical study of slightly viscous flow. *Journal of Fluid Mechanics*, 57(04):785–796, 1973.

[34] E Rivoalen and S Huberson. The particle strength exchange method applied to axisymmetric viscous flows. *Journal of Computational Physics*, 168(2):519–526, 2001.

[35] J.D. Eldredge, T. Colonius, and A. Leonard. A vortex particle method for two-dimensional compressible flow. *J. Computational Physics*, 179:371–399, 2002.

[36] Monika Nitsche and James H Strickland. Extension of the gridless vortex method into the compressible flow regime*. *Journal of Turbulence*, 3:40–40, 2002.

[37] S Mas-Gallic, M Louaked, and O Pironneau. A particle in cell method for the 2-d compressible euler equations. In *Vortex flows and related numerical methods*, pages 373–387. Springer, 1993.

[38] R Meske. *The vortex-in-cell method for compressible flow.* PhD thesis, PhD thesis, Imperial College of Science and Technology, London, UK, 1994.

[39] P. K. Chaviaropoulos, I. G. Nikolaou, K. A. Aggelis, M. N. N. Soerensen andJ. Johansen, O. L. Hansen, Mac Gaunaa, T. Hambraus, Heiko Frhr. von Geyr, Ch. Hirsch, Kang Shun, S. G. Voutsinas, G. Tzabiras, Y. Perivolaris, and S. Z. Dyrmose. Viscous and Aeroelastic Effects on Wind Turbine Blades. The VISCEL project. Part I: 3D Navier-Stokes Rotor simulations. *Wind Energy*, 6:365–385, 2003.

[40] Sven Schmitz and Jean-Jacques Chattot. A coupled Navier–Stokes/Vortex–Panel solver for the numerical analysis of wind turbines. *Computers & Fluids*, 35:742–745, 2006.

[41] G. S. Oxley. *A 2-D Hybrid Euler-Compressible Vortex Particle Method For Transonic Rotorcraft Flows.* PhD thesis, Dept. Mech. & Aero. Engineering, Carleton University, Ottawa, Canada, 2009.

[42] F.R. Menter. Zonal Two Equation k-omega Turbulence Models for Aerodynamic Flows. *AIAA Paper 93-2906*, 1993.

[43] P.R. Spalart and S.R. Allmaras. One-Equation Turbulence Model for Aerodynamic Flows. *Recherche Aerospatiale*, pages 5–21, 1994.

[44] RA James. The solution of poisson's equation for isolated source distributions. *Journal of Computational Physics*, 25(2):71–93, 1977.

[45] Karl Lackner. Computation of ideal mhd equilibria. *Computer Physics Communications*, 12(1):33–44, 1976.

[46] V. Venkatakrishnan. On the Accuracy of Limiters and Convergence to Steady State Solutions. *AIAA paper 93-0880*, 1993.

[47] Jiri Blazek. *Computational Fluid Dynamics: Principles and Applications.* Elsevier Science, 2001.

[48] Gregory T. Balls and Phillip Colella. A Finite Difference Domain Decomposition Method Using Local Corrections for the Solution of Poisson's Equation. *J. Computational Physics*, 180:25–53, 2002.

[49] R. Boisvert. A fourth order accurate fast direct method for the Helmholtz equation. *Academic Press*, pages 35–44, 1984.

[50] J. Guermond, S. Huberson, and W. Shen. Simulation of 2D external viscous flows by means of a domain decomposition method. *J. Computational Physics*, 108:343–352, 1993.

[51] P. Plouhmans, G.S. Winckelmans, J.K. Salmon, A. Leonard, and M.S.Warren. Vortex methods for direct numerical simulation of three-dimensional bluff body flows: Application to the sphere at Re=300, 500, and 1000. *J. Computational Physics*, 178:427–436, 2002.

[52] NASA Turbulence Modeling Resource. `http://turbmodels.larc.nasa.gov/`.

[53] CFL3D Navier Stokes Code. `http://cfl3d.larc.nasa.gov/`.

[54] N. Gregory and C.L. O'Reilly. Low-Speed Aerodynamic Characteristics of NACA 0012 Aerofoil Sections, including the Effects of Upper-Surface Roughness Simulation Hoar Frost. *NASA R&M 3726*, Jan 1970.

[55] NPARC Alliance Validation Archive. `http://www.grc.nasa.gov/WWW/wind/valid/homepage.html`.

[56] *AGARD Advisory Report No. 138, Experimental data base for computer program assessment.* 1979.

[57] M Maureen Hand, DA Simms, L Fingersh, D Jager, J Cotrell, S Schreck, and S Larwood. *Unsteady aerodynamics experiment phase V: test configuration and available data campaigns.* National Renewable Energy Laboratory, 2001.

[58] Niels N Sørensen, JA Michelsen, and S Schreck. Navier–stokes predictions of the nrel phase vi rotor in the nasa ames 80 ft× 120 ft wind tunnel. *Wind Energy*, 5(2-3):151–169, 2002.

[59] Mukesh Marutrao Yelmule and EswaraRao Anjuri VSJ. Cfd predictions of nrel phase vi rotor experiments in nasa/ames wind tunnel. *International Journal of Renewable Energy Research (IJRER)*, 3(2):261–269, 2013.

[60] David A Simms, S Schreck, M Hand, and LJ Fingersh. *NREL unsteady aerodynamics experiment in the NASA-Ames wind tunnel: a comparison of predictions to measurements.* National Renewable Energy Laboratory Colorado, USA, 2001.

[61] R. Landon. NACA0012, oscillatory and transient pitching. *AGARD Report No.707*, pages 3.1–3.25, 1982.

[62] Jonathon P Baker, CP Van Dam, and Benson L Gilbert. Flatback airfoil wind tunnel experiment. *Report, Sandia National Laboratories, Albuquerque, NM and Livermore, CA, USA*, 2008.

[63] JP Baker and CP van Dam. Drag Reduction of Blunt Trailing-Edge Airfoils, BBAA VI International Colloquium on: Bluff Bodies Aerodynamics & Applications. Technical report, Milano, Italy, 2008.

[64] Edward A Mayda, CP van Dam, David D Chao, and Dale E Berg. Computational design and analysis of flatback airfoil wind tunnel experiment. Technical report, Sandia National Laboratories, 2008.

[65] Niels N Sørensen. A Small Study of Flatback Airfoils. In *Aeroelastic Workshop*, pages 153–195.

[66] F.X. Caradonna and C. Tung. Experimental and Analytical Studies of a Model Helicopter Rotor in Hover. *NASA Technical Memorandum 81232, National Aeronautics and Space Administration*, 1981.