POLITECNICO DI MILANO
Corso di Laurea Magistrale in Ingegneria Aeronautica
Scuola di Ingegneria Industriale e dell'Informazione
Dipartimento di Scienze e Tecnologie Aerospaziali

Computing pressure from PIV velocity measurements: a novel approach

Relatore: Prof. Franco Auteri

Tesi di Laurea di:
Davide Montagnani, matricola 783329
Daniele Zagaglia, matricola 782836

Anno Accademico 2012-2013

Contents

List of Figures

List of Tables

Abstract

The purpose of this work is to develop an innovative procedure for computing the pressure field from PIV velocity measurements for unsteady, incompressible flows. This new technique aims at solving some of the problems affecting the methods proposed so far in the literature, such as the use of inaccurate boundary conditions for the pressure and insufficient robustness with respect to measurement errors. The method is based on a generalization of the Glowinski-Pironneau method for the uncoupled pressure-velocity solution of the incompressible Navier–Stokes equations. A finite element approximation of the problem is introduced, and a Fortran90 program for the proposed method has been developed. The method is first applied on an exact solution of the Navier–Stokes equations, in order to check its convergence properties. The robustness of the method with respect to the error in the velocity measurements is tested for both stochastic and deterministic perturbations. Then the proposed technique has been applied to the PIV database of a classical aeronautical problem: the pitching airfoil for the study of dynamic stall. The computed pressure is compared with direct pressure measurements, showing very encouraging results. As a final, challenging application, the pressure field around a Gurney flap mounted on the trailing edge of a pitching airfoil has been reconstructed.

Keywords: Pressure computation, Particle Image Velocimetry (PIV), Uncoupled Navier–Stokes equations, Glowinski–Pironneau method, Finite Element Method (FEM), Dynamic stall.

Sommario

Lo scopo di questo lavoro di tesi è lo sviluppo di una procedura innovativa per il calcolo del campo di pressione da misure di velocità PIV, per flussi instazionari ed incomprimibili. Questa nuova tecnica cerca di risolvere alcuni problemi riscontrati nei metodi proposti fino ad oggi, quali l'uso di condizioni al contorno inaccurate per la pressione o la sensibilità all'errore di misura della velocità. Il metodo si basa su una generalizzazione del metodi Glowinski–Pironneau per il disaccoppiamento delle equazioni di Navier Stokes. Si introduce quindi un'approssimazione ad elementi finiti del problema, che è stata implementata in un programma Fortran90. Si procede quindi a provarlo su una soluzione esatta delle equazioni di Navier–Stokes, mostrando le sue proprietà di convergenza. Successivamente viene testata la robustezza del metodo agli errori di misura di velocità, con perturbazioni sia stocastiche che deterministiche. Il metodo è quindi applicato al database PIV di un problema tipicamente aeronautico: il profilo oscillante per lo studio dello stallo dinamico. La pressione così calcolata viene confrontata con misure dirette di pressione, mostrando risultati molto incoraggianti. Come applicazione finale, il metodo è usato per calcolare la distribuzione di pressione attorno a un'aletta Gurney montata sul bordo d'uscita di un profilo oscillante.

Keywords: Calcolo della pressione, Particle Image Velocimetry (PIV), Equazioni di Navier–Stokes disaccoppiate, Metodo di Glowinski–Pironneau, Metodo agli elementi finiti (FEM), Stallo Dinamico.

Acknowledgements

We would really like to thank our supervisor Prof. Franco Auteri, whose continuous support and help has been of invaluable importance to us.

We would also like to express our gratitude to Prof. Luigi Quartapelle for his precious advice on the theoretical aspects and for providing us with the FEM Fortran90 primitives.

Eventually we would like to thank Ing. Alex Zanotti and Ing. Marco Carini for their help throughout our thesis work.

Chapter 1

Introduction

In this chapter the challenge of computing pressure from PIV velocity measurements is introduced. In section 1.1 a brief description of the Particle Image Velocimetry technique is reported, explaining the working principle and the major advantages of this technique. Then in section 1.2 the problem of computing pressure from PIV measurements is approached, explaining the motivation for this relatively new issue and the state of the art so far. In section 1.3 an innovative approach to this problem is proposed, which has been developed in the present thesis. Eventually in section 1.4 the outline of the thesis is illustrated.

1.1 Particle Image Velocimetry (PIV)

1.1.1 The working principle

The rapid evolution of both digital image acquisition and computing capability in the past few decades has determined a dramatic development of image-based experimental techniques for the characterization of fluid flows. Among these techniques, Particle Image Velocimetry (PIV) is definitely the most important, since it is able to provide two or three-component velocity fields with acceptable accuracy (up to 1 %) at many measurement points in the flow, simultaneously. In fact, differently from standard flow visualizations, which provide global but only qualitative information, or laser-Doppler velocimetry, which gives accurate but local velocity information, PIV is able to provide at the same time global and quantitative instantaneous information about the flow field.

The working principle of PIV is based on the measurement of the displacement of small tracer particles convected by the fluid by comparing two digital images taken with a short time delay between them. By measuring

the displacement of the particles in pixels (as it will be discussed in section 1.1.2) in the $i-$th direction (Δs_i) and given the time delay between the two successive frames (Δt) and the meters/pixel ratio of the image (C), is it possible to reconstruct locally the $i-$th velocity component as follows:

$$U_i = C\frac{\Delta s_i}{\Delta t} \tag{1.1}$$

The typical PIV experimental set-up is depicted in Figure 1.1, which is available at `http://www.dlr.de/as/en/Portaldata/5/Resources/images/abteilungen/abt_ev/artikel/PIV_img1_xl.jpg`. PIV can be seen as a non-intrusive measurement technique, since apart the tracer particles and the laser light sheet no other perturbation is introduced inside the test section. The tracer particles must be sufficiently small ($\leq 1\,\mu$m in air) in order to follow properly the fluid motion without any appreciable modification to the properties of the flow. In order to obtain robust unbiased measurements over the flow domain, the tracer particles have to be homogeneously distributed within the observed flow region. The tracer particles are illuminated by means of a thin light sheet, which is pulsed in order to freeze the particle motion. The use of a laser light source is determined by the fact that it can produce a high-energy, pulsed, collimated monochromatic light beam that can be easily shaped into a thin light sheet. The light scattered by the tracer particles is recorded at two instants in time using a digital camera. By placing more than a camera with a proper angle with respect to the laser sheet, is it possible to resolve all the three velocity components (stereoscopic PIV). Otherwise, by placing only one camera with its recording plane parallel to the laser sheet, it is possible to resolve only the in-plane components of the velocity field . In the present work only this second configuration will be examined in detail. Synchronization among all the system components is crucial. The laser pulses, camera acquisition and exposure triggers must be accurately synchronized, especially for high speed flow.

Each pair of PIV frames is then stored in order to be post-processed, as described in the following section 1.1.2.

1.1.2 Image processing

Once acquired, each gray-level image can be regarded as an array of $M \times N$ pixels, each one being characterized by a discrete value of the gray level (corresponding to the local light intensity). The number of gray levels is typically 256 (8 bit images).

The two images of a pair of consecutive frames are divided into smaller regions, called interrogation windows. An example of two sub-images, cor-

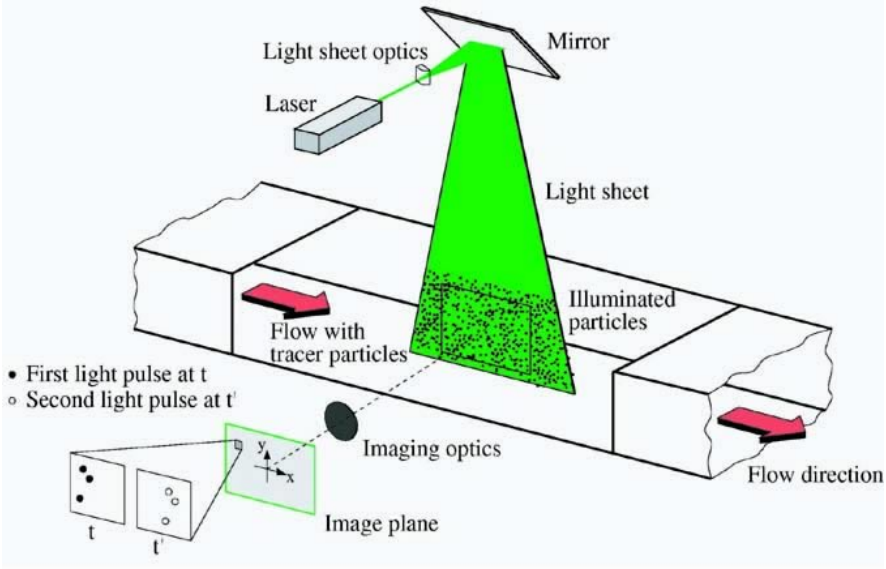
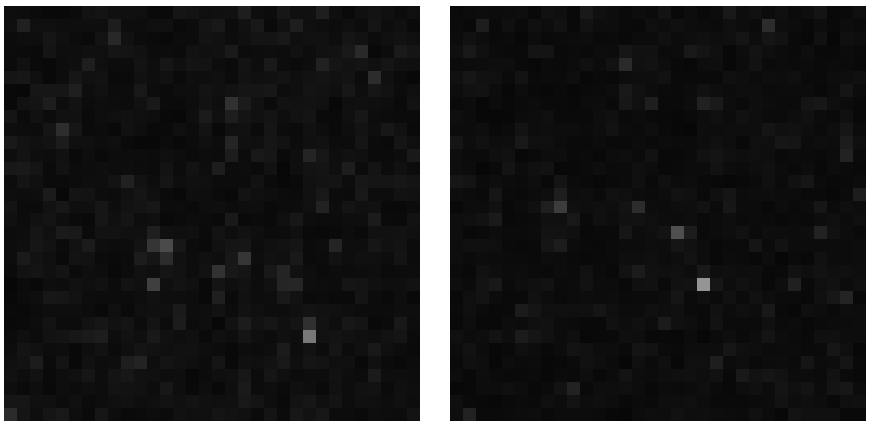


Figure 1.1: PIV measurement system

responding to the same interrogation window at different time instants, is depicted in Figure 1.2. Comparing two consecutive images we can appreciate that most of the tracer particles seem to move towards the same direction, in this example North-West. The mean displacement of all the tracer particles can be computed by a spatial cross-correlation among the two images corresponding to the same measurement window. The cross correlation among the two images previously described is depicted in figure 1.3. As can be appreciated, the correlation function shows a peak, whose coordinates represents the average displacement of the tracing particles, which represents an approximation of the velocity field, once divided by the time interval, see Eq. 1.1.

The signal to noise ratio (SNR) of the correlation, which can be defined as the ratio between the maximum value reached at the main peak and the mean value of the background, can be used as a criterion for the acceptance of the computed velocity: if the SNR is less than a suitable threshold, the local velocity measurement is refused. The missing measurement can be replaced at a later stage by a suitable average of the data coming from the neighbouring interrogation windows.

With the method above, the displacement can be computed up to a pixel uncertainty. This will lead to a 10 % uncertainty in the velocity computation considering standard values of Δt and C, which is generally not acceptable. As described in [1], page 327, a sub-pixel interpolation can be performed in



(a) Interrogation window at t (b) Interrogation window at t + Δt

Figure 1.2: Interrogation windows

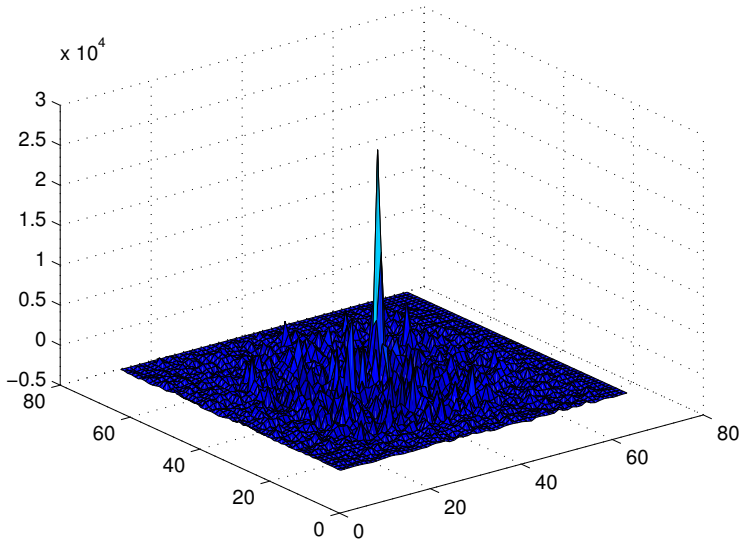


Figure 1.3: Image spatial cross-correlation

order to achieve higher accuracy. With this expedient, accuracies up to 1 % can be achieved.

The above procedure is then repeated for every interrogation window in which the measurement domain has been divided, giving the global velocity field.

Before computing the correlation, several different pre-processings can be carried out on the two images in order to improve the correlation quality, such as histogram stretching, equalization, min-max filtering. For further details, see [1], pages 318-320.

1.2 Computing pressure from PIV measurements

1.2.1 Motivation

The problem of reconstructing the pressure field starting from velocity measurements is quite recent, since only using Particle Image Velocimetry the required resolution of the velocity vector field can be achieved. Berton et al. in [2] computed the airloads on an helicopter blade using Laser Doppler velocity measurements. However in order to measure an entire velocity field, the experiment had to be repeated for every different measurement volume, each one using a different positioning of the laser head, which is very time consuming. The fast development of PIV rapidly increased the interest in techniques able to estimate the aerodynamic loads on immersed bodies exploiting the measured velocity field, since, as described in section 1.1.1, the PIV measurement technique is able to provide the instantaneous flow field with a sufficient accuracy.

Generally speaking the possibility of computing the pressure field directly from the PIV measurements is relevant for several reasons that are:

- the possibility to have information on the pressure field over the whole domain, highlighting, for instance, strong vortical structures.
- the possibility to have a non intrusive measurement technique for computing the pressure on an immersed body and therefore the airloads by direct integration.
- the possibility to have information of the pressure value in situations where a direct measurement of pressure is not possible, such as for instance the regions far away from the immersed body and the regions of the body where the pressure taps can not be easily placed, such as the trailing edge.

1.2.2 State of the art

In the past years, two different approaches have been proposed to compute the pressure field from the PIV measurements. The first one is based on the solution of a Poisson problem for the pressure, while the second one involves the direct integration of the momentum equation by means of finite differences. The two methodologies are discussed in the following sections, highlighting their respective advantages and flaws.

Solution of a Poisson problem for pressure

In 1999 Gurka et al. proposed in [3] a method in which the pressure field is computed as the solution of a Poisson equation. This equation is obtained by taking the divergence of the momentum equation:

$$-\nabla^2 p = \rho \boldsymbol{\nabla} \cdot (\boldsymbol{u} \cdot \boldsymbol{\nabla}) \boldsymbol{u}, \tag{1.2}$$

and it is supplemented with boundary conditions which are based on heuristic assumptions about the physics of the problem, but are nevertheless arbitrary. As an example, a Neumann boundary condition can be derived by evaluating the momentum equation on the boundary in order to estimate $\boldsymbol{\nabla} p$ and then $\frac{\partial p}{\partial n}$. The aim of the authors was mainly to smooth the velocity computational error through the smearing properties of the laplacian term. In their work the method was applied to a steady laminar pipe flow and to a steady turbulent impinging jet, taking into account the Reynolds stress terms.

The same procedure has been applied by other authors to different problems. In [4], the method was applied to the flow around a circular cylinder with and without rotational oscillation and the fluid forces were studied in the lower Reynolds number range. In [5] Obi et al. investigated the flow around a pair of bluff bodies set in tandem in a uniform flow.

In any case, all these solutions depend significantly from the type of boundary conditions enforced.

Integration of the pressure gradient

An alternative technique, proposed by Bauer et al. in [6] consists in the spatial integration of the pressure gradient obtained from the momentum equation of the Navier-Stokes equations by means of finite differences:

$$\boldsymbol{\nabla} p = -\rho \left(\frac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u} \cdot \boldsymbol{\nabla}) \boldsymbol{u} \right) + \mu \nabla^2 \boldsymbol{u}. \tag{1.3}$$

Starting from a point, where the value of pressure can be imposed arbi-

trarily, the pressure gradient is integrated in both directions, thus generating the pressure values in all the other nodes. The main problem with this method is due to the fact that the pressure error is accumulated at each integration. In this work the method was applied to the unsteady pressure field generated by a wall-mounted obstacle.

This method has been extended by Van Oudheusden, Scarano et al. to the compressible case in [7], exploiting the gas law and the adiabatic flow condition instead of the continuity equation, since errors in the divergence computation due to tridimensionality and to out-of-plane velocity gradients could accumulate.

In [8], Liu and Katz computed the pressure field of a cavity using a Lagrangian approach in order to estimate the material derivative of velocity, instead of the separate computation of the time derivative and of the advection term. As shown at a later stage by Violato et al. in [9], this approach yields more accurate estimates of the pressure gradient when compared with the Eulerian one.

This method has been applied to several different flow configurations and geometries. In [10], de Kat and van Oudheusden applied the method to the turbulent flow past a square cylinder. In [11] Ghaemi et al. analyzed the unsteady pressure field generated by a turbulent boundary layer, with the use of a tomographic PIV system and with the application of three-dimensional momentum equation.

More details about the above mentioned techniques can be found in the work of Charonko et al. [12], where a detailed comparison of several methods for calculating the pressure field from 2D time-resolved velocity fields in incompressible flows is presented and the need of proper filtering to mitigate the velocity error propagation in the pressure estimation is analyzed.

1.3 Description of the proposed method

As reported in section 1.2, the methods proposed so far present both positive and negative aspects. On one hand, the method which employs the solution of the Poisson equation for pressure can smooth the measurements errors on the velocity through the smearing properties of the laplacian, but lacks of consistent boundary conditions, which makes the solution extremely sensitive to the type of boundary condition which is enforced. On the other hand, the method which integrates directly the pressure gradient from the momentum equation is definitely consistent, but it is not very accurate due to the accumulation of the error at every integration.

The innovative method proposed in this work aims at combining both

the advantages of the two above methods. Indeed, through a generalization of the Glowinski-Pironneau method for the uncoupling of the Navier-Stokes equations, the use of pressure integral condition and a decomposition of the variables, it is indeed possible to obtain a new uncoupled formulation of the problem, where the boundary condition for pressure can be enforced exactly, not depending on the type of flow considered. Moreover, since every equation involved in this formulation ivolves a laplacian term, the method can provide a smoothing offect on the errors affecting the velocity measurements.

The present methodology is intended to be applied to 2D/3D, unsteady, incompressible flows either laminar or turbulent. The problem, once recast in a variational formulation, is solved by means of Finite Elements. A Fortran90 program which implements this new technique has been produced, and then, after its validation, employed to analyse the PIV database of a typical aeronautical problem: the flow around a pitching airfoil, which is intended to replicate the dynamic stall process.

1.4 Outline of this work

This work is organized as follows. In Chapter 1, the problem of computing the pressure field from PIV velocity measurements is introduced. A brief description of the Particle Image Velocimetry technique is reported, explaining the working principle and the major advantages of this technique. Then the problem of computing pressure from PIV measurements is approached, motivating the need for such technique and summarizing the state of the art on this topic.

In Chapter 2 the mathematical formulation of the method is considered. Once the Navier-Stokes equations and their Reynolds-averaged formulation have been introduced, two different methods to uncouple the computation of velocity and pressure are described, both involving pressure integral conditions in order to impose the incompressibility costraint over the whole domain. Then a procedure to project the experimental velocity field on a divergence free space is described. Eventually, the equations of the second uncoupling method (Glowinski-Pironneau) are recast in weak form in order to be discretized in the following chapter.

In Chapter 3 the discretization of the continuous problem is performed. First, the computational domain is discretized, employing the PIV measurement points. Then, a finite element (FEM) discretization of the problem is introduced and it is used to obtain a set of algebraic linear systems. Eventually, the implementation in a Fortran90 program is briefly described.

In Chapter 4 the results of the program verification and validation are

illustrated. Each component of the program is individually validated and its convergence properties are evaluated. Then, the whole program is validated on an unsteady exact solution of the Navier-Stokes equation. The robustness of the method is then investigated for both deterministic and stochastic errors.

In Chapter 5 the test rig and the experimental set-up necessary to obtain the velocity field database, which will be post-processed at a later stage in order to compute the pressure field, are described. The dynamic stall experimental set-up, comprising both PIV and unsteady pressure measurements, is introduced. Then, the test matrix and the reference test conditions (steady/unstedy, light/deep stall, angle of attack, PIV measurement windows) are thoroughly described. A modified set-up, with the Gurney flap attached on the airfoil trailing edge, is described. Eventually a procedure to estimate the uncertainty on the pressure coefficient, obtained with the proposed method, is described.

In Chapter 6, the pressure field around a NACA 23012 airfoil is computed starting from experimental data. Several steady and unsteady flow conditions are considered, and the computed pressure distribution on the airfoil surface is compared with the pressure distribution measured by Kulite sensors. A few different version of the proposed method are tested and the results are compared in order to evaluate their respective performance.

In Chapter 7 we draw some conclusions and outline possible future developments of the proposed method.

Chapter 2

Mathematical formulation

In this chapter the mathematical formulation of the method is described. In section 2.1 the Navier-Stokes equations and their Reynolds-averaged form are introduced. Afterwards in section 2.2 and 2.3, two different methods to uncouple the computation of velocity and pressure are described, both involving pressure integral conditions in order to impose the incompressibility costraint over the whole domain. Then a procedure to project the experimentally measured velocity field on a divergence free functional space is described in section 2.4. Eventually in section 2.5 the equations of the second method are recast in their weak formulation that will be discretized at a later stage in chapter 3.

2.1 Equations for incompressible flows

2.1.1 Navier–Stokes and RANS equations

Let us consider a control volume V and its surface $S = \partial V$. The unsteady incompressible Navier–Stokes equations which model the flow incompressible behaviour, together with their boundary condition are:

$$\begin{cases} \dfrac{\partial \boldsymbol{U}}{\partial t} + (\boldsymbol{U} \cdot \boldsymbol{\nabla})\boldsymbol{U} - \nu \nabla^2 \boldsymbol{U} + \boldsymbol{\nabla} P = 0 & \text{in V} \\ \boldsymbol{\nabla} \cdot \boldsymbol{U} = 0 & \\ \boldsymbol{U}\big|_S = \boldsymbol{B} & \text{on S,} \end{cases} \tag{2.1}$$

where $\boldsymbol{U}(\boldsymbol{x},t)$ represents the velocity vector field and $P(\boldsymbol{x},t)$ represents the pressure field divided by the fluid density. It is useful to note here that, in the particular application we are envisaging, $\boldsymbol{B} = \boldsymbol{B}(\boldsymbol{x},t^n)$ is the velocity measured on the boundary of the computational domain and therefore is known, so that we can enforce full Dirichlet boundary conditions on the

whole boundary ∂V. Since we are interested in the phase averaged of the pressure field around the boundary, we introduce the corresponding version of the Reynolds averaged Navier-Stokes (RANS) equations,

$$\begin{cases} \dfrac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u}\cdot\boldsymbol{\nabla})\boldsymbol{u} + \langle(\boldsymbol{u}'\cdot\boldsymbol{\nabla})\boldsymbol{u}'\rangle - \nu\nabla^2\boldsymbol{u} + \boldsymbol{\nabla}p = 0 & \text{in V} \\ \boldsymbol{\nabla}\cdot\boldsymbol{u} = 0 & \\ \boldsymbol{u}\big|_S = \boldsymbol{b} & \text{on S,} \end{cases} \tag{2.2}$$

where $\boldsymbol{u} = \langle \boldsymbol{U} \rangle$ and $\boldsymbol{u}' = \boldsymbol{U} - \boldsymbol{u}$ and similarly for pressure. In addition, the compatibility condition $\oint_S \boldsymbol{b}\cdot\boldsymbol{n} = 0$ is assumed to be satisfied. However this is not true in our particular application, since the measured velocity field is affected by the measurement error.

2.1.2 Time discretized RANS equations

Since the velocity field is known only at discrete time intervals, we introduce the time discretization of RANS equations:

$$\begin{cases} \dfrac{\boldsymbol{u}^n - \boldsymbol{u}^{n-1}}{\Delta t} + (\boldsymbol{u}^n\cdot\boldsymbol{\nabla})\boldsymbol{u}^n + \left\langle(\boldsymbol{u}'^{,n}\cdot\boldsymbol{\nabla})\boldsymbol{u}'^{,n}\right\rangle - \nu\nabla^2\boldsymbol{u}^n = -\boldsymbol{\nabla}p^n & \text{in V} \\ \boldsymbol{\nabla}\cdot\boldsymbol{u}^n = 0 & \\ \boldsymbol{u}^n\big|_S = \boldsymbol{b}^n & \text{on S.} \end{cases} \tag{2.3}$$

An implicit Euler scheme has been adopted for simplicity, but other discretization schemes could be used without changing the substance of the method.

Let us divide the momentum equation by the cinematic viscosity ν and denote the unknown fields at the new time level by $\boldsymbol{u} = \boldsymbol{u}^n$ and $p = p^n$, where now the letter p indicates the scaled pressure $p \leftarrow \nu^{-1}p$. Then, we have

$$\begin{cases} (-\nabla^2 + \gamma)\boldsymbol{u} + \boldsymbol{\nabla}p = \boldsymbol{g} & \text{in V} \\ \boldsymbol{\nabla}\cdot\boldsymbol{u} = 0, & \\ \boldsymbol{u}\big|_S = \boldsymbol{b} & \text{on S,} \end{cases} \tag{2.4}$$

where $\gamma = \frac{1}{\nu\Delta t}$ and $\boldsymbol{g} = \gamma\boldsymbol{u}^{n-1} - \nu^{-1}[(\boldsymbol{u}^n\cdot\boldsymbol{\nabla})\boldsymbol{u}^n + \langle(\boldsymbol{u}'^n\cdot\boldsymbol{\nabla})\boldsymbol{u}'^n\rangle]$, which is an experimentally known quantity.

Since the current velocity field is experimentally known, the time derivative of the velocity field can also be treated explicitly and moved to the right

hand side of the first equation. One obtains:

$$\begin{cases} -\nabla^2 \boldsymbol{u} + \boldsymbol{\nabla} p = \tilde{\boldsymbol{g}} & \text{in V} \\ \boldsymbol{\nabla} \cdot \boldsymbol{u} = 0, & \\ \boldsymbol{u}\big|_S = \boldsymbol{b} & \text{on S,} \end{cases} \tag{2.5}$$

where $\tilde{\boldsymbol{g}} = \gamma(\boldsymbol{u}^{n-1} - \boldsymbol{u}^n) - \nu^{-1}[(\boldsymbol{u}^n \cdot \boldsymbol{\nabla})\boldsymbol{u}^n + \langle(\boldsymbol{u'}^n \cdot \boldsymbol{\nabla})\boldsymbol{u'}^n\rangle]$. The consequences of this explicit treatment of the time derivative will be discussed later, in Section 2.3.5. In the following sections the method will be derived using the implicit scheme 2.4.

2.2 First Decoupling Method (Quartapelle–Napolitano)

2.2.1 Equations

By taking the divergence of the (time-discretized) momentum equation (2.4) and using the continuity equation, we obtain the following Poisson equation for p:

$$-\nabla^2 p = -\boldsymbol{\nabla} \cdot \boldsymbol{g}. \tag{2.6}$$

As shown in [13], the fulfillment of this equation and of the momentum equation does not assure that the incompressibility condition is satisfied. In fact, let $\boldsymbol{v}$ be a velocity field, not necessarily solenoidal, solution of the momentum equation with p solution of $\nabla^2 p = -\boldsymbol{\nabla} \cdot \boldsymbol{g}$. By taking the divergence of the momentum equation one obtains $(-\nabla^2 + \gamma)\boldsymbol{\nabla} \cdot \boldsymbol{v} + \nabla^2 p = \boldsymbol{\nabla} \cdot \boldsymbol{g}$, that is using 2.6, $(-\nabla^2 + \gamma)\boldsymbol{\nabla} \cdot \boldsymbol{v} = 0$. Thus, $\boldsymbol{\nabla} \cdot \boldsymbol{v}$ is only metaharmonic, but not necessarily equal to zero. However, if one imposes the boundary condition $\boldsymbol{\nabla} \cdot \boldsymbol{v}\big|_S = 0$, then the metaharmonic character of $\boldsymbol{\nabla} \cdot \boldsymbol{v}$ implies that $\boldsymbol{\nabla} \cdot \boldsymbol{v} = 0$ everywhere in V. Thus the uncoupled system equivalent to system 2.4 reads:

$$\begin{cases} (-\nabla^2 + \gamma)\boldsymbol{u} + \boldsymbol{\nabla} p = \boldsymbol{g} & \text{in V} \\ -\nabla^2 p = -\boldsymbol{\nabla} \cdot \boldsymbol{g} & \\ \boldsymbol{u}\big|_S = \boldsymbol{b}, \qquad \boldsymbol{\nabla} \cdot \boldsymbol{u}\big|_S = 0 & \text{on S.} \end{cases} \tag{2.7}$$

As usual, the compatibility condition $\oint_S \hat{\boldsymbol{n}} \cdot \boldsymbol{b} = 0$ is assumed to be satisfied. The problem with the system (2.7) is that there are too many boundary conditions for one variable, the velocity, and not enough for the other, the pressure. Thus, strictly speaking, problem (2.7) consisting of the two elliptic equations subject to the indicated boundary conditions has to be considered

as a *coupled* system of equations, in the sense that the pressure and velocity should be computed simultaneously as a consequence of the coupling brought about by the redundant boundary conditions for $\boldsymbol{u}$. Moreover, should the velocity be known, the pressure equation would lack a boundary condition. For more details about the uncoupling of Navier–Stokes equation, see [14].

2.2.2 Pressure integral condition for the uncoupled equations

The following vector identities are collected here for convenience, since they will be widely used in the following sections:

$$\begin{aligned} \boldsymbol{\nabla}(\boldsymbol{\nabla}\cdot\boldsymbol{w}) &= \boldsymbol{\nabla}\times\boldsymbol{\nabla}\times\boldsymbol{w}+\nabla^2\boldsymbol{w} \\ \boldsymbol{\nabla}\cdot(a\boldsymbol{w}) &= a\boldsymbol{\nabla}\cdot\boldsymbol{w}+\boldsymbol{w}\cdot\boldsymbol{\nabla}a \\ \boldsymbol{\nabla}\cdot(\boldsymbol{v}\times\boldsymbol{w}) &= \boldsymbol{w}\cdot\boldsymbol{\nabla}\times\boldsymbol{v}-\boldsymbol{v}\cdot\boldsymbol{\nabla}\times\boldsymbol{w}. \end{aligned} \tag{2.8}$$

To uncouple the Poisson equation for the pressure from the velocity equation, let us first consider the adjoint problem of the Helmholtz equation for $\boldsymbol{u}$. Its Green identity reads, using properties 2.8:

$$\begin{aligned} &-\int_V \boldsymbol{u}\cdot(-\nabla^2+\gamma)\boldsymbol{v}+\int_V \boldsymbol{v}\cdot(-\nabla^2+\gamma)\boldsymbol{u}=\int_V\Big(\boldsymbol{u}\cdot\nabla^2\boldsymbol{v}-\boldsymbol{v}\cdot\nabla^2\boldsymbol{u}\Big)= \\ &=\int_V\Big[\boldsymbol{u}\cdot\big(\boldsymbol{\nabla}(\boldsymbol{\nabla}\cdot\boldsymbol{v})-\boldsymbol{\nabla}\times\boldsymbol{\nabla}\times\boldsymbol{v}\big)\Big]-\int_V\Big[\boldsymbol{v}\cdot\big(\boldsymbol{\nabla}(\boldsymbol{\nabla}\cdot\boldsymbol{u})-\boldsymbol{\nabla}\times\boldsymbol{\nabla}\times\boldsymbol{u}\big)\Big]= \\ &=\int_V\Big[\boldsymbol{\nabla}\cdot(\boldsymbol{u}\boldsymbol{\nabla}\cdot\boldsymbol{v})-\boldsymbol{\nabla}\cdot\boldsymbol{u}\boldsymbol{\nabla}\cdot\boldsymbol{v}+\boldsymbol{\nabla}\cdot\big(\boldsymbol{u}\times(\boldsymbol{\nabla}\times\boldsymbol{v})\big)-\boldsymbol{\nabla}\times\boldsymbol{u}\cdot\boldsymbol{\nabla}\times\boldsymbol{v}\Big]+ \\ &\quad-\int_V\Big[\boldsymbol{\nabla}\cdot(\boldsymbol{v}\boldsymbol{\nabla}\cdot\boldsymbol{u})-\boldsymbol{\nabla}\cdot\boldsymbol{v}\boldsymbol{\nabla}\cdot\boldsymbol{u}+\boldsymbol{\nabla}\cdot\big(\boldsymbol{v}\times(\boldsymbol{\nabla}\times\boldsymbol{u})\big)-\boldsymbol{\nabla}\times\boldsymbol{v}\cdot\boldsymbol{\nabla}\times\boldsymbol{u}\Big]= \\ &=\int_V\Big[\boldsymbol{\nabla}\cdot(\boldsymbol{u}\boldsymbol{\nabla}\cdot\boldsymbol{v})+\boldsymbol{\nabla}\cdot\big(\boldsymbol{u}\times(\boldsymbol{\nabla}\times\boldsymbol{v})\big)-\boldsymbol{\nabla}\cdot(\boldsymbol{v}\boldsymbol{\nabla}\cdot\boldsymbol{u})-\boldsymbol{\nabla}\cdot\big(\boldsymbol{v}\times(\boldsymbol{\nabla}\times\boldsymbol{u})\big)\Big]= \\ &=\oint_S\Big(\hat{\boldsymbol{n}}\cdot\boldsymbol{u}\boldsymbol{\nabla}\cdot\boldsymbol{v}+\hat{\boldsymbol{n}}\times\boldsymbol{u}\cdot\boldsymbol{\nabla}\times\boldsymbol{v}-\hat{\boldsymbol{n}}\cdot\boldsymbol{v}\boldsymbol{\nabla}\cdot\boldsymbol{u}-\hat{\boldsymbol{n}}\times\boldsymbol{v}\cdot\boldsymbol{\nabla}\times\boldsymbol{u}\Big), \end{aligned} \tag{2.9}$$

which can be rewritten as:

$$\int_V\Big(\boldsymbol{v}\cdot(-\nabla^2+\gamma)\boldsymbol{u}-\boldsymbol{u}\cdot(-\nabla^2+\gamma)\boldsymbol{v}\Big)=\oint_S\Big(\hat{\boldsymbol{n}}\times\boldsymbol{u}\cdot\boldsymbol{\nabla}\times\boldsymbol{v}+\hat{\boldsymbol{n}}\cdot\boldsymbol{u}\boldsymbol{\nabla}\cdot\boldsymbol{v}-\hat{\boldsymbol{n}}\times\boldsymbol{v}\cdot\boldsymbol{\nabla}\times\boldsymbol{u}-\hat{\boldsymbol{n}}\cdot\boldsymbol{v}\boldsymbol{\nabla}\cdot\boldsymbol{u}\Big). \tag{2.10}$$

If $\boldsymbol{v}$ satisfies the following differential problem

$$\begin{cases} (-\nabla^2 + \gamma)\boldsymbol{v} = 0 \quad \text{in V} \\ \hat{\boldsymbol{n}} \cdot \boldsymbol{v}\big|_S \neq 0 \\ \hat{\boldsymbol{n}} \times \boldsymbol{v}\big|_S = 0, \end{cases} \tag{2.11}$$

then the integral condition can be expressed as:

$$\int_V \boldsymbol{v} \cdot (\boldsymbol{g} - \boldsymbol{\nabla} p) = \oint_S \Big(\hat{\boldsymbol{n}} \times \boldsymbol{b} \cdot \boldsymbol{\nabla} \times \boldsymbol{v} + \hat{\boldsymbol{n}} \cdot \boldsymbol{b} \boldsymbol{\nabla} \cdot \boldsymbol{v} \Big), \tag{2.12}$$

where the boundary condition $\boldsymbol{u} = \boldsymbol{b}$ has been exploited. The important point about these integral conditions is that they depend only on the source term $\boldsymbol{g}$ of the momentum equation and on the value $\boldsymbol{b}$ of the velocity prescribed on the boundary. Furthermore, since the "number" of linearly independent metaharmonic fields $\boldsymbol{v}$ is "equal" to that of boundary points minus one, the projection integral conditions (2.12) provide the correct number of conditions to be combined with the Poisson equation in order to obtain a well determined problem for the pressure field, p being defined up to an arbitrary additive constant. It follows that, as shown by [15], problem (2.7) can be reformulated equivalently in the uncoupled or split form :

$$\begin{cases} \begin{cases} -\nabla^2 p = -\boldsymbol{\nabla} \cdot \boldsymbol{g} & \text{in V} \\ -\int_V \boldsymbol{v} \cdot \boldsymbol{\nabla} p = -\int_V \boldsymbol{v} \cdot \boldsymbol{g} + \oint_S \big(\hat{\boldsymbol{n}} \times \boldsymbol{b} \cdot \boldsymbol{\nabla} \times \boldsymbol{v} + \hat{\boldsymbol{n}} \cdot \boldsymbol{b} \boldsymbol{\nabla} \cdot \boldsymbol{v} \big) \\ [\text{where} \quad (-\nabla^2 + \gamma)\boldsymbol{v} = 0 \quad \text{in V}, \quad \hat{\boldsymbol{n}} \cdot \boldsymbol{v}\big|_S \neq 0, \quad \hat{\boldsymbol{n}} \times \boldsymbol{v}\big|_S = 0] \end{cases} \\ (-\nabla^2 + \gamma)\boldsymbol{u} = -\boldsymbol{\nabla} p + \boldsymbol{g} \quad \text{in V}, \quad \boldsymbol{u}\big|_S = \boldsymbol{b} \end{cases} \tag{2.13}$$

2.2.3 Decomposition method in the continuum

Decomposition

To obtain the pressure field satisfying the integral condition 2.12, the solution $p(\boldsymbol{x})$ is decomposed into its harmonic and non harmonic components as follows:

$$p(\boldsymbol{x}) = p^0(\boldsymbol{x}) + \oint_S p'(\boldsymbol{x}; \boldsymbol{\sigma}') \lambda(\boldsymbol{\sigma}') dS(\boldsymbol{\sigma}'), \tag{2.14}$$

where λ represents a surface unknown function with zero average on S, being the pressure field defined up to an additive constant. Thanks to the linearity of the problem the nonharmonic component will take into account the source term while the harmonic component will provide the boundary values.

Equations

The fields $p^0(\boldsymbol{x})$ and $p'(\boldsymbol{x};\boldsymbol{\sigma'})$ are, for every $\boldsymbol{\sigma'} \in S \setminus \boldsymbol{\sigma^\star}$, the solutions to the family of problems:

$$\begin{array}{ll} -\nabla^2 p^0 = -\boldsymbol{\nabla}\cdot\boldsymbol{g} \quad \text{in V}, & p^0\big|_S = 0 \\ -\nabla^2 p' = 0 \quad \text{in V}, & p'(\boldsymbol{x};\boldsymbol{\sigma'})\big|_S = \delta^{(2)}(\boldsymbol{x}-\boldsymbol{\sigma}) - \delta^{(2)}(\boldsymbol{x}-\boldsymbol{\sigma^*}) \end{array} \tag{2.15}$$

Integral condition and linear variational problem

The surface unknown $\lambda(\boldsymbol{\sigma})$ is determined by using the decomposition 2.14 in the integral condition 2.12:

$$\begin{aligned} \int_V \boldsymbol{\chi}(\boldsymbol{x};\boldsymbol{\sigma})\cdot\left[\boldsymbol{g} - \boldsymbol{\nabla}\left(p^0(\boldsymbol{x}) + \oint_S p'(\boldsymbol{x};\boldsymbol{\sigma'})\lambda(\boldsymbol{\sigma'})dS(\boldsymbol{\sigma'})\right)\right] = \\ = \oint_S \left(\hat{\boldsymbol{n}}\times\boldsymbol{b}\cdot\boldsymbol{\nabla}\times\boldsymbol{\chi}(\boldsymbol{x};\boldsymbol{\sigma}) + \hat{\boldsymbol{n}}\cdot\boldsymbol{b}\boldsymbol{\nabla}\cdot\boldsymbol{\chi}(\boldsymbol{x};\boldsymbol{\sigma})\right). \end{aligned} \tag{2.16}$$

By reordering the terms, keeping the unknown terms on the left-hand side and taking the known terms on the right-hand side of the equation, the following linear problem is obtained:

$$\oint_{S(\sigma')} B(\boldsymbol{\sigma},\boldsymbol{\sigma'})\lambda(\boldsymbol{\sigma'})dS_{\boldsymbol{\sigma'}} = \beta(\boldsymbol{\sigma}), \tag{2.17}$$

where:

$$\begin{aligned} B(\boldsymbol{\sigma},\boldsymbol{\sigma'}) &= -\int_V \boldsymbol{\nabla}p'(\boldsymbol{x};\boldsymbol{\sigma'})\cdot\boldsymbol{\chi}(\boldsymbol{x};\boldsymbol{\sigma})dV \\ \beta(\boldsymbol{\sigma}) &= \int_V (\boldsymbol{\nabla}p^0 - \boldsymbol{g})\cdot\boldsymbol{\chi}(\boldsymbol{x};\boldsymbol{\sigma}) + \oint_S \big(\hat{\boldsymbol{n}}\times\boldsymbol{b}\cdot\boldsymbol{\nabla}\times\boldsymbol{\chi}(\boldsymbol{x};\boldsymbol{\sigma}) + \hat{\boldsymbol{n}}\cdot\boldsymbol{b}\boldsymbol{\nabla}\cdot\boldsymbol{\chi}(\boldsymbol{x};\boldsymbol{\sigma})\big). \end{aligned} \tag{2.18}$$

The vector fields $\boldsymbol{\chi}(\boldsymbol{x};\boldsymbol{\sigma})$ are the solution of the following metaharmonic problem for any $\boldsymbol{\sigma} \in S \setminus \boldsymbol{\sigma^\star}$:

$$\begin{cases} (-\nabla^2 + \gamma)\boldsymbol{\chi} = 0 & \text{in V} \\ \hat{\boldsymbol{n}}\cdot\boldsymbol{\chi}\big|_S = \delta^{(2)}(\boldsymbol{x}-\boldsymbol{\sigma}) - \delta^{(2)}(\boldsymbol{x}-\boldsymbol{\sigma^*}) & \\ \hat{\boldsymbol{n}}\times\boldsymbol{\chi}\big|_S = 0. & \end{cases} \tag{2.19}$$

2.2.4 Efficient numerical implementation of the decomposition

The method described in section 2.2.3 is not numerically efficient, because it needs to solve over the whole domain of as many 2.19 systems as the number of Dirichlet boundary nodes and to store them. A more efficient method can be envisaged. In order to do so, auxiliary vector fields $\boldsymbol{v}(\boldsymbol{x};\boldsymbol{\sigma})$ are introduced which are only required to satisfy the boundary conditions of problem 2.19:

$$\begin{cases} \boldsymbol{v} = \text{arbitrary} \quad \text{in V} \\ \hat{\boldsymbol{n}} \cdot \boldsymbol{v}\big|_S = \delta^{(2)}(\boldsymbol{x}-\boldsymbol{\sigma}) - \delta^{(2)}(\boldsymbol{x}-\boldsymbol{\sigma}^*) \\ \hat{\boldsymbol{n}} \times \boldsymbol{V}\big|_S = 0. \end{cases} \tag{2.20}$$

After solving Poisson equations 2.15 for p^0 and p', one has to solve two Helmholtz equations for $\boldsymbol{u}^0$ and $\boldsymbol{u}'$:

$$\begin{array}{lll} -(\nabla^2+\gamma)\boldsymbol{u}^0 = -\boldsymbol{\nabla}p^0 + \boldsymbol{g} & \text{in V}, & \boldsymbol{u}^0\big|_S = \boldsymbol{b}, \\ -(\nabla^2+\gamma)\boldsymbol{u}' = -\boldsymbol{\nabla}p' & \text{in V}, & \boldsymbol{u}'\big|_S = 0. \end{array} \tag{2.21}$$

B and β can be obtained by similar expressions involving $\boldsymbol{v}$ instead of $\boldsymbol{\chi}$. Let us consider first the expression of B. Taking in account the differential problems for $\boldsymbol{u}'$ (equation 2.21) and using the Green identity 2.10 one obtains:

$$\begin{aligned} B(\boldsymbol{\sigma},\boldsymbol{\sigma}') &= -\int_V \boldsymbol{\nabla}p' \cdot \boldsymbol{\chi} = \int_V (-\nabla^2+\gamma)\boldsymbol{u}' \cdot \boldsymbol{\chi} = \\ &= \int_V \boldsymbol{u}' \cdot (-\nabla^2+\gamma)\boldsymbol{\chi} + \\ &+ \oint_S \Big(\hat{\boldsymbol{n}} \times \boldsymbol{u}' \cdot \boldsymbol{\nabla} \times \boldsymbol{\chi} + \hat{\boldsymbol{n}} \cdot \boldsymbol{u}' \boldsymbol{\nabla} \cdot \boldsymbol{\chi} - \hat{\boldsymbol{n}} \times \boldsymbol{\chi} \cdot \boldsymbol{\nabla} \times \boldsymbol{u}' - \hat{\boldsymbol{n}} \cdot \boldsymbol{\chi} \boldsymbol{\nabla} \cdot \boldsymbol{u}' \Big) \end{aligned} \tag{2.22}$$

By using $\boldsymbol{u}'$ homogeneous boundary conditions and $\boldsymbol{\chi}$ differential equation 2.19:

$$B(\boldsymbol{\sigma},\boldsymbol{\sigma}') = -\oint_S \hat{\boldsymbol{n}} \cdot \boldsymbol{\chi} \boldsymbol{\nabla} \cdot \boldsymbol{u}' = -\oint_S \hat{\boldsymbol{n}} \cdot \boldsymbol{v} \boldsymbol{\nabla} \cdot \boldsymbol{u}'. \tag{2.23}$$

Applying the divergence theorem and integrating by parts we obtain:

$$\begin{aligned} B(\boldsymbol{\sigma},\boldsymbol{\sigma}') &= -\oint_S \hat{\boldsymbol{n}} \cdot \boldsymbol{v} \boldsymbol{\nabla} \cdot \boldsymbol{u}' = -\int_V \boldsymbol{\nabla} \cdot (\boldsymbol{v} \boldsymbol{\nabla} \cdot \boldsymbol{u}') = \\ &= -\int_V [\boldsymbol{\nabla} \cdot \boldsymbol{v} \boldsymbol{\nabla} \cdot \boldsymbol{u}' + \boldsymbol{v} \cdot \boldsymbol{\nabla}(\boldsymbol{\nabla} \cdot \boldsymbol{u}')] = \\ &= -\int_V [\boldsymbol{\nabla} \cdot \boldsymbol{v} \boldsymbol{\nabla} \cdot \boldsymbol{u}' + \boldsymbol{v} \cdot (\boldsymbol{\nabla} \times \boldsymbol{\nabla} \times \boldsymbol{u}' + \nabla^2 \boldsymbol{u}')]. \end{aligned} \tag{2.24}$$

Using vector identities 2.8 and boundary conditions on $\boldsymbol{v}$:

$$\begin{aligned} B(\boldsymbol{\sigma},\boldsymbol{\sigma}') &= -\int_V [\boldsymbol{\nabla}\cdot\boldsymbol{v}\boldsymbol{\nabla}\cdot\boldsymbol{u}' + \boldsymbol{v}\cdot\nabla^2\boldsymbol{u}' + \boldsymbol{\nabla}\cdot((\boldsymbol{\nabla}\times\boldsymbol{u}')\times v) + \boldsymbol{\nabla}\times\boldsymbol{u}'\cdot\boldsymbol{\nabla}\times\boldsymbol{v}] = \\ &= -\int_V [\boldsymbol{\nabla}\cdot\boldsymbol{v}\boldsymbol{\nabla}\cdot\boldsymbol{u}' + \boldsymbol{v}\cdot\nabla^2\boldsymbol{u}' + \boldsymbol{\nabla}\times\boldsymbol{u}'\cdot\boldsymbol{\nabla}\times\boldsymbol{v}] + \oint_S \hat{\boldsymbol{n}}\cdot((\boldsymbol{\nabla}\times\boldsymbol{u}')\times\boldsymbol{v}) = \\ &= -\int_V [\boldsymbol{\nabla}\cdot\boldsymbol{v}\boldsymbol{\nabla}\cdot\boldsymbol{u}' + \boldsymbol{v}\cdot\nabla^2\boldsymbol{u}' + \boldsymbol{\nabla}\times\boldsymbol{u}'\cdot\boldsymbol{\nabla}\times\boldsymbol{v}]. \end{aligned} \tag{2.25}$$

Considering the Helmholtz problem for $\boldsymbol{u}'$, adding and subtracting $\gamma\boldsymbol{u}'$, we finally obtain:

$$B(\boldsymbol{\sigma},\boldsymbol{\sigma}') = -\int_V [\boldsymbol{\nabla}\cdot\boldsymbol{v}\boldsymbol{\nabla}\cdot\boldsymbol{u}' + \boldsymbol{\nabla}\times\boldsymbol{v}\cdot\boldsymbol{\nabla}\times\boldsymbol{u}' + \boldsymbol{v}\cdot\boldsymbol{\nabla}p' + \gamma\boldsymbol{v}\cdot\boldsymbol{u}'] \tag{2.26}$$

A similar calculation can be carried out for β:

$$\begin{aligned} \beta(\boldsymbol{\sigma}) &= \int_V (\boldsymbol{\nabla}p^0 - \boldsymbol{g})\cdot\boldsymbol{\chi} + \oint_S \big(\hat{\boldsymbol{n}}\times\boldsymbol{b}\cdot\boldsymbol{\nabla}\times\boldsymbol{\chi} + \hat{\boldsymbol{n}}\cdot\boldsymbol{b}\boldsymbol{\nabla}\cdot\boldsymbol{\chi}\big) = \\ &= -\int_V \boldsymbol{\chi}\cdot(-\nabla^2+\gamma)\boldsymbol{u}^0 + \oint_S \big(\hat{\boldsymbol{n}}\times\boldsymbol{u}^0\cdot\boldsymbol{\nabla}\times\boldsymbol{\chi} + \hat{\boldsymbol{n}}\cdot\boldsymbol{u}^0\boldsymbol{\nabla}\cdot\boldsymbol{\chi}\big) = \\ &= -\int_V \boldsymbol{u}^0\cdot(-\nabla^2+\gamma)\boldsymbol{\chi} + \\ &- \oint_S \Big(\hat{\boldsymbol{n}}\times\boldsymbol{u}^0\cdot\boldsymbol{\nabla}\times\boldsymbol{\chi} + \hat{\boldsymbol{n}}\cdot\boldsymbol{u}^0\boldsymbol{\nabla}\cdot\boldsymbol{\chi} - \hat{\boldsymbol{n}}\times\boldsymbol{\chi}\cdot\boldsymbol{\nabla}\times\boldsymbol{u}^0 - \hat{\boldsymbol{n}}\cdot\boldsymbol{\chi}\boldsymbol{\nabla}\cdot\boldsymbol{u}^0\Big) + \\ &+ \oint_S \big(\hat{\boldsymbol{n}}\times\boldsymbol{u}^0\cdot\boldsymbol{\nabla}\times\boldsymbol{\chi} + \hat{\boldsymbol{n}}\cdot\boldsymbol{u}^0\boldsymbol{\nabla}\cdot\boldsymbol{\chi}\big). \end{aligned} \tag{2.27}$$

Using boundary conditions and $\boldsymbol{\chi}$ differential equation 2.19:

$$\beta(\boldsymbol{\sigma}) = \oint_S \hat{\boldsymbol{n}}\cdot\boldsymbol{\chi}\boldsymbol{\nabla}\cdot\boldsymbol{u}^0 = \oint_S \hat{\boldsymbol{n}}\cdot\boldsymbol{v}\boldsymbol{\nabla}\cdot\boldsymbol{u}^0. \tag{2.28}$$

Applying the divergence theorem and integrating by parts we obtain:

$$\begin{aligned} \beta(\boldsymbol{\sigma}) &= \oint_S \hat{\boldsymbol{n}}\cdot\boldsymbol{v}\boldsymbol{\nabla}\cdot\boldsymbol{u}^0 = \int_V \boldsymbol{\nabla}\cdot(\boldsymbol{v}\boldsymbol{\nabla}\cdot\boldsymbol{u}^0) = \\ &= \int_V [\boldsymbol{\nabla}\cdot\boldsymbol{v}\boldsymbol{\nabla}\cdot\boldsymbol{u}^0 + \boldsymbol{v}\cdot\boldsymbol{\nabla}(\boldsymbol{\nabla}\cdot\boldsymbol{u}^0)] = \\ &= \int_V [\boldsymbol{\nabla}\cdot\boldsymbol{v}\boldsymbol{\nabla}\cdot\boldsymbol{u}^0 + \boldsymbol{v}\cdot(\boldsymbol{\nabla}\times\boldsymbol{\nabla}\times\boldsymbol{u}^0 + \nabla^2\boldsymbol{u}^0)]. \end{aligned} \tag{2.29}$$

Using vector identities 2.8 and boundary conditions on $\boldsymbol{v}$:

$$\begin{aligned}\beta(\boldsymbol{\sigma}) &= \int_V [\boldsymbol{\nabla}\cdot\boldsymbol{v}\boldsymbol{\nabla}\cdot\boldsymbol{u}^0 + \boldsymbol{v}\cdot\nabla^2\boldsymbol{u}^0 + \boldsymbol{\nabla}\cdot((\boldsymbol{\nabla}\times\boldsymbol{u}^0)\times v) + \boldsymbol{\nabla}\times\boldsymbol{u}^0\cdot\boldsymbol{\nabla}\times\boldsymbol{v}] = \\ &= \int_V [\boldsymbol{\nabla}\cdot\boldsymbol{v}\boldsymbol{\nabla}\cdot\boldsymbol{u}^0 + \boldsymbol{v}\cdot\nabla^2\boldsymbol{u}^0 + \boldsymbol{\nabla}\times\boldsymbol{u}^0\cdot\boldsymbol{\nabla}\times\boldsymbol{v}] + \oint_S \hat{\boldsymbol{n}}\cdot((\boldsymbol{\nabla}\times\boldsymbol{u}^0)\times\boldsymbol{v}) = \\ &= \int_V [\boldsymbol{\nabla}\cdot\boldsymbol{v}\boldsymbol{\nabla}\cdot\boldsymbol{u}^0 + \boldsymbol{v}\cdot\nabla^2\boldsymbol{u}^0 + \boldsymbol{\nabla}\times\boldsymbol{u}^0\cdot\boldsymbol{\nabla}\times\boldsymbol{v}].\end{aligned} \tag{2.30}$$

Considering the Helmholtz problem for $\boldsymbol{u}^0$, adding and subtracting $\gamma\boldsymbol{u}^0$, we obtain:

$$\beta(\boldsymbol{\sigma}) = \int_V [\boldsymbol{\nabla}\cdot\boldsymbol{v}\boldsymbol{\nabla}\cdot\boldsymbol{u}^0 + \boldsymbol{\nabla}\times\boldsymbol{v}\cdot\boldsymbol{\nabla}\times\boldsymbol{u}^0 + \boldsymbol{v}\cdot\boldsymbol{\nabla}p^0 + \gamma\boldsymbol{v}\cdot\boldsymbol{u}^0 - \boldsymbol{v}\cdot\boldsymbol{g}]. \tag{2.31}$$

The arbitrariness of the field $\boldsymbol{v}$ at all internal point of the domain can be exploited by setting $\boldsymbol{v} = 0$, so that the integration domain reduces to a narrow shell along the boundary, with obvious advantages from the viewpoint of the computational efficiency. Unfortunately this method leads to an operator $B(\boldsymbol{\sigma},\boldsymbol{\sigma'})$ which is full and not symmetric. Obviuosly the operator is singular because the pressure is determined only up to an arbitrary additive constant. To sum up, we have obtained the problem:

$$\oint_{S^{(\sigma')}} B(\boldsymbol{\sigma},\boldsymbol{\sigma'})\lambda(\boldsymbol{\sigma'})dS_{\boldsymbol{\sigma'}} = \beta(\boldsymbol{\sigma}), \tag{2.32}$$

where:

$$\begin{aligned}B(\boldsymbol{\sigma},\boldsymbol{\sigma}') &= -\int_V [\boldsymbol{\nabla}\cdot\boldsymbol{v}\boldsymbol{\nabla}\cdot\boldsymbol{u}' + \boldsymbol{\nabla}\times\boldsymbol{v}\cdot\boldsymbol{\nabla}\times\boldsymbol{u}' + \boldsymbol{v}\cdot\boldsymbol{\nabla}p' + \gamma\boldsymbol{v}\cdot\boldsymbol{u}'], \\ \beta(\boldsymbol{\sigma}) &= \int_V [\boldsymbol{\nabla}\cdot\boldsymbol{v}\boldsymbol{\nabla}\cdot\boldsymbol{u}^0 + \boldsymbol{\nabla}\times\boldsymbol{v}\cdot\boldsymbol{\nabla}\times\boldsymbol{u}^0 + \boldsymbol{v}\cdot\boldsymbol{\nabla}p^0 + \gamma\boldsymbol{v}\cdot\boldsymbol{u}^0 - \boldsymbol{v}\cdot\boldsymbol{g}].\end{aligned} \tag{2.33}$$

Once λ, representing the reconstructed value of the pressure on the boundary, has been computed, the pressure field $p(\boldsymbol{x})$ can be determined.

2.3 Second Decoupling Method (Glowinski–Pironneau)

Glowinski and Pironneau have introduced an alternative uncoupling method which is presented in this section. For more details, see [16]. As we will see

later, this method does not require any additional constraint over the choice of the test function $\boldsymbol{v}$, in particular it does not require the fullfilment of $\hat{\boldsymbol{n}} \times \boldsymbol{v}\big|_S = 0$, which is quite constrictive.

2.3.1 Equations

This second uncoupling method, whose derivation will be discussed in depth in section 2.3.2, reads:

$$\begin{cases} -\nabla^2 p = -\boldsymbol{\nabla} \cdot \boldsymbol{g} & \\ (-\nabla^2 + \gamma)\boldsymbol{u} = -\boldsymbol{\nabla} p + \boldsymbol{g}, & \boldsymbol{u}\big|_S = \boldsymbol{b} \\ -\nabla^2 \varphi = \boldsymbol{\nabla} \cdot \boldsymbol{u}, & \varphi\big|_S = 0, \qquad \frac{\partial \varphi}{\partial n}\big|_S = 0. \end{cases} \tag{2.34}$$

In the above formulation the Poisson equation $-\nabla^2\varphi = \boldsymbol{\nabla} \cdot \boldsymbol{u}$ for φ has been introduced, supplemented by homogeneous Dirichlet condition $\varphi\big|_S = 0$ and homogeneous Neumann condition $\frac{\partial \varphi}{\partial n}\big|_S = 0$. As it will be shown in section 2.3.2, these boundary conditions lead to $\varphi = 0$ and $\boldsymbol{\nabla} \cdot \boldsymbol{u} = 0$ everywhere in the domain.

2.3.2 φ Poisson equation and boundary conditions

We will now show the derivation of the φ equation and boundary conditions. Taking the divergence of the Helmholtz equation for the velocity and taking into account the Poisson equations for p and for φ we have :

$$\boldsymbol{\nabla} \cdot [(-\nabla^2 + \gamma)\boldsymbol{u} + \boldsymbol{\nabla} p] = \boldsymbol{\nabla} \cdot \boldsymbol{g} \tag{2.35a}$$

$$(-\nabla^2 + \gamma)\boldsymbol{\nabla} \cdot \boldsymbol{u} + \nabla^2 p = \boldsymbol{\nabla} \cdot \boldsymbol{g} \tag{2.35b}$$

$$(-\nabla^2 + \gamma)\boldsymbol{\nabla} \cdot \boldsymbol{u} = 0 \tag{2.35c}$$

$$(-\nabla^2 + \gamma)\nabla^2 \varphi = 0 \tag{2.35d}$$

As we can see, φ has to satisfy a fourth-order elliptic equation. Since the velocity field has to be divergence free, from the Poisson equation for φ, it must be $\nabla^2\varphi = 0$ in the domain. One can assure $\boldsymbol{\nabla} \cdot \boldsymbol{u} = 0$ by imposing both Dirichlet and Neumann homogeneous boundary conditions on the whole boundary and forcing $\varphi = 0$ in the whole domain. In fact it can be demonstrated, under the assumption $\gamma > 0$, that:

$$\begin{cases} (-\nabla^2 + \gamma)\nabla^2 \varphi = 0 \quad \text{in V} \\ \varphi\big|_S = 0 \\ \frac{\partial \varphi}{\partial n}\big|_S = 0 \end{cases} \quad \Rightarrow \quad \varphi = 0 \quad \text{in V.} \tag{2.36}$$

Let's first multiply the fourth-order equation by φ and then integrate over the domain:

$$0 = \int_V \varphi(-\nabla^2 + \gamma)\nabla^2\varphi = \int_V \gamma\varphi\nabla^2\varphi - \int_V \varphi\nabla^4\varphi. \tag{2.37}$$

By means of integration by parts and homogeneous boundary conditions, one can write these terms as:

$$\begin{aligned}\int_V \gamma\varphi\nabla^2\varphi &= -\int_V \gamma\boldsymbol{\nabla}\varphi\cdot\boldsymbol{\nabla}\varphi + \oint_S \gamma\varphi\frac{\partial\varphi}{\partial n} = \\ &= -\int_V \gamma\boldsymbol{\nabla}\varphi\cdot\boldsymbol{\nabla}\varphi = \\ &= -\int_V \gamma|\boldsymbol{\nabla}\varphi|^2 \leq 0\end{aligned} \tag{2.38}$$

$$\begin{aligned}-\int_V \varphi\nabla^4\varphi &= -\int_V \varphi\nabla^2\nabla^2\varphi = \\ &= \int_V \boldsymbol{\nabla}\varphi\cdot\boldsymbol{\nabla}(\nabla\varphi) - \oint_S \varphi\frac{\partial\nabla^2\varphi}{\partial n} = \\ &= \int_V \boldsymbol{\nabla}\varphi\cdot\boldsymbol{\nabla}(\nabla\varphi) = \\ &= -\int_V \nabla^2\varphi\nabla^2\varphi + \oint_S \nabla^2\varphi\frac{\partial\varphi}{\partial n} = \\ &= -\int_V \nabla^2\varphi\nabla^2\varphi \\ &= -\int_V |\nabla^2\varphi|^2 \leq 0\end{aligned} \tag{2.39}$$

Thus:

$$0 = \int_V \varphi(-\nabla^2 + \gamma)\nabla^2\varphi = -\int_V \gamma|\boldsymbol{\nabla}\varphi|^2 - \int_V |\nabla^2\varphi|^2 \leq 0 \tag{2.40}$$

Homogeneous boundary conditions make the solution to be $\varphi = 0$ everywhere in V.

2.3.3 Decomposition method in the continuum

Owing to the lack of an explicit boundary condition for p, system 2.34 defines a set of equations coupled together by the imposition of two boundary conditions on φ. To determine the boundary value of p assuring that the incompressibility constraint is satisfied, we proceed as follows.

Decomposition and equations

Let assume that the solution $(p, \boldsymbol{u}, \varphi)$ can be decomposed as:

$$\begin{Bmatrix} p(\boldsymbol{x}) \\ \boldsymbol{u}(\boldsymbol{x}) \\ \varphi(\boldsymbol{x}) \end{Bmatrix} = \begin{Bmatrix} p^0(\boldsymbol{x}) \\ \boldsymbol{u}^0(\boldsymbol{x}) \\ \varphi^0(\boldsymbol{x}) \end{Bmatrix} + \begin{Bmatrix} p_\lambda(\boldsymbol{x}) \\ \boldsymbol{u}_\lambda(\boldsymbol{x}) \\ \varphi_\lambda(\boldsymbol{x}) \end{Bmatrix}, \tag{2.41}$$

where p^0 satisfies homogeneous boundary conditions on S while p_λ fulfills non homogeneous boundary data. Because of the linearity of the problem, the original system 2.34 can be split in the following homogeneous and non-homogeneous problems:

$$\begin{cases} -\nabla^2 p^0 = -\boldsymbol{\nabla} \cdot \boldsymbol{g}, & p^0|_S = 0 \\ (-\nabla^2 + \gamma)\boldsymbol{u^0} = -\boldsymbol{\nabla} p^0 + \boldsymbol{g}, & \boldsymbol{u}^0|_S = \boldsymbol{b} \\ -\nabla^2 \varphi^0 = \boldsymbol{\nabla} \cdot \boldsymbol{u^0}, & \varphi^0|_S = 0, \end{cases} \tag{2.42}$$

$$\begin{cases} -\nabla^2 p_\lambda = 0, & p_\lambda|_S = \lambda \\ (-\nabla^2 + \gamma)\boldsymbol{u}_\lambda = -\boldsymbol{\nabla} p_\lambda, & \boldsymbol{u}_\lambda|_S = 0 \\ -\nabla^2 \varphi_\lambda = \boldsymbol{\nabla} \cdot \boldsymbol{u}_\lambda, & \varphi_\lambda|_S = 0. \end{cases} \tag{2.43}$$

Here λ denotes an unknown surface function unknown with zero average to be determined so that the following extra boundary condition on φ is satisfied:

$$\frac{\partial \varphi}{\partial n}\Big|_S = 0. \tag{2.44}$$

Since p^0 satisfies homogeneous Dirichlet boundary conditions, λ corresponds to the value of the pressure field over the boundary up to an additive constant.

Integral condition and linear variational problem

As already mentioned, the condition:

$$\frac{\partial \varphi}{\partial n}\Big|_S = 0 \qquad \Rightarrow \qquad -\frac{\partial \varphi_\lambda}{\partial n}\Big|_S = \frac{\partial \varphi^0}{\partial n}\Big|_S \tag{2.45}$$

has to be satisfied. This can be re-written in the integral (variational) form:

$$-\oint_S \mu \frac{\partial \varphi_\lambda}{\partial n} = \oint_S \mu \frac{\partial \varphi^0}{\partial n}. \tag{2.46}$$

for any test function μ of zero average defined on S. Now each of the two terms can be developed. To do so orthogonality properties, boundary conditions and the μ-counterpart of λ-equations are used together with the following useful properties:

- Green identity

$$-\int_V \boldsymbol{u}\cdot\nabla^2\boldsymbol{v} = -\int_V \boldsymbol{v}\cdot\nabla^2\boldsymbol{u} + \oint_S (\hat{\boldsymbol{n}}\times\boldsymbol{v}\cdot\boldsymbol{\nabla}\times\boldsymbol{u} + \hat{\boldsymbol{n}}\cdot\boldsymbol{v}\boldsymbol{\nabla}\cdot\boldsymbol{u} - \hat{\boldsymbol{n}}\times\boldsymbol{u}\cdot\boldsymbol{\nabla}\times\boldsymbol{v} - \hat{\boldsymbol{n}}\cdot\boldsymbol{u}\boldsymbol{\nabla}\cdot\boldsymbol{v}) \tag{2.47}$$

- Orthogonality

$$\int_V \boldsymbol{\nabla}\varphi_\lambda\cdot\boldsymbol{\nabla}p_\mu = \int_V \boldsymbol{\nabla}\cdot(\varphi_\lambda\boldsymbol{\nabla}p_\mu) - \int_V \varphi_\lambda\nabla^2 p_\mu = \oint_S \varphi_\lambda\boldsymbol{\nabla}p_\mu\cdot\hat{\boldsymbol{n}} - \int_V \varphi_\lambda\nabla^2 p_\mu = 0 \tag{2.48}$$

given $\nabla^2 p_\mu = 0$ in V and $\varphi_\lambda = 0$ on the boundary. The μ subfix indicates the μ-counterpart problem of the λ system. Similarly, it can be shown that:

$$\int_V \boldsymbol{\nabla}\varphi^0\cdot\boldsymbol{\nabla}p_\mu = 0 \tag{2.49}$$

- Vector identity

$$\boldsymbol{\nabla}(\boldsymbol{\nabla}\cdot\boldsymbol{v}) = \boldsymbol{\nabla}\times\boldsymbol{\nabla}\times\boldsymbol{v} + \nabla^2\boldsymbol{v} \tag{2.50}$$

First let us focus on the left-hand side of equation 2.46. Let $\mu = p_\mu\big|_S$, as it can be deduced from the μ-counterpart problem of the λ system. By exploiting the identity 2.45, the divergence theorem and the othogonality property 2.48, one can write:

$$\begin{aligned} -\oint_S \frac{\partial\varphi_\lambda}{\partial n}\mu &= -\oint_S \frac{\partial\varphi_\lambda}{\partial n}p_\mu \\ &= -\int_V \boldsymbol{\nabla}\cdot(p_\mu\boldsymbol{\nabla}\varphi_\lambda) \\ &= -\int_V p_\mu\nabla^2\varphi_\lambda - \int_V \boldsymbol{\nabla}\varphi_\lambda\cdot\boldsymbol{\nabla}p_\mu \\ &= -\int_V p_\mu\nabla^2\varphi_\lambda. \end{aligned} \tag{2.51}$$

Then, using the Poisson equation for φ_λ, integrating by parts, exploiting homogeneous boundary conditions for $\boldsymbol{u}_\lambda$ and eventually introducing the

Helmholtz equation for $\boldsymbol{u}_\mu$, we have:

$$\begin{aligned} -\oint_S \frac{\partial \varphi_\lambda}{\partial n}\mu &= -\int_V p_\mu \nabla^2 \varphi_\lambda \\ &= \int_V p_\mu \boldsymbol{\nabla}\cdot\boldsymbol{u}_\lambda \\ &= \int_V [-\boldsymbol{u}_\lambda\cdot\boldsymbol{\nabla}p_\mu + \boldsymbol{\nabla}\cdot(p_\mu \boldsymbol{u}_\lambda)] \\ &= \int_V \boldsymbol{u}_\lambda\cdot(-\boldsymbol{\nabla}p_\mu) + \oint_S p_\mu \boldsymbol{u}_\lambda\cdot\hat{\boldsymbol{n}} \\ &= \int_V \boldsymbol{u}_\lambda\cdot(-\nabla^2+\gamma)\boldsymbol{u}_\mu. \end{aligned} \tag{2.52}$$

By introducing the vectorial identity 2.50, we have:

$$-\oint_S \frac{\partial \varphi_\lambda}{\partial n}\mu = \int_V \gamma \boldsymbol{u}_\lambda\cdot\boldsymbol{u}_\mu + \int_V \boldsymbol{u}_\lambda\cdot[\boldsymbol{\nabla}\times\boldsymbol{\nabla}\times\boldsymbol{u_\mu} - \boldsymbol{\nabla}(\boldsymbol{\nabla}\cdot\boldsymbol{u}_\mu)]. \tag{2.53}$$

Thanks to integration by parts, the divergence theorem and homogeneous boundary conditions these two terms can be re-written as:

$$\begin{aligned} -\int_V \boldsymbol{u}_\lambda\boldsymbol{\nabla}(\boldsymbol{\nabla}\cdot\boldsymbol{u}_\mu) &= -\int_V \boldsymbol{\nabla}\cdot(\boldsymbol{u}_\lambda\boldsymbol{\nabla}\cdot\boldsymbol{u}_\mu) + \int_V \boldsymbol{\nabla}\cdot\boldsymbol{u}_\lambda\boldsymbol{\nabla}\cdot\boldsymbol{u}_\mu \\ &= \int_V \boldsymbol{\nabla}\cdot\boldsymbol{u}_\lambda\boldsymbol{\nabla}\cdot\boldsymbol{u}_\mu, \\ \int_V \boldsymbol{u}_\lambda\cdot(\boldsymbol{\nabla}\times\boldsymbol{\nabla}\times\boldsymbol{u_\mu}) &= \int_V \boldsymbol{\nabla}\cdot[(\boldsymbol{\nabla}\times\boldsymbol{u}_\mu)\times\boldsymbol{u}_\lambda] + \int_V \boldsymbol{\nabla}\times\boldsymbol{u}_\mu\cdot\boldsymbol{\nabla}\times\boldsymbol{u}_\lambda \\ &= \int_V \boldsymbol{\nabla}\times\boldsymbol{u}_\mu\cdot\boldsymbol{\nabla}\times\boldsymbol{u}_\lambda, \end{aligned} \tag{2.54}$$

so that:

$$-\oint_S \frac{\partial \varphi_\lambda}{\partial n}\mu = \int_V (\gamma\boldsymbol{u}_\lambda\cdot\boldsymbol{u}_\mu + \boldsymbol{\nabla}\times\boldsymbol{u}_\mu\cdot\boldsymbol{\nabla}\times\boldsymbol{u}_\lambda + \boldsymbol{\nabla}\cdot\boldsymbol{u}_\lambda\boldsymbol{\nabla}\cdot\boldsymbol{u}_\mu). \tag{2.55}$$

This result reveals the symmetry property of the operator associated with this variational problem. Let us now focus on the right-hand side of the equation 2.46. Similarly to what has been done for the left-hand side, we

have:

$$\begin{aligned} -\oint_S \frac{\partial \varphi^0}{\partial n}\mu &= -\oint_S \frac{\partial \varphi^0}{\partial n} p_\mu \\ &= -\int_V \boldsymbol{\nabla}\cdot(p_\mu \boldsymbol{\nabla}\varphi^0) \\ &= -\int_V p_\mu \nabla^2 \varphi^0 - \int_V \boldsymbol{\nabla}\varphi^0 \cdot \boldsymbol{\nabla} p_\mu \\ &= -\int_V p_\mu \nabla^2 \varphi^0. \end{aligned} \tag{2.56}$$

Then, using the Poisson equation for φ^0 , integrating by parts, exploiting Dirichlet boundary conditions ($\boldsymbol{u}^0|_S = \boldsymbol{b}$) and eventually introducing the Helmholtz equation for $\boldsymbol{u}_\mu$, we have:

$$\begin{aligned} -\oint_S \frac{\partial \varphi^0}{\partial n}\mu &= -\int_V p_\mu \nabla^2 \varphi^0 \\ &= \int_V p_\mu \boldsymbol{\nabla}\cdot \boldsymbol{u}^0 \\ &= \int_V [-\boldsymbol{u}^0 \cdot \boldsymbol{\nabla} p_\mu + \boldsymbol{\nabla}\cdot(p_\mu \boldsymbol{u}^0)] \\ &= \int_V \boldsymbol{u}^0 \cdot (-\boldsymbol{\nabla} p_\mu) + \oint_S p_\mu \boldsymbol{u}^0 \cdot \hat{\boldsymbol{n}} \\ &= \int_V \boldsymbol{u}^0 \cdot (-\nabla^2 + \gamma)\boldsymbol{u}_\mu + \oint_S p_\mu \boldsymbol{b}\cdot\hat{\boldsymbol{n}}. \end{aligned} \tag{2.57}$$

Exploiting the Green identity 2.47 and the homogeneous boundary conditions for $\boldsymbol{u}_\mu$ we have:

$$\begin{aligned} -\oint_S \frac{\partial \varphi^0}{\partial n}\mu &= \int_V \boldsymbol{u}^0 \cdot (-\nabla^2 + \gamma)\boldsymbol{u}_\mu + \oint_S p_\mu \boldsymbol{b}\cdot\hat{\boldsymbol{n}} = \\ &= \int_V \boldsymbol{u}_\mu \cdot (-\nabla^2 + \gamma)\boldsymbol{u}^0 + \\ &+ \oint_S (\hat{\boldsymbol{n}} \times \boldsymbol{u}_\mu \cdot \boldsymbol{\nabla} \times \boldsymbol{u}^0 + \hat{\boldsymbol{n}}\cdot\boldsymbol{u}_\mu \boldsymbol{\nabla}\cdot\boldsymbol{u}^0 - \hat{\boldsymbol{n}}\times\boldsymbol{u}^0\cdot\boldsymbol{\nabla}\times\boldsymbol{u}_\mu - \hat{\boldsymbol{n}}\cdot\boldsymbol{u}^0\boldsymbol{\nabla}\cdot\boldsymbol{u}_\mu) + \\ &+ \oint_S p_\mu \boldsymbol{b}\cdot\hat{\boldsymbol{n}} = \\ &= \int_V \boldsymbol{u}_\mu \cdot (-\nabla^2 + \gamma)\boldsymbol{u}^0 - \oint_S (\hat{\boldsymbol{n}}\times\boldsymbol{u}^0\cdot\boldsymbol{\nabla}\times\boldsymbol{u}_\mu + \hat{\boldsymbol{n}}\cdot\boldsymbol{u}^0\boldsymbol{\nabla}\cdot\boldsymbol{u}_\mu) + \oint_S p_\mu \boldsymbol{b}\cdot\hat{\boldsymbol{n}}. \end{aligned} \tag{2.58}$$

Given the Helmholtz equation for $\boldsymbol{u}^0$ and the boundary conditions $\boldsymbol{u}^0\big|_S = \boldsymbol{b}$, $p_\mu\big|_S = \mu$:

$$\begin{aligned}
-\oint_S \frac{\partial \varphi^0}{\partial n}\mu &= \int_V \boldsymbol{u}_\mu \cdot (-\boldsymbol{\nabla} p^0 + \boldsymbol{g}) - \oint_S (\hat{\boldsymbol{n}} \times \boldsymbol{b} \cdot \boldsymbol{\nabla} \times \boldsymbol{u}_\mu + \hat{\boldsymbol{n}} \cdot \boldsymbol{b} \boldsymbol{\nabla} \cdot \boldsymbol{u}_\mu) + \oint_S p_\mu \boldsymbol{b} \cdot \hat{\boldsymbol{n}} = \\
&= \int_V \boldsymbol{u}_\mu \cdot (-\boldsymbol{\nabla} p^0 + \boldsymbol{g}) - \oint_S [\hat{\boldsymbol{n}} \times \boldsymbol{b} \cdot \boldsymbol{\nabla} \times \boldsymbol{u}_\mu + \hat{\boldsymbol{n}} \cdot \boldsymbol{b} (\boldsymbol{\nabla} \cdot \boldsymbol{u}_\mu - \mu)] = \\
&= \int_V \varphi_\mu (-\nabla^2 p^0) + \int_V \boldsymbol{u}_\mu \cdot \boldsymbol{g} - \oint_S [\hat{\boldsymbol{n}} \times \boldsymbol{b} \cdot \boldsymbol{\nabla} \times \boldsymbol{u}_\mu + \hat{\boldsymbol{n}} \cdot \boldsymbol{b} (\boldsymbol{\nabla} \cdot \boldsymbol{u}_\mu - \mu)].
\end{aligned} \tag{2.59}$$

Given the Poisson equation for p^0:

$$-\oint_S \frac{\partial \varphi^0}{\partial n}\mu = \int_V (-\boldsymbol{\nabla} \cdot \boldsymbol{g}) \varphi_\mu + \int_V \boldsymbol{g} \cdot \boldsymbol{u}_\mu - \oint_S [\hat{\boldsymbol{n}} \times \boldsymbol{b} \cdot \boldsymbol{\nabla} \times \boldsymbol{u}_\mu + \hat{\boldsymbol{n}} \cdot \boldsymbol{b} (\boldsymbol{\nabla} \cdot \boldsymbol{u}_\mu - \mu)]. \tag{2.60}$$

Finally by collecting together the new derived expressions for the left- and right-hand sides of 2.45:

$$\begin{aligned}
&\int_V (\gamma \boldsymbol{u}_\lambda \cdot \boldsymbol{u}_\mu + \boldsymbol{\nabla} \times \boldsymbol{u}_\mu \cdot \boldsymbol{\nabla} \times \boldsymbol{u}_\lambda + \boldsymbol{\nabla} \cdot \boldsymbol{u}_\lambda \boldsymbol{\nabla} \cdot \boldsymbol{u}_\mu) = \\
&= \int_V [(\boldsymbol{\nabla} \cdot \boldsymbol{g}) \varphi_\mu - \boldsymbol{g} \cdot \boldsymbol{u}_\mu] + \oint_S [\hat{\boldsymbol{n}} \times \boldsymbol{b} \cdot \boldsymbol{\nabla} \times \boldsymbol{u}_\mu + \hat{\boldsymbol{n}} \cdot \boldsymbol{b} (\boldsymbol{\nabla} \cdot \boldsymbol{u}_\mu - \mu)].
\end{aligned} \tag{2.61}$$

The above formulation is not convenient for computational purposes since it would require to determine and store u_μ and φ_μ for every linearly independent function μ.

2.3.4 Efficient numerical implementation of the decomposition

Decomposition and equations

A more convenient decomposition of the solution can be expressed as:

$$\begin{Bmatrix} p(\boldsymbol{x}) \\ \boldsymbol{u}(\boldsymbol{x}) \\ \varphi(\boldsymbol{x}) \end{Bmatrix} = \begin{Bmatrix} p^0(\boldsymbol{x}) \\ \boldsymbol{u}^0(\boldsymbol{x}) \\ \varphi^0(\boldsymbol{x}) \end{Bmatrix} + \oint_S \begin{Bmatrix} p'(\boldsymbol{x}; \boldsymbol{\sigma}') \\ \boldsymbol{u}'(\boldsymbol{x}; \boldsymbol{\sigma}') \\ \varphi'(\boldsymbol{x}; \boldsymbol{\sigma}') \end{Bmatrix} \lambda(\boldsymbol{\sigma}') dS(\boldsymbol{\sigma}'), \tag{2.62}$$

where the functions with superscript 0 are the solution to the same equations previously introduced (2.42), and the functions with superscript ' are solutions to the following equations for any $\boldsymbol{\sigma'} \in S \setminus \boldsymbol{\sigma}^*$ ($\boldsymbol{\sigma}^*$ being a fixed

point arbitrarily chosen on S):

$$\begin{cases} -\nabla^2 p' = 0, & p'|_S = \delta^2(\boldsymbol{x} - \boldsymbol{\sigma'}) - \delta^2(\boldsymbol{x} - \boldsymbol{\sigma^*}) \\ (-\nabla^2 + \gamma)\boldsymbol{u'} = -\boldsymbol{\nabla} p', & \boldsymbol{u'}|_S = 0 \\ -\nabla^2 \varphi' = \boldsymbol{\nabla} \cdot \boldsymbol{u'}, & \varphi'|_S = 0. \end{cases} \tag{2.63}$$

Moreover the auxiliary scalar funcions $w(\boldsymbol{x};\boldsymbol{\sigma})$ defined over the domain are introduced:

$$w(\boldsymbol{x};\boldsymbol{\sigma}) \quad \text{arbitrary in V}, \quad w(\boldsymbol{x};\boldsymbol{\sigma})\big|_S = \delta^{(2)}(\boldsymbol{x} - \boldsymbol{\sigma}) - \delta^{(2)}(\boldsymbol{x} - \boldsymbol{\sigma^*}). \tag{2.64}$$

These functions are taken different from zero only in a narrow shell along the boundary, so that the volume integrals above can be evaluated much more efficiently than the corresponding expressions containing $\boldsymbol{u}_\mu$. Moreover these test functions must not satisfy any additional requirement, unlike those derived from the first uncoupling method (see equation 2.20).

Integral condition and linear variational problem

Let us write the boundary condition $\frac{\partial \varphi}{\partial n}\big|_S = 0$ in its weak formulation:

$$-\oint_{S^{(x)}} w(\boldsymbol{x};\boldsymbol{\sigma}) \oint_{S^{(\sigma')}} \frac{\partial \varphi'}{\partial n}(\boldsymbol{x};\boldsymbol{\sigma'})\lambda(\boldsymbol{\sigma'}) dS_{\boldsymbol{\sigma'}} dS_{\boldsymbol{x}} = \oint_S w(\boldsymbol{x};\boldsymbol{\sigma}) \frac{\partial \varphi^0}{\partial n}(\boldsymbol{x}) dS_{\boldsymbol{x}}, \tag{2.65}$$

which defines a variational problem for the boundary unknown $\lambda(\boldsymbol{\sigma'})$. First let us focus on the right-hand side of the above equation. By applying the divergence theorem and using the Poisson equation for φ^0, we have:

$$\begin{aligned} \beta(\boldsymbol{\sigma}) &= \oint_S w(\boldsymbol{x};\boldsymbol{\sigma}) \frac{\partial \varphi^0}{\partial n}(\boldsymbol{x}) dS_{\boldsymbol{x}} = \\ &= \int_V \boldsymbol{\nabla} \cdot [w(\boldsymbol{x};\boldsymbol{\sigma}) \boldsymbol{\nabla} \varphi^0(\boldsymbol{x})] dV_{\boldsymbol{x}} = \\ &= \int_V [w(\boldsymbol{x};\boldsymbol{\sigma}) \nabla^2 \varphi^0(\boldsymbol{x}) + \boldsymbol{\nabla} w(\boldsymbol{x};\boldsymbol{\sigma}) \cdot \boldsymbol{\nabla} \varphi^0(\boldsymbol{x})] dV_{\boldsymbol{x}} = \\ &= \int_V [-w(\boldsymbol{x};\boldsymbol{\sigma}) \boldsymbol{\nabla} \cdot \boldsymbol{u}^0(\boldsymbol{x}) + \boldsymbol{\nabla} w(\boldsymbol{x};\boldsymbol{\sigma}) \cdot \boldsymbol{\nabla} \varphi^0(\boldsymbol{x})] dV_{\boldsymbol{x}}. \end{aligned} \tag{2.66}$$

At the same time, expanding the the left-hand side we have:

$$
\begin{aligned}
&- \oint_{S^{(x)}} w(\boldsymbol{x};\boldsymbol{\sigma}) \oint_{S^{(\sigma')}} \frac{\partial \varphi'}{\partial n}(\boldsymbol{x};\boldsymbol{\sigma'})\lambda(\boldsymbol{\sigma'})dS_{\boldsymbol{\sigma'}}dS_{\boldsymbol{x}} = \\
&= - \oint_{S^{(\sigma')}} \left(\oint_{S^{(x)}} \frac{\partial \varphi'}{\partial n}(\boldsymbol{x};\boldsymbol{\sigma'})w(\boldsymbol{x};\boldsymbol{\sigma})dS_{\boldsymbol{x}} \right) \lambda(\boldsymbol{\sigma'})dS_{\boldsymbol{\sigma'}} = \\
&= \oint_{S^{(\sigma')}} \left(\int_{V^{(x)}} \left(w(\boldsymbol{x};\boldsymbol{\sigma})\boldsymbol{\nabla}\cdot\boldsymbol{u}'(\boldsymbol{x}) - \boldsymbol{\nabla}w(\boldsymbol{x};\boldsymbol{\sigma})\cdot\boldsymbol{\nabla}\varphi'(\boldsymbol{x}) \right) dV_{\boldsymbol{x}} \right) \lambda(\boldsymbol{\sigma'})dS_{\boldsymbol{\sigma'}} = \\
&= \oint_{S^{(\sigma')}} B(\boldsymbol{\sigma},\boldsymbol{\sigma'})\lambda(\boldsymbol{\sigma'})dS_{\boldsymbol{\sigma'}}.
\end{aligned}
\tag{2.67}
$$

Eventually the linear variational problem for $\lambda(\boldsymbol{\sigma'})$ can be wrtiten as:

$$
\oint_{S^{(\sigma')}} B(\boldsymbol{\sigma},\boldsymbol{\sigma'})\lambda(\boldsymbol{\sigma'})dS_{\boldsymbol{\sigma'}} = \beta(\boldsymbol{\sigma}), \tag{2.68}
$$

with:

$$
\begin{aligned}
B(\boldsymbol{\sigma},\boldsymbol{\sigma'}) &= \int_V [w\boldsymbol{\nabla}\cdot\boldsymbol{u}' - \boldsymbol{\nabla}w\cdot\boldsymbol{\nabla}\varphi']dV \\
\beta(\boldsymbol{\sigma}) &= \int_V [-w\boldsymbol{\nabla}\cdot\boldsymbol{u}^0 + \boldsymbol{\nabla}w\cdot\boldsymbol{\nabla}\varphi^0]dV.
\end{aligned}
\tag{2.69}
$$

2.3.5 Discrete formulation

As described in the previous section 2.3.4, the final linear variational problem is function of $\boldsymbol{\sigma}$ and $\boldsymbol{\sigma'}$, namely the spatial coordinates where the Dirac delta associated with the definition of w and the boundary conditions for p' is placed, respectively. In a discretized FEM framework, the Dirac delta functions can be consistently replaced by the space basis functions, which are equal to one in the considered node and zero elsewhere. In this way the test functions w are replaced by the chosen FEM basis functions. Moreover we can write the linear problem 2.3.4 as many times as the number of the Dirichlet boundary nodes, each time changing the boundary conditions for p' by replacing the 'discretized' Dirac delta function in a different node of the boundary. By doing so we obtain an algebraic linear system whose solution $\underline{\lambda}$ represents the expected value of the pressure field on the boundary, since $p^0|_S = 0$.

In this section the discrete formulation of the above method is illustrated. Further details about the actual discretization of the linear problem are given in section 3.3.3.

Decomposition

In the discrete setting, the solution can be expressed as follows:

$$\begin{Bmatrix} p(\boldsymbol{x}) \\ \boldsymbol{u}(\boldsymbol{x}) \\ \varphi(\boldsymbol{x}) \end{Bmatrix} = \begin{Bmatrix} p^0(\boldsymbol{x}) \\ \boldsymbol{u}^0(\boldsymbol{x}) \\ \varphi^0(\boldsymbol{x}) \end{Bmatrix} + \sum_{k=1}^{N_D} \begin{Bmatrix} p^k(\boldsymbol{x}) \\ \boldsymbol{u}^k(\boldsymbol{x}) \\ \varphi^k(\boldsymbol{x}) \end{Bmatrix} \lambda^k, \tag{2.70}$$

It can be pointed out that the expansion coefficient λ_k corresponds to the pressure value on the boundary node k of the pressure field solution, while $p^k(\boldsymbol{x})$, $\boldsymbol{u}^k(\boldsymbol{x})$ and $\varphi^k(\boldsymbol{x})$ represent suitable basis functions.

Equations

By virtue of its linearity, the original problem 2.34 can be split in one non homogeneous problem and as many homogeneous problems as the number of Dirichlet boundary nodes, N_D, each one being associated with a different boundary value for p^k:

$$\begin{cases} -\nabla^2 p^0 = -\boldsymbol{\nabla} \cdot \boldsymbol{g}, & p^0|_S = 0 \\ (-\nabla^2 + \gamma)\boldsymbol{u^0} = -\boldsymbol{\nabla} p^0 + \boldsymbol{g}, & \boldsymbol{u}^0|_S = \boldsymbol{b} \\ -\nabla^2 \varphi^0 = \boldsymbol{\nabla} \cdot \boldsymbol{u^0}, & \varphi^0|_S = 0, \end{cases} \tag{2.71}$$

$$\begin{cases} -\nabla^2 p^k = 0, & p^k|_S = \mu^k \\ (-\nabla^2 + \gamma)\boldsymbol{u}^k = -\boldsymbol{\nabla} p^k, & \boldsymbol{u}^k|_S = 0 \qquad k = 1 : N_D \\ -\nabla^2 \varphi^k = \boldsymbol{\nabla} \cdot \boldsymbol{u}^k, & \varphi^k|_S = 0, \end{cases} \tag{2.72}$$

where μ^k is a function which is set to one on the k-th boundary node and zero elsewhere (equivalent to the w^k basis function).

Equations for the explicit treatment of the time derivative

As introduced in section 2.1.2, an explicit treatment of the time derivative can be adopted. The corresponding equations are:

$$\begin{cases} -\nabla^2 p^0 = -\boldsymbol{\nabla} \cdot \tilde{\boldsymbol{g}}, & p^0|_S = 0 \\ -\nabla^2 \boldsymbol{u^0} = -\boldsymbol{\nabla} p^0 + \tilde{\boldsymbol{g}}, & \boldsymbol{u}^0|_S = \boldsymbol{b} \\ -\nabla^2 \varphi^0 = \boldsymbol{\nabla} \cdot \boldsymbol{u^0}, & \varphi^0|_S = 0, \end{cases} \tag{2.73}$$

$$\begin{cases} -\nabla^2 p^k = 0, & p^k|_S = \mu^k \\ -\nabla^2 \boldsymbol{u}^k = -\boldsymbol{\nabla} p^k, & \boldsymbol{u}^k|_S = 0 \qquad k = 1 : N_D \\ -\nabla^2 \varphi^k = \boldsymbol{\nabla} \cdot \boldsymbol{u}^k, & \varphi^k|_S = 0. \end{cases} \tag{2.74}$$

As it can be seen from the previous equations, in this formulation in addition to the pressure also the velocity must satisfy a Poisson equation, instead of a Helmholtz equation. This can increase the accuracy of the method especially when both Δt and ν are small, hence when the coefficient γ is large. In such situation, an implicit treatment of the time derivative would cause the laplacian term of the Helmholtz equation to be multiplied by a small coefficient leading to a singular perturbation problem. As a consequence of that, when measurement errors are introduced through the velocity boundary data, they could be greatly amplified resulting in severe oscillations of the reconstructed pressure field. For this reason this second method, with an explicit treatment of the time derivative, seems more suitable to process time resolved velocity fields where the measurement error has not been filtered out by averaging.

Integral condition and linear variational problem

From the condition $\frac{\partial \varphi}{\partial n}\big|_S = 0$, with $\varphi(\boldsymbol{x}) = \varphi^0(\boldsymbol{x}) + \sum_{k=1}^{N_D} \varphi^k(\boldsymbol{x})\lambda^k$, we can re-write the variational condition $\oint_S w\frac{\partial \varphi}{\partial n} = 0$:

$$\oint_S w\frac{\partial \varphi^0}{\partial n} + \sum_{k=1}^{N_D} \oint_S w\frac{\partial \varphi^k}{\partial n}\lambda^k = 0 \tag{2.75}$$

Introducing the divergence theorem and the Poisson equation for φ^0, we have:

$$\begin{aligned} \oint_S w\frac{\partial \varphi^0}{\partial n} &= \int_V \boldsymbol{\nabla}\cdot(w\boldsymbol{\nabla}\varphi^0) = \\ &= \int_V (w\nabla^2\varphi^0 + \boldsymbol{\nabla}w\cdot\boldsymbol{\nabla}\varphi^0) = \\ &= \int_V (-w\boldsymbol{\nabla}\cdot\boldsymbol{u}^0 + \boldsymbol{\nabla}w\cdot\boldsymbol{\nabla}\varphi^0) \end{aligned} \tag{2.76}$$

and similarly for $\oint_S w\frac{\partial \varphi^k}{\partial n}$. It follows that:

$$\sum_{k=1}^{N_D} \int_V (w\boldsymbol{\nabla}\cdot\boldsymbol{u}^k - \boldsymbol{\nabla}w\cdot\boldsymbol{\nabla}\varphi^k)\lambda^k = -\int_V (w\boldsymbol{\nabla}\cdot\boldsymbol{u}^0 - \boldsymbol{\nabla}w\cdot\boldsymbol{\nabla}\varphi^0) \tag{2.77}$$

This linear problem is the "discrete" counterpart of equations 2.68 and 2.69. The numerical discretization by introducing the finite element test functions and the derivation of the associated algebraic linear problem are reported in section 3.3.3.

Once all the λ^k have been computed, the reconstructed pressure and

velocity field are given by:

$$p(\boldsymbol{x}) = p^0(\boldsymbol{x}) + \sum_{k=1}^{N_D} p^k(\boldsymbol{x})\lambda^k, \tag{2.78}$$

$$\boldsymbol{u}(\boldsymbol{x}) = \boldsymbol{u}^0(\boldsymbol{x}) + \sum_{k=1}^{N_D} \boldsymbol{u}^k(\boldsymbol{x})\lambda^k \tag{2.79}$$

2.4 Projection on a divergence-free space

Since the described method computes the pressure field by enforcing the incompressibility constraint all over the domain, any possible velocity measurement affected by an error which is not divergence free could result in a potentially inaccurate computation of the pressure field. Before the method is applied, a projection of the measured velocity field on a divergence free space could be performed to improve the accuracy of the method. In this subsection the mathematical formulation of a projector on the divergence free function space is derived. Let introduce $\tilde{\boldsymbol{u}}$, namely a measured velocity field which is affected by measurement errors, hence it is not divergence free. Our aim is to find a solenoidal velocity field $\boldsymbol{u}$ that is close to $\tilde{\boldsymbol{u}}$ in some sense. In this work we quantify the distance between $\boldsymbol{u}$ and $\tilde{\boldsymbol{u}}$ using the L^2 norm of $\boldsymbol{u} - \tilde{\boldsymbol{u}}$, and we find $\boldsymbol{u}$ in order to minimize such distance. To do so, the following functional is introduced, where the incompressibility constraint is enforced by means of a Lagrange multiplier λ,

$$J = \frac{1}{2}\int_V \|\boldsymbol{u} - \tilde{\boldsymbol{u}}\|_2 - \int_V \lambda \boldsymbol{\nabla}\cdot\boldsymbol{u} = \frac{1}{2}\int_V (\boldsymbol{u} - \tilde{\boldsymbol{u}})\cdot(\boldsymbol{u} - \tilde{\boldsymbol{u}}) - \int_V \lambda\boldsymbol{\nabla}\cdot\boldsymbol{u}. \tag{2.80}$$

By imposing a null functional variation ($\delta J = 0$), the functional can be minimized:

$$\begin{aligned}\delta J &= \frac{1}{2}\int_V \delta\boldsymbol{u}\cdot(\boldsymbol{u} - \tilde{\boldsymbol{u}}) + (\boldsymbol{u} - \tilde{\boldsymbol{u}})\cdot\delta\boldsymbol{u} - \int_V \delta\lambda\boldsymbol{\nabla}\cdot\boldsymbol{u} - \int_V \lambda\boldsymbol{\nabla}\cdot\delta\boldsymbol{u} \\ &= \int_V \delta\boldsymbol{u}\cdot(\boldsymbol{u} - \tilde{\boldsymbol{u}}) - \int_V \delta\lambda\boldsymbol{\nabla}\cdot\boldsymbol{u} - \int_V \lambda\boldsymbol{\nabla}\cdot\delta\boldsymbol{u} = 0.\end{aligned} \tag{2.81}$$

Now, by replacing the variations $\delta\boldsymbol{u}$ and $\delta\lambda$ with the corresponding test functions $\boldsymbol{v} \in [H^1(V)]^2$ and $q \in L^2(V)$, respectively, and moving the known integrals on the right-hand side, we obtain:

$$\int_V \boldsymbol{v}\cdot\boldsymbol{u} - \int_V q\boldsymbol{\nabla}\cdot\boldsymbol{u} - \int_V \lambda\boldsymbol{\nabla}\cdot v = \int_V \boldsymbol{v}\cdot\tilde{\boldsymbol{u}}, \tag{2.82}$$

The weak formulation of the problem states:
Find $\boldsymbol{u} \in [H^1(V)]^2$ and $\lambda \in L^2(V)$ such that

$$\int_V \boldsymbol{v} \cdot \boldsymbol{u} - \int_V q \boldsymbol{\nabla} \cdot \boldsymbol{u} - \int_V \lambda \boldsymbol{\nabla} \cdot v = \int_V \boldsymbol{v} \cdot \tilde{\boldsymbol{u}}, \tag{2.83}$$

$\forall \boldsymbol{v} \in [H^1(V)]^2$ and $q \in L^2(V)$.

No boundary conditions are required. The numerical implementation of the projector will be analyzed in section 3.3.4.

2.5 Weak formulation

Before being discretized, equations 2.71 and 2.72 must be recasted in their variational (weak) formulation.

2.5.1 p^0 equation

Let us first consider the Poisson equation for p^0. Its strong formulation reads:
Find $p^0(\boldsymbol{x}) : \mathbb{R}^{n_d} \to \mathbb{R}$ such that:

$$\begin{cases} -\nabla^2 p^0 = -\boldsymbol{\nabla} \cdot \boldsymbol{g} & \text{on } V \\ p^0 = 0 & \text{on } \partial V. \end{cases} \tag{2.84}$$

To obtain the weak formulation of the problem, we multiply the Poisson equation for a scalar test function v, whose appropriate functional space will be discussed later, and then we integrate over the domain V:

$$-\int_V v \nabla^2 p^0 = -\int_V v \nabla \cdot \boldsymbol{g}.$$

By integrating by parts and applying the divergence theorem, the left-hand side of the equation can be rewritten as:

$$-\int_V v \nabla^2 p^0 = \int_V \nabla v \cdot \nabla p^0 - \int_V \nabla \cdot (v \nabla p^0) = \int_V \nabla v \cdot \nabla p^0 - \int_S v \frac{\partial p^0}{\partial n}.$$

By choosing v such that it is zero on the whole Dirichlet boundary, we have:

$$\int_V \nabla v \cdot \nabla p^0 = -\int_V v \nabla \cdot \boldsymbol{g}.$$

Therefore, in order to have a well posed problem, the weak formulation of the problem reads:

Find $p^0 \in H_0^1 : \{\, p^0 \in H^1(V) \mid p^0 = 0 \quad \text{on} \quad S \,\}$ such that :

$$\int_V \nabla v \cdot \nabla p^0 = -\int_V v \nabla \cdot \boldsymbol{g} \qquad \forall v \in H_0^1 : \left\{\, v \in H^1(V) \;\middle|\; v = 0 \quad \text{on} \quad S \,\right\}.$$

As we can see, the pressure, which in the uncoupled formulation must satisfy a Poisson equation, is required to have an higher degree of regularity than in the standard coupled version (L^2).

2.5.2 $\boldsymbol{u}^0$ equation

Since the non linear term $(\boldsymbol{u} \cdot \boldsymbol{\nabla})\boldsymbol{u}$ is known and can be treated as a standard forcing term inside $\boldsymbol{g}$, the Helmholtz equations for the n_d velocity components are indipendent, where n_d is the number of spatial dimensions, so that the Helmholtz vectorial problem for $\boldsymbol{u}^0$ can be reduced to n_d scalar Helmholtz problems, each one for each different velocity component. In this section the weak formulation for the generic component u_l^0 , $l = 1 : n_d$, will be derived. Let us start from the strong formulation of the problem:

Find $u_l^0(\boldsymbol{x}) : \mathbb{R}^{n_d} \to \mathbb{R}$ such that:

$$\begin{cases} -\nabla^2 u_l^0 + \gamma u_l^0 = -\dfrac{\partial p^0}{\partial x_l} + g_l & \text{on } V \\ u_l^0 = b_l & \text{on } \partial V. \end{cases} \tag{2.85}$$

Similarly to what has been done for the Poisson equation for p^0, the Helmholtz equation is multiplied by a scalar test function v, whose appropriate functional space will be discussed later, and then integrated over the domain V. The part containing the Laplace operator can be integrated by parts, and applying the divergence thoerem eventually one obtains:

$$\int_V \nabla v \cdot \nabla u_l^0 + \gamma \int_V v u_l^0 = -\int_V v \left(-\frac{\partial p^0}{\partial x_l} + g_l \right),$$

since v is null on the boundary. In this case non-homogeneous Dirichlet boundary conditions have to be considered. To do so, a lifting procedure of the boundary values must be carried out in order to find a solution in the same vector space of the test function ($\mathbb{H}_0^1$). Let us introduce a lifting function R_b:

$$R_b \in H^1(V), \quad R_b|_S = b_l.$$

From the above definition, it follows that u_l^0 can be decomposes as $u_l^0 = \bar{u}_l^0 + R_b$, with $\bar{u}_l^0 \in H_0^1(V)$, since both u_l^0 and R_b are equal to b_l on the boundary. Then the weak formulation reads:

Find $\bar{u}_l^0 \in H_0^1 : \{\, \bar{u}_l^0 \in H^1(V) \mid \bar{u}_l^0 = 0 \quad \text{on} \quad S \,\}$ such that :

$$\int_V \nabla v \cdot \nabla u_l^0 + \gamma \int_V v u_l^0 = -\int_V v \left(-\frac{\partial p^0}{\partial x_l} + g_l \right) - \int_V \nabla v \cdot \nabla R_b - \gamma \int_V v R_b$$
$$\forall v \in H_0^1 : \left\{\, v \in H^1(V) \;\middle|\; v = 0 \quad \text{on} \quad S \,\right\}. \quad (2.86)$$

In the present work the boundary value lifting is performed only formally. As it will be pointed out in section 3.3.2, the discretization of the problem will be carried out without considering non homogeneous Dirichlet boundary conditions, which will be enforced using an algebraic trick at a later stage.

2.5.3 Weak problems summary

All the other weak formulations (for φ^0, p^k, $\boldsymbol{u}^k$ and φ^k) can be carried out similarly to those derived for p^0 and $\boldsymbol{u}^0$. Their associated form with homogeneous boundary data can be summarized as follows:

$$\begin{cases} \text{Find} \quad p^0 \in V & \int_V \boldsymbol{\nabla} v \cdot \boldsymbol{\nabla} p^0 = -\int_V v \boldsymbol{\nabla} \cdot \boldsymbol{g} & \forall v \in \mathbb{H}_0^1 \\ \text{Find} \quad u_l^0 \in V & \int_V (\boldsymbol{\nabla} v \cdot \boldsymbol{\nabla} u_l^0 + \gamma v u_l^0) = \int_V v \left(-\frac{\partial p^0}{\partial x_l} + g_l \right) & \forall v \in \mathbb{H}_0^1, \quad l = 1 : n_D \\ \text{Find} \quad \varphi^0 \in V & \int_V \boldsymbol{\nabla} v \cdot \boldsymbol{\nabla} \varphi^0 = \int_V v \boldsymbol{\nabla} \cdot \boldsymbol{u}^0 & \forall v \in \mathbb{H}_0^1 \end{cases} \quad (2.87)$$

$$\begin{cases} \text{Find} \quad p^k \in V & \int_V \boldsymbol{\nabla} v \cdot \boldsymbol{\nabla} p^k = 0 & \forall v \in \mathbb{H}_0^1 \\ \text{Find} \quad u_l^k \in V & \int_V (\boldsymbol{\nabla} v \cdot \boldsymbol{\nabla} u_l^k + \gamma v u_l^k) = \int_V v \left(-\frac{\partial p^k}{\partial x_l} \right) & \forall v \in \mathbb{H}_0^1, \quad l = 1 : n_D \\ \text{Find} \quad \varphi^k \in V & \int_V \boldsymbol{\nabla} v \cdot \boldsymbol{\nabla} \varphi^k = \int_V v \boldsymbol{\nabla} \cdot \boldsymbol{u}^k & \forall v \in \mathbb{H}_0^1, \end{cases} \quad (2.88)$$

where V is a suitable vector space with $V \subset \mathbb{H}^1$.

Chapter 3

Numerical Discretization

In this chapter the numerical discretization of the continuous problem is described. First, in section 3.1, the computational domain is discretized, employing the PIV measure points. Then in section 3.2 a finite element (FEM) discretization of the problem is introduced and it is used in section 3.3 in order to obtain a set of algebraic linear systems. Eventually in section 3.4 a brief description of the corresponding Fortran90 program is provided.

3.1 Mesh Generation

3.1.1 Grid data structure

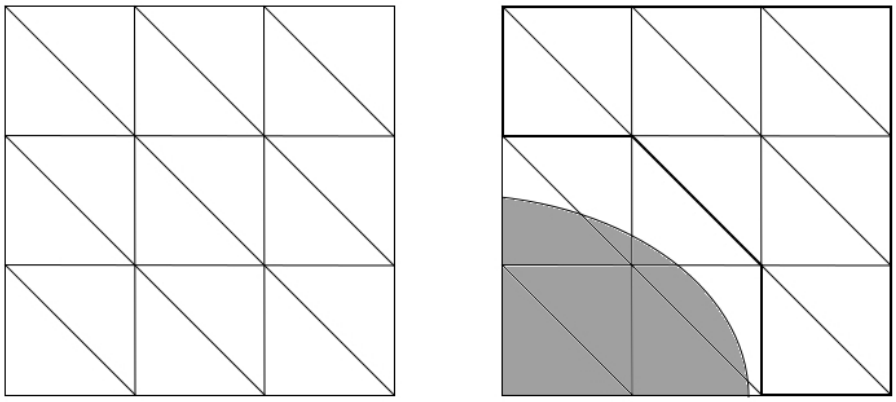
The grid data structure is composed of:

- **nodes data** collecting:
  - the number of the domain and boundary nodes;
  - an array containing the coordinates of the nodes;
  - a vector containing the global numbering of the boundary nodes;
  - a vector containing the boundary numbering of the boundary nodes;
  - a vector containing the numbering of the boundary side to which the boundary nodes belong;
- **grid data** collecting:
  - the number of the domain and boundary elements;
  - a vector containing the numbering of the domain elements;

- an array containing the global numbering of the nodes belonging to a given domain element;
- an array containing the neighboring elements;
- a vector containing the numbering of the boundary elements;
- two vectors containing the boundary elements type and the label of the boundary portion they belong to;
- an array containing the boundary numbering of the boundary nodes belonging to a given boundary element;
- an array containing the neighboring boundary elements;

3.1.2 Domain clipping

The PIV image postprocessing produces one file for each image couple containing the coordinates corresponding to the nodes of a rectangular structured grid. If there is a body or part of it in the image, PIV cannot correlate inside this part of the domain. The measures in this region are spurious and the corresponding nodes have to be left out from the grid.

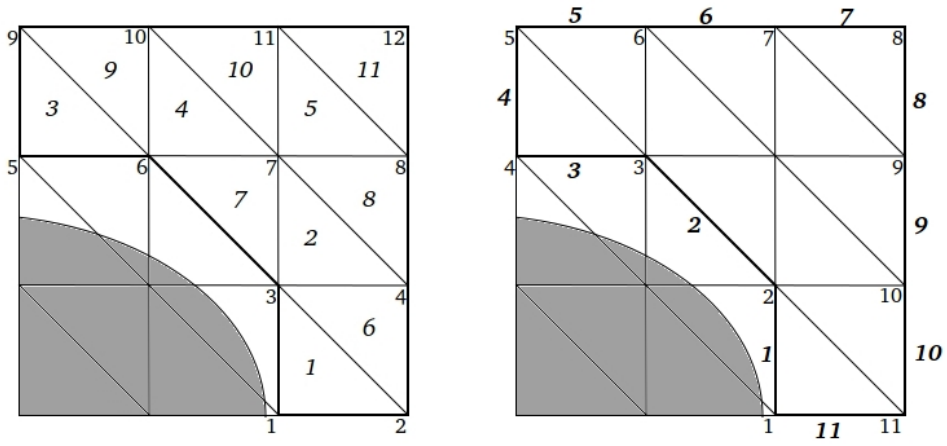


(a) *Non-shaped grid* (b) *Shaped grid (bold)*

Figure 3.1: Comparison between rectangular and shaped-domain (example)

3.1.3 Example

With reference to the small grid in Figure 3.2, an example of the grid data structure and of the procedure by which is constructed is illustrated in the following.



(a) *Domain nodes and elements numbering*

(b) *Boundary nodes and elements numbering*

Figure 3.2: Domain and boundary nodes and elements in the example

Umesh2d output

The dimensions of the UMesh2d data structures relative to the previous example are listed in Table 3.1.

Number of domain nodes	np_D	12
Number of domain elements	ne_D	11
Number of boundary nodes	np_B	11
Number of boundary elements	ne_B	11

Table 3.1: Summary of data structures dimensions

UMesh2d output consists of two files: nodes data file and grid data file. Each one of the two Umesh2d output files contains a *domain section* and a *boundary section*.

nodes.* Umesh2d file

The domain section in the nodes.* file contains the numbering and the coordinates of all the nodes (see Table 3.2).

The boundary section in the nodes.* file contains the boundary nodes both indicating their global and boundary numbering together with the side they belong to (see Table 3.3).

rrr	1.0	1.5	1.0	1.5	0.0	0.5	1.0	1.5	0.0	0.5	1.0	1.5
	0.0	0.0	0.5	0.5	1.0	1.0	1.0	1.0	1.5	1.5	1.5	1.5

Table 3.2: nodes. file: domain section*

$jD_{jB}(1:11)$	1	3	3	6	6	5	5	9	9	10	10
$jD_{jB}(12:22)$	11	11	12	12	8	8	4	4	2	2	1
$jB(1:11)$	1	2	3	4	5	6	7	8	9	10	11
$jB(12:22)$	12	13	14	15	16	17	18	19	20	21	22
$side(1:11)$	1	1	1	1	1	1	1	1	1	1	1
$side(12:22)$	1	1	1	1	1	1	1	1	1	1	1

Table 3.3: nodes. file: boundary section*

grid.* Umesh2d file

The domain section in the grid.* file contains the domain elements numbering, their type (triangles are two dimensional elements), global numbering of the nodes belonging to the elements and their neighboring elements (see Table 3.4).

The boundary section in the grid.* file contains boundary elements numbering, their type (line segments are one dimensional elements), the indices of the side they belong to, the boundary numbering of the nodes belonging to the boundary elements and their neighboring boundary elements (see Table 3.5).

m	1	2	3	4	5	6	7	8	9	10	11
$type$	2	2	2	2	2	2	2	2	2	2	2
j_m	1	3	5	6	7	4	7	8	10	11	12
	2	4	6	7	8	3	6	3	7	10	11
	3	7	9	10	11	2	3	4	6	7	8
ma_m	6	8	9	10	11	1	0	2	3	4	5
	0	7	0	9	10	0	2	0	4	5	0
	0	6	0	7	8	2	4	5	0	0	0

Table 3.4: grid. file: domain section*

mb	1	2	3	4	5	6	7	8	9	10	11
$typeB$	1	1	1	1	1	1	1	1	1	1	1
$sideB$	1	1	1	1	1	1	1	1	1	1	1
jB_{mB}	1	3	5	7	9	11	13	15	17	19	21
	2	4	6	8	10	12	14	16	18	20	22
mBa_{mB}	2	3	4	5	6	7	8	9	10	11	0
	0	1	2	3	4	5	6	7	8	9	10

Table 3.5: grid. file: boundary section*

3.1.4 Grid data construction

Domain

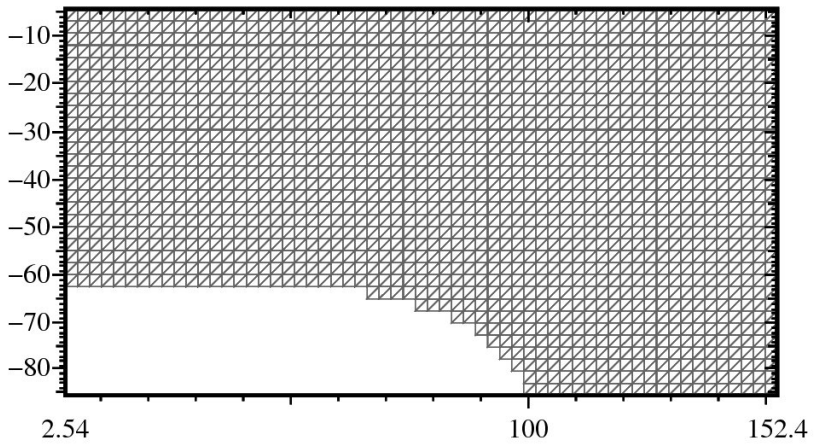
The method to evaluate the pressure field from PIV velocity measurements is tested here on a classical aeronautical problem: the flow around a wing section. Both flow at the leading edge and at the trailing edge will be investigated. One of the goals of this method is the indirect measure of the loads acting on the wing, for which it is very important to calculate the pressure field as close as possible to the wall. A portion of the wing section is captured in the image and therefore the grid has to be shaped around it.

First of all, all the nodes of the PIV rectangular domain are listed. A flag array collects the information about the position of each node with respect to the boundary. The zero entries indicate the nodes lying outside the flow region. The number of grid nodes is equal to the number of non-zero entry of the array. Then grid nodes are selected from the original PIV nodes and are sorted with increasing order from bottom to top and from left to right.

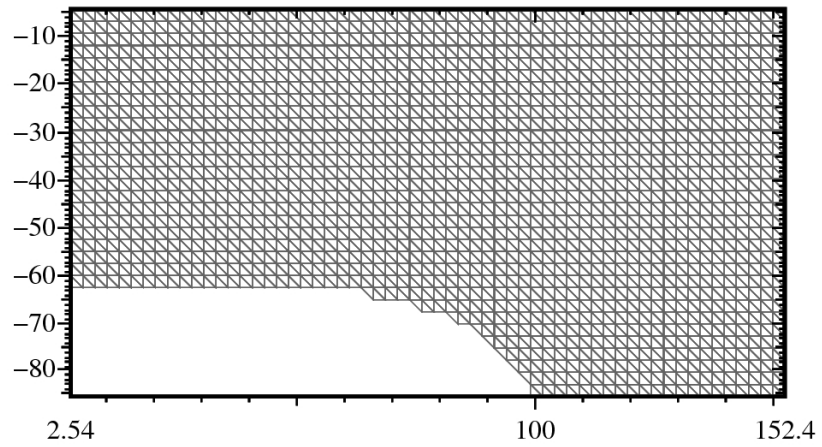
Due to the structured type of the mesh, domain elements are generated quite easily. The square Cartesian grid of PIV experimental data is divided into square isosceles triangular elements (the grid is not a Delaunay triangulation), then the connectivity array and the array of neighbouring elements are built. Depending on how triangular elements are built (i.e. depending on which diagonal of the square element is considered) some domain elements next to the boundary can be lost. For some domains (e.g. leading edge or trailing edge of an airfoil), a uniform choice through the domain can give globally as many elements as possible (Figure 3.3), but this cannot be considered as a rule.

Boundary

After building domain data structure, the boundary data structure can be built too. For two dimensional problems both the number of boundary



(a) *From bottom left to top right diagonal*



(b) *From top left to bottom right diagonal*

Figure 3.3: Results of different choices of the diagonal orientation

elements and the number of boundary nodes are equal to the number of zero elements in the array collecting the domain neighbouring elements.

The array containing the global numbering of the boundary nodes is then built. This array follows from by the choice of two contiguous boundary nodes. Once the first two boundary nodes are set, the starting point, the boundary orientation and the whole boundary nodes numbering are defined.

Eventually, the side indices, the boundary numbering of boundary elements and the boundary connectivity array are stored.

3.1.5 Grid smoothing over the airfoil boundary

The finite element method is quite sensitive to the presence of convex corners in the boundary, which can lead to local oscillations of the solution. Such kind of corners are located only on the airfoil portion of the boundary, where it is of crucial importance to have good results in order to accurately estimate airloads on the body. In order to improve the results and reduce the oscillations, the nodes on this part of the boundary can be moved on a more regular curve, approximating the airfoil profile. For example one can get the airfoil surface from the PIV image or build a curve from the available data.

If PIV pictures of good quality are available and the airfoil contour can be easily identified, the actual position of the profile is known: there is no need to build the boundary approximating the original sharp-cornered domain.

If PIV pictures are not available and there is no way to know the actual position of the airfoil surface, a smooth approximation of the profile boundary is needed. For example, one can choose a set of nodes, "the most inner" nodes, and approximate the airfoil profile by means of linear interpolation or cubic spline. Both methods could be of poor quality: in general linear interpolation is not smooth enough for the leading edge, while some oscillations could occur with cubic spline interpolation. These three different procedures are resumed in Figure 3.7.

Other "holistic" approximation can get better results (see figure 3.4). For example, once set the fixed nodes (those ones belonging to the airfoil portion of the domain which won't be moved), one can calculate an approximation of the local slope of the smooth boundary curve in these nodes, and then build the curve. Once the regular curve has been built, every node is moved to the closest point belonging to the regular curve. As a consequence of this procedure, the boundary elements change their shape.

A method to build a smooth approximation of the jagged boundary could consist in the procedure summarized in figure 3.5. The slope of the desired boundary curve is approximated in each one of the fixed nodes; then lines

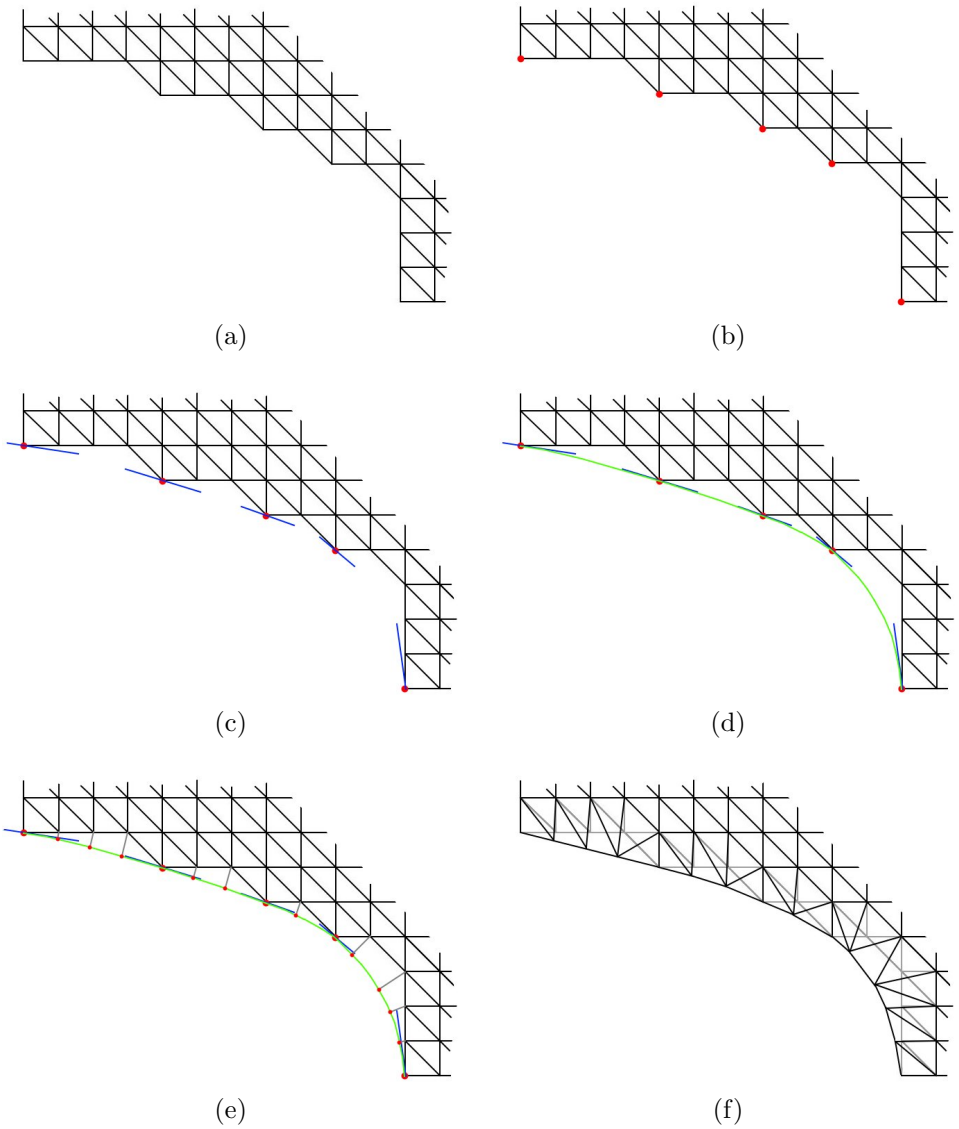


Figure 3.4: Smoothing grid border corresponding to the airfoil

with these slopes, passing through the relative nodes, are drawn and intersections between consecutive lines are found. Taking two consecutive lines, the segments lying between the fixed nodes and the intersection are divided by a set of equispaced points; these points are connected as shown in figure 3.5.Then the smooth curve is obtained by taking the envelope of these lines.

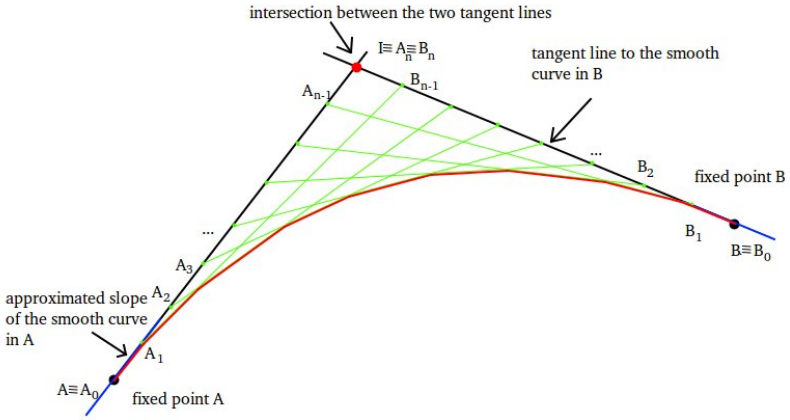


Figure 3.5: Smooth curve creation

After modifying the grid, velocity measures on the new nodes have to be approximated, for example through extrapolation of the available measures (3.6):

$$f(I_i) = \frac{\overline{B_i I_i}}{\overline{B_{i-1} B_i}} f(B_{i-1}) + \frac{\overline{B_{i-1} I_i}}{\overline{B_{i-1} B_i}} f(B_i) \tag{3.1}$$

$$f(N_i) = f(I_i) + \frac{\overline{N_i I_i}}{\overline{P_i B_i}} (f(P_i) - f(I_i)) \tag{3.2}$$

In general the extrapolation does not produce a divergence free velocity field, so oscillations are not guarantee to disappear. When the airfoil boundary is extracted from the PIV images, the velocity on the boundary nodes can be computed by imposing the no-slip condition or by extrapolation from internal points. Results will be presented in the following chapters.

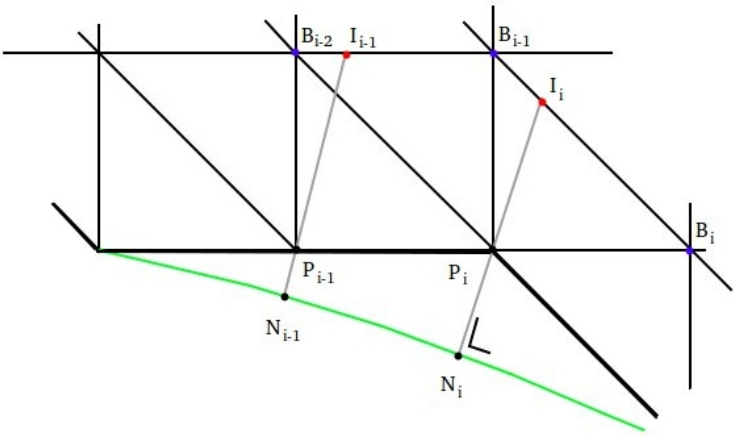
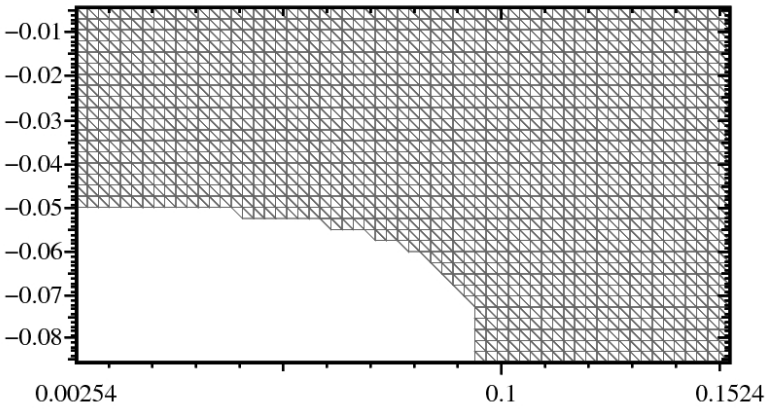


Figure 3.6: Linear extrapolation

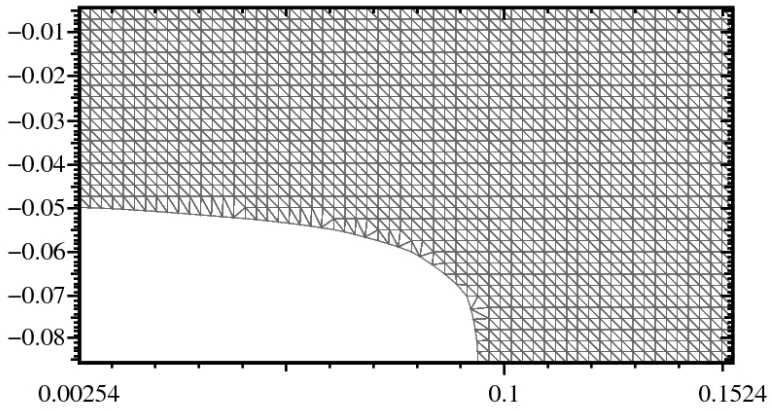
3.1.6 FEM grids for the unsteady problem

When the method is applied to turbulent flows, several PIV measures have to be collected and the velocity field should be (phase) averaged to calculate first and second order statistics, i.e. Reynolds stresses. If n velocity fields are available, a threshold $m \leq n$ is set: if m or more measures are valid, the node will be included in the numerical grid.

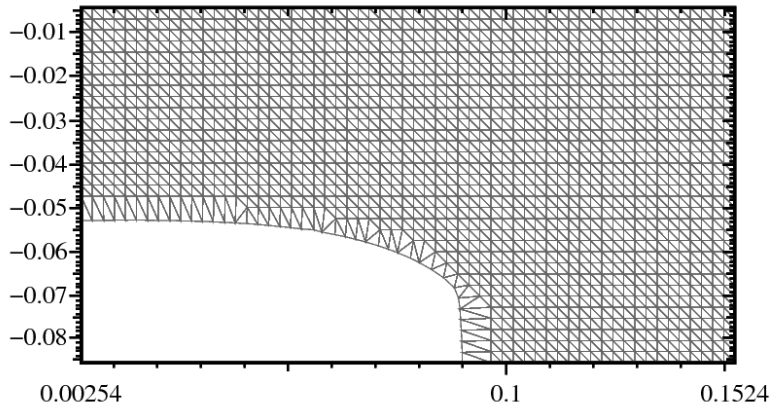
If a moving body is present in the PIV window, grids at different instants could be different, even thuogh the time interval between consecutive PIV measures is small ($\approx 10^{-2}$ s). The actual grid is composed of the nodes belonging to both grids.



(a) *Original sharp-cornered grid*



(b) *Grid without using PIV image informations*



(c) *Grid calculated from PIV image*

Figure 3.7: FEM grid

3.2 Finite Element Method

Let $\mathcal{T}_h$ be a regular triangulation of the domain V, as discussed in section 3.1, where h is the typical radius of the elements. We can now find an approximate solution to the weak problems (2.87, 2.88) in a finite-dimensional space. We choose X_h^r, namely the space of piecewise polynomial functions built over $\mathcal{T}_h$, which is defined as :

$$X_h^r = \left\{ u_h \in C^0(\bar{V}) \;\middle|\; u_h|_K \in \mathbb{P}^r \quad \forall \quad K \in \mathcal{T}_h \right\},$$

where $\mathbb{P}^r$ is the space of polynomial functions of degree r and K are the elements of grid $\mathcal{T}_h$. For this problem we will choose a linear piecewise approximation for p and φ and a quadratic piecewise approximation for $\boldsymbol{u}$, namely we select the vector spaces X_h^1 and X_h^2 respectively. The choice of two different vector spaces is necessary to satisfy the LBB condition, which must be satisfied even if the pressure and velocity equations are uncoupled (For further details see [17]). Introducing the basis ϕ_j, $1 \leq j \leq N$ for the vector space X_h^1 and the basis ψ_j, $1 \leq j \leq M$ for the vector space X_h^2, we can now express the approximate solutions by means of these bases:

$$\begin{cases} p_h(\boldsymbol{x}) = \sum_{j=1}^{N} \phi_j(\boldsymbol{x}) p_j, & p_h(\boldsymbol{x}) \in X_h^1 \\ u_{l_h}(\boldsymbol{x}) = \sum_{j=1}^{M} \psi_j(\boldsymbol{x}) u_{l_j}, & u_{l_h}(\boldsymbol{x}) \in X_h^2, \quad l = 1 : n_D \\ \varphi_h(\boldsymbol{x}) = \sum_{j=1}^{N} \phi_j(\boldsymbol{x}) \varphi_j, & \varphi_h(\boldsymbol{x}) \in X_h^1, \end{cases} \tag{3.3}$$

where N and M are the number of $\mathbb{P}^1$ and $\mathbb{P}^2$ nodes respectively and n_D is the number of spatial dimensions. By substituting these expressions in the previously developed weak formulation, one obtains a set of algebraic sparse linear systems whose solution represents the nodal values of the unknowns. This procedure will be widely discussed in the following section 3.3.

3.3 Discretized equations

3.3.1 Discretized g problem

Before discretizing the problems 2.87 and 2.88, we have to compute $\boldsymbol{g}$ from the experimentally known velocity fields $\boldsymbol{u}^n$ and $\boldsymbol{u}^{n-1}$. Let us recall the definition of l $\boldsymbol{g}$:

$$\boldsymbol{g} = \gamma \boldsymbol{u}^{n-1} - \frac{1}{\nu}(\boldsymbol{u}^n \cdot \boldsymbol{\nabla})\boldsymbol{u}^n - \frac{1}{\nu} \left\langle (\boldsymbol{u}^{',n} \cdot \boldsymbol{\nabla})\boldsymbol{u}^{',n} \right\rangle . \tag{3.4}$$

One can not compute directly $\boldsymbol{g}$ since the advection term $(\boldsymbol{u}^n \cdot \boldsymbol{\nabla})\boldsymbol{u}^n$ and the Reynolds stress term $\langle(\boldsymbol{u}^{'n} \cdot \boldsymbol{\nabla})\boldsymbol{u}^{'n}\rangle$ are to be evaluated yet. Let us first concentrate on the first part, neglecting the Reynolds stresses term for the moment being. $\boldsymbol{g}$ can be rewritten as follows:

$$\boldsymbol{g} = \gamma \boldsymbol{u}^{n-1} - \frac{1}{\nu}(\boldsymbol{u}^n \cdot \boldsymbol{\nabla})\boldsymbol{u}^n. \tag{3.5}$$

One can now recast the problem in a variational formulation, by multiplying the previous expression for a vector test function $\boldsymbol{v}$ and integrating over the domain V:

$$\int_V \boldsymbol{v} \cdot \boldsymbol{g} = \gamma \int_V \boldsymbol{v} \cdot \boldsymbol{u}^{n-1} - \frac{1}{\nu} \int_V \boldsymbol{v} \cdot (\boldsymbol{u}^n \cdot \boldsymbol{\nabla})\boldsymbol{u}^n. \tag{3.6}$$

Let now express each component of the approximate solution $\boldsymbol{g}_h$ by means of the $\mathbb{P}^2$ space basis:

$$g_{l_h}(\boldsymbol{x}) = \sum_{j=1}^{M} \psi_j(\boldsymbol{x}) g_{l_j}, \quad g_{l_h}(\boldsymbol{x}) \in X_h^2, \quad l = 1 : n_D \tag{3.7}$$

We can now substitute the expression 3.7, together with the analogous expressions for $u_{l_h}^n$ and $u_{l_h}^{n-1}$, in the variational formulation 3.6. Since the weak formulation must hold for every v , it is sufficient to verify it for every component of the space basis. So we can substitute v with the generic $\boldsymbol{g}$ basis element ψ_i. As an example, the x component of $\boldsymbol{g}$ reads, in a 2D case:

$$\sum_{j=1}^{M} \int_V \psi_i \psi_j g_{x_j} = \gamma \sum_{j=1}^{M} \int_V \psi_i \psi_j u_{x_j}^{n-1} - \frac{1}{\nu} \sum_{j=1}^{M} \int_V \psi_i \psi_j \left(u_{x_j}^{n-1} \frac{\partial \psi_j}{\partial x} + u_{y_j}^{n-1} \frac{\partial \psi_j}{\partial y} \right), \quad i = 1 : M. \tag{3.8}$$

Equation 3.8 represents the i-th row of the linear system $M\mathbf{g} = \mathbf{f}$, where :

$$M_{ij} = \int_V \psi_i \psi_j$$

$$\mathbf{f}_i = \gamma \sum_{j=1}^{M} \int_V \psi_i \psi_j u_{x_j}^{n-1} - \frac{1}{\nu} \sum_{j=1}^{M} \int_V \psi_i \psi_j \left(u_{x_j}^{n-1} \frac{\partial \psi_j}{\partial x} + u_{y_j}^{n-1} \frac{\partial \psi_j}{\partial y} \right),$$

Therefore, the weak problem reduces to the solution of a linear algebraic problem, with the mass matrix M, the right-hand side $\mathbf{f}$ and the unknown

$\mathbf{g}$, namely a vector containing the approximate solution evaluated in each $\mathbb{P}^2$ node. No boundary condition enforcement is needed. We are assuming here that the velocity fields are known on every $\mathbb{P}^2$ node. Unfortunately they are experimentally known only over the PIV measurement grid whose nodes are coincident with those of the $\mathbb{P}^1$ grid. Therefore a linear interpolation of the functions over the quadratic nodes is needed in order to cope with this problem.

However we still need to take into account the Reynolds stress term in order to complete the description of $\boldsymbol{g}$. As stated in section 2.1.1, $\boldsymbol{u}^n$ and $\boldsymbol{u}^{n-1}$ represent a phase average of the experimentally measured velocity fields for the actual and the previous time instant. As a consequence, for each experimentally measured velocity field with the same phase, one can compute the fluctuating vector field $\boldsymbol{u'}$ as the difference between the instantaneous velocity field and its phase-average. Then the quantity $(\boldsymbol{u}^{'n} \cdot \boldsymbol{\nabla})\boldsymbol{u}^{'n}$ can be evaluated solving a mass problem similar to the one previously introduced. Eventually all the terms $(\boldsymbol{u}^{'n} \cdot \boldsymbol{\nabla})\boldsymbol{u}^{'n}$ can be averaged in order to obtain $\langle(\boldsymbol{u}^{'n} \cdot \boldsymbol{\nabla})\boldsymbol{u}^{'n}\rangle$.

3.3.2 Discretized equations for the p, u, φ problems

Algebraic linear systems

Let us now discretize all the other weak formulations. Similarly to what has been done for the $\boldsymbol{g}$ equations, one can substitute expansions 3.3 into the weak formulations 2.87 and 2.88, getting rid of the boundary condition enforcement for the moment being. Since the weak formulations must hold for every test function of the associated functional space, it is sufficient to verify them for every element of their respective space basis. By substituting the test functions with their generic basis element ϕ_i or ψ_i , we have:

$$\begin{cases} \displaystyle\sum_{j=1}^{N} \int_V \boldsymbol{\nabla}\phi_i \cdot \boldsymbol{\nabla}\phi_j p_j^0 = -\sum_{j=1}^{M} \int_V \phi_i \sum_{l=1}^{n_D} \frac{\partial \psi_j}{\partial x_l} g_{l_j} & i = 1 : N \\ \displaystyle\sum_{j=1}^{M} \int_V (\boldsymbol{\nabla}\psi_i \cdot \boldsymbol{\nabla}\psi_j + \gamma\psi_i\psi_j) u_{l_j}^0 = -\sum_{j=1}^{N} \int_V \psi_i \frac{\partial \phi_j}{\partial x_l} p_j^0 + \sum_{j=1}^{M} \int_V \psi_i\psi_j g_{l_j} & \begin{matrix} i = 1 : M, \\ l = 1 : n_D \end{matrix} \\ \displaystyle\sum_{j=1}^{M} \int_V \boldsymbol{\nabla}\phi_i \cdot \boldsymbol{\nabla}\phi_j \varphi_j^0 = \sum_{j=1}^{M} \int_V \phi_i \sum_{l=1}^{n_D} \frac{\partial \psi_j}{\partial x_l} u_{l_j}^0 & i = 1 : N \end{cases} \tag{3.9}$$

$$\begin{cases} \sum_{j=1}^{N} \int_V \boldsymbol{\nabla}\phi_i \cdot \boldsymbol{\nabla}\phi_j p_j^k = 0 & i = 1:N \\ \sum_{j=1}^{M} \int_V (\boldsymbol{\nabla}\psi_i \cdot \boldsymbol{\nabla}\psi_j + \gamma\psi_i\psi_j) u_{l_j}^k = -\sum_{j=1}^{N} \int_V \psi_i \dfrac{\partial\phi_j}{\partial x_l} p_j^k & \begin{matrix} i = 1:M, \\ l = 1:n_D \end{matrix} \\ \sum_{j=1}^{M} \int_V \boldsymbol{\nabla}\phi_i \cdot \boldsymbol{\nabla}\phi_j \varphi_j^k = \sum_{j=1}^{M} \int_V \phi_i \sum_{l=1}^{n_D} \dfrac{\partial\psi_j}{\partial x_l} u_{l_j}^k & i = 1:N \end{cases} \tag{3.10}$$

Every equation above represents the i-th row of an algebraic linear system. Let us define the mass and the stiffness matrices concerning $\mathbb{P}^1$ and $\mathbb{P}^2$ nodes as follows :

$$K_{ij}^{(1)} = \int_V \boldsymbol{\nabla}\phi_i \cdot \boldsymbol{\nabla}\phi_j, \quad K_{ij}^{(2)} = \int_V \boldsymbol{\nabla}\psi_i \cdot \boldsymbol{\nabla}\psi_j \tag{3.11}$$

$$M_{ij}^{(2)} = \int_V \psi_i\psi_j \tag{3.12}$$

The algebraic linear problems can be written as follows:

$$\begin{cases} K^{(1)}\mathbf{p^0} = \mathbf{f}^{p^0} \\ (K^{(2)} + \gamma M^{(2)})\mathbf{u}_l^0 = \mathbf{f}^{u_l^0}, \quad l = 1:n_D \\ K^{(1)}\boldsymbol{\varphi^0} = \mathbf{f}^{\varphi^0} \end{cases} \tag{3.13}$$

$$\begin{cases} K^{(1)}\mathbf{p^k} = \mathbf{f}^{p^k} \\ (K^{(2)} + \gamma M^{(2)})\mathbf{u}_l^k = \mathbf{f}^{u_l^k}, \quad l = 1:n_D \\ K^{(1)}\boldsymbol{\varphi^k} = \mathbf{f}^{\varphi^k}, \end{cases} \tag{3.14}$$

The solution of each linear system is a vector containing the approximate solution evaluated in each grid node.

All the system matrices are symmetric definite-positive (since $\gamma > 0$). Moreover the matrices are sparse, due to the compact support of the finite element basis functions and a sparse direct solver (e.g. MUMPS) is used in order to reduce the computational effort. It can be highlighted the fact that the stiffness matrices corresponding to the Poisson equations for p^o, φ^0, p^k and φ^k are the same, so we can generate and factorize the K matrix once for all. The same property applies to the matrices associated with the Helmohltz problem.

Boundary conditions enforcement

As previously pointed out, the solving matrices and right-hand side terms have been generated neglecting the Dirichlet boundary conditions. In order to enforce boundary conditions, we can act directly on the derived algebraic system. Let us consider the boundary value assignment on boundary node i. The modification simply consists in setting every coefficient of the i-th row to zero, apart for the diagonal one which is set to one, and then modify the corresponding element of the right hand-side by setting there the value prescribed by the boundary condition.

3.3.3 Discretized integral condition

Algebraic linear system

In section 2.3.5 the variational problem relative to the integral condition has been formulated:

$$\sum_{k=1}^{N_D} \int_V (w\boldsymbol{\nabla}\cdot\boldsymbol{u}^k - \boldsymbol{\nabla}w\cdot\boldsymbol{\nabla}\varphi^k)\lambda^k = -\int_V (w\boldsymbol{\nabla}\cdot\boldsymbol{u}^0 - \boldsymbol{\nabla}w\cdot\boldsymbol{\nabla}\varphi^0) \quad (3.15)$$

Writing equation 3.15 for every element of the basis w^i corresponding to each Dirichlet boundary node, we have:

$$\sum_{k=1}^{N_D} \int_V (w^i\boldsymbol{\nabla}\cdot\boldsymbol{u}^k - \boldsymbol{\nabla}w^i\cdot\boldsymbol{\nabla}\varphi^k)\lambda^k = -\int_V (w^i\boldsymbol{\nabla}\cdot\boldsymbol{u}^0 - \boldsymbol{\nabla}w^i\cdot\boldsymbol{\nabla}\varphi^0), \quad i = 1:1:N_D. \quad (3.16)$$

Equation 3.16 represents the i-th row of the linear sistem $B\lambda = \beta$, where:

$$B_{ik} = \int_V (w^i\boldsymbol{\nabla}\cdot\boldsymbol{u}^k - \boldsymbol{\nabla}w^i\cdot\boldsymbol{\nabla}\varphi^k), \quad (3.17)$$

$$b_i = -\int_V (w^i\boldsymbol{\nabla}\cdot\boldsymbol{u}^0 - \boldsymbol{\nabla}w^i\cdot\boldsymbol{\nabla}\varphi^0). \quad (3.18)$$

To sum up, in order to compute the influence Matrix B one has to solve N_D 2.72 problems, each one with a different boundary condition for p^k. Each problem will give u^k and φ^k, which are necessary to compute the k-th column of the influence matrix. The right-hand side β can be computed similarly by solving the problem 2.71.

Influence matrix properties

The derived influence matrix B has the following properties:

- The influence matrix is singular. Because of full Dirichlet boundary conditions imposed on the velocity field, the pressure is determined up to an additive arbitrary constant. This singularity can be eliminated both by imposing a prescribed mean value or the value of pressure in a node.

- The influence matrix has full pattern.

- The influence matrix is symmetric positive-semidefinite. Therefore once desingularized the linear system can be solved iteratively by means of the conjugate gradient method. This property can be demonstrated as follows. By using the orthogonality property 2.48, one obtains:

$$\begin{aligned} B_{ik} &= \int_V (w^i \boldsymbol{\nabla} \cdot \boldsymbol{u}^k - \boldsymbol{\nabla} w^i \cdot \boldsymbol{\nabla} \varphi^k) \\ &= \int_V (w^i \boldsymbol{\nabla} \cdot \boldsymbol{u}^k + (-\boldsymbol{\nabla} w^i + \boldsymbol{\nabla} p^i) \cdot \boldsymbol{\nabla} \varphi^k) \end{aligned} \tag{3.19}$$

Let us now focus on the term $-\int_V (\boldsymbol{\nabla} w^i - \boldsymbol{\nabla} p^i) \cdot \boldsymbol{\nabla} \varphi^k)$. Using twice the divergence theorem and integration by parts, introducing the homogeneous boundary conditions for $\boldsymbol{u}^k$ and considering $p^i|_S = w^i|_S$ and the Poisson equation for φ^k, we have:

$$\begin{aligned} -\int_V (\boldsymbol{\nabla} w^i - \boldsymbol{\nabla} p^i) \cdot \boldsymbol{\nabla} \varphi^k) &= -\oint_S (w^i - p^i) \boldsymbol{\nabla} \varphi^k \cdot \hat{\boldsymbol{n}} + \int_V (w^i - p^i) \nabla^2 \varphi^k \\ &= \int_V (w^i - p^i) \nabla^2 \varphi^k = -\int_V (w^i - p^i) \boldsymbol{\nabla} \cdot \boldsymbol{u}^k \\ &= -\oint_S (w^i - p^i) \boldsymbol{u}^k \cdot \hat{\boldsymbol{n}} + \int_V (\boldsymbol{\nabla} w^i - \boldsymbol{\nabla} p^i) \cdot \boldsymbol{u}^k \\ &= \int_V (\boldsymbol{\nabla} w^i - \boldsymbol{\nabla} p^i) \cdot \boldsymbol{u}^k \end{aligned}$$

Using again the divergence theorem and introducing the Helmholtz equation for $\boldsymbol{u}^k$ with the associated boundary conditions:

$$
\begin{aligned}
B_{ik} &= \int_V (w^i \boldsymbol{\nabla} \cdot \boldsymbol{u}^k + (\boldsymbol{\nabla} w^i - \boldsymbol{\nabla} p^i) \cdot \boldsymbol{u}^k) \\
&= \int_V \boldsymbol{\nabla} \cdot (w^i \boldsymbol{u}^k) - \int_V \boldsymbol{\nabla} p^i \cdot \boldsymbol{u}^k \\
&= \oint_S w^i \boldsymbol{u}^k \cdot \hat{\boldsymbol{n}} - \int_V \boldsymbol{\nabla} p^i \cdot \boldsymbol{u}^k \\
&= -\int_V \boldsymbol{\nabla} p^i \cdot \boldsymbol{u}^k = \int_V (-\nabla^2 + \gamma) \boldsymbol{u}^i \cdot \boldsymbol{u}^k = \\
&= \int_V (\boldsymbol{\nabla} \boldsymbol{u}^i : \boldsymbol{\nabla} \boldsymbol{u}^k + \gamma \boldsymbol{u}^i \cdot \boldsymbol{u}^k) - \oint_S \boldsymbol{u}^k \cdot (\boldsymbol{\nabla} \boldsymbol{u}^i \hat{\boldsymbol{n}}) \\
&= \int_V (\boldsymbol{\nabla} \boldsymbol{u}^i : \boldsymbol{\nabla} \boldsymbol{u}^k + \gamma \boldsymbol{u}^i \cdot \boldsymbol{u}^k) \\
&= B_{ki}
\end{aligned}
$$

Solution of the algebraic linear system

As highlighted in the previous subsection, the solution of the algebraic linear system presents some criticalities. First of all, the fact that the influence matrix is full, so one can take no advantage by using sparse linear solvers such as MUMPS, makes the computational cost of the operation very expansive. Moreover, since the influence matrix B is singular due to the fact that the pressure field is determined up to an arbitrary additive constant, we have to desingularize the matrix before its factorization.

A first method useful to eliminate the singularity is to artificially impose the pressure value on one node or, alternatively, to impose the mean value of the pressure. To do so, it is sufficient to replace the row corresponding to the node where we want to impose the pressure value with a row of zeros, but for the diagonal term which is set to one. The corresponding term of the right-hand is set equal to the pressure value to be imposed. Alternatively, to impose a zero mean value of the pressure on the boundary, one can replace any row of the matrix B with a row of ones, and then put the corresponding term of the right-hand side to zero. This method is definitely of simple implementation, however it can be seen that the modified matrix B, especially for low Δt and ν, is extremely ill-conditioned, comporting a bad solution computation.

In order to avoid this problem, a spectral decomposition can be performed. Since B is symmetric positive-semidefinite, all its eigenvalues are real and non negative. Moreover the right eigenvector matrix X is orthogonal, hence $X^{-1} = X^T$. Therefore the matrix B can be decomposed as follows:

$$B = X \Lambda X^T, \tag{3.20}$$

where Λ is a diagonal matrix containing the eigenvalues of the matrix B. Since the matrix B is singular only once, one of the eigenvalues is equal to zero. Moreover the corresponding eigenvector is of the form $\alpha\mathbf{1}$, where α corresponds to the pressure arbitrary additive constant. Let us rename the solution of the linear problem $\mathbf{x} = \lambda$ in order not to confuse it with the eigenvalue vector. By introducing expression 3.20 in $B\mathbf{x} = \beta$ and pre-multiplying by X^T, one obtains:

$$B\mathbf{x} = \beta, \qquad X\Lambda X^T\mathbf{x} = \beta, \qquad \Lambda X^T\mathbf{x} = X^T\beta \tag{3.21}$$

Let us rename $\tilde{\mathbf{x}} = X^T\mathbf{x}$ and $\tilde{\beta} = X^T\beta$. One obtains the following linear system:

$$\Lambda\tilde{\mathbf{x}} = \tilde{\beta} \tag{3.22}$$

Since the matrix Λ is diagonal, every equation of this linear system can be solved separately. However we must take into account that the first eigenvalue, if we order the eigenvalues in ascending order, will be zero ($\Lambda_{11} = 0$) since the matrix is positive semi-definite. Hence the solution of system 3.22 reads:

$$\tilde{\mathbf{x}}_1 = 0, \qquad \tilde{\mathbf{x}}_i = \frac{\tilde{\beta}_i}{\Lambda_{ii}}, \quad i = 2 : N_D \tag{3.23}$$

Eventually the solution of the influence matrix linear system can be retrieved by :

$$\mathbf{x} = X\tilde{\mathbf{x}} \tag{3.24}$$

The spectral decomposition is generally more onerous from the computational point of view than a normal factorization, but the relatively coarse mesh of the PIV measurements and the fact that the system has as many equations as the boundary nodes (not all the grid nodes), make it affordable.

3.3.4 Discretized equations for the projector

The projection problem presented in section 2.4 can be simply discretized by substituting expansions 3.3 in equation 2.83 and then replacing the test functions by a generic element of their respective basis, namely ϕ_i for λ and ψ_i for $\boldsymbol{u}$.

As an example, the equations for the discretized x component of the projection problem reads:

$$\sum_{j=1}^{M}\int_V \psi_i\psi_j u_{x_j} - \sum_{j=1}^{N}\int_V \phi_j\frac{\partial\psi_i}{\partial x}\lambda_j = \sum_{j=1}^{M}\int_V \psi_i\psi_j\tilde{u}_{x_j}, \quad i = 1,\ldots,M. \tag{3.25}$$

while the discretized divergence term reads:

$$-\sum_{j=1}^{M}\int_V \phi_i \frac{\partial \psi_j}{\partial x} u_{x_j} - \sum_{j=1}^{M}\int_V \phi_i \frac{\partial \psi_j}{\partial y} u_{y_j} = 0, \quad i = 1, \dots, N. \tag{3.26}$$

As we can see, the problem for the two velocity components is coupled by the Lagrange multipliers. By recalling the definition of the $\mathbb{P}^2$ mass matrix and introducing the following matrices associated with the gradient operator

$$G_x = \int_V \phi_j \frac{\partial \psi_i}{\partial x}, \quad G_y = \int_V \phi_j \frac{\partial \psi_i}{\partial y}, \tag{3.27}$$

the linear problem derived from the discretized equations 2.83 reads:

$$\begin{bmatrix} M^{(2)} & 0 & -G_x \\ 0 & M^{(2)} & -G_y \\ -G_x^T & -G_y^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{u}_x \\ \mathbf{u}_y \\ \lambda \end{bmatrix} = \begin{bmatrix} M^{(2)}\tilde{\mathbf{u}}_x \\ M^{(2)}\tilde{\mathbf{u}}_y \\ 0 \end{bmatrix}. \tag{3.28}$$

The present linear system does not require to impose any boundary condition. However, it is singular once due to the fact that λ is determined up to an arbitrary additive constant, similarly to what happens to the pressure field in the incompressible Navier–Stokes equations.

In order to be solved, the right hand side of this problem must belong to the matrix range. To do so, the following projector can be used: $(I - \mathbf{x}\mathbf{x}^*)$, where I is the identity matrix and $\mathbf{x}$ is the right eigenvector corresponding to the null eigenvalue, hence a basis for the kernel of the matrix, and has the following form

$$\mathbf{x} = \begin{bmatrix} 0 & \dots & 0,0 & \dots & 0,1 & \dots & 1 \end{bmatrix}^T, \tag{3.29}$$

with $\mathbf{x}^*$ being the transpose of the corresponding left eigenvector, which is simply equal to $\mathbf{x}^T$ since the problem is self-adjoint. By applying the projector to the right-hand side, we can notice that it returns the same input vector, since the only nonzero entries of $\mathbf{x}\mathbf{x}^T$ multiply the part of the right-hand side corresponding to the Lagrange multipliers, which is zero. That means that the right hand side is already in the operator range, and by using a solver which can handle this kind of singularities, such as MUMPS for instance, one of the infinitely many exact solutions is returned. All the possible solutions share the same velocity field, the λ fields differing for an additive constant.

3.4 F90 Program description

In this section a brief description of the Fortran90 Finite Element program which has been developed is reported.

PIV database processing The PIV output is processed. The velocity fields corresponding to the same phase are averaged, and the Reynolds stress term $\langle(\boldsymbol{u}^{'n}\cdot\boldsymbol{\nabla})\boldsymbol{u}^{'n}\rangle$ is computed.

Wind tunnel data processing The program reads the wind tunnel data of the considered experiment. These data include the asymptotic value of pressure, temperature, density and dynamic pressure, plus the Reynolds number computed for a unit chord. From these data the kinematic viscosity can be computed.

Mesh and 'belt' Generation Starting from the PIV output, the shaped mesh is generated, as shown in section 3.1. Moreover the computational domain for the influence problem, which , thanks to w basis function choice, is a thin strip composed by all the elements with at least one node belonging to the boundary, is extracted from the complete mesh.

0 problems solution The problems:

$$\begin{cases} -\nabla^2 p^0 = -\boldsymbol{\nabla}\cdot\boldsymbol{g}, & p^0|_S = 0 \\ (-\nabla^2+\gamma)\boldsymbol{u^0} = -\boldsymbol{\nabla}p^0+\boldsymbol{g}, & \boldsymbol{u}^0|_S = \boldsymbol{b} \\ -\nabla^2\varphi^0 = \boldsymbol{\nabla}\cdot\boldsymbol{u^0}, & \varphi^0|_S = 0, \end{cases} \tag{3.30}$$

are solved by means of the Finite Element Method as described in section 3.3.2. The solution is stored in order to compute the right-hand side of the influence problem.

k **problems solution** The following problems, as many as the number of the Dirichlet nodes, are solved:

$$\begin{cases} -\nabla^2 p^k = 0, & p^k|_S = \mu^k \\ (-\nabla^2+\gamma)\boldsymbol{u}^k = -\boldsymbol{\nabla}p^k, & \boldsymbol{u}^k|_S = 0 \qquad k = 1:N_D \\ -\nabla^2\varphi^k = \boldsymbol{\nabla}\cdot\boldsymbol{u}^k, & \varphi^k|_S = 0. \end{cases} \tag{3.31}$$

Each problem has a different Dirichlet boundary condition for p^k. The solution is stored in order to compute the left-hand side of the influence problem and to reconstruct the pressure and velocity fields at a later stage.

Influence problem solution The influence matrix and the corresponding right-hand side are computed:

$$B_{ik} = \int_V (w^i \boldsymbol{\nabla} \cdot \boldsymbol{u}^k - \boldsymbol{\nabla} w^i \cdot \boldsymbol{\nabla} \varphi^k), \tag{3.32}$$

$$b_i = -\int_V (w^i \boldsymbol{\nabla} \cdot \boldsymbol{u}^0 - \boldsymbol{\nabla} w^i \cdot \boldsymbol{\nabla} \varphi^0). \tag{3.33}$$

The problem is then solved by means of its spectral decomposition, obtaining the vector λ, which represents at the same time the value of the pressure field on the boundary and the vector of the coefficients of the pressure expansion.

Solution reconstruction Finally the pressure field is reconstructed:

$$p(\boldsymbol{x}) = p^0(\boldsymbol{x}) + \sum_{k=1}^{N_D} p^k(\boldsymbol{x}) \lambda^k, \tag{3.34}$$

$$\boldsymbol{u}(\boldsymbol{x}) = \boldsymbol{u}^0(\boldsymbol{x}) + \sum_{k=1}^{N_D} \boldsymbol{u}^k(\boldsymbol{x}) \lambda^k \tag{3.35}$$

Note that p does exactly not corresponds to the physical pressure field, since the equations have been scaled by the kinematic viscosity and density. The physical pressure values are obtained by multiplying the computed pressure field by ν and ρ:

$$p = \rho \nu \tilde{p} \tag{3.36}$$

Chapter 4

Program Validation

The aim of this chapter is to show the results of the program verification and validation process. In section 4.1 the single parts of the program are individually validated and their convergence properties are shown.Then in section 4.2 the whole program is validated on an unsteady test-case. Finally in 4.3 the robustness of the method is investigated for both deterministic and stochastic errors.

4.1 Convergence check for the solver components

In this section the single parts of the program are individually tested. In the first part, the Poisson solver for pressure and φ, the Helmholtz solver for velocity and the mass solver for $\mathbf{g}$ are applied to meshes with different values of h in order to test the convergence of the method. Afterwards, the right-hand side generation for the different equations is tested too.

4.1.1 Poisson $\mathbb{P}^1$ scalar solver validation

Test case definition

The Poisson $\mathbb{P}^1$ scalar solver is tested on the following test case over the rectangular domain $\Omega \subset \mathbb{R}^2$: $[0,A] \times [0,B]$, $(A = 3, B = 2)$. The fully Dirichelet problem reads:

Find $u(\boldsymbol{x}) : \mathbb{R}^2 \to \mathbb{R}$ such that

$$\begin{cases} -\nabla^2 u = f(\boldsymbol{x}) & \text{on } \Omega \\ u = b(s) & \text{on } \partial\Omega. \end{cases} \tag{4.1}$$

We consider the following choices for the forcing term, f, and for the prescribed solution on the Dirichlet boundary, b:

$$f = 4\pi^2 \left(\frac{1}{A^2} + \frac{1}{B^2} \right) \cos\left(2\pi \frac{x}{A}\right) \cos\left(2\pi \frac{y}{B}\right),$$

$$b = \cos\left(2\pi \frac{x}{A}\right) \cos\left(2\pi \frac{y}{B}\right).$$

The exact solution for this choice of f and b is :

$$u_{ex} = \cos\left(2\pi \frac{x}{A}\right) \cos\left(2\pi \frac{y}{B}\right).$$

Convergence analysis

For the convergence analysis, we consider the normalized infinity norm of the error, defined as:

$$Err = \frac{\|u - u_{ex}\|_\infty}{\|u_{ex}\|_\infty} = \frac{\max\limits_{\boldsymbol{x}\in\Omega} |u - u_{ex}|}{\max\limits_{\boldsymbol{x}\in\Omega} |u_{ex}|}, \tag{4.2}$$

where u_{ex} is the exact solution. The L^∞ norm has been used throughout in this work since it is more apt to evaluate the performance of the proposed technique from the viewpoint of the experimentalist. The results are shown in Table (4.1) :

h	$Err[-]$	$Err\%$
0.5	0.4571	45.71
0.25	0.1400	14.00
0.125	3.6721 E-002	3.67
0.0625	9.2890 E-003	0.928
0.03125	2.3290 E-003	0.232
0.015625	5.8269 E-004	0.0582

Table 4.1: Convergence analysis for the Poisson scalar solver

The same results are represented in logarithmic scale in Figure 4.1. As it can be appreciated from the figure, this method provides a quadratic convergence if the error is calculated in L^∞ norm.

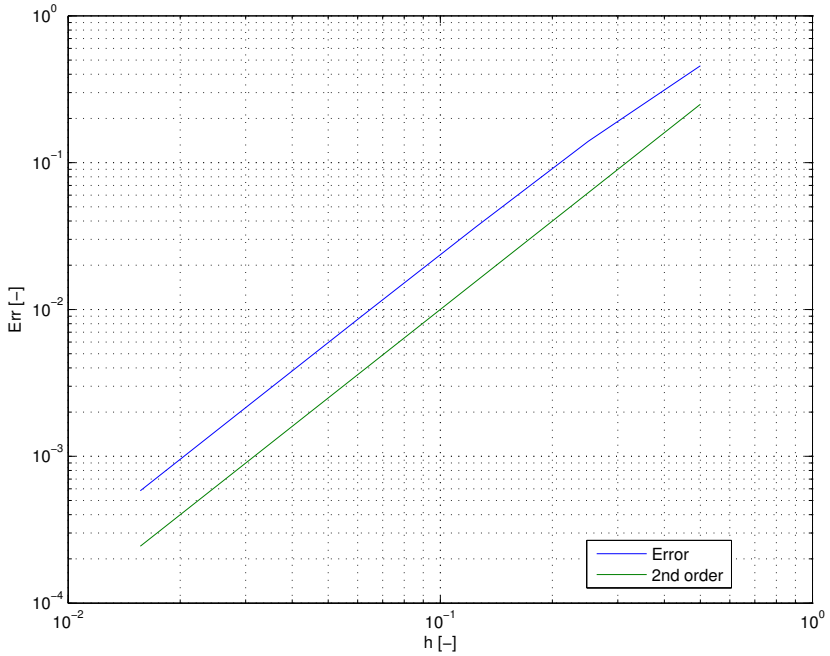


Figure 4.1: Convergence analysis for the Poisson scalar solver

4.1.2 Helmholtz $\mathbb{P}^2$ vector solver validation

Test case definition

Similarly to the Poisson solver, the Helmholtz $\mathbb{P}^2$ vector solver is tested on the rectangular domain $\Omega \subset \mathbb{R}^2 : [0,A] \times [0,B]$ $(A = 3, B = 2)$ for the following Dirichlet problem:

Find $\boldsymbol{u}(\boldsymbol{x}) : \mathbb{R}^2 \to \mathbb{R}^2$ such that

$$\begin{cases} -\nabla^2 \boldsymbol{u} + \gamma \boldsymbol{u} = \boldsymbol{f}(\boldsymbol{x}) & \text{on } \Omega \\ \boldsymbol{u} = \boldsymbol{b}(s) & \text{on } \partial\Omega. \end{cases} \tag{4.3}$$

We consider the following choice for the forcing term $\boldsymbol{f}$ and for the prescribed solution on the Dirichlet boundary $\boldsymbol{b}$:

$$f_x = f_y = \left(4\pi^2 \left(\frac{1}{A^2} + \frac{1}{B^2}\right) + \gamma\right) \cos\left(2\pi \frac{x}{A}\right) \cos\left(2\pi \frac{y}{B}\right),$$

$$b_x = b_y = \cos\left(2\pi \frac{x}{A}\right) \cos\left(2\pi \frac{y}{B}\right).$$

The exact solution for this choice of $\boldsymbol{f}$ and $\boldsymbol{g}$ is :

$$u_{ex,x} = u_{ex,y} = \cos\left(2\pi\frac{x}{A}\right)\cos\left(2\pi\frac{y}{B}\right).$$

Convergence analysis

For the convergence analysis we consider the normalized infinity norm of the error also in this case, defined as:

$$Err = \frac{\sqrt{\sum_{i=1}^{2}\|u_i - u_{ex,i}\|_\infty^2}}{\sqrt{\sum_{i=1}^{2}\|u_{ex,i}\|_\infty^2}}, \tag{4.4}$$

where $u_{ex,i}$ is the exact solution for the $i-th$ component of vector $\boldsymbol{u}$. The results are shown in Table (4.2) for $\gamma = 1$:

h	$Err[-]$	$Err\%$
0.5	2.7325 E-002	2.732
0.25	2.1659 E-003	0.217
0.125	1.4319 E-004	0.0143
0.0625	9.0745 E-006	0.001 91
0.03125	5.6911 E-007	0.000 569
0.015625	3.5600 E-008	0.000 035 6

Table 4.2: Convergence analysis for the Helmholtz vector solver

The same results are represented in the logarithmic-scale plot of Figure (4.2). As it can be appreciated from the figure, this method provides 4th order convergence with this measure of the error.

4.1.3 Mass $\mathbb{P}^2$ $\boldsymbol{g}$ solver validation

Test case definition

As discussed in section 3.3.1, we now analyse the mass solver for $\boldsymbol{g}$, which solves equation 4.5 given the present and the past step velocity field by means of $\mathbb{P}^2$ finite elements:

$$\boldsymbol{g} = \gamma\boldsymbol{u}^{n-1} - \frac{1}{\nu}(\boldsymbol{u}^n\cdot\boldsymbol{\nabla})\boldsymbol{u}^n - \frac{1}{\nu}\left\langle(\boldsymbol{u}^{',n}\cdot\boldsymbol{\nabla})\boldsymbol{u}^{',n}\right\rangle. \tag{4.5}$$

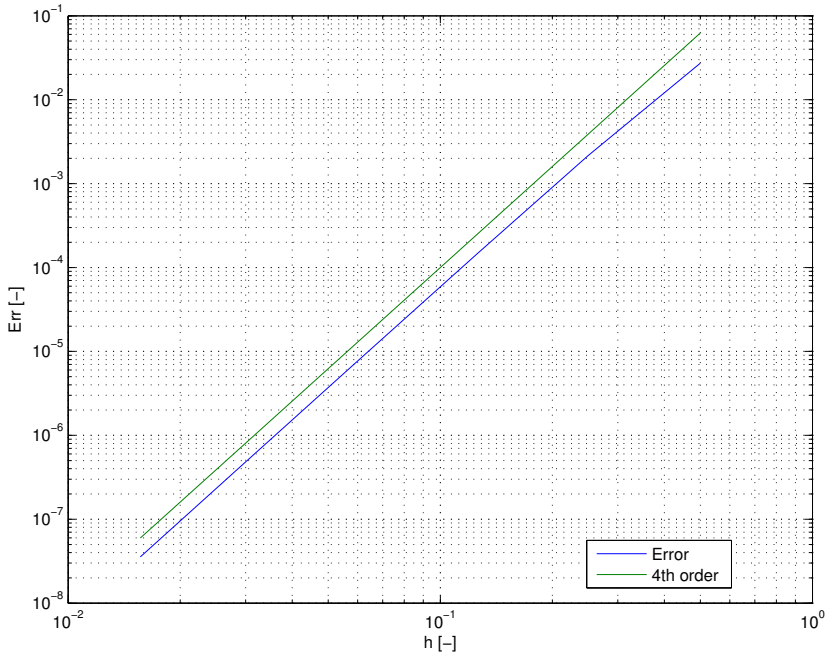


Figure 4.2: Convergence analysis for the Helmholtz vectorial solver

For the moment being, the Reynolds stress term is neglected, as it would only turn into a standard forcing term. The following choices of $\boldsymbol{u}^n$, $\boldsymbol{u}^{n-1}$ are considered for the test case:

$$\boldsymbol{u}^n = \left(\cos\left(2\pi\frac{x}{A}\right)\cos\left(2\pi\frac{y}{B}\right)\right)\hat{\mathbf{i}} + \left(\sin\left(2\pi\frac{x}{A}\right)\sin\left(2\pi\frac{y}{B}\right)\right)\hat{\mathbf{j}},$$

$$\boldsymbol{u}^{n-1} = \boldsymbol{u}^n + 0.1.$$

The exact solution for this choice of $\boldsymbol{u}^{n-1}$ and $\boldsymbol{u}^n$ is :

$$\boldsymbol{g}_{x,ex} = \gamma\boldsymbol{u}_x^{n-1} + \frac{2\pi}{\nu A}\boldsymbol{u}_x^n\left(\sin\left(2\pi\frac{x}{A}\right)\cos\left(2\pi\frac{y}{B}\right)\right) + \frac{2\pi}{\nu B}\boldsymbol{u}_y^n\left(\cos\left(2\pi\frac{x}{A}\right)\sin\left(2\pi\frac{y}{B}\right)\right),$$

$$\boldsymbol{g}_{y,ex} = \gamma\boldsymbol{u}_y^{n-1} - \frac{2\pi}{\nu A}\boldsymbol{u}_x^n\left(\cos\left(2\pi\frac{x}{A}\right)\sin\left(2\pi\frac{y}{B}\right)\right) - \frac{2\pi}{\nu B}\boldsymbol{u}_y^n\left(\sin\left(2\pi\frac{x}{A}\right)\cos\left(2\pi\frac{y}{B}\right)\right).$$

Convergence analysis

For the convergence analysis we consider the normalized L^∞ norm of the error, as defined in Equation 4.4. The results are shown in Table (4.3),

computed for $\nu = 1.5 \cdot 10^{-5}$ and $dt = 0.001$.

h	$Err[-]$	$Err\%$
0.5	3.5116 E-004	0.0351
0.25	1.0471 E-004	0.0105
0.125	3.0701 E-005	0.003 07
0.0625	8.1286 E-006	0.000 813
0.03125	2.0656 E-006	0.000 207
0.015625	5.1918 E-007	0.000 051 9

Table 4.3: Convergence analysis for the mass $\boldsymbol{g}$ solver

The same results are represented in the logarithmic-scale plot of Figure (4.3). As it can be appreciated from the figure, this method provides a second order convergence with this measure of the error.

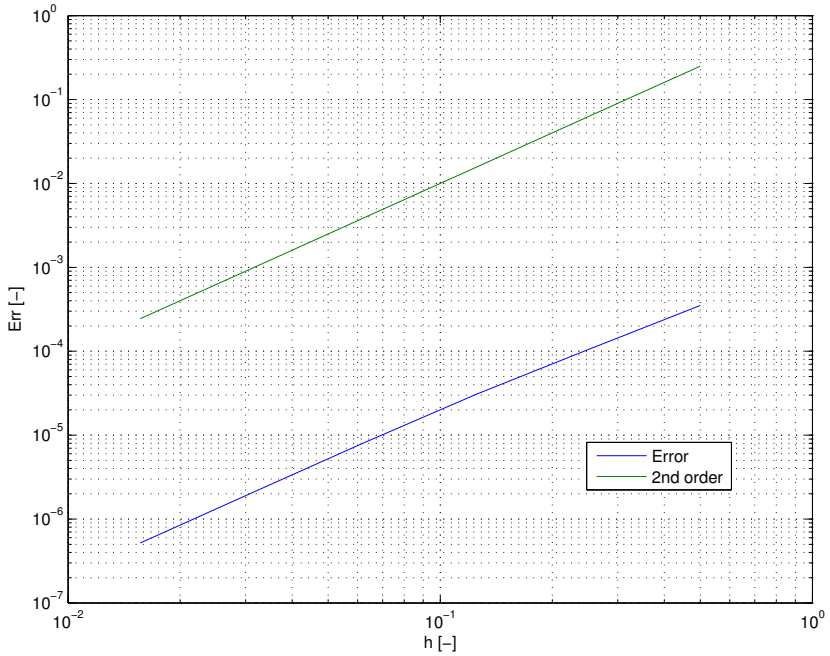


Figure 4.3: Convergence analysis for the mass $\boldsymbol{g}$ solver

4.1.4 RHS validation

Validation procedure

It is now shown a procedure whose purpose is to verify whether the RHS of the Helmohltz problem for velocity and the RHS of the Poisson problem for pressure (equations 2.71) are correctly generated. As an example, let us take the RHS of the pressure Poisson equation, that is $-\boldsymbol{\nabla} \cdot \boldsymbol{g}$, and rename it y, so that:

$$y = -\boldsymbol{\nabla} \cdot \boldsymbol{g}. \tag{4.6}$$

We now have to recast the problem in variational form and expand the unknown as a linear combination of the space basis functions, as shown in Section 3.2:

$$\int_\Omega vy = -\int_\Omega v\boldsymbol{\nabla} \cdot \boldsymbol{g} \tag{4.7}$$

$$\sum_{j=1}^{n} \left(\int_\Omega \phi_i \phi_j \right) y_j = -\int_\Omega \phi_i \boldsymbol{\nabla} \cdot \boldsymbol{g}. \tag{4.8}$$

Similarly to the mass $\boldsymbol{g}$ solver, equation 4.8 can be seen as the $i-th$ row of an algebraic linear system whose coefficient matrix is the $\mathbb{P}^1$ mass matrix. Once the linear system has been solved, we can estimate the error between y (computed) and $-\boldsymbol{\nabla} \cdot \boldsymbol{g}$ (assumed to be known analitically for the test case):

$$Err = \frac{\|y + \boldsymbol{\nabla} \cdot \boldsymbol{g}\|_\infty}{\|\boldsymbol{\nabla} \cdot \boldsymbol{g}\|_\infty}. \tag{4.9}$$

The very same procedure can be extended to every component of the right-hand side of the Helmholtz equation, either $-\boldsymbol{\nabla} p$ or $\boldsymbol{g}$.

Test Case

The same functions and parameters employed in subsection 4.1.3 have been used to test the convergence in the computation of $-\boldsymbol{\nabla} \cdot \boldsymbol{g}$ and $-\boldsymbol{\nabla} p + \boldsymbol{g}$, with the following choice for p:

$$p = \cos\left(2\pi \frac{x}{A}\right) \cos\left(2\pi \frac{y}{B}\right).$$

Convergence analysis

The results for $-\boldsymbol{\nabla} \cdot \boldsymbol{g}$ and $-\boldsymbol{\nabla} p + \boldsymbol{g}$ are shown in Table (4.4a) and Table (4.4b), respectively. The same results are represented in the logarithmic plot

h	$Err[-]$	$Err\%$
0.5	0.39264	39.3
0.25	7.8853 E-002	7.88
0.125	1.9091 E-002	1.91
0.0625	4.7310 E-003	0.473
0.03125	1.1801 E-003	0.118
0.015625	2.9486 E-004	0.0295

(a) $-\boldsymbol{\nabla} \cdot \boldsymbol{g}$ convergence analysis

h	$Err[-]$	$Err\%$
0.5	3.5116 E-004	0.0351
0.25	1.0471 E-004	0.0105
0.125	3.0701 E-005	0.00307
0.0625	8.1286 E-006	0.000813
0.03125	2.0656 E-006	0.000207
0.015625	5.1918 E-007	0.0000519

(b) $-\boldsymbol{\nabla} p + \boldsymbol{g}$ convergence analysis

Table 4.4: Convergence analysis for $-\boldsymbol{\nabla} \cdot \boldsymbol{g}$ and $-\boldsymbol{\nabla} p + \boldsymbol{g}$

of Figure (4.4) and Figure (4.5). As it can be appreciated, this method provides a second order convergence for both $-\boldsymbol{\nabla} \cdot \boldsymbol{g}$ and $-\boldsymbol{\nabla} p + \boldsymbol{g}$.

4.2 Convergence check for the complete solver

4.2.1 Test case definition

The whole method has been tested on the test-case described in this section in order to investigate its accuracy and convergence properties. The following divergence free velocity vector field and pressure field are considered:

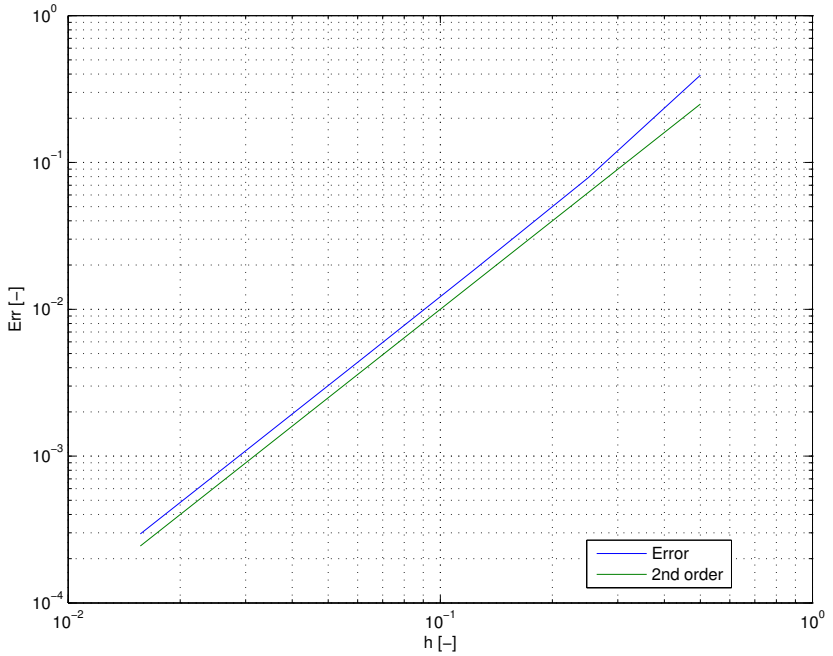


Figure 4.4: Convergence analysis for $-\boldsymbol{\nabla} \cdot \boldsymbol{g}$

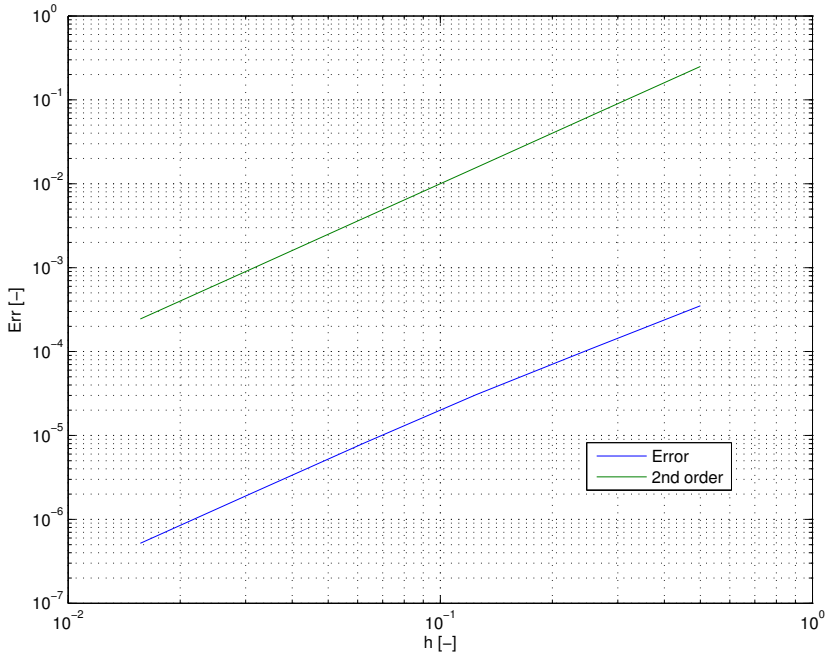


Figure 4.5: Convergence analysis for $-\boldsymbol{\nabla} p + \boldsymbol{g}$

$$
\begin{aligned}
&\boldsymbol{u}(\boldsymbol{x},t) = \tilde{\boldsymbol{u}}(\boldsymbol{x})g(t),\\
&\tilde{u}_x(\boldsymbol{x}) = -\cos x \sin y,\\
&\tilde{u}_y(\boldsymbol{x}) = \sin x \cos y,\\
&g(t) = 1 - \exp(-4t),\\
&p(\boldsymbol{x},t) = -\frac{1}{4}[\cos(2x) + cos(2y)]g^2(t),
\end{aligned}
\tag{4.10}
$$

which is the exact solution $(\boldsymbol{u},p)$ to the following unsteady, coupled problem:

Find $\boldsymbol{u}$, p such that:

$$
\begin{cases}
\dfrac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u}\cdot\boldsymbol{\nabla})\boldsymbol{u} - \nu\nabla^2\boldsymbol{u} + \boldsymbol{\nabla}p = \boldsymbol{f}(\boldsymbol{x},t) & \text{in V}\subset\mathbb{R}^2 : [-1,1]^2\\
\boldsymbol{\nabla}\cdot\boldsymbol{u} = 0 & \\
\boldsymbol{u}\big|_S = \boldsymbol{b} & \text{on S,}\\
\boldsymbol{u}(t=0) = 0, &
\end{cases}
\tag{4.11}
$$

where

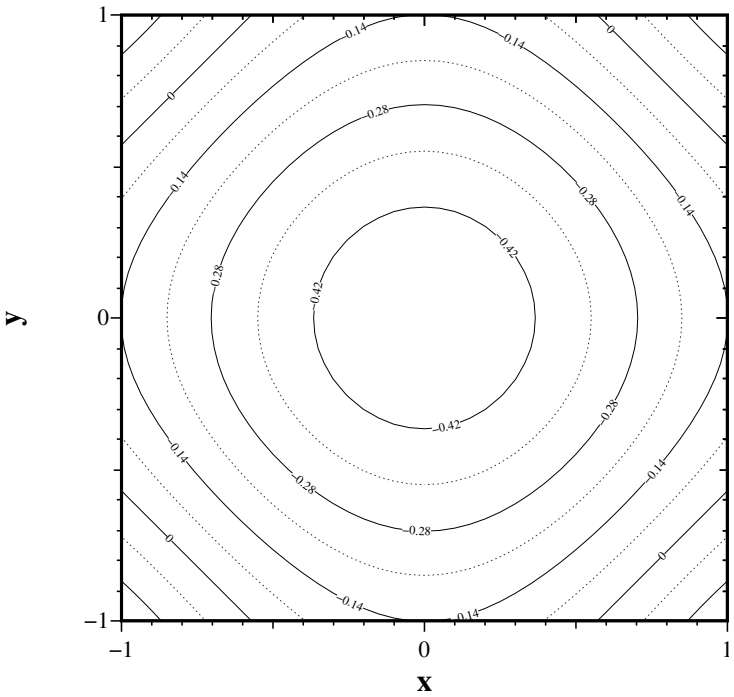
$$
\boldsymbol{f}(\boldsymbol{x},t) = \tilde{\boldsymbol{u}}(\boldsymbol{x})[g'(t) + 2\nu g(t)]. \tag{4.12}
$$

The exact pressure and velocity fields for $t = 1\ s$ are represented in figure 4.6.

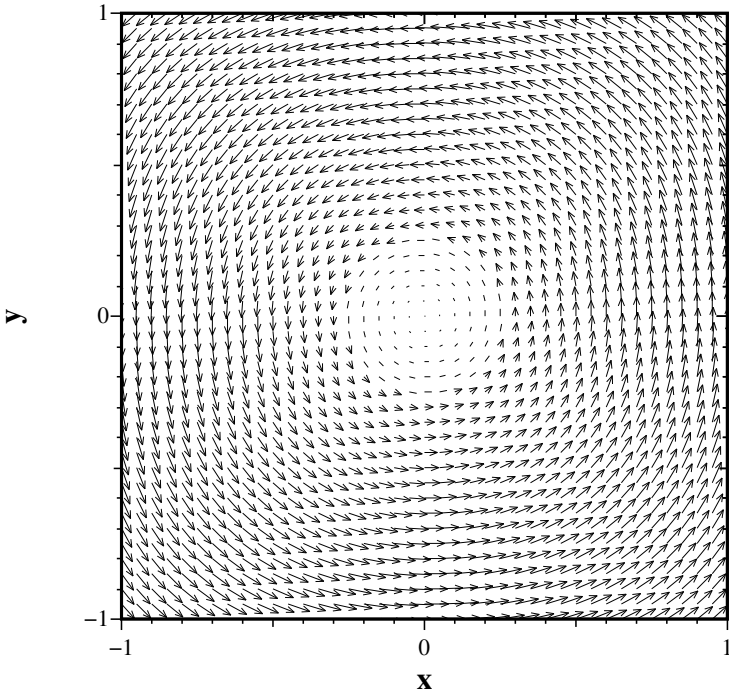
4.2.2 Convergence analysis

For the convergence analysis we consider the normalized L^∞ norm (maximum norm) of the error between the exact and the reconstructed solution, as defined in Equation 4.2 for p and equation 4.4 for $\boldsymbol{u}$. The results for $\Delta t = 10^{-3}$, $\nu = 10^{-5}$ are resumed in Table 4.5a and Table 4.5b. The same results are depicted in the logarithmic plots of Figure 4.7 and 4.8. As it can be appreciated from the figure, this method provides a second order convergence for both the reconstructed pressure and velocity in this error norm.

In table 4.6 the convergence results for pressure are presented in the case of an explicit treatment of the time derivative term. The same results are compared to those obtained with an implicit treatment of the time derivative term in the logarithmic-scale plot of Figure 4.7. As it could be expected, the second order convergence is guaranteed, but the error is higher for the explicit solver with respect to the implicit one.



(a) *Exact solution, pressure*



(b) *Exact solution, velocity*

Figure 4.6: Plot of the exact solution for the test case reported in eq. 4.11

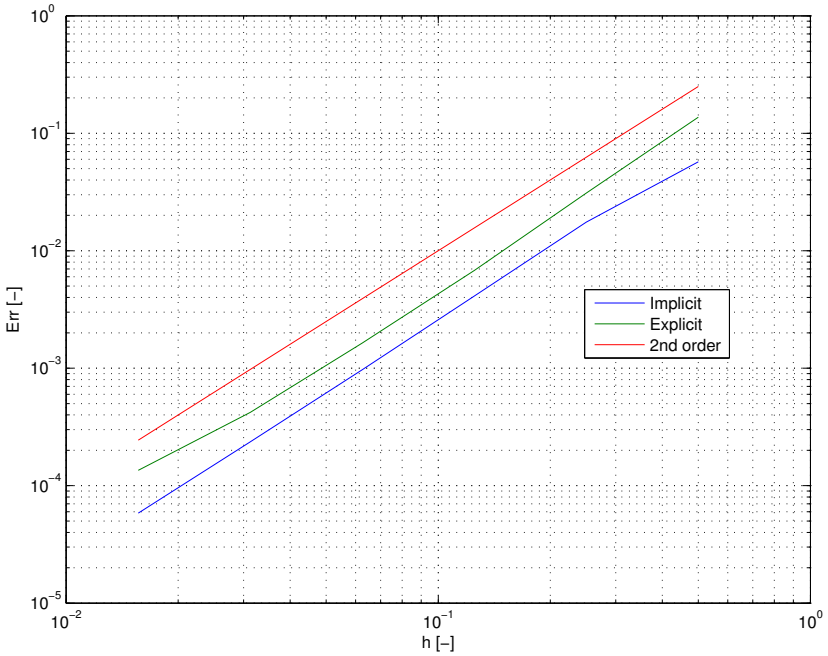


Figure 4.7: Convergence analysis for the reconstructed pressure

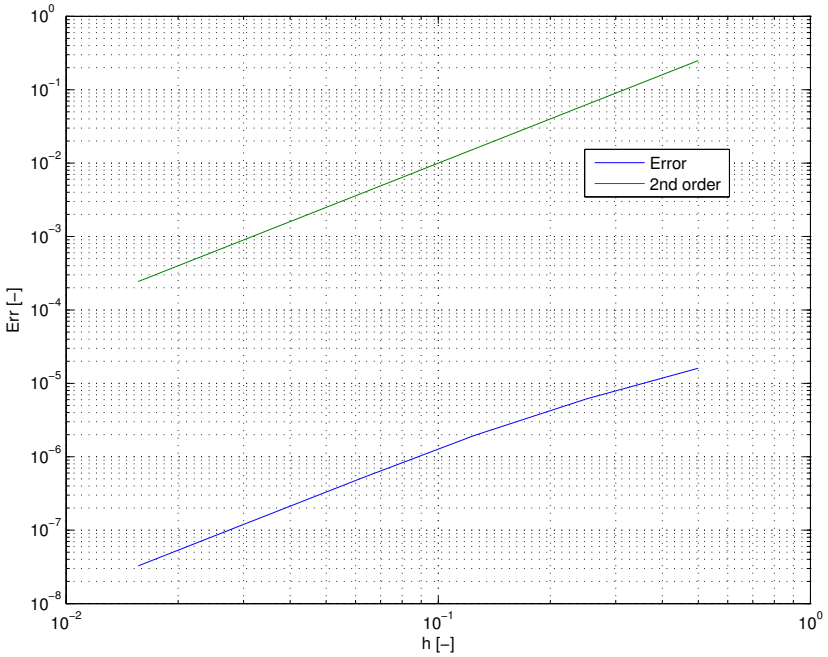


Figure 4.8: Convergence analysis for the reconstructed velocity

h	$Err[-]$	$Err\%$
0.5	5.718 E-002	5.72
0.25	1.754 E-002	1.73
0.125	4.113 E-003	0.411
0.0625	9.714 E-004	0.0971
0.03125	2.359 E-004	0.0236
0.015625	5.830 E-005	0.005 83

(a) Reconstructed pressure convergence analysis

h	$Err[-]$	$Err\%$
0.5	1.6056 E-005	0.00161
0.25	6.1180 E-006	0.000612
0.125	1.9451 E-006	0.000195
0.0625	5.1478 E-007	0.0000515
0.03125	1.3014 E-007	0.0000130
0.015625	3.2591 E-008	0.00000326

(b) Reconstructed velocity convergence analysis

Table 4.5: Convergence analysis fot the reconstructed pressure and velocity

h	$Err[-]$	$Err\%$
0.5	0.1372	13.72
0.25	3.0970 E-002	3.10
0.125	6.7891 E-003	0.678
0.0625	1.6383 E-003	0.164
0.03125	4.2114 E-004	0.0421
0.015625	1.3511 E-004	0.0135

Table 4.6: Pressure convergence analysis, explicit treatment of the time derivative

4.3 Error Analysis

4.3.1 Stochastic errors

Methodology

In order to test the robustness of the method in presence of velocity measurement errors, a Monte Carlo simulation on the Test case described in section 4.2 has been carried out with the following methodology:

1. Both the current and the previous step velocity fields are perturbed with a stochastic error. The error has a Gaussian distribution, hence we assume an ideally calibrated instrument, with null expected value and standard deviation $\sigma_{pert} = \frac{0.01}{2\sqrt{3}} \simeq 3 \cdot 10^{-3}$, and it has been generated using the Box-Muller method, for further details see [18]. The perturbation is uncorrelated between the two velocity components and in the spatial domain. The same perturbation has been applied to the two time steps, since an uncorrelated velocity perturbation in time, especially for small Δt, would be amplified too much.

2. The proposed method is applied to the perturbed velocity field, with the following choice of the program parameters: $\nu = 10^{-5}$, $\Delta t = 10^{-3}$, $h = 0.05$.

3. The procedure is repeated an adequate amount of times, each time with a different stochastic perturbation, which must satisfy the statistical properties just introduced. The computed pressure errors are stored for each realization of the Monte Carlo simulation.

4. The produced database is then post-processed in order to compute statistical properties of the pressure error.

For a deeper insight of the Monte Carlo method, see [19].

Preliminary analysis

A preliminary analysis with a reduced amount of simulations is conducted in order to check the robustness of the method with the following configurations:

- Explicit/implicit treatment of the time derivative
- With/without the projection on a divergence free space, as described in section 2.4.

Twenty simulations with different stochastic perturbations have been performed for each configuration. The average pressure error, computed in L^∞ norm and normalized by the L^∞ norm of the exact solution, is shown in Table 4.7.

-	With projection	Without projection
Implicit	1.33 E-002	2.81
Explicit	1.25 E-002	1.33 E-002

Table 4.7: Preliminary stochastic analysis: L^∞ norm of the pressure error

Let us first concentrate on the treatment of the time derivative, without the projection. As we can see, the implicit method presents an unacceptable error in the pressure computation. This is caused, as observed in section 2.3.5, by the fact that for small Δt and ν, hence great γ, the laplacian term of the Helmholtz equation for $\boldsymbol{u}$ is everywhere negligible with respect to the reaction term but for a thin layer near the boundary, leading to a singular perturbation problem. This fact generates oscillations in the velocity fields when discretized by the Bubnov–Galerkin method. If the boundary value of velocity is affected by an error, this would turn in its great amplification when the pressure is computed. The same problem is not present in the explicit method since the velocity has to satisfy a Poisson equation.

By means of the projection on a divergence free space, which can be thought as a sort of filtering of the perturbation, the pressure error gets back to acceptable values. In the case of the explicit treatment of the time derivative, the projection step does not improve significantly the accuracy.

Detailed analysis

A more accurate Monte Carlo analysis has been carried out, based on a set of 1000 realizations. The flow and program parameters are the same as for the preliminary analysis. An explicit treatment of the time derivative has been adopted since it guarantees a better robustness to velocity measurement errors. No projection has been applied since no major improvements have been observed with this procedure, while the factorization and solution of the relative coupled system would be onerous from the point of view of the computational time with such an amount of simulations.

At the end of each simulation, the local normalized pressure error, defined as follows

$$Err = \frac{p - p_{ex}}{\|p_{ex}\|_\infty} \tag{4.13}$$

is stored for every grid node. At the end of the whole Monte Carlo Simulation, the average and the standard deviation σ of Err is computed for each grid node. As we can appreciate from Figure 4.9 and 4.10, which represents the global standard deviation of the pressure error and of the perturbation versus the number of simulations considered, 1000 simulations seem to be sufficient to reach convergence up to the third decimal. The results are presented in Figure 4.11.

Let us first analyse the space distribution of the mean value of the error. As it can be appreciated from figure 4.11a the mean value of the pressure error is globally quite small, and follows the profile of the discretization error. This means that locally the mean value of the error due to the perturbation is approximately zero, which was expected since the perturbation has zero mean value too. In normal applications, where obviously the exact solution is not available, an a posteriori estimator could be used to compute an approximation of the numerical error. This would be very helpful in the estimation of the total error.

A more interesting piece of information can be deduced from the plot of the standard deviation of the error, Figure 4.11b. Here the light blue plane represents the standard deviation of the perturbation. As we can see, the standard deviation of the perturbation is amplified homogeneously on the inner nodes, with a ratio $\sigma/\sigma_{pert} \simeq 1.6$. The same phenomenon can be appreciated among the nodes belonging to the boundary, but here the ratio σ/σ_{pert} grows to approximately 2. This analysis shows that the most sensible parts for the amplification of the error deviation is the border of the domain, which is generally where we are more interested in knowing the pressure value (e.g. to compute the loads).

The spurious null value that can be observed on the boundary of the domain in Figure 4.11b can be explained by the fact that since pressure is defined up to an additive constant, one has to impose the value of the computed pressure on one node equal to the exact solution computed in the same position in order to compute the proper error, then in that point the error is always equal to zero, and then its variance is equal to zero too.

The very same Monte Carlo analysis has been repeated with different values of the perturbation standard deviation. The results are resumed in Table 4.8:

As it can be appreciated, the ratios between the error standard deviation and σ_{pert} are almost independent from the standard deviation of the perturbation, which is a typical behaviour of linear problems. This means that the non-linear advection term does not play a significant role in the error propagation.

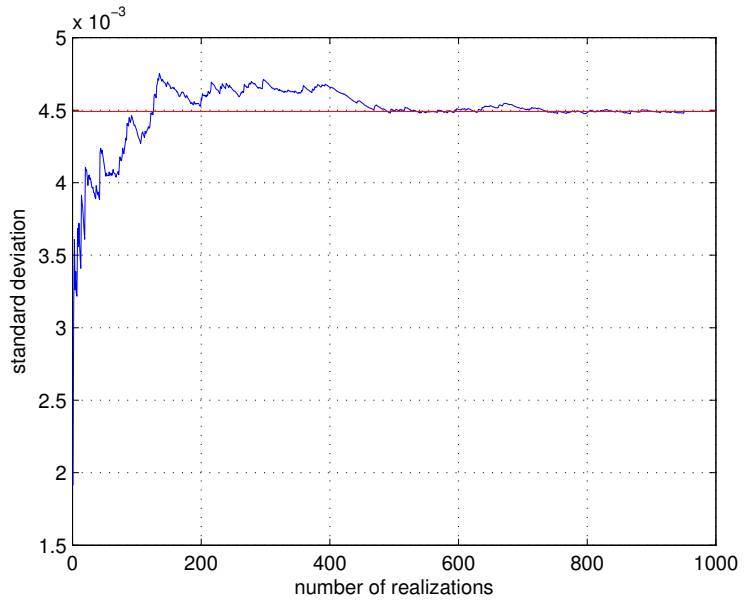


Figure 4.9: Convergence of the Monte Carlo analysis, standard deviation of the Pressure error

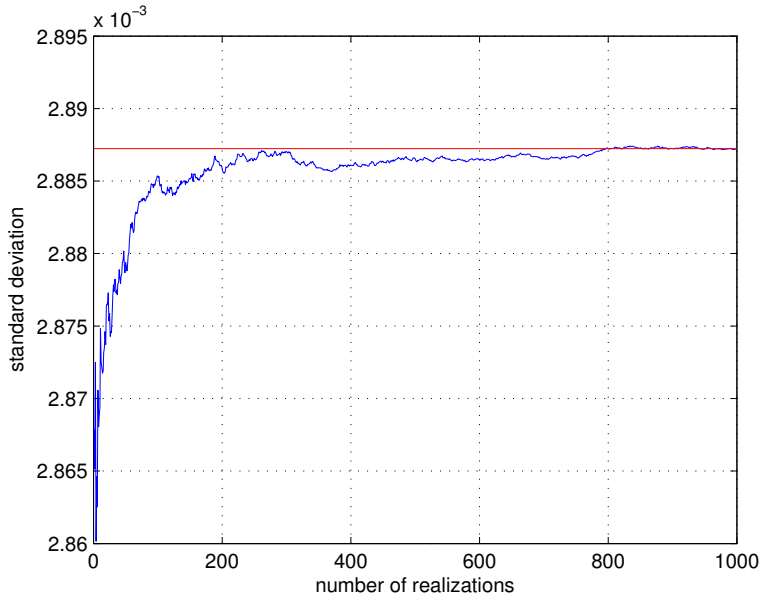
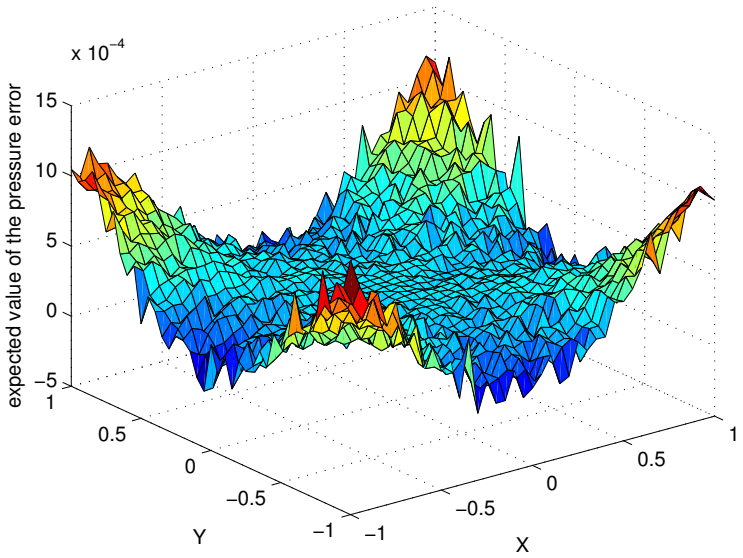
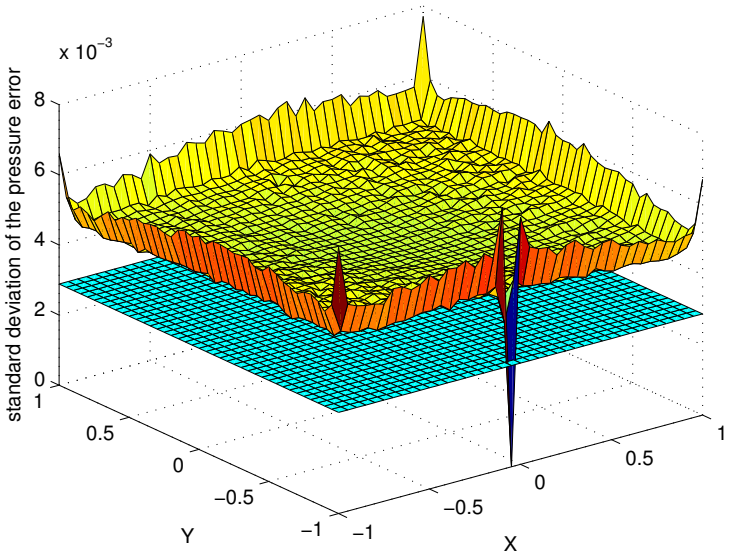


Figure 4.10: Convergence of the Monte Carlo analysis, standard deviation of the perturbation



(a) *Mean Value of the local pressure error*



(b) *Standard deviation of the local pressure error*

Figure 4.11: Expected value and standard deviation of the pressure error

σ_{pert}	$\sigma_{center}/\sigma_{pert}$	$\sigma_{border}/\sigma_{pert}$
0.01	1.54	1.98
0.03	1.6	2.02
0.001	1.54	1.98

Table 4.8: Stochastic analysis for different values of the standard deviation of the perturbation

4.3.2 Deterministic errors

The same test case has been also used to evaluate the error produced by deterministic perturbations as a function of their wavelength. The perturbation applied to both the current and the previous step velocity fields and to both velocity components, is the following:

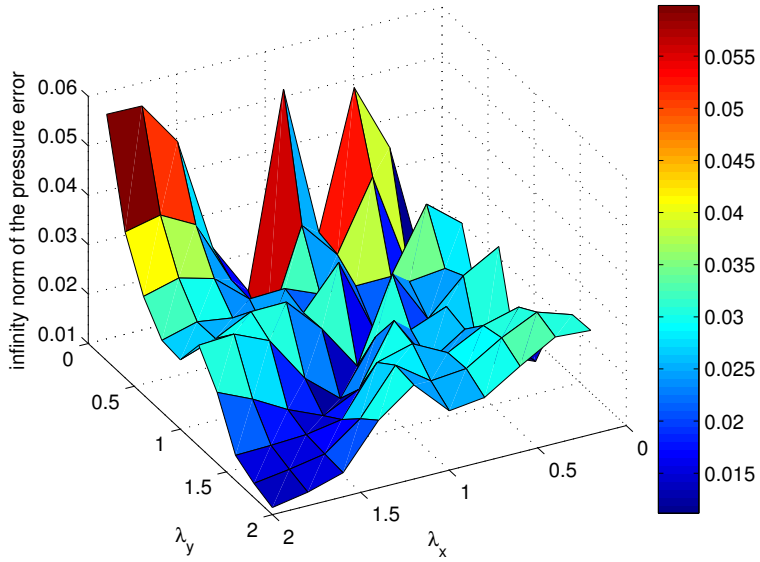
$$Pert = C \cos\left(2\pi \frac{x}{\lambda_x}\right) \cos\left(2\pi \frac{y}{\lambda_y}\right), \tag{4.14}$$

with C set to 0.01. The two parameters λ_x and λ_y, which represent the perturbation wavelength in the x and y direction, respectively, are varied from 0.2, that is 4 times the grid dimension ($h = 0.05$), to 2, which represents the side length of the square domain ($A = 2$). For each simulation with different λ_x and λ_y, the normalized infinity norm of the pressure error is stored. Results are reported in Figure 4.12.

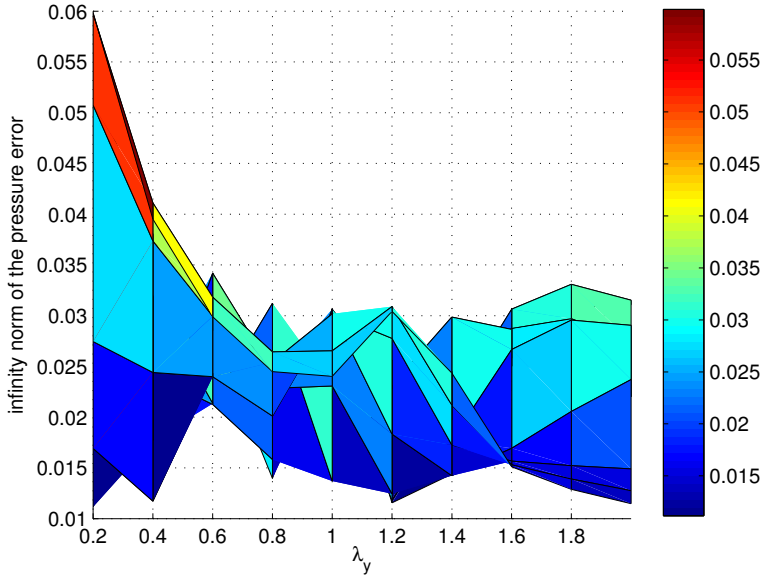
The results should be analyzed taking into account the following considerations. The error is the sum of two different contributions:

- The developed method is more sensitive to perturbations with large wavelength, since perturbations with small wavelength, are filtered and smoothed by the laplacian terms of the method.
- The numerical Finite Element discretization of the method is more sensitive to perturbations with small wavelength, since the approximation of the perturbation is less accurate as the wavelength is reduced, and oscillations can arise when its wavelength becomes small. This particular effect can be appreciated in Figure 4.12b.

The two opposite effects seem to compensate for $\lambda_x, \lambda_y > 0.8$, resulting in a region where the pressure error is almost constant.



(a) *Deterministic error analysis, view A*



(b) *Deterministic error analysis, view B*

Figure 4.12: Deterministic error analysis

Chapter 5

Test rig and experimental set-up

In this chapter the test rig and the experimental set-up necessary to obtain the velocity field database, which will be post-processed at a later stage in order to compute the pressure field, are described. In section 5.1 the test rig is presented, while in the experimental set-up to investigate the dynamic stall is described, including the apparatus for PIV and unsteady pressure measurements. Then, in section 5.3, the test matrix comprehensive of the reference test conditions is thoroughly described. In section 5.4, the modified set-up, with the Gurney flap attached on the airfoil trailing edge, is described. In this configuration our method is essential, since the presence of the moving Gurney flap and the low trailing edge thickness prevent pressure taps from being employed. Eventually, in section 5.5, a procedure to estimate the uncertainty of the pressure coefficient, obtained with the proposed method, is described.

5.1 Test rig for the oscillating airfoil

The wind Tunnel

The experimental campaign has been carried out at Politecnico di Milano, taking advantage of the low-speed closed-return wind tunnel of DAST (Department of Aerospace Science and Technology). This wind tunnel has a rectangular test section (heigth 1.5 m, width 1 m) and can achieve a maximum wind speed of 55 m/s, with a turbuence level lower than 0.1%

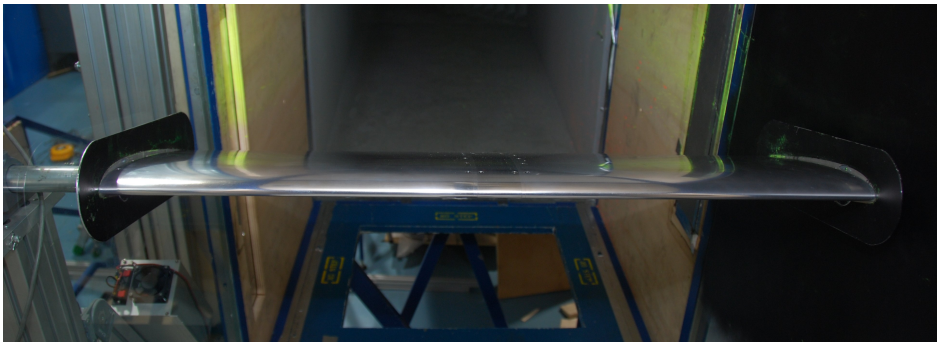


Figure 5.1: The blade section model mounted in the wind tunnel

The blade section model

The blade section model that was employed for the experimental campaign (Figure5.1) had been already employed in an experimental investigation about the dynamic stall process (for further details see [20]). The chosen airfoil is the NACA 23012 and the blade section model has a 0.3 m chord length and a 3.1 aspect ratio. The model is composed by three aluminium machined external sections (3 cm thick) connected to an internal metallic frame also made of aluminium. The internal frame is composed by four airfoil ribs connected to a steel tubular shaft, with its axis located at 25% of the chord. The model presents an interchangeable midspan section, which allows to use different central sections specifically designed for PIV surveys or unsteady pressure measurements. End plates were used during the tests to minimize interference with the boundary layer on the wall of the wind tunnel.

The supporting frame and the pitching mechanism control

The blade section model is mounted horizontally in the wind tunnel test section and is hinged about the quarter-chord axis on a tubular steel shaft positioned on self-aligning bearings. The model is installed on a heavy metallic supporting structure composed by steel beams and aluminium profiles to which the model is connected through the tubular shaft, ad depicted in Figure 5.2 and 5.3.

The oscillating airfoil motion is driven by a brushless servomotor through a gearbox drive.The sinusoidal pitching motion of the airfoil and the parameters of the pitching cycle, such as the mean pitch angle, the pitching amplitude and frequency are controlled by an interface software implemented in Labview. The motor voltage signal is produced by a proportional and derivative feedback controller during a period T that covers a number of oscillating

cycles selected by the user. The user can select the mean angular position, the peak angular velocity and the amplitude of the oscillation. In order to compute the input parameters, two encoders are mounted directly on the outer shaft: the first one is a 2048 imp/rev absolute digital encoder, used for the feedback control, whereas the second one is a 4096 imp/rev incremental analog encoder, used to determine the current model position.

Figure 5.2: The supporting frame and the pitching motor

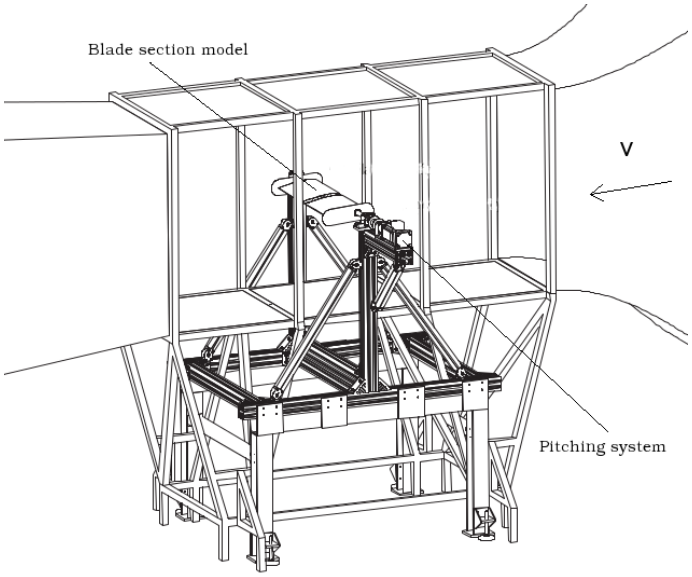


Figure 5.3: Schematic of the test rig

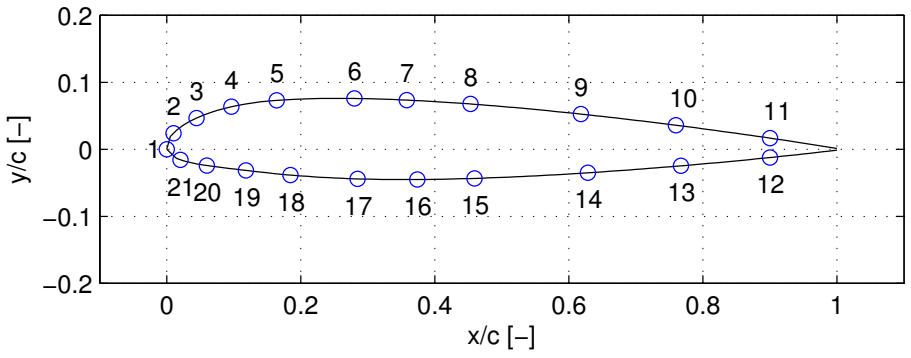
5.2 Experimental set-up for the oscillating airfoil

Unsteady pressure measurement set-up

As previously explained, the midspan section of the model can be changed depending on the kind of measurements that have to be carried out. The midspand section designed for unsteady pressure measurements is equipped with 21 Kulite XCS-093 pressure transducers, with a slight increase in concentration near the leading edge. The pressure tap positions are shown in Table 5.4a and in figure 5.4b. The pressure tap numbering starts from the leading edge and follows a closed loop from the upper to the lower surface.

#	x/c	#	x/c	#	x/c
1	0	8	0.4533	15	0.4593
2	0.01	9	0.618	16	0.3737
3	0.0443	10	0.7598	17	0.2848
4	0.0964	11	0.9	18	0.1847
5	0.1642	12	0.9	19	0.1182
6	0.28	13	0.7672	20	0.0598
7	0.3580	14	0.6282	21	0.0205

(a)



(b)

Figure 5.4: Pressure tap location on the NACA 23012 model midspan section

The signal cables pass through the internal frame of the model and come out from the external tubular shaft opposite to the motor, as depicted in Figure 5.5; this solution is interesting to reduce the stress on the cables dur-

ing the oscillation cycles. The pressure signals have been acquired using a National Instruments compact data acquisition system cDAQ-9172 equipped with six NI 9237, 24 bit sampling bridge modules. The transducer signals have been acquired with a 50 kHz simultaneous sampling rate on 21 channels for a time period corresponding to 30 complete pitching cycles. The high sampling rate is needed to capture the fine details of the dynamic stall phenomenon characterised by severe unsteadiness, especially for the tests with high reduced frequency ($k = 0.1$). The model angular position has been measured by the incremental encoder simultaneously to the pressure transducer signals.The simultaneous acquisition of the encoder signals has been used to evaluate of the phase averages.

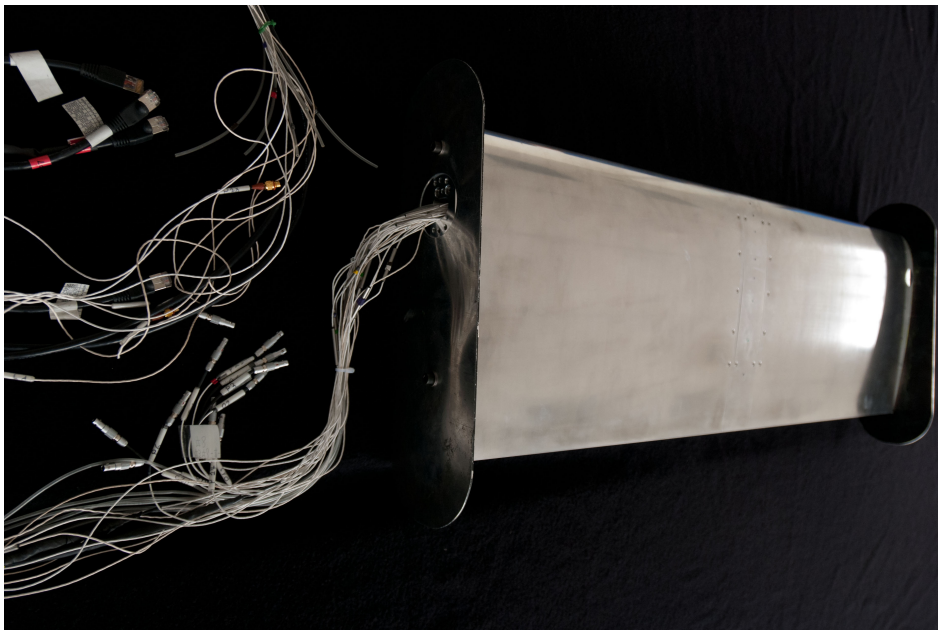


Figure 5.5: Wing section model with pressure transducers cables

PIV velocity measurement set-up

The PIV set-up consists of the following equipment :

- A Pixelfly PCO double shutter CCD camera with a 12 bit, 1360×1024 pixel array with a 55 mm Nikon lens, used to acquire the image pairs. The CCD camera has been mounted on a dual axis traversing system guided by two stepper motors that allowed to move the measurement window along two orthogonal directions.

- A Dantec Dynamics Nd:Yag double pulsed laser with 200 mJ output energy and a wavelength of 532 nm.The laser sheet passed through an opening in the wind tunnel roof aligned with the flow and positioned at the midspan of the test section width. Both the camera and the laser are connected to the heavy structure of the rig in order to avoid the transfer of wind tunnel vibrations, especially at high speed.

- A particle generator (PIVpart30 by PIVTEC) with Laskin atomizer nozzles, which injects small oil droplets (diameter $1 - 2$ μm) just after the fan section of the wind tunnel.

- A synchronizing system, whose main purpose is to control the synchronizarion of the two laser pulses with the exposure of the image pairs by the camera. The timing is controlled by an interface software coded in MATLAB, which sends the trigger signal to both camera and laser through a 6 channel Quantum Composer 9618 pulse generator. The inputs of the control interface are the angular position of the airfoil where one wants the image pairs to be taken, the camera exposure time and the time delay between the two laser pulses.

Before the PIV measurements are carried out, a calibration of the measuring window is needed in order to determine the image magnification factor ($m/pixel$). An equispaced grid located at the midspan section has been used for this purpose. The image post-processing has been carried out by means of the PIVviev 2C/3C software, which implements a multi-grid algorithm, starting from 96 $\times$ 96 pixel interrogation windows up to 32 $\times$ 32, with an overlap factor of 50 %. For further details, see [21].

5.3 Oscillating airfoil database

In the experimental campaign several different configurations have been considered. All the experiments have been carried out with the following flow parameters:

- Wind speed: approximately 30 m/s

- Reynolds number: approximately $6 \cdot 10^5$

- Mach number: approximately 0.09

5.3.1 Steady survey

At first, a steady survey has been conducted by clamping the model shaft, at two different angles of attack:

- 9° : fully attached flow
- 18° : fully separated flow

For every flow condition, two different PIV measurement windows (152 mm $\times$ 86 mm) have been considered: the first one the upper surface of the leading edge (fin1) and the second one on the upper surface of the trailing edge (fin2). The two measurement windows are represented in Figure 5.6:

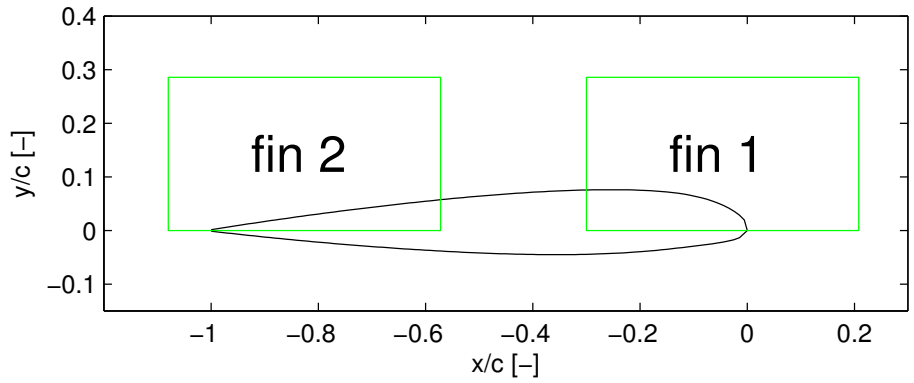


Figure 5.6: PIV measurement windows

For each configuration and measurement window, 10000 velocity fields have been sampled, that have been post-processed in order to compute the average field and the Reynolds stress tensor.

5.3.2 Unsteady survey

To test the proposed method on a challenging unsteady flow, a moving, pitching airfoil in dynamic stall conditions has been considered. The prescribed time history of the angle of attack for the oscillating airfoil is the following :

$$\alpha(t) = \alpha_0 + A\sin(\Omega t). \tag{5.1}$$

All the experiments are carried out with a reduced frequency $k = 0.1$, which, considering $U_\infty = 30\ m/s$ and $c = 0.3\ m$, corresponds to an oscillation frequency of approximately 3.2 Hz ($\Omega = 20\ rad/s$). Three pitching conditions are considered:

- $\alpha_0 = 5°, A = 5°$ (light stall); $\alpha = 10°$ upstroke. Fully attached flow.
- $\alpha_0 = 5°, A = 10°$ (deep stall); $\alpha = 19°$ upstroke. Flow still attached on the leading edge, it starts to separate on the trailing edge.
- $\alpha_0 = 5°, A = 10°$ (deep stall); $\alpha = 10°$ downstroke. Flow starts to reattach on the leading edge, completely separated on the trailing edge.

For every configuration 1000 image pairs are obtained and then postprocessed. The considered measurement windows are the same as for the steady test case.

Since we are interested in an unsteady computation of the pressure field, we have to acquire, for every configuration, a set of velocity fields at different timesteps before and after the considered instant, depending on which time integration scheme has been adopted. For example, considering the implicit Euler scheme described in section 2.1.2, one must acquire also:

- For the light stall, $\alpha = 10°$ upstroke case: the velocity field corresponding to $\alpha = 9.5°$
- For the deep stall, $\alpha = 19°$ upstroke case: the velocity field corresponding to $\alpha = 18.5°$
- For the deep stall, $\alpha = 10°$ downstroke case: the velocity field corresponding to $\alpha = 10.5°$

5.4 Experimental set-up and database for the oscillating airfoil with Gurney flap

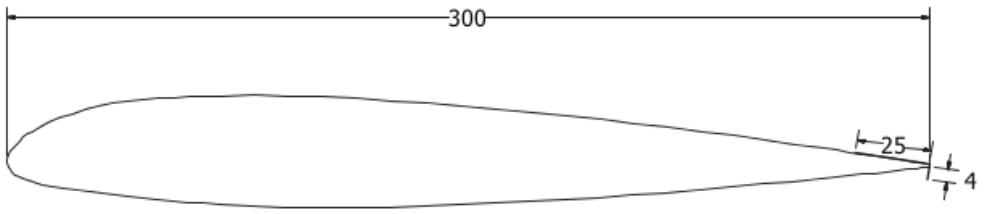
5.4.1 The Gurney flap

PIV measurements have also been carried out on a more challenging test case. In this second application a Gurney flap, which is a deployable L-shaped tab, has been attached to the trailing edge of the just described blade section model. The interest in this particular configuration lies in the possibility of an active control of the Gurney flap in order to alleviate the detrimental effects of the dynamic stall as well as in improving the retreating blade performance. As shown in [22], great performance enhancements can be obtained with the use of an active Gurney flap deployed on the retreating side of the helicopter rotor disk and retracted on the advancing side. The problem is challenging since direct pressure measurements on the Gurney Flap cannot be obtained due to the low thickness of the trailing edge and to the fact that generally

the Gurney flap is very thin and deployable. A non intrusive method for measuring pressure, like the one proposed in this thesis, is therefore crucial.

5.4.2 Experimental set-up and database

The L-shaped tab was manufactured using two carbon fiber skins. The tab is 0.5 mm thick and spans the entire blade section model with a 25 mm chord. Two main configurations are possible : deployed Gurney flap (Figure 5.7a) and retracted Gurney flap (Figure 5.7b). When deployed the tab is laid on the airfoil upper surface so that the end tab behaves as a Gurney flap in correspondence of the trailing edge. In this configuration the end of the tab protrudes 4 mm from the trailing edge corresponding to 1.3% of the airfoil chord. When retracted, the L-shaped tab forms an angle of 9.1°. with the airfoil upper surface since the tab tip touches the trailing edge. An intermediate configuration where the L-tab is deployed half-way could be investigated.



(a) Deployed Gurney flap

(b) Retracted Gurney flap

Figure 5.7: Gurney flap configurations

The set-up of the experiment is the same as for the previous case (Section 5.1), but for a IMPERX camera with a 1952 × 1112 pixel resolution, which has been substituted for the PCO Pixelfly camera. The 100 × 52 mm measurement window, centered on the trailing edge of the airfoil is shown in Figure 5.8.

Among all the possibilities, the following dynamic stall configurations have been selected:

- $\alpha_0 = 5°, A = 5°$ (ligth stall); $\alpha = 10°$ upstroke, extracted Gurney flap
- steady, $\alpha = 18°$, extracted/retracted Gurney flap

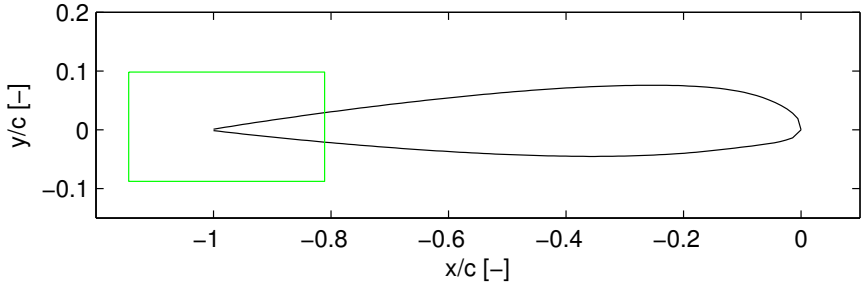


Figure 5.8: PIV measurement window for the Gurney flap experiment

5.5 Estimate of the error on the pressure coefficient

In this section a procedure aimed at estimating the error on the pressure coefficient, which will be later used for comparison with the experimental data, is described. Let us recall the definition of the pressure coefficient:

$$c_P = \frac{P - P_\infty}{\frac{1}{2}\rho U_\infty^2}. \tag{5.2}$$

As already stated in the previous sections, the pressure computed with the proposed method is defined up to an additive constant due to the full Dirichlet boundary condition employed for the velocity field. In particular, using the spectral decomposition method for the influence problem described in section 3.3.3, the method provides a pressure field with null average on the boundary. The arbitrary on the additive constant will be computed by using a least square algorithm which minimizes the error with respect to the experimental pressure data. For these reasons, for the moment being we will neglect how the c_P depends on P_∞, which will turn out only as an offset of the computed value of the c_P. The uncertainty of the c_P is provided by the following error propagation rule :

$$\sigma_{c_P} = \sqrt{\left(\frac{\partial c_P}{\partial P}\sigma_P\right)^2 + \left(\frac{\partial c_P}{\partial \rho}\sigma_\rho\right)^2 + \left(\frac{\partial c_P}{\partial U_\infty}\sigma_{U_\infty}\right)^2}, \tag{5.3}$$

where σ_P, σ_ρ and σ_{U_∞} are the uncertainties on the pressure, density and asymptotic velocity data. The sensitivity derivatives are the following:

$$\frac{\partial c_P}{\partial P} = \frac{2}{\rho U_\infty^2}, \qquad \frac{\partial c_P}{\partial \rho} = -\frac{2P}{\rho^2 U_\infty^2}, \qquad \frac{\partial c_P}{\partial U_\infty} = -\frac{4P}{\rho U_\infty^3} \tag{5.4}$$

While the uncertainty of the asymptotic velocity is available from the instrumentation datasheet and the uncertainty on density can be computed by its error propagation law, the uncertainty of pressure is not available, since the pressure value is an output of the program. As shown in section 4.3.1, the ratio between the adimensional standard deviation of the pressure error and the adimensional standard deviation of the velocity perturbation is almost constant on the boundary (approximately equal to 2) and not dependent on the amplitude of the perturbation. Hence we can use this relation as an approximation of the pressure uncertainty:

$$\sigma_P = 2\frac{\|p\|_\infty}{\|u\|_\infty}\sigma_u. \tag{5.5}$$

Again, the uncertainty on the velocity data is not known. It can nonetheless be approximated considering the PIV fundamental equation 1.1. If the expected displacement by means of a sub-pixel interpolation error is 0.1 pixel, we then have:

$$\sigma_u = 0.1\frac{C}{\Delta t}, \tag{5.6}$$

where Δt is the time delay between two successive frames and C is the meters/pixel ratio of the image.

An additional contribution to the error comes from the numerical error due to the FEM approximation of the partial differential equations. This error could be estimated by a-posteriori error estimate technique (see for example [23], [24]), but this aspect was beyond the scope of the present work.

Chapter 6

Results

6.1 Comparison with reference experimental results

6.1.1 Jagged grid

In this section, we present the results obtained by using the method introduced in §2 and 3 to reconstruct the pressure field around the NACA 23012 airfoil from PIV measurements of the related velocity field. Herein the method is applied without any modification of the jagged grid inherited from the PIV data postprocessing, i.e. the grid which is obtained by simply dropping the points which lie outside of the flow field. As described in §5, in the experimental setup the velocity field is sampled at the middle span of the employed wing section model with the light plane perpendicular to the wing axis. Two-dimensional velocity fields are collected within the two spatial windows fin1 and fin2, located close to the leading edge and the trailing edge, respectively, see Fig. 5.6. Both the flow past the airfoil at fixed angle of attack (*steady test case*) and the one developing around the oscillating airfoil at the reduced frequency $k = 0.1$ (*unsteady test case*) are considered. More precisely, the reported results are referred to the following test cases:

1. steady test case with $\alpha = 9°$;
2. steady test case with $\alpha = 18°$;
3. unsteady test case with $\alpha_0 = 10°$, $A = 10°$ and $\alpha = 19°$ upstroke;
4. unsteady test case with $\alpha_0 = 10°$, $A = 10°$ and $\alpha = 10°$ downstroke;
5. unsteady test case with $\alpha_0 = 5°$, $A = 10°$ and $\alpha = 10°$ upstroke.

Dealing with the Reynolds averaged Navier-Stokes equations, both the mean field and the Reynolds stress tensor components have been computed by phase–averaging over 1000 samples of the velocity field for each selected phase. This high number of samples allows to smooth out the high frequency errors and therefore to reduce the error associated with the time derivative term in the Navier–Stokes equations, which otherwise could be quite severe. In order to compare the obtained numerical results with the available measures gathered from Kulite transducers on the airfoil surface, the arbitrary reference level of the computed pressure field has been fixed usign a least squares regression with respect to the experimental data. Then, for each test case, the resulting distribution of the pressure coefficient c_P is compared with the experimental data, as reported in Fig. 6.1–6.5. The reconstructed pressure fields at the leading and trailing edges are illustrated in Fig. 6.6–6.10 and 6.11–6.15, respectively. In addition, the modulus of the velocity field at the leading edge is also depicted in Fig. 6.16–6.20.

The pressure and c_P uncertainties due to the measurement errors can be estimated as described in section 5.5. A 0.5% accuracy for the free stream velocity and a 1% accuracy for density are assumed. The results for the test cases considered are represented in Table 6.1:

Case	$\sigma_P(Pa)$	$\sigma_{c_P}(-)$
Case 1	29.72	0.0588
Case 2	22.34	0.0431
Case 3	51.60	0.1065
Case 4	26.14	0.0511
Case 5	28.18	0.0555

(a) First measurement window (Leading edge)

Case	$\sigma_P(Pa)$	$\sigma_{c_P}(-)$
Case 1	9.56	0.0183
Case 2	22.47	0.0433
Case 3	6.48	0.0124
Case 4	8.70	0.0165
Case 5	5.87	0.0113

(b) Second measurement window (Trailing edge)

Table 6.1: Estimated error for pressure and c_P

The just reported c_P uncertainties are represented also on the comparison plots in Figure 6.1–6.5.

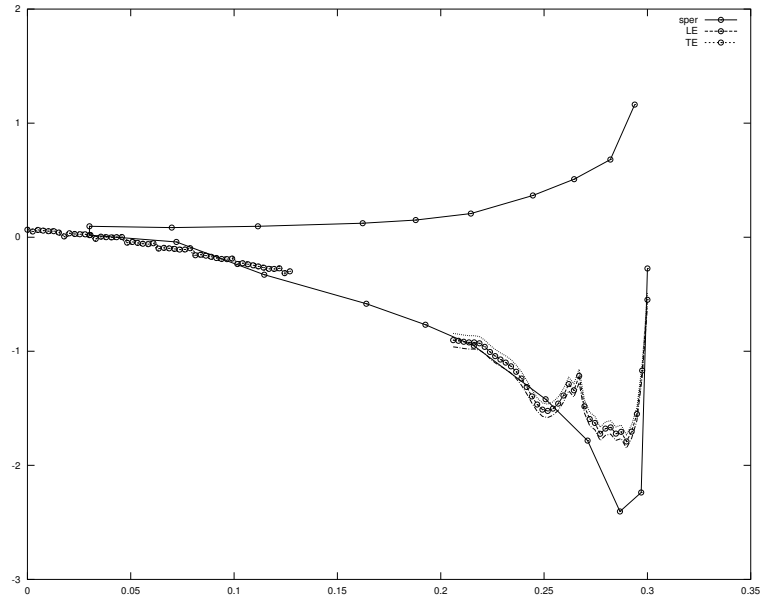


Figure 6.1: Steady test case with $\alpha = 9°$ (n. 1): c_P distribution.

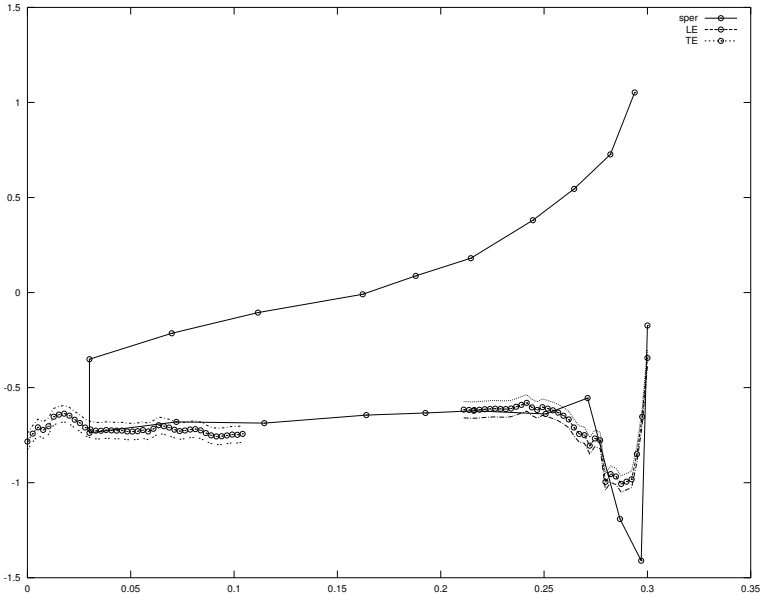


Figure 6.2: Steady test case with $\alpha = 18°$ (n. 2): c_P distribution.

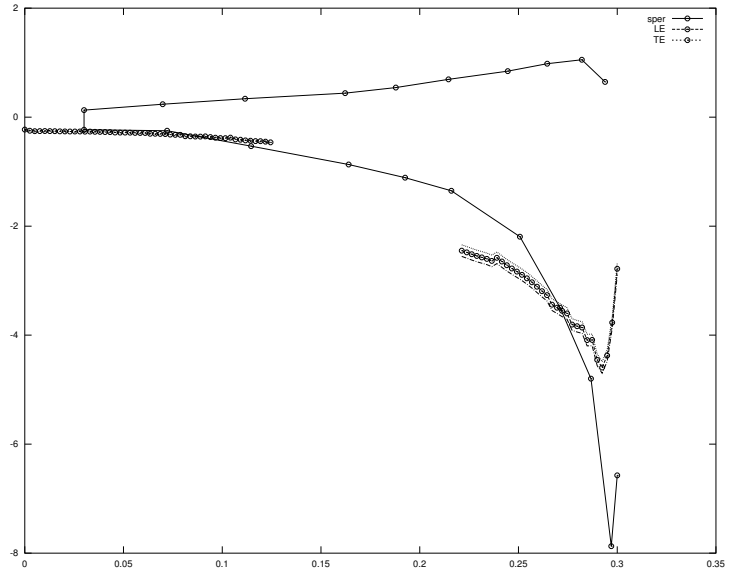


Figure 6.3: Unsteady test case with $\alpha_0 = 10°$, $A = 10°$ and $\alpha = 19°$ upstroke (n. 3): c_P distribution.

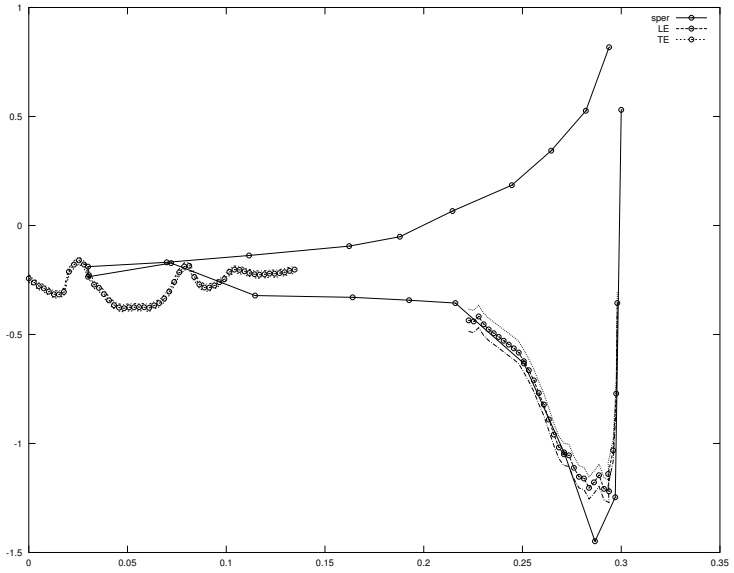


Figure 6.4: Unsteady test case with $\alpha_0 = 10°$, $A = 10°$ and $\alpha = 10°$ downstroke (n. 4): c_P distribution.

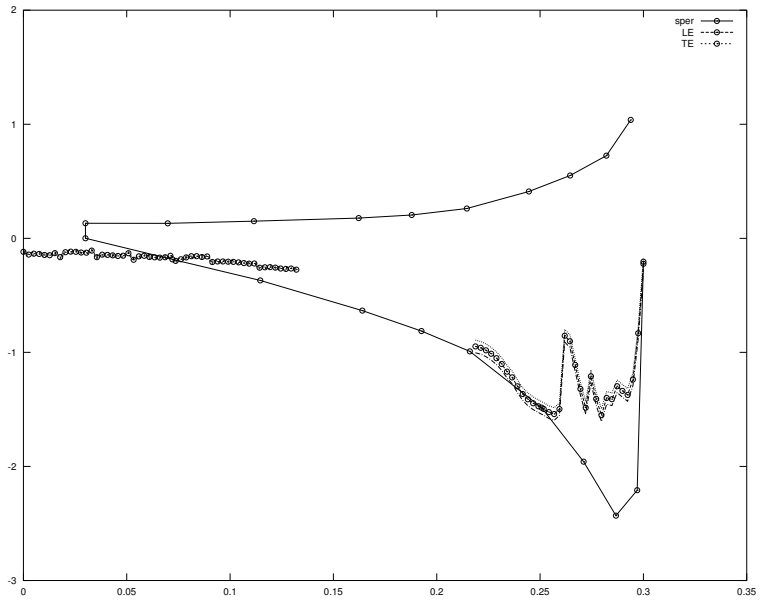


Figure 6.5: Unsteady test case with $\alpha_0 = 5°$, $A = 10°$ and $\alpha = 10°$ upstroke (n. 5): c_P distribution.

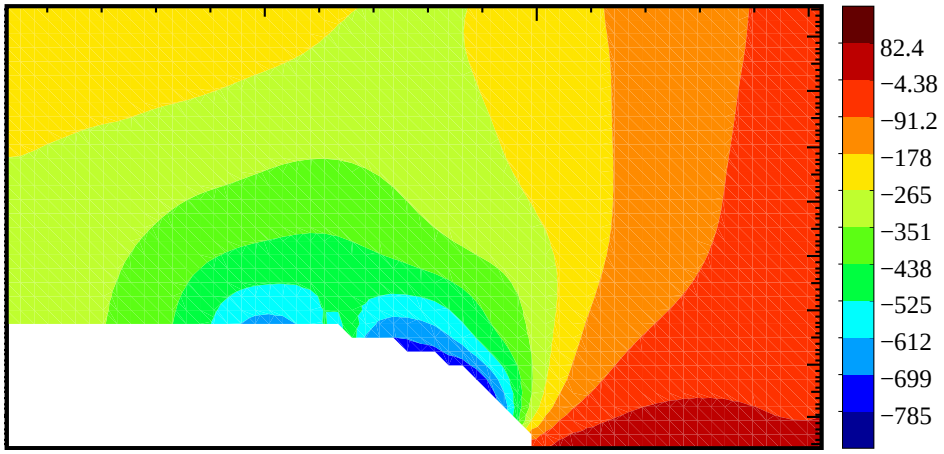


Figure 6.6: Steady test case with $\alpha = 9°$ (n. 1): pressure field at the leading edge.

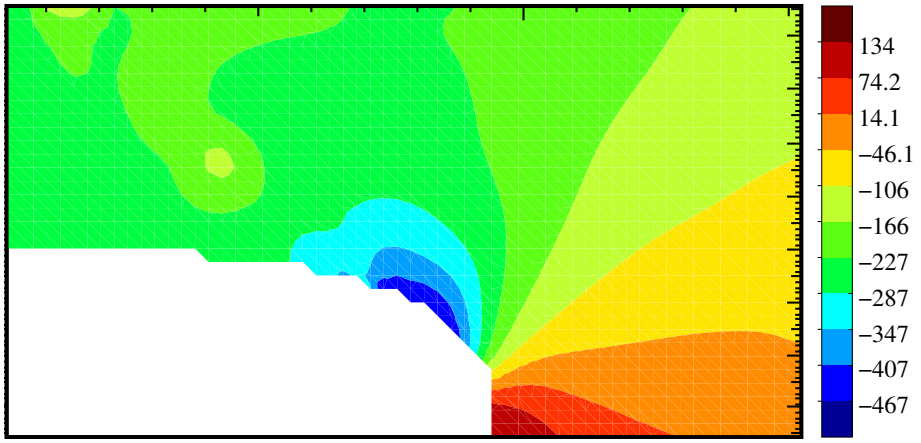


Figure 6.7: Steady test case with $\alpha = 18°$ (n. 2): pressure field at the leading edge.

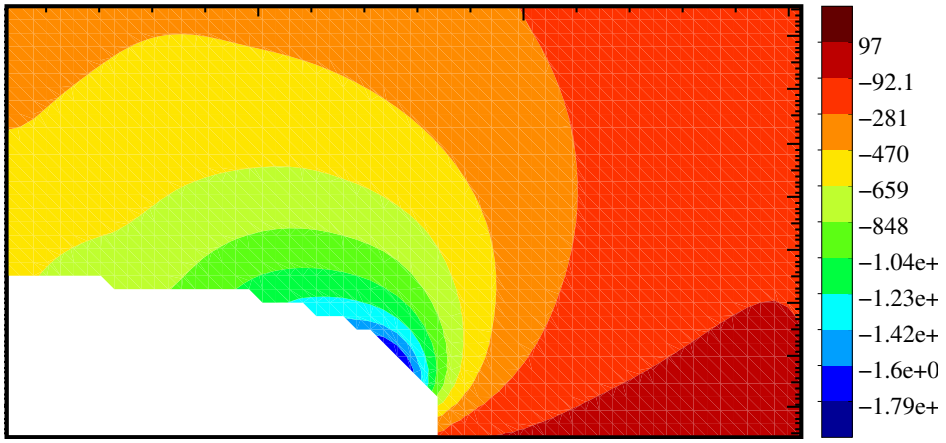


Figure 6.8: Unsteady test case with $\alpha_0 = 10°$, $A = 10°$ and $\alpha = 19°$ upstroke (n. 3): pressure field at the leading edge.

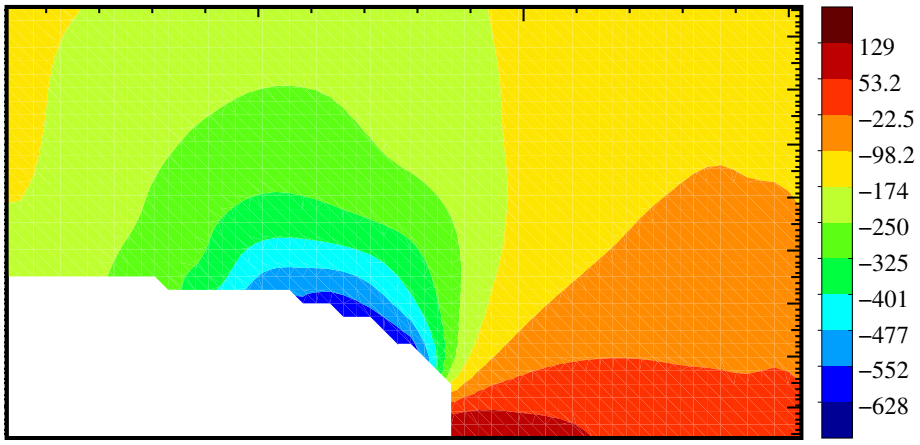


Figure 6.9: Unsteady test case with $\alpha_0 = 10°$, $A = 10°$ and $\alpha = 10°$ downstroke (n. 4): pressure field at the leading edge.

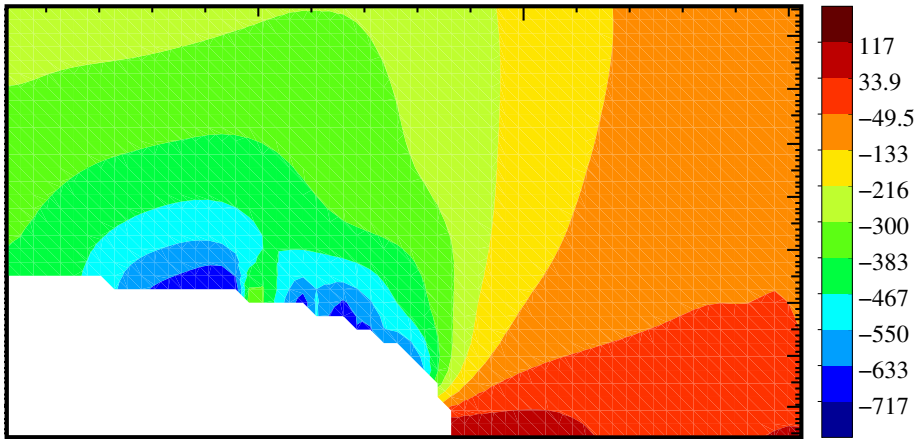


Figure 6.10: Unsteady test case with $\alpha_0 = 5°$, $A = 10°$ and $\alpha = 10°$ upstroke (n. 5): pressure field at the leading edge.

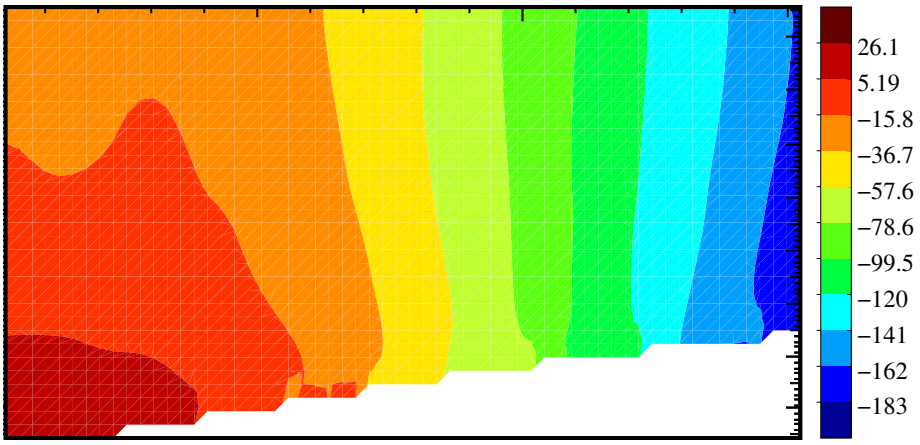


Figure 6.11: Steady test case with $\alpha = 9°$ (n. 1): pressure field at the trailing edge.

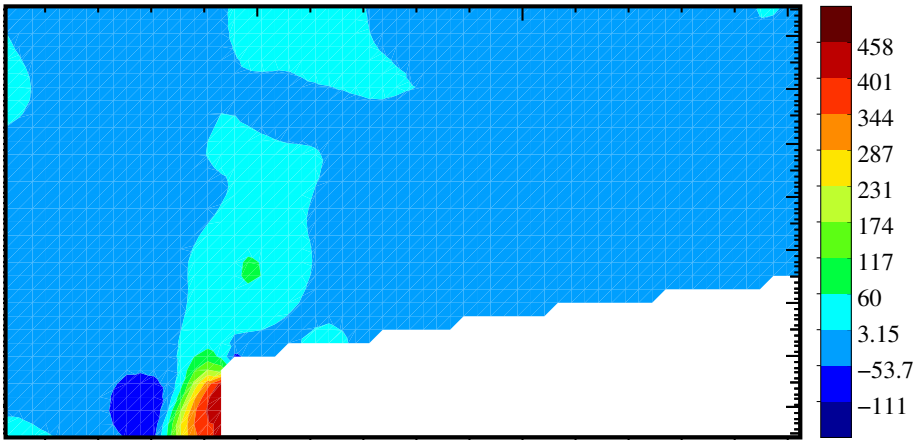


Figure 6.12: Steady test case with $\alpha = 18°$ (n. 2): pressure field at the trailing edge.

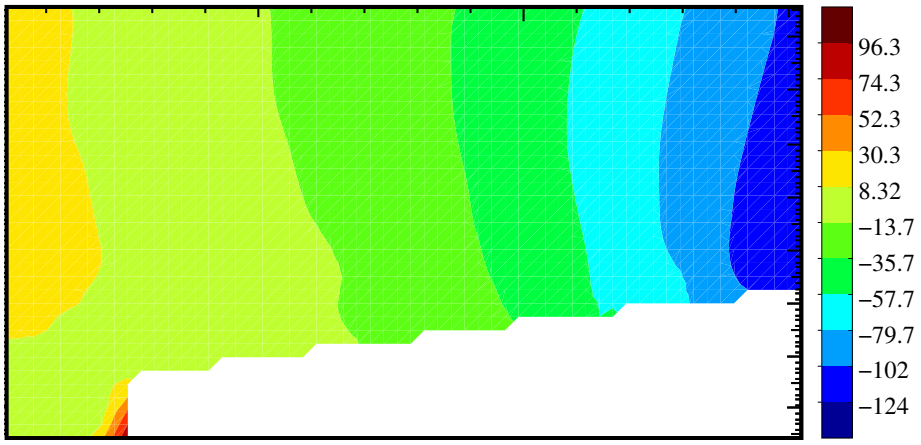


Figure 6.13: Unsteady test case with $\alpha_0 = 10°$, $A = 10°$ and $\alpha = 19°$ upstroke (n. 3): pressure field at the trailing edge.

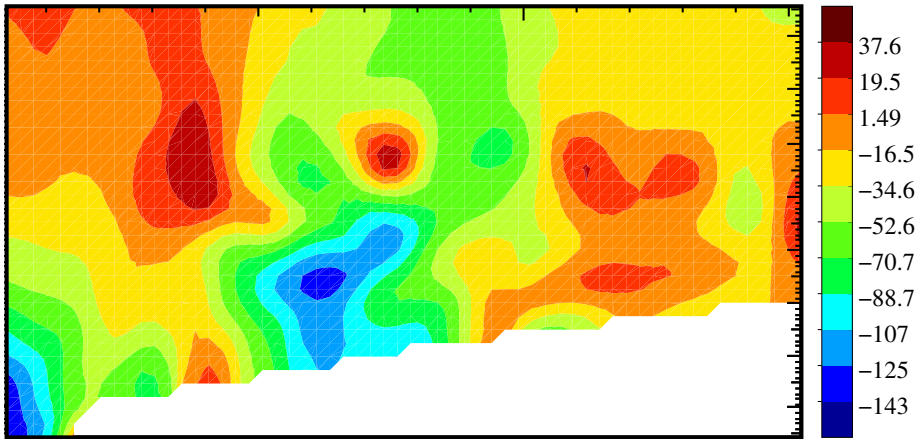


Figure 6.14: Unsteady test case with $\alpha_0 = 10°$, $A = 10°$ and $\alpha = 10°$ downstroke (n. 4): pressure field at the trailing edge.

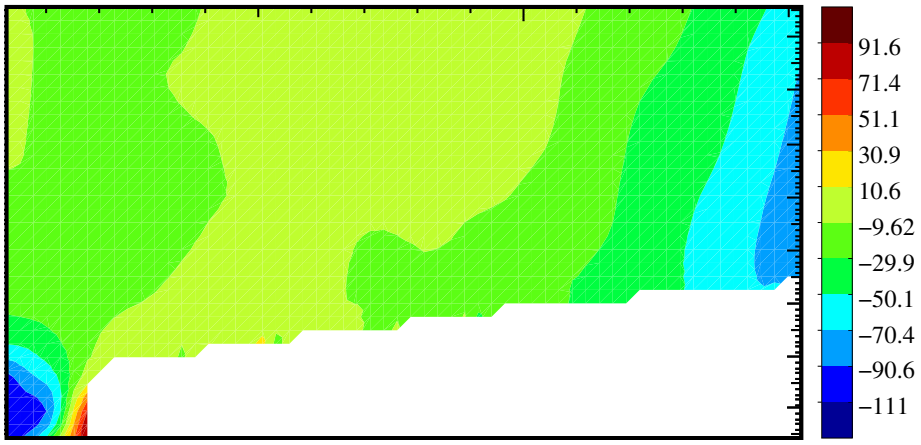


Figure 6.15: Unsteady test case with $\alpha_0 = 5°$, $A = 10°$ and $\alpha = 10°$ upstroke (n. 5): pressure field at the trailing edge.

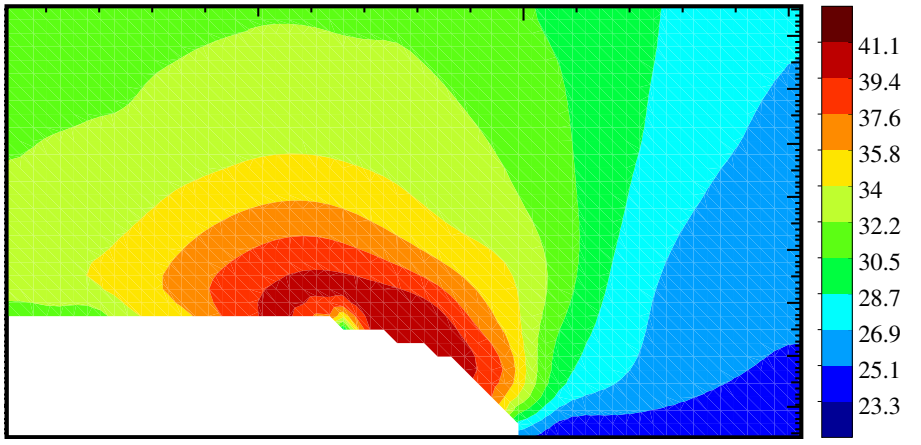


Figure 6.16: Steady test case with $\alpha = 9°$ (n. 1): modulus of the velocity field at the leading edge.

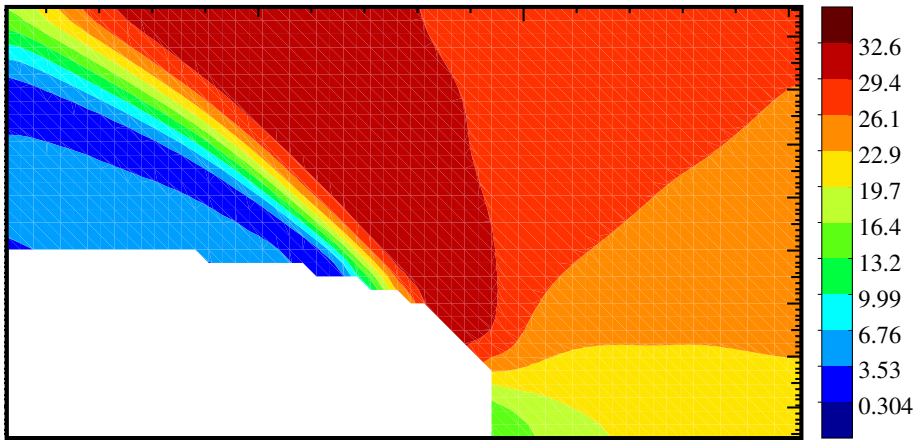


Figure 6.17: Steady test case with $\alpha = 9°$ (n. 2): modulus of the velocity field at the leading edge.

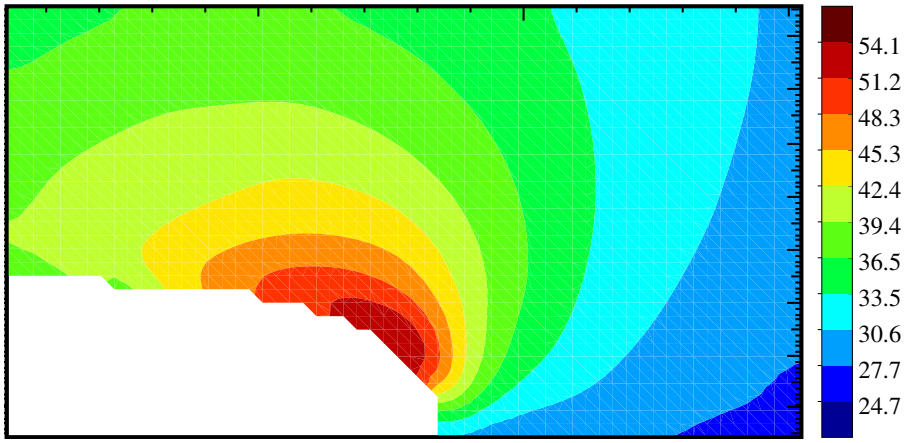


Figure 6.18: Unsteady test case with $\alpha_0 = 10°$, $A = 10°$ and $\alpha = 19°$ upstroke (n. 3): modulus of the velocity field at the leading edge.

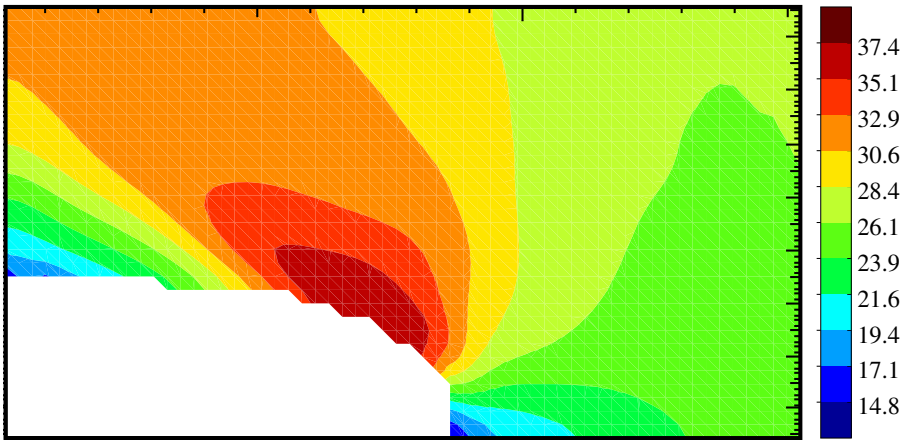


Figure 6.19: Unsteady test case with $\alpha_0 = 10°$, $A = 10°$ and $\alpha = 10°$ downstroke (n. 4): modulus of the velocity field at the leading edge.

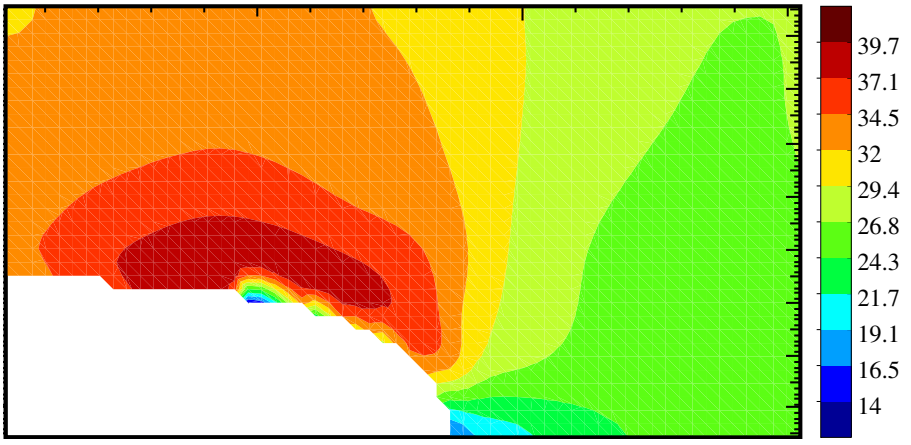


Figure 6.20: Unsteady test case with $\alpha_0 = 5°$, $A = 10°$ and $\alpha = 10°$ upstroke (n. 5): modulus of the velocity field at the leading edge.

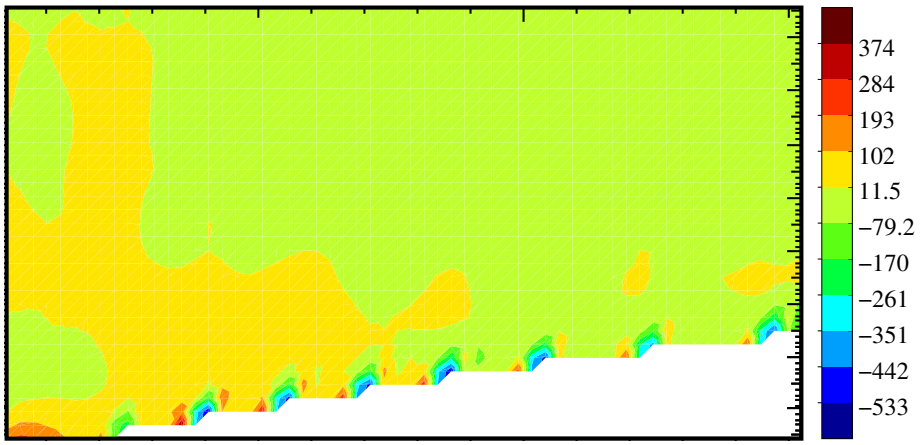


Figure 6.21: Steady test case with $\alpha = 9°$ (n. 1): divergence of the velocity field.

With reference to Fig. 6.1–6.5, fairly good agreement is obtained between experimental data and numerical results in most of the test cases, thus providing an assessment of the effectiveness of our technique. Nevertheless, in some cases, such as for instance the one reported in Fig. 6.3, the computed pressure value is considerably different from the corresponding measurement. These discrepancies can be ascribed to four main sources of error, namely:

1. the reduced reliability of the PIV measures close to the airfoil surface especially in presence of light scattering. Light scattering was an issue especially on the leading edge of the airfoil, and this is certainly a reason for the poor agreement between the present results and the reference ones near the velocity peak at the leading edge ;

2. the violation of the incompressibility constraint for the considered two-dimensional velocity field;

3. the inadequate spatial resolution of strong velocity and pressure gradients close to the wall at the leading edge;

4. the distance between the points where the pressure is computed and the real position of the airfoil surface.

The first source of errors leads to unphysical velocity measurements close to the wall. Correspondigly, also the computed pressure field results affected by the same unphysical behaviour. This is well illustrated in Fig. 6.16 for the test case 1, where an unphysical local minimum can be observed in the region of maximum velocity near to the leading edge. This local minimum is reflected in a local pressure maximum in the same region of the flow field,

as visible in Fig. 6.6, and in the surface pressure distribution, Fig. 6.1. The error on the velocity field measurement could have a severe impact on the computed pressure field, and could justify why the peak of the pressure coefficient near the nose of the airfoil is not captured correctly. The same arguments apply also for test case 5, as shown in Fig. 6.20, 6.5 and 6.10.

For what concerns the second source of errors, the effects due to the violation of the incompressibility constraint are mainly localized close to the trailing edge where spurious oscillations of the c_P are observed, Fig. 6.1. This is confirmed by inspecting the divergence of the inherent velocity field which is illustrated in Fig. 6.21: strong deviations from zero are only present close to the wall. Furthermore it has been checked that this behaviour is not induced by the corners of the jagged boundary since it has been observed also in presence when the boundary is smooth.

The third source of errors is related to the fact that strong velocity and pressure gradients which are present close to the wall at the leading edge, cannot be adequately resolved on the given Cartesian grid with uniform mesh size. This can produce detrimental effects on the numerical results, since we must recall that the final error is the sum of the numerical error and of the propagation of the error on the velocity measurement. In particular, as will be shown in the following (see Fig. 6.22, 6.26), the distance between the jagged and the true airfoil boundary can be of the order of the mesh size. It could be useful in this respect, to exploit an a-posteriori error estimator, such as those usually employed for the adaptive mesh refinement of the grid in finite element calculations [23, 24], to quantify the numerical error. This could be useful to have an indication of the PIV resolution required to obtain a result of predefined accuracy.

Finally, the fourth source of error is related to the fact that the reference pressure has been measured on the airfoil wall, while the present values are obtained on the grid points nearest to the airfoil surface but not lying on it. This error is significant especially in the region near the airfoil nose, where the pressure gradient normal to the wall is maximum owing to the small radius of curvature of the streamlines. This source of error can be attenuated by the techniques proposed in Section 6.1.2.

6.1.2 Smoothed grid

In this section we try to improve the results obtained for the pressure distribution at the airfoil leading edge by modifying the FEM mesh employed for the pressure computation. This is done by replacing the jagged grid with a smooth one which follows the airfoil boundary. For such purpose two different approaches are proposed:

***Interpolation* approach** : the grid nodes close to the real airfoil surface are smoothly interpolated to define a new boundary curve while the remaining grid nodes are moved on that curve, Fig. 6.24;

***Extrapolation* approach** : the airfoil shape is reconstructed directly from the PIV images over the whole rectangular Cartesian mesh and the boundary nodes of the jagged grid lying near to the body surface are then moved onto the airfoil boundary, Fig. 6.26.

A further strategy, which has not been tested here for lack of time, consists in extending the computational mesh up to the real airfoil surface by connecting the jagged boundary to the discretized airfoil boundary where the *no-slip* condition is imposed. In all these approaches, high-aspect ratio triangular elements can be produced close to the modified boundary. However standard smoothing techniques as well as topological mesh changes can be used to improve the mesh quality with additional interpolation of the original data when new grid nodes are introduced.

As an example, the interpolation and extrapolation approaches have been applied to the test case 4 and the obtained results are shown in Fig. 6.25 and 6.27, respectively. With respect to the pressure distribution computed on the jagged grid, Fig. 6.23, the results obtained using the two modified meshes provide a far better estimation of the negative pressure peak on the airfoil nose. In particular, the extrapolation approach seems to provide the best results.

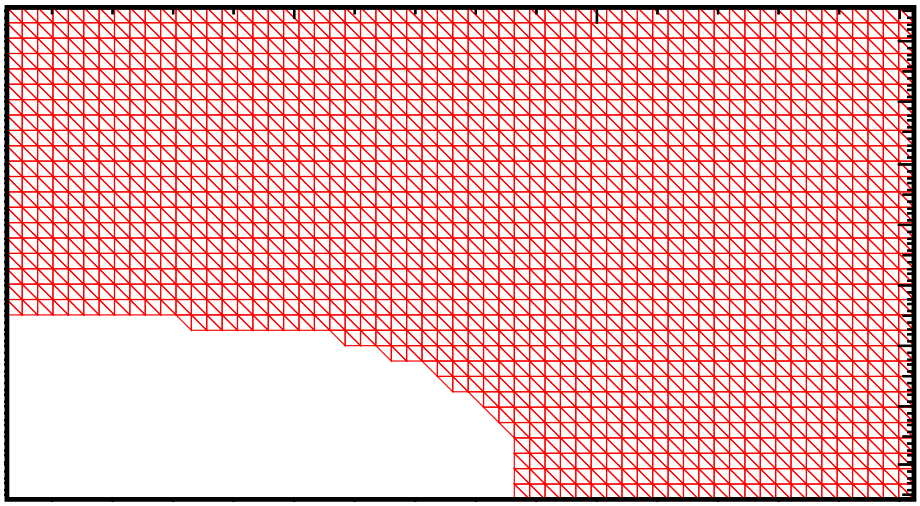


Figure 6.22: Original jagged grid for test case 4

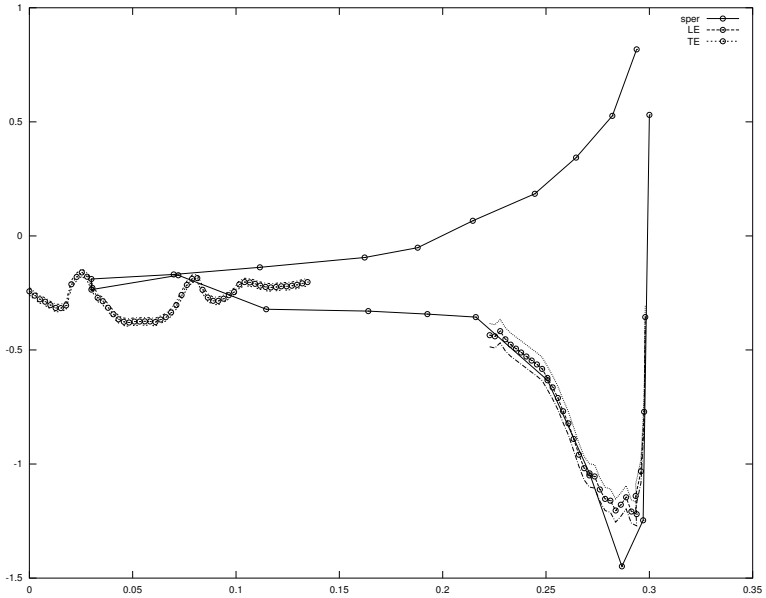


Figure 6.23: Results for test case 4 computed on the original jagged grid: c_P distribution.

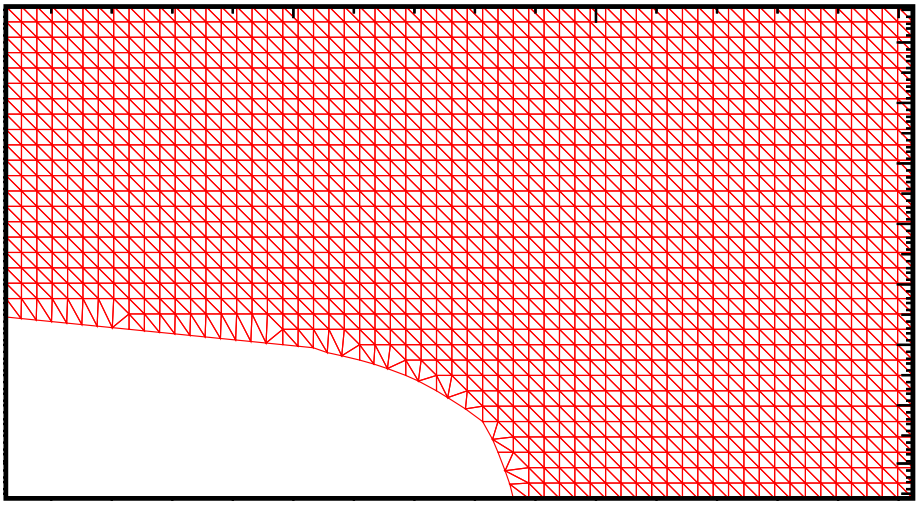


Figure 6.24: New grid generated for test case 4 using the interpolation approach.

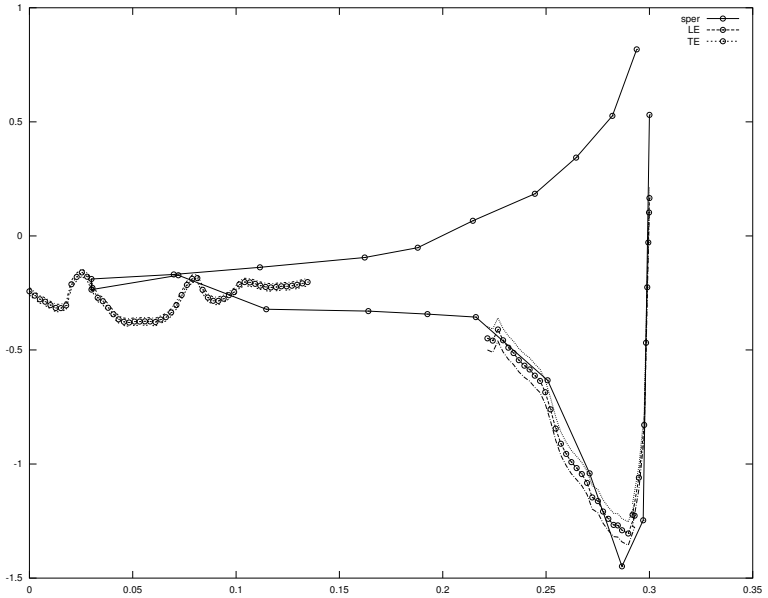


Figure 6.25: Results for the test case 4 computed on the modified grid using the interpolation approach: c_P distribution.

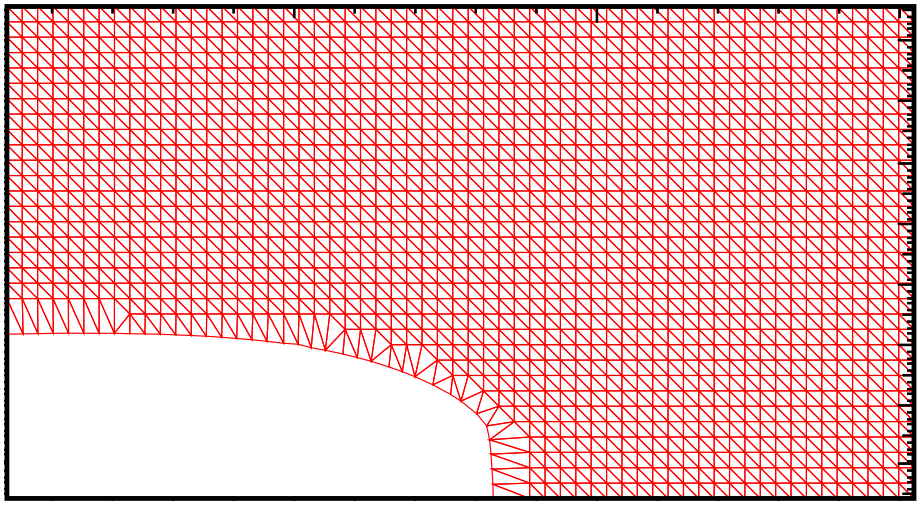


Figure 6.26: New grid generated for test case 4 using the extrapolation approach.

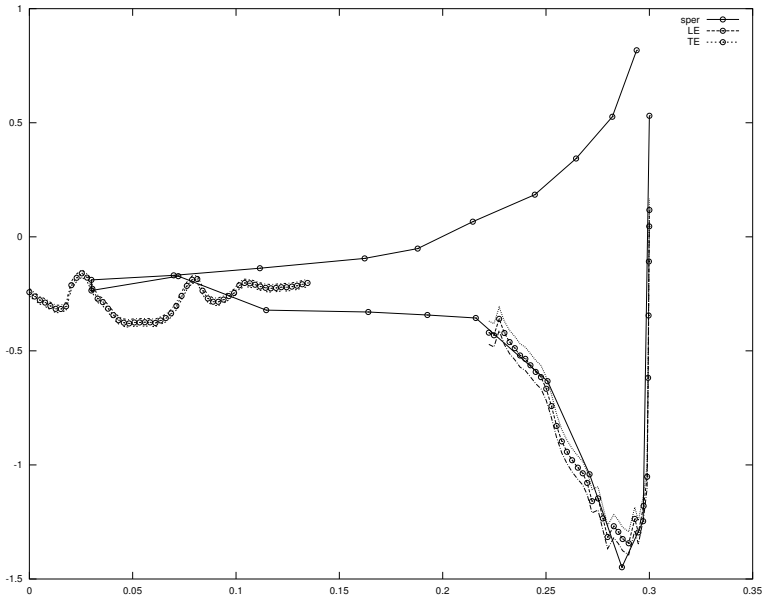


Figure 6.27: Results for the test case 4 computed on the modified grid using the extrapolation approach: c_P distribution.

6.2 Results for the Gurney flap modified set-up

In this section we present the results obtained by employing the method on a NACA 23012 airfoil equipped with a Gurney flap on the trailing edge. Direct pressure measurements are not feasible, because of the difficulty of placing pressure taps near the trailing edge and on the Gurney flap. Both non-averaged and averaged results are presented: in the former case one can identify vortices shed in the wake as minima in the pressure field, in the latter case the wake can be recognized as a region of relative maximum pressure.

The Gurney-on case corresponds to the extracted Gurney flap, while the Gurney-off case corresponds to the retracted configuration.

This section is just intented to contain few examples of how the method can be applied also to this peculiar configuration. For a deeper discussion of the Gurney flap effects, see [25].

As we can appreciate from Figure 6.31, when deployed, the L-shaped tab behaves as a Gurney flap producing a downward deflection of the wake, with reference to the clean configuration, with a consequent increase in lift.

With reference to the steady, 18° incidence case of Figure 6.33– 6.36, it is quite difficult to point out the difference among the extracted and retracted configuration, since the flow is completely separated on the upper surface of the airfoil for both cases. However, locally, the Gurney-off configuration seems to be more intrusive, since the pressure differences are higher.

A deeper analysis on the Gurney flap operating consequences should be performed in order to achieve more significant results. In particular, the complete flow field around the airfoil should be reconstructed to have indications about the global effects of the Gurney flap, but this is beyond the scope of the present work.

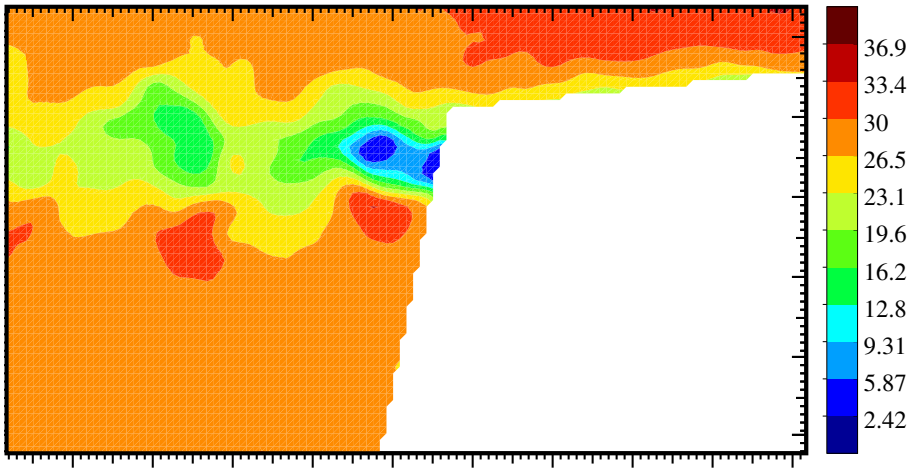


Figure 6.28: Unsteady test case with $\alpha_0 = 5°$, $A = 10°$ and $\alpha = 9°$, Gurney on: modulus of the velocity field.

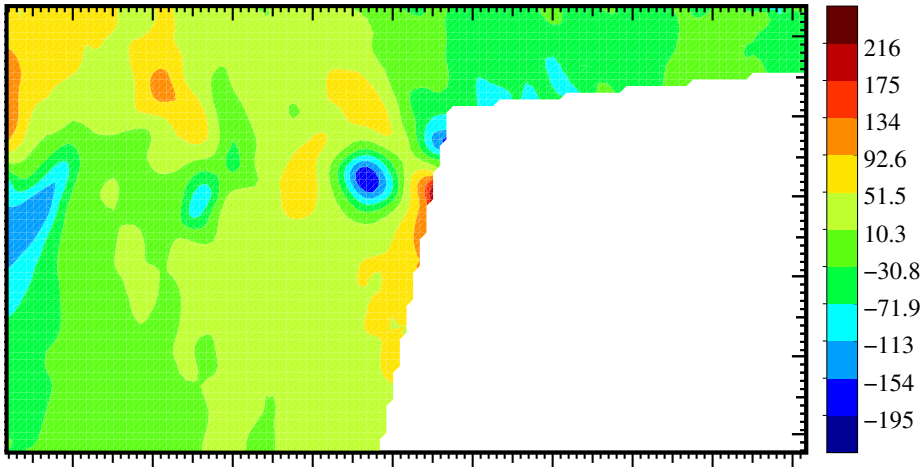


Figure 6.29: Unsteady test case with $\alpha_0 = 5°$, $A = 10°$ and $\alpha = 9°$, Gurney on: pressure field.

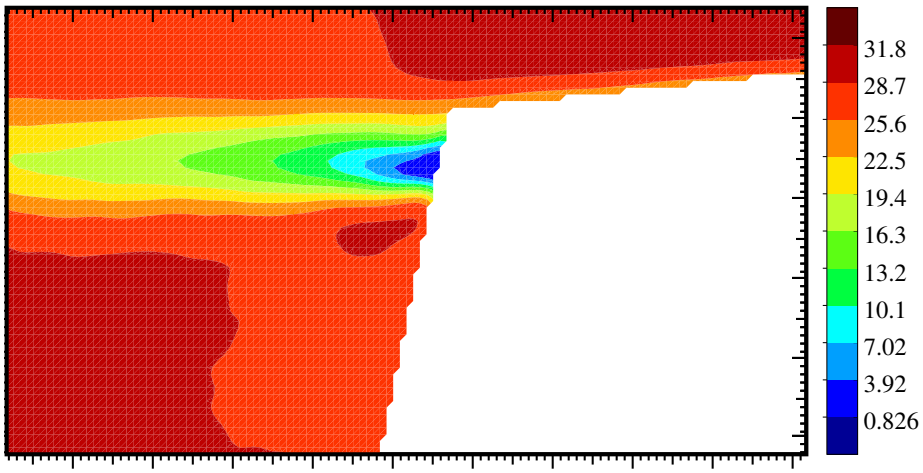


Figure 6.30: Unsteady test case with $\alpha_0 = 5°$, $A = 10°$ and $\alpha = 9°$, Gurney on: modulus of the velocity field (average quantities).

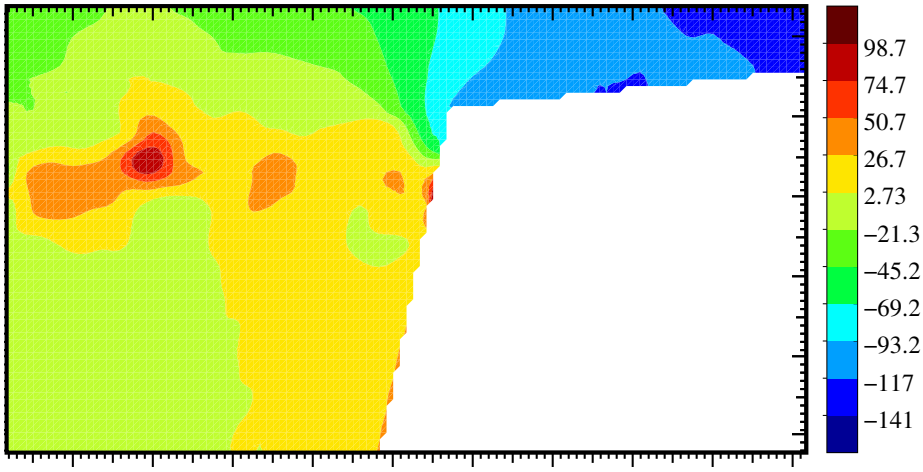


Figure 6.31: Unsteady test case with $\alpha_0 = 5°$, $A = 10°$ and $\alpha = 9°$, Gurney on: pressure field (average quantities).

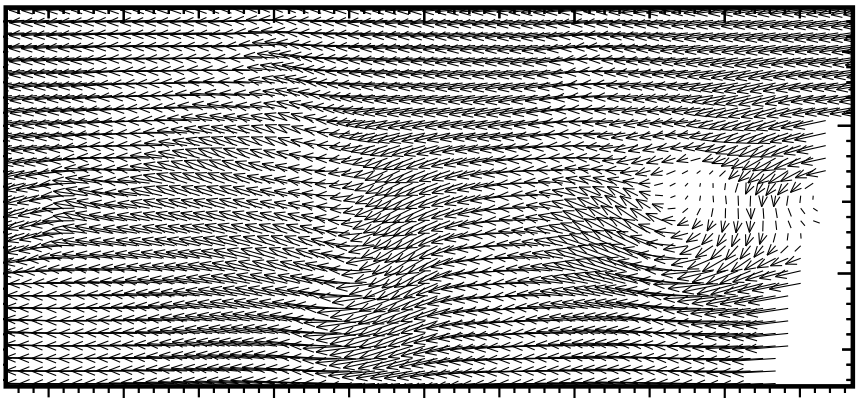


Figure 6.32: Unsteady test case with $\alpha_0 = 5°$, $A = 10°$ and $\alpha = 9°$, Gurney on: velocity field in the wake.

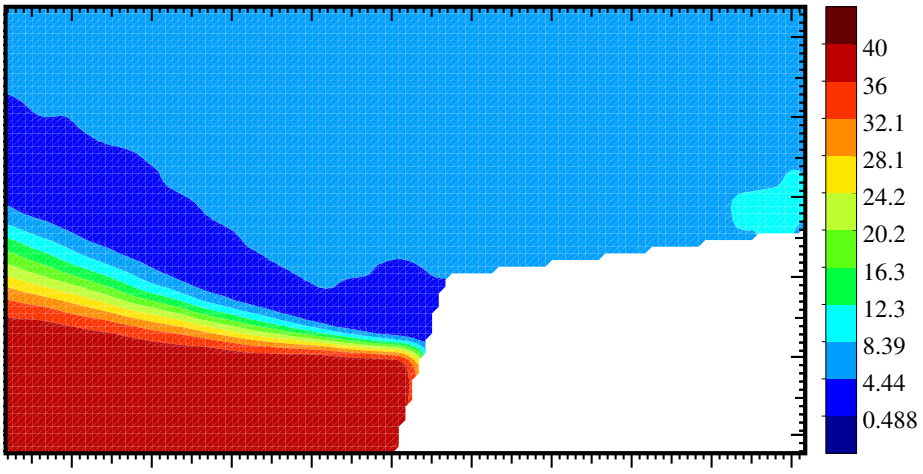


Figure 6.33: Steady test case with $\alpha = 18°$, Gurney on: modulus of the velocity field (average quantities).

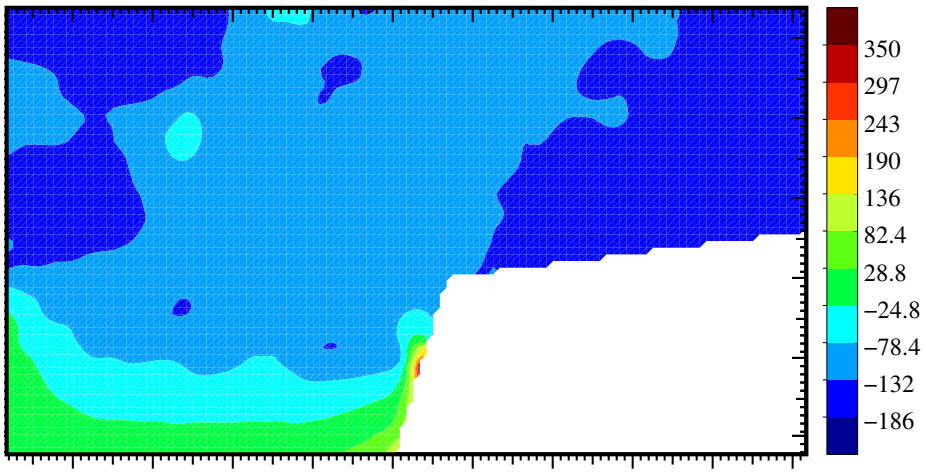


Figure 6.34: Steady test case with $\alpha = 18°$, Gurney on: pressure field (average quantities).

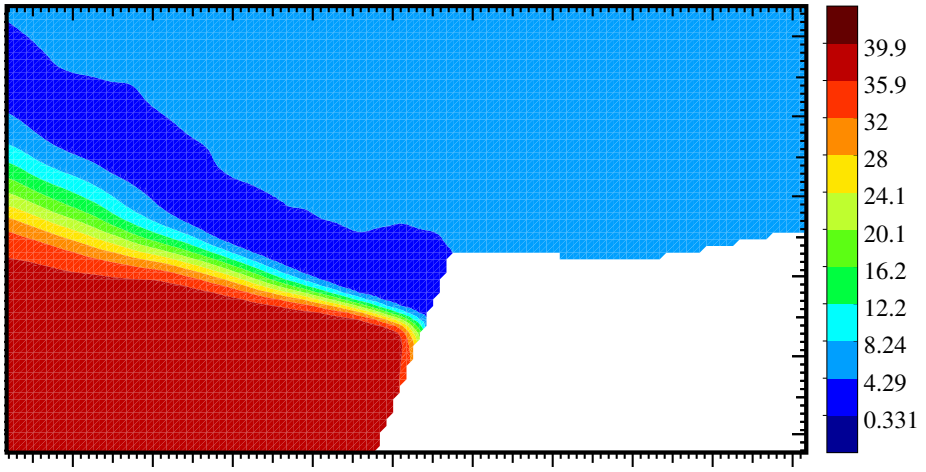


Figure 6.35: Steady test case with $\alpha = 18°$, Gurney off: modulus of the velocity field (average quantities).

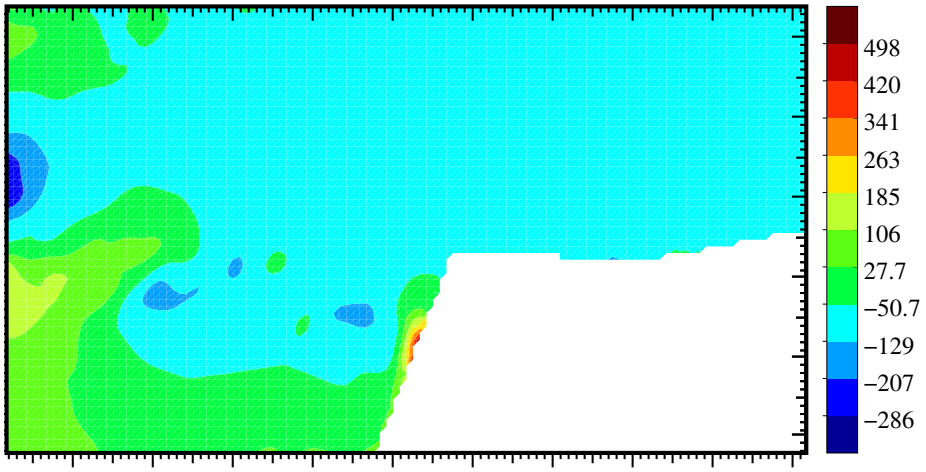


Figure 6.36: Steady test case with $\alpha = 18°$, Gurney off: pressure field (average quantities).

Chapter 7

Conclusions

In this work an innovative procedure to compute the pressure field from PIV velocity measurements has been developed which does not require the imposition of any, possibly unphysical, boundary condition on the pressure such as for other techniques already available in the literature. Based on the uncoupled method of Glowinski–Pironneau for the solution of the Navier–Stokes equations, a set of linear PDE problems for the unknown pressure field has been derived and its numerical solution by means of the finite element method implemented.

First the proposed method is tested against an exact solution of the Navier–Stokes equations over a square domain, showing second order accuracy with respect to the grid size h, with the pressure error being computed in the L^∞ norm. This norm has been chosen because it gives valuable information to appraise the properties of the method from the viewpoint of the experimentalist. Performed Monte Carlo simulations, with imposed stochastic perturbations of the velocity field, show that the regions which are most sensitive to error on the velocity data are the boundary and the corners of the square domain, with an approximate ratio between the standard deviation of the error and the standard deviation of the perturbation of almost 2. Moreover this ratio results to be quite insensitive to a change in the perturbation amplitude. In addition to the former statistical analysis, an error analysis with deterministic velocity perturbations at different wavelengths has been also performed. Such analysis indicates that counterbalancing effects, depending on the discretization and the method properties, contribute to the pressure error so that no well defined trends with respect to the imposed perturbation wavelength are observed.

The method has then been applied to the PIV measurements of a typical aeronautical problem: the oscillating airfoil. The comparison between the computed pressure distribution and the available pressure measurements on

the airfoil surface shows pretty encouraging results. The reconstructed pressure field presents the expected structures of the examined cases (e.g separated flow, reattaching flow, etc.). Nevertheless some discrepancies are still present between the numerical and the experimental pressure values. This discrepancy can be ascribed to four main different sources of error, namely:

- the reflection of the laser sheet in proximity of the airfoil nose, which causes a bad correlation in the PIV image post-processing and then spurious values of the computed velocity;
- the high pressure gradient in the region close to the airfoil nose. Since the pressure is not evaluated on the airfoil surface, but on the first measurement point, this will result into a significant difference;
- the violation of the incompressibility constraint for the two dimensional velocity field, especially in presence of flow separations, for which the three-dimensional character of the flow can not be neglected;
- the FEM sensitivity to mesh quality, especially with respect to sharp cornern on the boundary.

However the second and fourth effects can be both substantially reduced by both smoothing and reshaping the grid, placing the inner nodes on the profile, and interpolating there the velocity values, as shown in section 3.1.5.

Future developments

Further improvements of the proposed method can be developed by taking into account the actual deviation from the incompressibility constraint of the sampled two–dimensional velocity field, which is expected to play an important role when deep flow separation occurs. In principle, such a correction can be achieved by modifying the mathematical formulation of the governing Navier-Stokes equations introducing a source term in the continuity equation.

On the contrary, for mildly separated flow and more generally when the deviation from the exact incompressibility condition is rather small, such as for a nearly two-dimensional flow field, a correction to the initial velocity datum can be obtained by means of a projection onto a divergence-free functional space.

Another possible improvement, whose consequences have to be investigated, consists in adding supplementary nodes on the airfoil, where the no-slip condition can be imposed.

Finally the developed software could be extended to 3D flows using tomographic, stereo PIV.

Bibliography

[1] C. Tropea, A. Yarin, and J. Foss. Springer Handbook of Experimental Fluid Mechanics. *Springer-Verlag Berlin Heidelberg*, 2007.

[2] E. Berton, C. Maresca, and D. Favier. A new experimental method for determining local airloads on rotor blades in forward flight. *Experiments in fluids*, 37(3):455–457, 2004.

[3] R. Gurka, A. Liberzon, D. Hefetz, D. Rubinstein, and U. Shavit. Computation of pressure distribution using piv velocity data. In *Workshop on Particle Image Velocimetry*, 1999.

[4] N. Fujisawa, S. Tanahashi, and K. Srinivas. Evaluation of pressure field and fluid forces on a circular cylinder with and without rotational oscillation using velocity data from piv measurement. *Measurement Science and Technology*, 16(4):989, 2005.

[5] S. Obi and N. Tokai. The pressure–velocity correlation in oscillatory turbulent flow between a pair of bluff bodies. *International journal of heat and fluid flow*, 27(5):768–776, 2006.

[6] T. Baur and J. Köngeter. Piv with high temporal resolution for the determination of local pressure reductions from coherent turbulence phenomena. In *3rd International Workshop on Particle Image Velocimetry,(Santa Barbara, CA, USA),(1999-9)*, 1999.

[7] B. W. van Oudheusden, F. Scarano, E. Roosenboom, E. Casimiri, and L. Souverein. Evaluation of integral forces and pressure fields from planar velocimetry data for incompressible and compressible flows. *Experiments in Fluids*, 43(2-3):153–162, 2007.

[8] X. Liu and J. Katz. Instantaneous pressure and material acceleration measurements using a four-exposure piv system. *Experiments in Fluids*, 41(2):227–240, 2006.

[9] D. Violato, P. Moore, and F. Scarano. Lagrangian and eulerian pressure field evaluation of rod-airfoil flow from time-resolved tomographic piv. *Experiments in fluids*, 50(4):1057–1070, 2011.

[10] R. De Kat and B. Van Oudheusden. Instantaneous planar pressure determination from piv in turbulent flow. *Experiments in fluids*, 52(5):1089–1106, 2012.

[11] S. Ghaemi, D. Ragni, and F. Scarano. Piv-based pressure fluctuations in the turbulent boundary layer. *Experiments in fluids*, 53(6):1823–1840, 2012.

[12] J. Charonko, C. King, B. Smith, and P. Vlachos. Assessment of pressure field calculations from particle image velocimetry measurements. *Measurement Science and Technology*, 21(10):105401, 2010.

[13] L. Kleiser and U. Schumann. Treatment of incompressibility and boundary conditions in 3-d numerical spectral simulations of plane channel flows. In *3rd Conference on Numerical Methods in Fluid Mechanics*, volume 1, pages 165–173, 1980.

[14] L. Quartapelle. *Numerical solution of the incompressible Navier-Stokes equations*, volume 113. Springer, 1993.

[15] L. Quartapelle and M. Napolitano. Integral conditions for the pressure in the computation of incompressible viscous flows. *Journal of Computational Physics*, 62(2):340–348, 1986.

[16] R. Glowinski and O. Pironneau. On a mixed finite element approximation of the stokes problem (i). *Numerische Mathematik*, 33(4):397–424, 1979.

[17] J. Guermond and L. Quartapelle. On stability and convergence of projection methods based on pressure poisson equation. *International Journal for Numerical Methods in Fluids*, 26(9):1039–1053, 1998.

[18] G. EP Box and M. Muller. A note on the generation of random normal deviates. *The Annals of Mathematical Statistics*, 29(2):610–611, 1958.

[19] R. Y. Rubinstein and D. P. Kroese. *Simulation and the Monte Carlo method*, volume 707. Wiley. com, 2011.

[20] A. Zanotti. *Retreating blade dynamic stall.* PhD thesis, Politecnino di Milano, 2012.

[21] *PIVview User Manual, available at* `http://www.pivtec.com/`.

[22] H. Yeo. Assessment of active controls for rotor performance enhancement. *Journal of the American Helicopter Society*, 53(2):152–163, 2008.

[23] H. Jin and S. Prudhomme. A posteriori error estimation of steady-state finite element solutions of the navier–stokes equations by a subdomain residual method. *Computer methods in applied mechanics and engineering*, 159:19–48, 1998.

[24] J. de Frutos, B. Garcia-Archilla, and J. Novo. A posteriori error estimations for mixed finite-element approximations to the navier–stokes equations. *Journal of Computational and Applied Mathematics*, 236(6):1103–1122, 2011.

[25] A. Zanotti, D. Grassi, and G. Gibertini. Experimental evaluation of an l-shaped tab to be used as an active gurney flap for dynamic stall control. In *38th ERF*, 2013.

Appendix A

Estratto in lingua italiana

In questa appendice presentiamo un breve estratto in lingua italiana del contenuto della tesi.

A.1 Introduzione

La tecnica della Particle Image Velocimetry (PIV) è probabilmente al giorno d'oggi una delle tecniche di misura più utilizzate in campo fluidodinamico, in quando consente una misura non intrusiva e quantitativa, con sufficiente accuratezza, dell'intero campo di velocità all'interno di una finestra di misura. Avendo a disposizione il campo di velocità in un dominio, si può quindi pensare di calcolarne il corrispondente campo di pressione. Ciò permetterebbe di ottenere in modo non intrusivo il campo di pressione, ma soprattutto darebbe la possibilità di avere misure di pressione dove non è possibile, o pratico, posizionare dei trasduttori di pressione (ad esempio al bordo di uscita di un profilo alare).

In letteratura, fino a questo momento sono prevalsi due approcci. Il primo è quello di risolvere un'equazione di Poisson per la pressione, ottenuta applicando l'operatore di divergenza all'equazione della quantità di moto delle equazioni di Navier-Stokes, affidandosi alle proprietà di regolarizzazione del laplaciano per evitare di amplificare le incertezze di misura del campo di velocità (si veda ad esempio [3]). Tuttavia i risultati dipendono dal tipo di condizioni al contorno per la pressione usate, non sempre definite in maniera consistente. Un secondo approccio (si veda [6]) è invece quello di integrare il gradiente di pressione dell'equazione della quantità di moto attraverso le differenze finite. Questo secondo metodo è del tutto consistente, ma tende ad accumulare l'errore di misura ad ogni passo di integrazione.

La tecnica da noi proposta ha come obbiettivo quello di combinare gli aspetti positivi dei due metodi, ovvero la consistenza dell'imposizione al con-

torno per la pressione ed una maggiore refrattarietà all'errore di misura della velocità grazie all'operatore Laplaciano.

A.2 Formulazione matematica

Il metodo si basa sul disaccoppiamento delle equazioni di Navier–Stokes con il metodo di Glowinski–Pironneau, ovvero con l'introduzione del vincolo di incomprimibilità tramite l'utilizzo della funzione φ (per approfondimenti di veda [16]) :

$$\begin{cases} -\nabla^2 p = -\boldsymbol{\nabla}\cdot\boldsymbol{g} & \\ (-\nabla^2+\gamma)\boldsymbol{u} = -\boldsymbol{\nabla}p+\boldsymbol{g}, & \boldsymbol{u}\big|_S = \boldsymbol{b} \\ -\nabla^2\varphi = \boldsymbol{\nabla}\cdot\boldsymbol{u}, & \varphi\big|_S = 0, \qquad \frac{\partial\varphi}{\partial n}\big|_S = 0. \end{cases} \tag{A.1}$$

Come si può notare l'equazione di φ deve soddisfare condizioni al contorno sovrabbondanti, di cui la prima può essere imposta direttamente, mentre la seconda sarà trattata in un momento successivo. L'equazione della pressione è invece priva di condizioni al contorno, il sistema è quindi accoppiato attraverso le condizioni al contorno. È possibile introdurre la seguente decomposizione delle variabili:

$$\begin{Bmatrix} p(\boldsymbol{x}) \\ \boldsymbol{u}(\boldsymbol{x}) \\ \varphi(\boldsymbol{x}) \end{Bmatrix} = \begin{Bmatrix} p^0(\boldsymbol{x}) \\ \boldsymbol{u}^0(\boldsymbol{x}) \\ \varphi^0(\boldsymbol{x}) \end{Bmatrix} + \sum_{k=1}^{N_D} \begin{Bmatrix} p^k(\boldsymbol{x}) \\ \boldsymbol{u}^k(\boldsymbol{x}) \\ \varphi^k(\boldsymbol{x}) \end{Bmatrix} \lambda^k, \tag{A.2}$$

che porta alle seguenti equazioni:

$$\begin{cases} -\nabla^2 p^0 = -\boldsymbol{\nabla}\cdot\boldsymbol{g}, & p^0|_S = 0 \\ (-\nabla^2+\gamma)\boldsymbol{u^0} = -\boldsymbol{\nabla}p^0+\boldsymbol{g}, & \boldsymbol{u}^0|_S = \boldsymbol{b} \\ -\nabla^2\varphi^0 = \boldsymbol{\nabla}\cdot\boldsymbol{u^0}, & \varphi^0|_S = 0, \end{cases} \tag{A.3}$$

$$\begin{cases} -\nabla^2 p^k = 0, & p^k|_S = \mu^k & \\ (-\nabla^2+\gamma)\boldsymbol{u}^k = -\boldsymbol{\nabla}p^k, & \boldsymbol{u}^k|_S = 0 & k = 1 : N_D \\ -\nabla^2\varphi^k = \boldsymbol{\nabla}\cdot\boldsymbol{u}^k, & \varphi^k|_S = 0, & \end{cases} \tag{A.4}$$

dove N_D è il numero di gradi di libertà sul contorno e μ^k è un'opportuna funzione di base per lo spazio delle funzioni traccia, la cui controparte nel caso discretizzato, è la funzione di base degli elementi finiti. Bisogna quindi risolvere un sistema di equazioni non omogeneo e tanti sistemi di equazioni k quanti sono i gradi di libertà sul contorno. Per poter ricostruire la pressione è però necessario trovare i λ^k, coefficienti dell'espansione succitate, che

rappresentano anche i valori della pressione sul bordo essendo p^0 nulla sul bordo. Per far ciò possiamo imporre in modo debole la seconda condizione al contorno su φ come accennato prima:

$$\oint_S w \frac{\partial \varphi^0}{\partial n} + \sum_{k=1}^{N_D} \oint_S w \frac{\partial \varphi^k}{\partial n} \lambda^k = 0. \tag{A.5}$$

Con qualche manipolazione matematica si ottiene il sistema di equazioni:

$$\sum_{k=1}^{N_D} \int_V (w \boldsymbol{\nabla} \cdot \boldsymbol{u}^k - \boldsymbol{\nabla} w \cdot \boldsymbol{\nabla} \varphi^k) \lambda^k = - \int_V (w \boldsymbol{\nabla} \cdot \boldsymbol{u}^0 - \boldsymbol{\nabla} w \cdot \boldsymbol{\nabla} \varphi^0), \tag{A.6}$$

che, una volta discretizzato, produrrà un sistema lineare la cui soluzione è un vettore contenente i valori della pressione sul bordo.

A.3 Discretizzazione numerica

Il passo successivo è l'introduzione di una discretizzazione ad elementi finiti del problema. Innanzitutto è necessario costruire una griglia a partire dai dati di misura della PIV. Si è optato per una triangolazione del dominio, con i nodi posti nel centro delle finestre di interrogazione della PIV, il che produce una griglia di triangoli isosceli e rettangoli. Nel caso di domini con corpi immersi, è necessario togliere dalla mesh tutti i nodi in cui la correlazione della PIV non è potuta avvenire. Per arginare problemi dovuti ad angoli pronunciati degli elementi sul contorno, può essere introdotta una regolarizzazione della griglia con successiva re-interpolazione dei dati di velocità sul bordo.

Una volta definita la griglia, si introduce una discretizzazione ad elementi finiti del problema, utilizzando elementi $\mathbb{P}^2$ per la velocità e $\mathbb{P}^1$ per la pressione e φ.

È stato quindi prodotto un programma Fortran90 che implementa il metodo appena descritto.

A.4 Validazione

Si è quindi passati alla validazione del programma e del metodo prodotto. Dopo la validazione delle singole componenti, l'intero solutore è stato provato su una soluzione esatta delle equazioni di Navier–Stokes per vedere se era in grado di ricostruire opportunamente il campo di pressione, con due diversi trattamenti della derivata temporale. Le curve di convergenza sono riportate

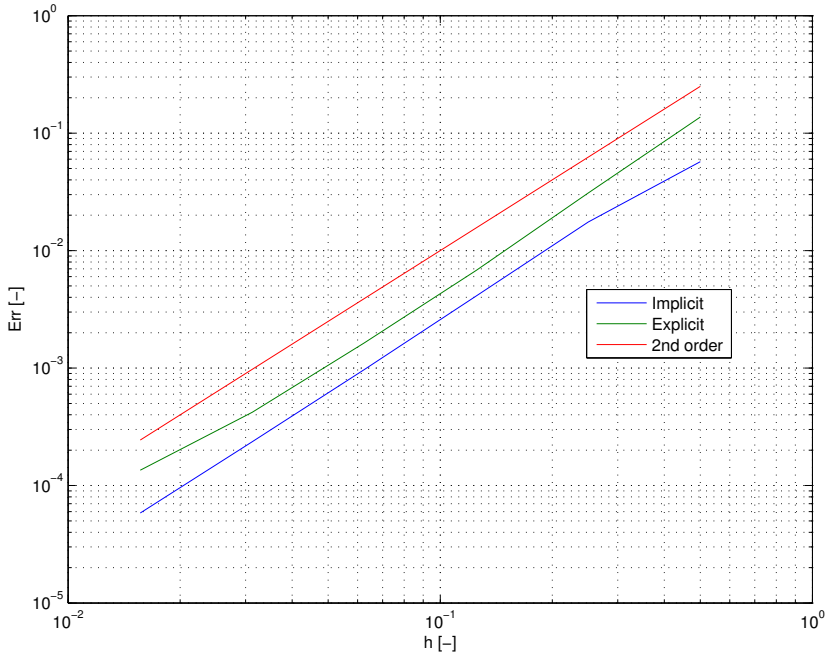


Figure A.1: Analisi di convergenza per la pressione ricostruita

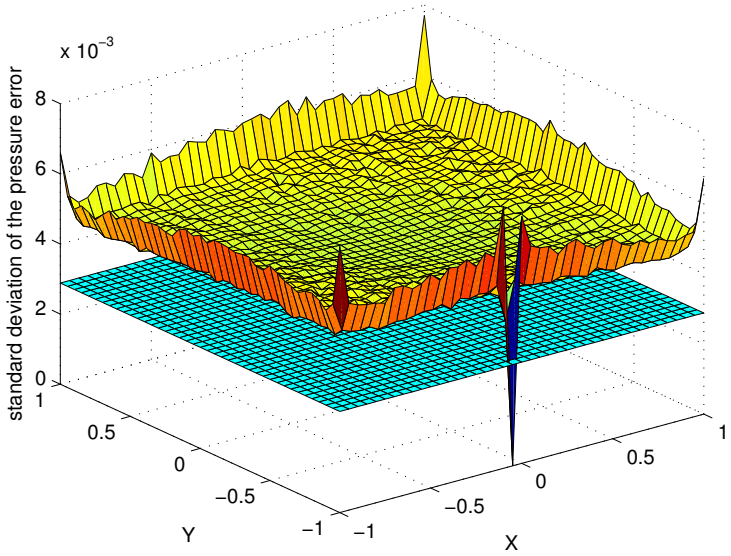


Figure A.2: Distribuzione della deviazione standard dell'errore nel dominio

in Figura A.1. Come si può notare il metodo garantisce una convergenza del second'ordine al variare della dimensione della mesh, con l'errore della pressione misurato nella norma del massimo.

Successivamente si è provata la robustezza del metodo ad errori di misura della velocità. Per far ciò è stata effettuata una simulazione Monte Carlo perturbando il campo di velocità con un errore a distribuzione gaussiana a media nulla e con deviazione standard assegnata. Alla fine l'errore della pressione calcolato localmente è stato mediato e ne è stata calcolata la deviazione standard. Per quanto riguarda il valore atteso dell'errore, si è riscontrato che questo converge sull'errore di discretizzazione numerica. Per quanto riguarda la varianza dell'errore, i risultati sono rappresentati in Figura A.2. Come si può notare, il metodo risulta più sensibile agli errori di velocitÃ sul bordo, con un rapporto tra la varianza dell'errore e quello della perturbazione di circa 2. Inoltre, ripetendo la simulazione per diversi valori della perturbazione, si è visto come questo rapporto sia insensibile alla variazione delle deviazione standard della perturbazione. Per questi motivi quest'ultimo risultato può essere usato per stimare l'incertezza sulla pressione nota l'incertezza sulla velocità.

A.5 La campagna sperimentale

La campagna sperimentale per la produzione del database PIV sullo stallo dinamico si è svolta presso la galleria del vento dipartimentale. L'allestimento sperimentale consisteva in un modello di profilo NACA 23012 mosso in angolo di incidenza da un motore elettrico, a sua volta controllato da un sistema di controllo PID. La galleria è stata opportunamente allestita per misure PIV, mentre il modello è stato dotato di trasduttori per l'acquisizione instazionaria delle misure di pressione, utili poi per il confronto con i valori ottenuti dal metodo. Le prove sono state svolte ad un Reynolds di circa 600000 ed una frequenza ridotta di circa 0.1. Le immagini ottenute dalla PIV sono state analizzate e i campi di moto ottenuti sono stati mediati in fase, in modo da filtrare parte del rumore di misura. Sono state effettuate misure sia stazionarie che instazionarie, a diversi angoli di incidenza e con finestre di misura centrate sul bordo d'attacco e di uscita del profilo.

Per i dettagli specifici sulla strumentazione utilizzata si rimanda al corrispondente capitolo della tesi.

A.6 Risultati

Infine la base di dati PIV è stata analizzata dal programma sviluppato. Tra i vari casi possibili, in questo estratto si presenta il caso instazionario del profilo ad angolo di incidenza di $\alpha = 10°$ in discesa, con riferimento alla finestra di misura sul bordo di attacco. Nelle Figure A.3 e A.4 sono rappresentati rispettivamente il campo di pressione ottenuto e il confronto tra i coefficienti di pressione calcolati e quelli ottenuti per via sperimentale. Come si può notare, l'accordo sul bordo di attacco è buono. Risultati ancora migliori sono stati ottenuti con una regolarizzazione della griglia. In altre configurazioni qui non presentate i risultati ottenuti non sono stati altrettanto buoni, per motivi dovuti più alla qualità delle misure PIV più che non al metodo sviluppato. Tra questi sicuramente il fatto che in alcune immagini era presente il riflesso del laser sul profilo, che inficia la correlazione delle misure PIV, producendo risultati di velocità non attendibili. Inoltre il fatto che i dati della PIV non sono noti sul profilo, ma a una distanza da esso che è funzione della posizione della finestra di interrogazione più vicina, comporta scostamenti rispetto al dato misurato specialmente in zone con forte gradiente di pressione normale (ad esempio il naso del profilo).

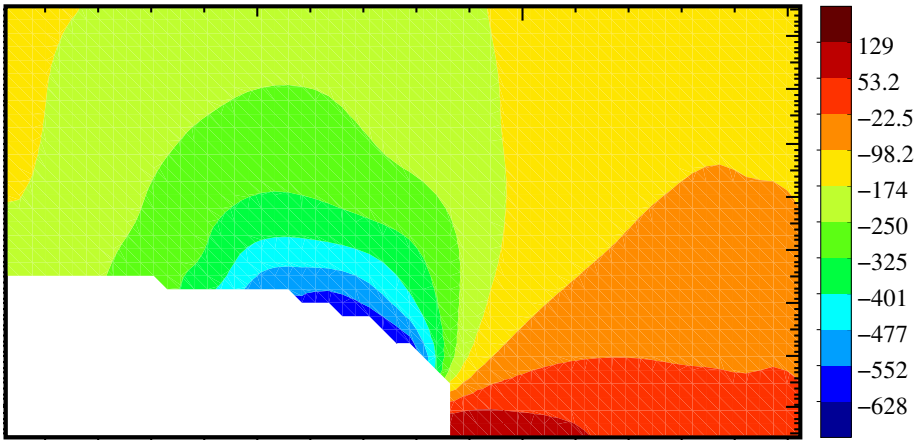


Figure A.3: Caso instazionario, $\alpha = 10°$ in discesa : campo di pressione al bordo di attacco.

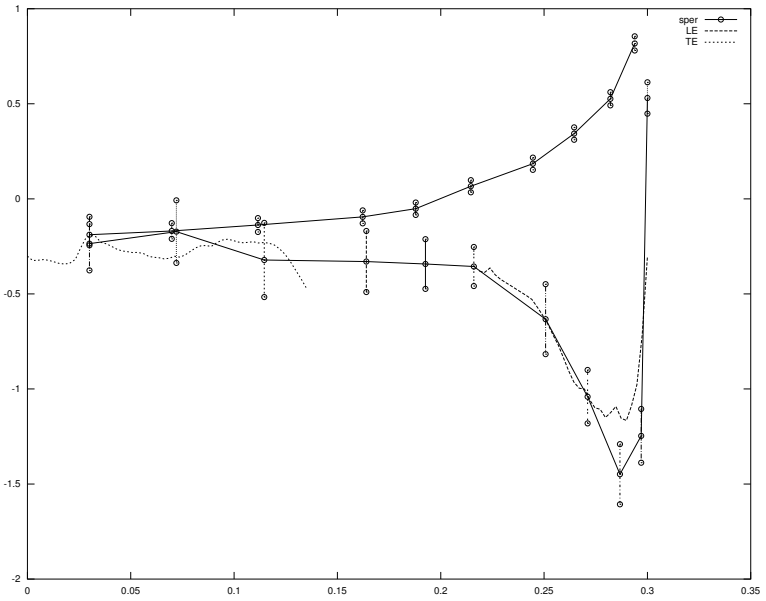


Figure A.4: Risultati per il Test case 4 con griglia originale: distribuzione del c_P .