École polytechnique de Louvain

Comparison of 2D Brinkman penalization methods applied to an impulsively started flow past a cylinder through the Vortex Particle-Mesh framework

Author: **Thomas Crucitti**
Supervisor: **Philippe Chatelain**
Readers: **Jean-François Remacle, Grégoire Winckelmans**
Academic year 2018–2019
Master [120] in Mechanical Engineering

Acknowledgements

Ce document est l'aboutissement de mes études en sciences de l'ingénieur à l'École Polytechnique de Louvain-la-Neuve. Mes remerciements vont à toutes les personnes ayant contribué, sur le plan académique comme personnel, à ma formation et à la réussite de mes études universitaires. Je tiens à remercier plus particulièrement les personnes m'ayant permis de mener à bien ce travail de fin d'études.

J'exprime tout d'abord ma gratitude à mes promoteurs, les Professeurs Philippe Chatelain et Grégoire Winckelmans, qui se sont toujous montrés disponibles et motivés tout au long de ce travail.

Je remercie sincèrement Thomas Gillis pour sa disponibilité, sa patience et ses précieuses indications.

Finalement, je voudrais remercier ma famille pour leur soutien moral, ainsi que l'intérêt qu'ils portaient à mon mémoire.

Contents

Introduction

The proper way to introduce this thesis is definitely a summary, a state-of-the-art so to say, of the different techniques and methods of penalization which have been developed over time, as well as the reasons for which they are still used. Wherever penalization is cited throughout this document, it actually refers to the manner whose obstacles are taken into account in incompressible viscous flows. In other words, it is about penalizing the no-slip boundary condition on the surface of a body surrounded by a viscous fluid.

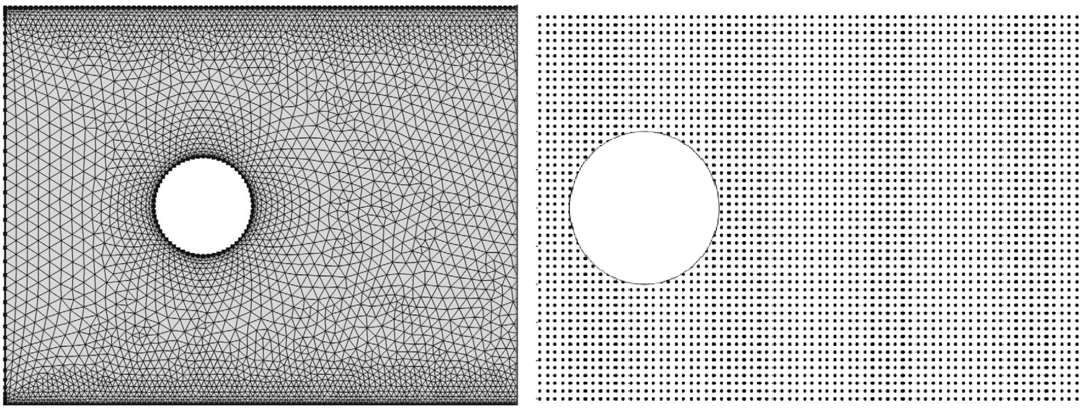

Figure 1: Unstructured (left) and uniform (right) meshes [1]

The aim of developing penalization methods is to avoid body-fitted unstructured meshes and therefore, to prefer uniform Cartesian grids (as shown on FIGURE 1) on which fast and efficient spectral or finite differences can be applied. A way to implement that is the addition of a penalized velocity term in the Navier-Stokes momentum equation, and it is also the unique technique and its consideration in the model that this thesis focuses on.

The described method is actually called the *continuous forcing* approach. In fact, the treatment of inner bodies for flow problems on uniform grids can be split between *discrete* and *continuous forcing* approaches, whose the differentiation is introduced by Mittal and Iaccarino [13]. While the *continuous forcing* approach uses a mathematical artefact in the Navier-Stokes equations, the *discrete forcing* techniques modify the discretization of the latter such as the immersed body is taken into account. Methods of this second class are briefly introduced since, as it is explained further, recent works have proven their efficiency and accuracy.

This introduction is thus structured as follows: the timeline of the *continuous forcing* approach is firstly done, then comes a brief presentation of the Vortex Particle-Mesh methods and finally, a summary of the development of this thesis and its purpose.

Mathematical model - *Continuous forcing approach*

About forty years ago, several authors (for instance Goldstein et al. [2]) followed the former work of Peskin [3], [4] consisting in the numerical analysis of blood flow in the heart. In his study, Peskin provides the solution of the Navier-Stokes equations for moving immersed boundaries that interact with the fluid, those ones being the muscular heart walls. To this end, he adds both a time integral of the velocity and a velocity penalization term only on the boundary, that method being therefore called the Immersed Boundary approach.

FIGURE 2 shows Peskin's drawings of the pattern of blood flow in heart valves. One can appreciate the carefulness and accuracy brought to this handmade representation.

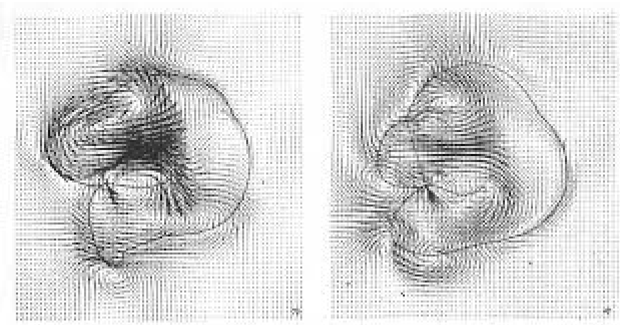

Figure 2: Peskin's drawings, 1977 [4]

A bit later, many other studies had shown that the Peskin's method was not sufficient and did not lead to a correct solution at high Reynolds numbers (E.M. Saiki and S. Biringen [5]). Thus, in order to overcome that issue, those works rightly suggest the extension of the penalization to the volume of the obstacle. E.M. Saiki and S. Biringen validate their approach by providing accurate agreements between their calculations and previous computational and experimental results for an immersed cylinder in an uniform flow.

In 1984, Arquis and Caltagirone [6] independently published similar works in which they add a penalization term on the velocity defined on the volume of the immersed body. Their method actually corresponds to a Brinkman model which applies a Lagrangian relaxation of the body constraint $\mathbf{u} = \mathbf{u}_b$ to impose the inner boundary condition [10], [14]. J.P. Caltagirone went further [7] and suggested that this approach was also able to calculate the lift and drag coefficients by integrating the penalization term inside the obstacle.

Thus, those ideas were successfully used by Angot [8], [9] to deal with fluid-porous-solid systems. He actually validated the mathematical modelling by means of several numerical tests that also confirmed the efficiency and accuracy of the method. The same methodology was therefore widely adopted for various works whose the computation of incompressible flows around a cylinder or behind a step [8], [11], [12].

In 2003, Carbou resumes the work of P. Angot, C-H Bruneau, and P. Fabrie wherein error estimates are established for a L^2 penalization, but are not in agreement with experimental results [10]. Thus, Carbou rectifies those error estimates and rigorously shows that a boundary layer appears in the obstacle, not in the fluid region [11], which is also verified in this document.

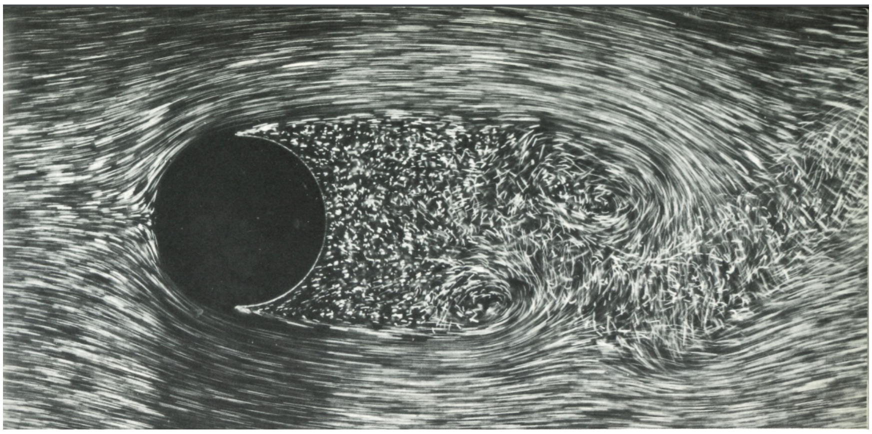

Figure 3: Example of experimental data, 1972 [23]
Immersed cylinder at $Re = 2000$
Visualization by air bubbles in water

Hence, the mathematical model of penalization has been generalized by the addition of a penalized velocity term in the momentum equation of Navier-Stokes, defined on the whole volume of the immersed obstacle. As previously mentioned, that modelling is the only one discussed in that thesis and this, because of its widespread use by the scientific community and the industry, but also because of its proven efficiency and accuracy over time.

Vortex Particle-Mesh methods

Vortex methods use the incompressible Navier-Stokes equations in their vorticity-velocity formulation. Those methods, coupled to a remeshing operation, have shown their ability to efficiently and accurately simulate complex viscous vortex flows [15],[17]. Therefore, the combination of the advantages of a vortex-particle method with those of a mesh-based approach is here called Vortex Particle-Mesh method (VPM), as mentioned in [16],[17].

However, the application of the *continuous* and *discrete forcing* approaches to the VPM framework is not straightforward and has been only partially studied.

In the *discrete forcing* approach, the Boundary Element Method (BEM), also referred as the panel method, is the most widespread technique that has been associated to the VPM background [15]. It firstly consists in the convection and diffusion of the vorticity field without handling the vorticity generated at the wall. Then the diffusion of the vortex sheet related to the residual slip velocity is applied into the flow [20].

Still in this family of methods, recent works have proposed the Immersed Interface Method (IIM) which uses modified Taylor series to take the inner body into account [19]. The Institute of Mechanics, Materials and Civil Engineering (*Université catholique de Louvain - UCLouvain*) is a leader of this technique and provides a validation of the latter in 2D, showing that the Immersed Interface Vortex Particle-Mesh (IIVPM) solver offers a sharp treatment of boundaries while conserving the temporal convergence order [19]. It is actually the only method cited in this document which allows that treatment of boundaries, while the other approaches either smear the obstacle interface, which degrades the spatial accuracy (as the Brinkman penalization technique) or their diffusion operator is split, that limiting the temporal convergence to the first order (as the BEM-VPM).

As mentioned in the timeline of the *continuous forcing* approach, the Brinkman penalization method has been directly applied to VPM framework [21]. Other methods iterate on the body constraint to converge the velocity and vorticity fields, that making the no-slip condition enforced at the obstacle surface, which actually improves the solution quality [18], [19]. Nevertheless, this approach involves only a first order at the boundary and requires several resolutions of the Poisson equation on both the computational domain and inside the obstacle at each iteration, which implies a large computational cost.

Comparison of VPM methods

The aim of this Master's thesis is the comparison of three widely used VPM methods of the *continuous forcing* approach in 2D: 2 non-iterative methods, i.e. the implicit approaches on the velocity and the vorticity, and the iterative penalization method proposed by Hejlesen et al. [22] (also used and studied in [18]).

First of all, the methodology and computations for generating the numerical results presented in this document are briefly summarized. Then a parametric study on the penalized velocity term, without the Navier-Stokes model, is done and compared to the perfect analytical case for an uniform flow around a cylinder, that being called the potential flow. Finally, the incompressible Navier-Stokes equations are penalized and processed, also for an immersed cylinder, through the considered 2D-VPM methods at a moderate Reynolds number ($Re = 550$).

Chapter 1

Methodology and tools

This first chapter exposes the 2D Vortex Particle-Mesh methodology, the use of penalization techniques, as well as the way the penalization step is processed. It also briefly introduces the tools enabling the computation and visualization of the numerical results.

In fact, this chapter is meant to provide the required knowledge in order to clearly and precisely understand the mechanism of numerical methods behind the presented graphical results.

Contents

1.1 The 2D Vortex Particle-Mesh method

This section presents the Vortex Particle-Mesh (VPM) framework as it is exploited in this work. Then the Brinkman penalization is exposed and defined through the VPM method.

1.1.1 General considerations

The Vortex Particle-Mesh method as used in this document takes basis on the 2-dimension Navier-Stokes equations in their vorticity-velocity formulation (Eq. 1.1) for an incompressible flow (Eq. 1.2). In 2D, the only vorticity is the one out-of-plane, that being $\boldsymbol{\omega} = \omega \mathbf{e}_z$.

$$\frac{D\omega}{Dt} = \frac{\partial \omega}{\partial t} + \mathbf{u} \cdot \nabla \omega = \nu \nabla^2 \omega \tag{1.1}$$

$$\nabla \cdot \mathbf{u} = 0 \tag{1.2}$$

where ν is the kinematic viscosity and the velocity $\mathbf{u}$ has to be recovered from the Poisson equation (see [17], [19] and references therein),

$$\nabla^2 \psi_\omega = -\omega \qquad\qquad \mathbf{u}_\omega = \nabla \times (\psi_\omega \mathbf{e}_z) \tag{1.3}$$

Those equations being reduced to,

$$\nabla^2 \mathbf{u}_\omega = -\nabla \times (\omega \mathbf{e}_z) \tag{1.4}$$

Here the velocity $\mathbf{u}_\omega$ is thus the one recovered from Eq. 1.4 such as $\mathbf{u} = \mathbf{u}_\infty + \mathbf{u}_\omega$, with $\mathbf{u}_\infty$ the free flow velocity.

Through the VPM method, the vorticity field ω is discretized by means of particles having positions $\mathbf{x}_p$, material surfaces S_p and strengths $\boldsymbol{\alpha}_p = \int_{S_p} \boldsymbol{\omega} d\mathbf{x} = \omega_p S_p$. Therefore, over a particle surface, one can obtain from Eq. 1.1 the system which has to be integrated,

$$\frac{d}{dt}\begin{bmatrix} \mathbf{x}_p \\ \boldsymbol{\alpha}_p \end{bmatrix} = \begin{bmatrix} \mathbf{u}_p \\ \nu(\nabla^2 \boldsymbol{\omega})_p S_p \end{bmatrix} \tag{1.5}$$

Here is the main interest of the VPM method: the right-hand-side of Eq. 1.5 is evaluated on the mesh, as well as all the other operator computations, which therefore improves the global efficiency. The interpolation of the particles to the mesh is done using high order interpolation formulas [24].

Over an uniform mesh on which the particle quantities have been interpolated, the differential operators of the system 1.5 can be calculated by means of standard finite differences. In this work, all the spatial derivatives are centered finite differences of second order.

Thus, this mesh is also used to solve the very costly Poisson problem (Eq. 1.4), through highly efficient FFT-based solvers [25], [26].

Once these computations have been processed, a new interpolation from the mesh to the particles is done in order to properly integrate Eq. 1.5 in time. The time integration scheme is therefore applied onto the particles and in this work, it consists in a standard Runge-Kutta scheme of third order. More precisely, it is the low-storage RK3 scheme suggested by Williamson [28].

In terms of stability, the time step is firstly constrained by the diffusion through the Fourier number, that being

$$r = \frac{\nu \Delta t}{h^2} < r_{max}$$

where r_{max} is defined for the given time and space discretization.

Then two criterions measure the accuracy condition for the convection of particles, on basis of Lagrangian deformation,

$$\Delta t \left\|\mathbf{S}\right\|_1 \leqslant C_{LCFL,S} \qquad \text{and} \qquad \Delta t \left\|\boldsymbol{\omega}\right\|_1 \leqslant C_{LCFL,\omega} \tag{1.6}$$

where the strain tensor is defined as $\mathbf{S} = \left[\nabla \mathbf{u} + (\nabla \mathbf{u})^t\right]$ and the constants are fixed for the simulations such as $C_{LCFL,S} \simeq C_{LCFL,\omega} \simeq 0.15$, which are typical values [19].

The VPM method also applies a procedure of remeshing [16], [17] in order to prevent the particles from clustering and depletion, which are consequences of the Lagrangian deformation. Briefly, the remeshing process consists in the periodic redistribution of the particles set onto the mesh.

One can refer to [26] for a more detailed and extensive description of the VPM framework and its implementation for large parallel architectures.

1.1.2 Brinkman penalization

Brinkman penalization methods [29] implant the obstacle Ω_b in the fluid domain and enforce the velocity inside the object, that imposing $\mathbf{u} = \mathbf{u}_b$, with $\mathbf{u}_b$ the velocity of the immersed obstacle. Thus it adds a constraint to the Navier-Stokes equations, consisting in the addition of a penalized velocity term. This term actually behaves as a Lagrangian relaxation and therefore, deletes undesirable velocity inside the obstacle to properly set $\mathbf{u} = \mathbf{u}_b$.

Since the constraint is on the velocity, the Brinkman methods are naturally applied to pressure-velocity formulation of the momentum Navier-Stokes equation [18] (Eq. 1.7).

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = -\nabla P + \nu \nabla^2 \mathbf{u} + \lambda \chi (\mathbf{u}_b - \mathbf{u}) \tag{1.7}$$

where λ is the Lagrangian multiplier and χ the mask function such as

$$\chi = \begin{cases} 1 & \text{if } \mathbf{x} \in \Omega_b \\ 0 & \text{otherwise} \end{cases}$$

One can obtain the 2-dimension vorticity-velocity formulation by applying the curl operator to Eq. 1.7, as follows

$$\frac{\partial \omega}{\partial t} + \mathbf{u} \cdot \nabla \omega = \nu \nabla^2 \omega + \lambda \nabla \times [\chi (\mathbf{u}_b - \mathbf{u})] \tag{1.8}$$

Hence, the Brinkman penalization methods through the VPM framework consists in the resolution of the following set of equations for each particle

$$\frac{d}{dt} \begin{bmatrix} \mathbf{x}_p \\ \omega_p \end{bmatrix} = \begin{bmatrix} \mathbf{u} \\ \nu(\nabla^2 \omega) + \lambda \nabla \times [\chi (\mathbf{u}_b - \mathbf{u})] \end{bmatrix}_p \tag{1.9}$$

The VPM method is then applied as it is described in the previous subsection. Since the domain considered in this document is unbounded, one can refer to [27] that proposes the Hockney–Eastwood algorithm which works on the Fourier transform of the zero-padded doubled-domain source field [18] in order to accurately solve the Poisson problem (Eq. 1.4).

1.2 Time step algorithm

This section presents an overview of the computation to be processed during one time step for the VPM methodology. It can also be seen as a sub-step of the multi-step time integration scheme used in this work, that being RK3.

FIGURE 1.1 shows the procedure of one time step (from n to $n+1$) and these are the comments associated to this scheme:

1. The particles (characterized by a position $\mathbf{x}_p^n$ and a vorticity ω_p^n) are convected and the vorticity ω_p^n is interpolated on the mesh (or remeshing process is performed), and its result is then marked ω^n.

$$\mathbf{x}_p^{n+1} \longrightarrow \mathbf{x}_p^n \qquad \omega_p^{n+1} \longrightarrow \omega_p^n$$

$$\omega_p^n \longrightarrow \omega^n$$

2. Vorticity is diffused onto the mesh (here presented using the Euler scheme)
$$\tilde{\omega}^{n+1} = \omega^n + \Delta t \left[\nu \nabla^2 \omega^n\right]$$
3. Computation of the unpenalized velocity field $\tilde{\mathbf{u}}^{n+1}$ on the mesh, by solving
$$\nabla^2 \tilde{\mathbf{u}}^{n+1} = -\nabla \times \tilde{\omega}^{n+1}$$
4. On the mesh, calculation of the correction vorticity $\delta\omega$,
$$\delta\omega = \mathcal{F}(\mathbf{u}_b, \tilde{\mathbf{u}}^{n+1})$$
with $\mathcal{F}$ the function performing the desired penalization technique.
5. Vorticity is updated on the mesh (a). Eq. (b) is solved to obtain the penalized velocity $\mathbf{u}^{n+1}$ and to properly convect the particles (c). Finally, vorticity is interpolated on the particles (d).
$$\omega^{n+1} = \tilde{\omega}^{n+1} + \delta\omega \quad \text{(a)} \qquad \nabla^2 \mathbf{u}^{n+1} = -\nabla \times \omega^{n+1} \quad \text{(b)}$$
$$\mathbf{x}_p^n \xrightarrow{\mathbf{u}^{n+1}} \mathbf{x}_p^{n+1} \quad \text{(c)} \qquad \omega^{n+1} \longrightarrow \omega_p^{n+1} \quad \text{(d)}$$

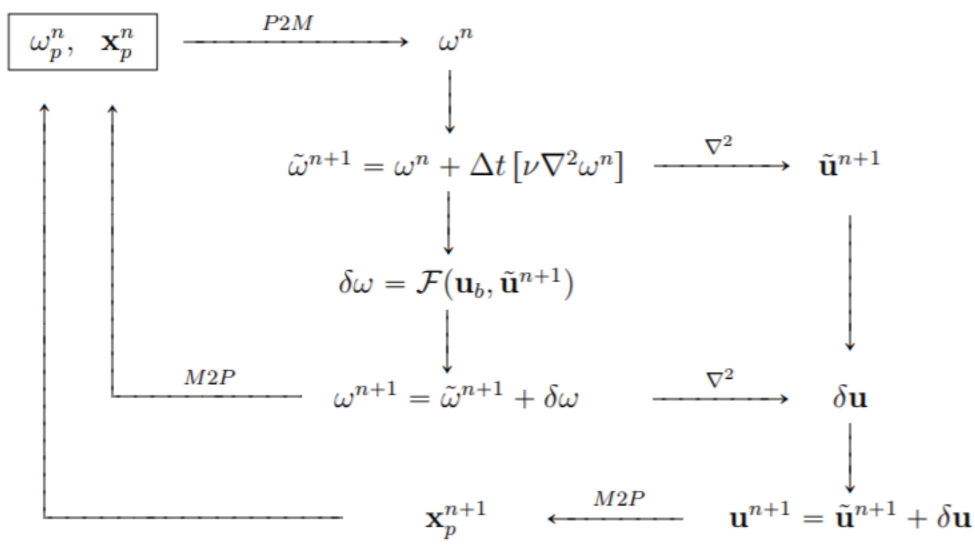

Figure 1.1: $P2M$: Interpolation from particles to the mesh
$M2P$: Interpolation from the mesh to particles
∇^2: Resolution of the Poisson equation (Eq.1.4)

1.3 Penalization techniques

This part focuses on the penalization function $\mathcal{F}$ introduced in the previous section. In other words, it is about the correction of vorticity which allows to take into account the immersed obstacle, that being

$$\delta\omega = \mathcal{F}(\mathbf{u}_b, \tilde{\mathbf{u}}^{n+1})$$

Three distinct numerical methods, subject of comparison in the next chapters, are here developed, explained and detailed. For the sake of ease, only the additional penalized term is treated, the Navier-Stokes model being separately processed according the integration scheme previously presented.

Therefore, the following subsections take basis on these equations

$$\frac{\partial \mathbf{u}}{\partial t} = \lambda\chi(\mathbf{u}_b - \mathbf{u}) \tag{1.10}$$

and the latter with the curl operator applied,

$$\frac{\partial \omega}{\partial t} = \nabla \times [\lambda\chi(\mathbf{u}_b - \mathbf{u})] \tag{1.11}$$

1.3.1 Implicit approach on velocity

In the implicit approach on velocity, the equation 1.10 is thus integrated according to the implicit Euler scheme, as follows

$$\mathbf{u}^{n+1} = \mathbf{u}^n + \Delta t \left[\lambda\chi(\mathbf{u}_b - \mathbf{u})\right]^{n+1}$$

That can be reduced to,

$$\tilde{\mathbf{u}}^{n+1} = \frac{\mathbf{u}^n + \lambda\Delta t\chi^{n+1}\mathbf{u}_b^{n+1}}{1 + \lambda\Delta t\chi^{n+1}} \tag{1.12}$$

where $\tilde{\mathbf{u}}^{n+1}$ is an intermediate velocity from which the vorticity ω^{n+1} is calculated. Since a fixed immersed body is considered in this work, one can shorten the notation $\chi^{n+1} = \chi$ and $\mathbf{u}_b^{n+1} = \mathbf{u}_b$.

Since the implicit Euler scheme is unconditionally stable, an arbitrary high value could be chosen for the Lagrangian factor λ, which could therefore enhance the consideration in the flow of the obstacle. However, two major drawbacks can be noted, as cited by T. Gillis et al. [18]:

- Significant diffusion errors are generated, as reported by Rasmussen et al. [31];
- According to Hejlesen et al. [22], it does not provide a global relation between vorticity and velocity which leads to a drastically decreasing time step in order to provide accurate results.

This method consists therefore in applying the curl operator to Eq. 1.12,

$$\begin{aligned}
\omega^{n+1} &= \nabla \times \tilde{\mathbf{u}}^{n+1} \\
&= \nabla \times \left(\frac{\mathbf{u}^n + \lambda \Delta t \chi \mathbf{u}_b}{1 + \lambda \Delta t \chi} \right) \\
&= \frac{\nabla \times \mathbf{u}^n + \lambda \Delta t \nabla \times (\chi \mathbf{u}_b)}{1 + \lambda \Delta t \chi} + \left[1 + \lambda \Delta t \nabla \chi\right]^{-1} \times (\mathbf{u}^n + \lambda \Delta t \chi \mathbf{u}_b) \\
&= \frac{\omega^n + \lambda \Delta t \nabla \times (\chi \mathbf{u}_b)}{1 + \lambda \Delta t \chi} + \left[1 + \lambda \Delta t (\nabla \chi)\right]^{-1} \times (\mathbf{u}^n + \lambda \Delta t \chi \mathbf{u}_b)
\end{aligned}$$

Hence, one has to compute the following equation

$$\omega^{n+1} = \frac{\omega^n + \lambda \Delta t \nabla \times (\chi \mathbf{u}_b)}{1 + \lambda \Delta t \chi} + \left[1 + \lambda \Delta t (\nabla \chi)\right]^{-1} \times (\mathbf{u}^n + \lambda \Delta t \chi \mathbf{u}_b) \qquad (1.13)$$

where the spatial derivatives are performed as finite centered differences of second order.

One can easily derive the error done by comparing the result of equation 1.13 to the result of the perfect implicit integration of equation 1.11, that being

$$\begin{aligned}
\omega^{n+1} &= \omega^n + \lambda \Delta t \left[\nabla \times (\chi(\mathbf{u}_b - \mathbf{u}^{n+1}))\right] \\
&= \omega^n + \lambda \Delta t \left[\chi(\nabla \times (\mathbf{u}_b - \mathbf{u}^{n+1})) + (\nabla \chi) \times (\mathbf{u}_b - \mathbf{u}^{n+1})\right] \\
&= \omega^n + \lambda \Delta t \left[-\chi \omega^{n+1} + \chi \nabla \times \mathbf{u}_b + (\nabla \chi) \times (\mathbf{u}_b - \mathbf{u}^{n+1})\right]
\end{aligned}$$

That being reformulated,

$$\omega^{n+1} = \frac{\omega^n + \lambda \Delta t \left[\nabla \chi \times (\mathbf{u}_b - \mathbf{u}^{n+1}) + \chi \nabla \times \mathbf{u}_b\right]}{1 + \lambda \Delta t \chi} \qquad (1.14)$$

Therefore, the error e_ω on this implicit approach is the difference between equations 1.13 and 1.14,

$$e_\omega = \left(\frac{1 + \lambda \Delta t \chi}{1 + \lambda \Delta t (\nabla \chi)} \right) \times (\mathbf{u}^n + \lambda \Delta t \chi \mathbf{u}_b) - \nabla \chi \times \mathbf{u}^{n+1}$$

1.3.2 Implicit approach on vorticity

Let us take back the equation 1.14 which implicitly integrates the vorticity

$$\omega^{n+1} = \frac{\omega^n + \lambda\Delta t\left[\nabla\chi \times (\mathbf{u}_b - \mathbf{u}^{n+1}) + \chi\nabla \times \mathbf{u}_b\right]}{1 + \lambda\Delta t\chi} \tag{1.15}$$

Since the term $(\nabla\chi \times \mathbf{u}^{n+1})$ is unknown a priori, it has to be neglected.

One can notice that, since the considered body has no velocity $\mathbf{u}_b = 0$ in the cases discussed in the next chapters, the Eq. 1.15 is therefore reduced to

$$\omega^{n+1} = \frac{\omega^n}{1 + \lambda\Delta t\chi} \tag{1.16}$$

However, since the vorticity is initialized to zero on the whole computational domain, integrating Eq. 1.16 actually leads to a zero vorticity, then it also implies a zero diffusion, etc... Therefore, the vorticity field would remain zero in time and space by starting with a zero vorticity. This is why we cannot afford to pose negligible the term $(\nabla\chi \times \mathbf{u}^{n+1})$. Therefore, we use the implicit scheme, Eq. 1.12, in the velocity formulation to substitute $\mathbf{u}^{n+1}$ for the establishment of the implicit approach on vorticity. It finally results to

$$\omega^{n+1} = \frac{\omega^n + \lambda\Delta t\left[\nabla\chi \times (\frac{-\mathbf{u}^n}{1+\lambda\Delta t\chi})\right]}{1 + \lambda\Delta t\chi} \tag{1.17}$$

1.3.3 Iterative method proposed by Hejlesen

The Hejlesen method of penalization can be explained as follows: at time step n, from a given non-penalized vorticity field $\tilde{\omega}^{n+1}$ and its corresponding non-penalized velocity field $\tilde{\mathbf{u}}^{n+1}$, this iterative penalization method aims at finding the vorticity correction $\delta\omega$, which generates the velocity correction, $\delta\mathbf{u}$, such that $\tilde{\mathbf{u}}^{n+1} + \delta\mathbf{u} = \mathbf{u}^{n+1} = \mathbf{u}_b$.

In this method, the vorticity correction is thus iterated by solving a linear system such as

$$A(\delta\omega\mathbf{e_z}) = \nabla \times \left[\chi(\mathbf{u}_b - \tilde{\mathbf{u}}^{n+1})\right] \tag{1.18}$$

where the linear operator A is defined as

$$A : \delta\omega_k \rightarrow \nabla \times [\chi\delta\mathbf{u}_k] = \nabla \times \left[\chi\left(\left(\nabla^2\right)^{-1}(-\nabla \times (\delta\omega_k\mathbf{e}_z))\right)\right] \tag{1.19}$$

Several algorithms allow the resolution of this linear system, two of them will be introduced: the Jacobi method and the Generalized Minimal Residual (GMRes).

Jacobi method

In fact, the iterative method suggested by Hejlesen et al. [22] is nothing but an over-relaxed Jacobi algorithm. Hence, the stability constraint that those authors refer can be interpreted as the well known stability constraint for that type of methods, that being

$$0 < \alpha = \lambda \Delta t < 2 \tag{1.20}$$

Obviously, since lies the resolution of the Poisson problem in the operator A (Eq. 1.19), the matrix-vector product ($A\delta\omega_k$) constitutes an expensive step at each iteration. In order to decrease that cost, Hejlesen et al. [22] reduce the domain on which the penalization method is computed. As a FFT-solver is used for the Poisson equation, the domain has to be reduced in the unbounded direction [25]. Despite the decrease of the computational and memory cost, the Jacobi method remains a greedy algorithm compared to more advanced ones. Therefore, here is presented an other algorithm, the GMRes method, whose the main interest is the reduction of the number of iterations needed to reach convergence.

Algorithm: Jacobi(α) - Hejlesen method

Given: $\tilde{\omega}^{n+1}$, $\tilde{\mathbf{u}}^{n+1}$
Set: $\delta\omega_0 = 0$; $\delta\mathbf{u}_0 = 0$
Compute: $b = \nabla \times [\chi(\mathbf{u}_b - \tilde{\mathbf{u}}^{n+1})]$; $r_0 = b$

for $k = 1, 2, ...$ **do**

$$\begin{aligned} \delta\omega_k &= (1-\alpha)\delta\omega_{k-1} + \alpha r_{k-1} \\ \nabla^2(\delta\mathbf{u}_k) &= -\nabla \times (\delta\omega_k \mathbf{e_z}) \\ r_k &= \nabla \times [\chi(\mathbf{u}_b - (\tilde{\mathbf{u}}^{n+1} + \delta\mathbf{u}_k))] \end{aligned}$$

end
$\omega^{n+1} = \tilde{\omega}^{n+1} + \delta\omega$

GMRes method

The generalized minimal residual method (GMRES) is an iterative method for the numerical solution (of a nonsymmetric system, as considered in our case) of linear equations. The method approximates the solution by the vector in a Krylov subspace with minimal residual [18].

One can use the PETSc library, which is a suite of data structures and routines for the scalable (parallel) solution of scientific applications modeled by partial differential equations [32]. Hence, it thus offers a suitable function of GMRes in our case.

Convergence criterion for iterative penalization method

Obviously, those presented algorithms need a stopping criterion, that being an estimator of the solution quality. In Hejlesen et al. [22], a way to compute the error at each iteration as the correction vorticity field's first moment integral (that being the impulse) is suggested. According to T. Gillis et al. [18], this stopping criterion is however not general as it allows spurious solutions, which are even-symmetric with respect to the inflow direction.

Therefore, still as referred to T. Gillis et al. [18], a dimensionless penalization residual is employed. At a given iteration, k, it is computed as

$$E_2^k(\mathbf{r}_k) = \frac{1}{\omega_\infty}\sqrt{\frac{1}{S_{\Omega_b}}\,\|\mathbf{r}_k\|_2^2} \tag{1.21}$$

where $\omega_\infty = U_\infty/L_{\Omega_b}$, S_{Ω_b} is the obstacle reference surface, L_{Ω_b} the obstacle reference length and the 2-norm is processed as $\|a\|_2^2 = \sum_{i,j} a_{i,j}^2 h^2$.

The residual $\mathbf{r}_k$ is defined as

$$\mathbf{r}_k = \nabla \times \left[\chi(\mathbf{u}_b - (\tilde{\mathbf{u}}^{n+1} + \delta\mathbf{u}_k))\right] \tag{1.22}$$

1.4 Tools used for the simulations

Here are exposed the different tools which have enabled the simulations of this thesis. Among all those tools, the way that the forces are evaluated is briefly introduced.

1.4.1 Global computation

The computation is done in **C++** language. All the calculations are based on the 2D Vortex Particle-Mesh framework which has been provided by the sector of thermodynamics and fluid mechanics of the *Université catholique de Louvain* (UCLouvain). Several numerical libraries are used, among them, PETSc has already been reported.

Although most of the simulations in this work have been performed on a private computer, some of them have been processed thanks to CÉCI, a Consortium providing a powerful computing equipment (clusters) in Belgium.

All the post-processing works are computed with the software Matlab. It is also employed for the visualization of the numerical results.

1.4.2 Force evaluation

In order to compute the forces exerted by the fluid on the immersed obstacle, one can use the momentum formulation from Noca et al. [30], as it is also suggested by Mimeau et al. [29] and used by T. Gillis et al. [18]: a control volume Ω_c containing the obstacle is considered, such that

$$\mathbf{F} = -\frac{d}{dt}\int_{\Omega_c} \mathbf{u} d\mathbf{x} + \int_{\partial\Omega_c} \boldsymbol{\gamma} \cdot \hat{\mathbf{n}} d\mathbf{x} \tag{1.23}$$

where $\hat{\mathbf{n}}$ is the normal to $S = \partial\Omega_c$ and $\boldsymbol{\gamma}$ is defined as

$$\begin{aligned} \boldsymbol{\gamma} \;=\; & \frac{1}{2}\left|\mathbf{u}\right|^2 \mathbf{I} - \frac{1}{N-1}\mathbf{u}(\mathbf{x}\times\boldsymbol{\omega}) + \frac{1}{N-1}\boldsymbol{\omega}(\mathbf{x}\times\mathbf{u}) \\ & -\frac{1}{N-1}\left[\left(\mathbf{x}\cdot\frac{\partial\mathbf{u}}{\partial t}\right)\mathbf{I} - \mathbf{x}\frac{\partial\mathbf{u}}{\partial t}\right] \\ & +\frac{1}{N-1}\left[(\mathbf{x}\cdot(\nabla\cdot\mathbf{T}))\mathbf{I} - \mathbf{x}(\nabla\cdot\mathbf{T})\right] + \mathbf{T} \end{aligned} \tag{1.24}$$

with N the spatial dimension. $\mathbf{I}$ is the unit tensor and $\mathbf{T} = \mu\left[\nabla\mathbf{u} + (\nabla\mathbf{u})^T\right]$ is the viscous stress tensor. In our case, $N = 2$ and Ω_c is a rectangular box that contains the body Ω_b.

Chapter 2

Parametric study on the Brinkman penalization term

The second chapter of this thesis focuses on the penalized velocity term, specific to the Brinkman penalization methods. The aim of this chapter is actually to provide a better understanding of this additional velocity term alone and also its potential impact on the Navier-Stokes equations through the three considered VPM methods.

A short recall on the potential flow in the case of a flow past a circular cylinder is firstly done. Then a parametric study on the additional penalized velocity term, referred as a Lagrangian relaxation, is introduced and performed for several values of the Lagrangian multiplier λ and mesh sizes.

Contents

2.1 Potential flow around a circular cylinder

In this section, the analytic solution of a flow past a circular cylinder is recalled, that being the perfect case of flow without any drag (the lift is zero since it is an immersed symmetric geometry), also called a potential flow.

The free flow velocity is marked $\mathbf{U}_\infty$ and is defined such that $\lim_{|\mathbf{x}|\to\infty} \mathbf{u} = \mathbf{U}_\infty$. FIGURE 2.1 shows an immersed cylinder of a radius $R = 0.2$ in a flow at $\mathbf{U}_\infty = [U_\infty, 0]^T$ and centered in $(x_0, y_0) = (0.3{,}0.5)$. One can use the complex variable z in order to compute this potential flow $f(z)$, since it actually corresponds to the superposition of a horizontal uniform flow and a horizontal dipole,

$$f(z) = U_\infty \left(z + \frac{R^2}{z} \right) \tag{2.1}$$

The velocity field $\mathbf{u} = [u, v]^T$ is derived as follows

$$\overline{w} = \frac{df}{dz}(z) = u(z) - iv(z) \tag{2.2}$$

That being in Cartesian variables (x, y),

$$\begin{aligned} u &= U_\infty \left(1 - R^2 \frac{(x^2 - y^2)}{(x^2 + y^2)^2} \right) \\ v &= -U_\infty R^2 \frac{2xy}{(x^2 + y^2)^2} \end{aligned} \tag{2.3}$$

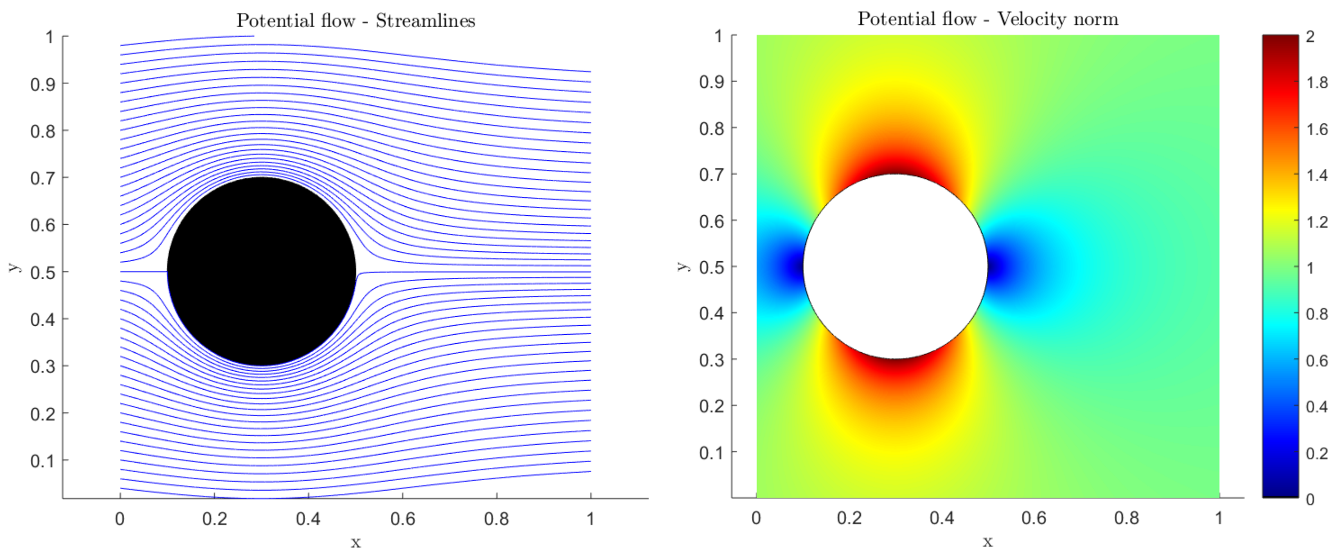

Figure 2.1: Potential flow around a circular cylinder

One can refer to the course *LMECA2322: Fluid mechanics and transfers 2* to have an extensive development of Eq. 2.1, 2.2 and the way they are built and derived [33].

2.2 Parametric analysis

This section reports the parametric study of the additional penalized velocity term, in its vorticity-velocity formulation. That steady partial differential equation 2.4 provides numerical results given by its integration according to the VPM method, those ones being then compared to the analytic solution, the potential flow previously presented.

$$\frac{\partial \omega}{\partial t} = \nabla \times [\lambda \chi (\mathbf{u}_b - \mathbf{u})] \tag{2.4}$$

Unless it is explicitly provided, the time step Δt used is such that $\Delta t = 10^{-6}$ and the computational domain is a regular 512×512 mesh defined on $(x, y) \times (x, y) = [0, 1] \times [0, 1]$.

2.2.1 Lagrangian relaxation

As already mentioned, the Lagrangian relaxation is a relaxation technique that consists in removing difficult constraints by integrating them into the objective function by penalizing it if these constraints are not respected. This additional term is thus presented in Eq. 2.4 where λ is the Lagrangian multiplier.

In order to clearly observe the impact of this relaxation factor λ, here are shown on Figures 2.2, 2.3, 2.4 and 2.5, the streamlines and velocity profiles resulting from the integration of Eq. 2.4 by the explicit Euler scheme, that being

$$\omega^{n+1} = \omega^n + \lambda \Delta t \nabla \times [\chi(\mathbf{u}_b - \mathbf{u}^n)] \tag{2.5}$$

for which the stability is ensured when using $0 < \lambda \Delta t < 2$.

The explicit Euler scheme is actually the non-iterative version of the method proposed by Hejlesen et al [22], which has therefore the same stability constraint. Using this explicit scheme to illustrate the effect of the λ factor is thus intrinsically consistent as part of our comparison.

Hence, as exposed on Figures 2.2, 2.3, 2.4 and 2.5, using higher values of λ allows to better take into account the body in the flow. One can note that, except for the case $\lambda \Delta t = 2$, the cylinder is increasingly cut by streamlines for decreasing values of $\lambda \Delta t$. This is actually the inherent problem of the non-iterative penalization method: it fails to deflect the flow field around the cylinder, whereas the flow penetrates the solid boundary and remains active within the penalized region. The velocity profiles thus only approach the perfect case of potential flow for $\lambda\Delta t = 2$.

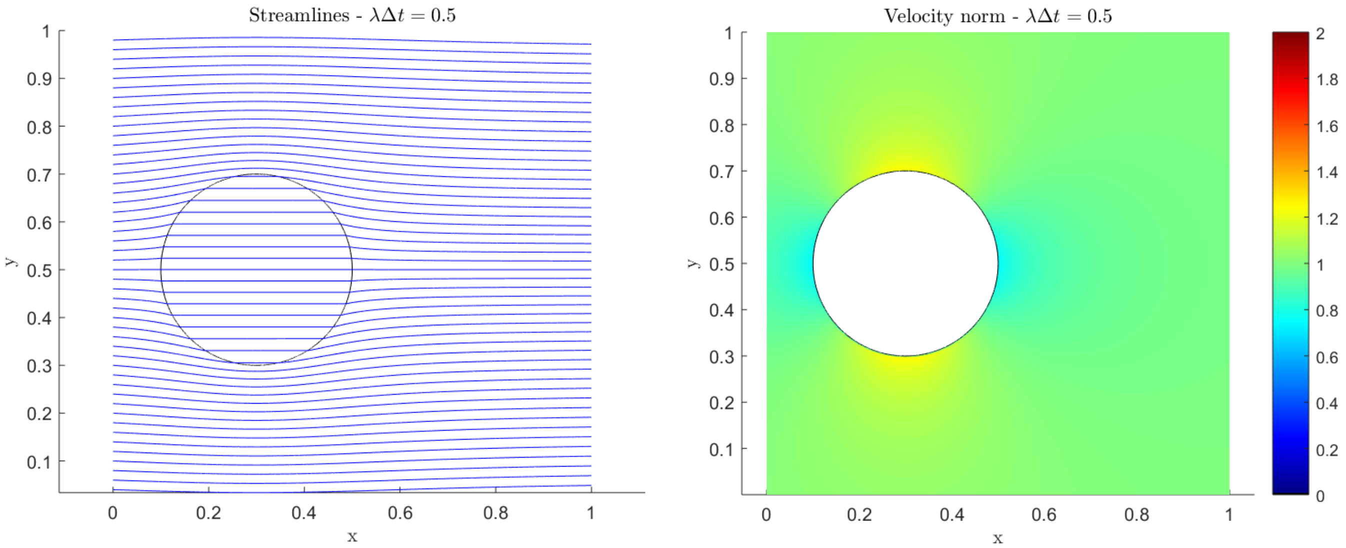

Figure 2.2: Streamlines and velocity profile for $\lambda\Delta t = 0.5$

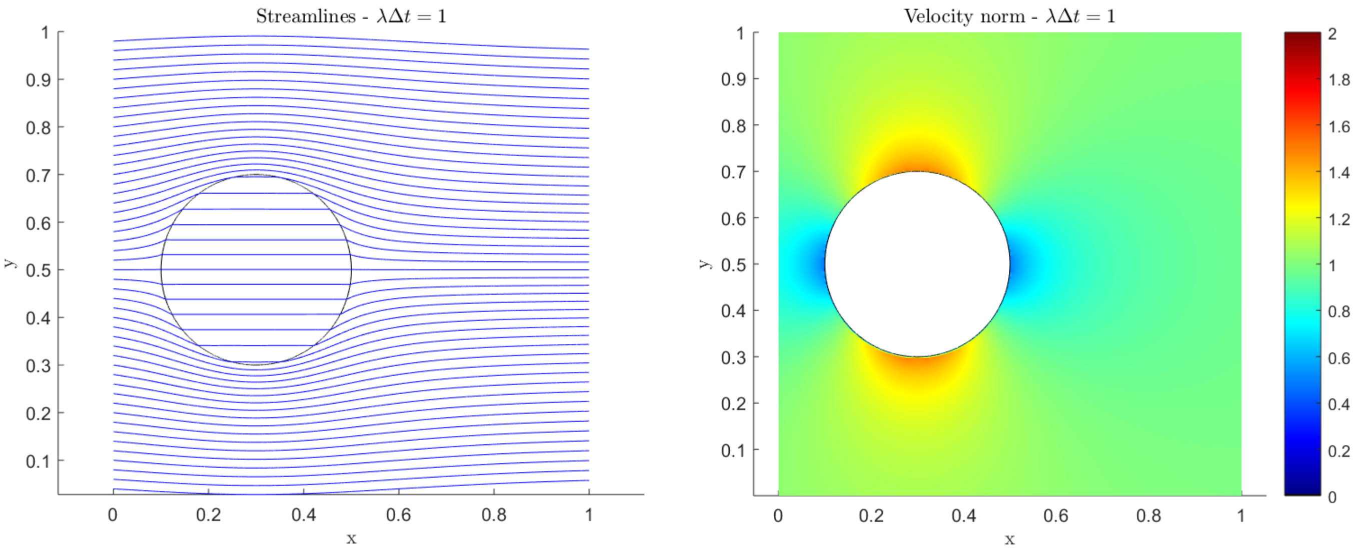

Figure 2.3: Streamlines and velocity profile for $\lambda\Delta t = 1$

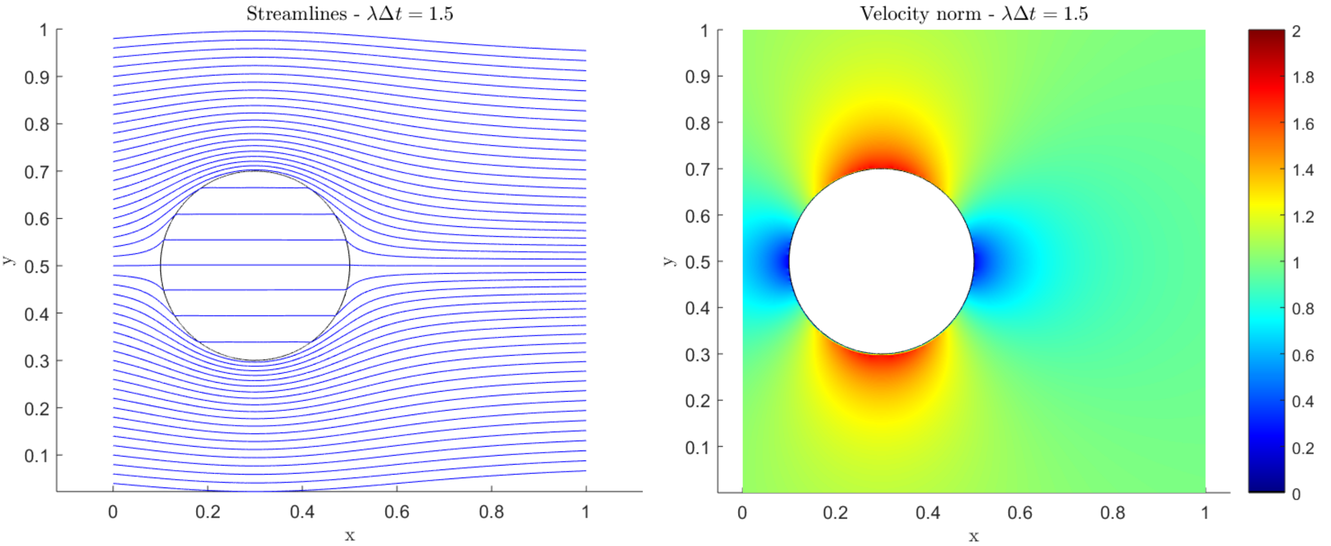

Figure 2.4: Streamlines and velocity profile for $\lambda\Delta t = 1.5$

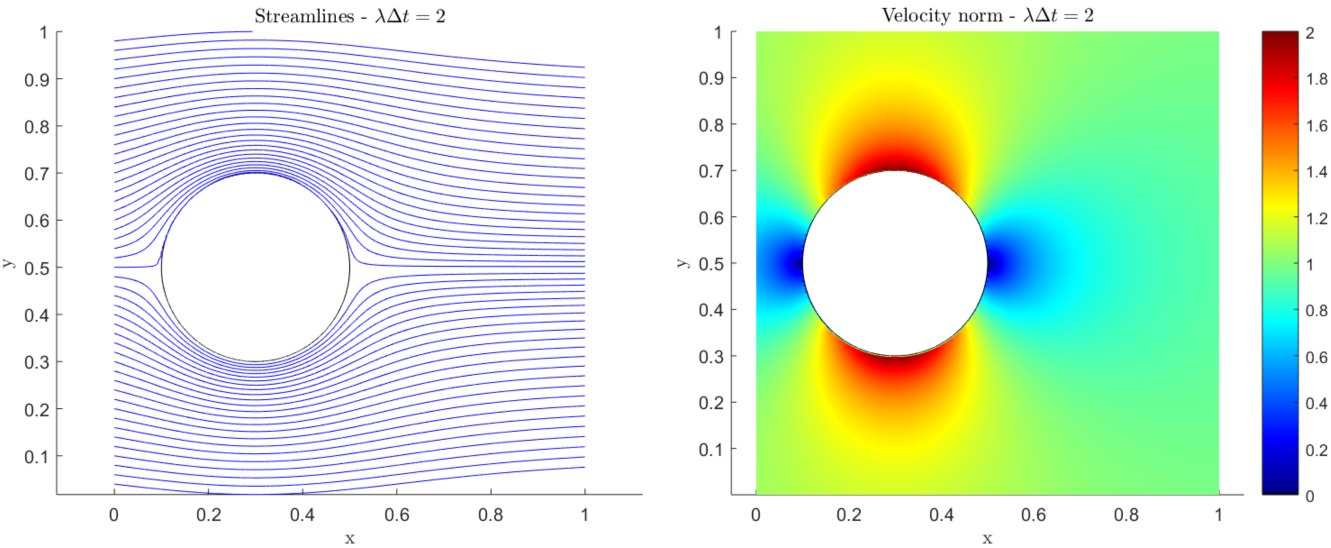

Figure 2.5: Streamlines and velocity profile for $\lambda\Delta t = 2$

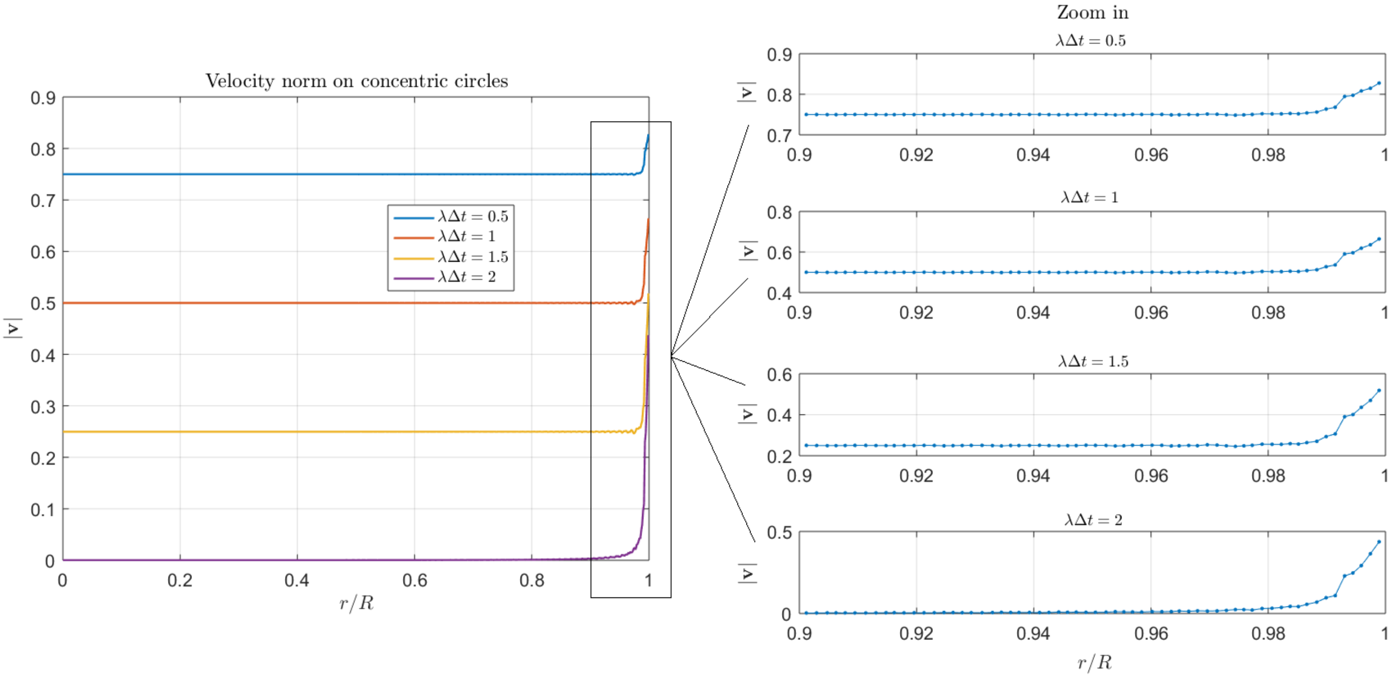

Figure 2.6: Velocity error on concentric circles

The FIGURE 2.6 confirms what has been previously explained, by calculating the mean velocity norm error done on concentric (to the immersed cylinder) circles. One can note a constant velocity norm on 90% of the radius R until it goes up on the remaining 10%. This visually corresponds to the straight cutting streamlines across the body. The increasing velocity on the last 10% can actually be interpreted as a boundary layer appearing inside the immersed obstacle, as referred by Carbou [11].

2.3 Application to the implicit VPM methods

In terms of stability, since the two first considered numerical methods are unconditionally stable (the *implicit approaches on the velocity and vorticity*), one can use any values of λ. However, as it is explained in the previous chapter, it has been shown that too high values of λ imply substantial diffusion errors [18]. The choice of λ has thus to be high enough to rightly take into account the obstacle and sufficiently low to avoid diffusion error.

The third method, the one proposed by Hejlesen et al. [22], is constrained to ensure its stability, as seen in the previous section. In fact, the numerical results provided by the latter are only used in order to evaluate how accurate are the two implicit VPM methods compared to an iterative one for which the error can be tuned.

Ideally, at the first iteration before any diffusion operation, the results of the two implicit methods should approach the fields of the corresponding potential flow, just as the Hejlesen method approximately does [22], and thus ensure a total deflection of the flow around the solid boundaries of the cylinder.

Thus, this section presents several graphical results for each method, and finishes on a brief comparison between them.

2.3.1 Implicit approach on velocity

As shown on FIGURE 2.7, the streamlines generated by the *implicit approach on velocity* get close to the potential flow from the third iteration for the relaxation factor $\lambda\Delta t = 1000$. It remains a non-negligible residual velocity inside the body at the first iteration which, even if the method finally succeeds at deflecting the flow around the cylinder, has an accelerating effect on the downstream flow being diffused since it has taken a shortcut via the inside of the obstacle, as it is mentioned in [22] and verified in the last chapter.

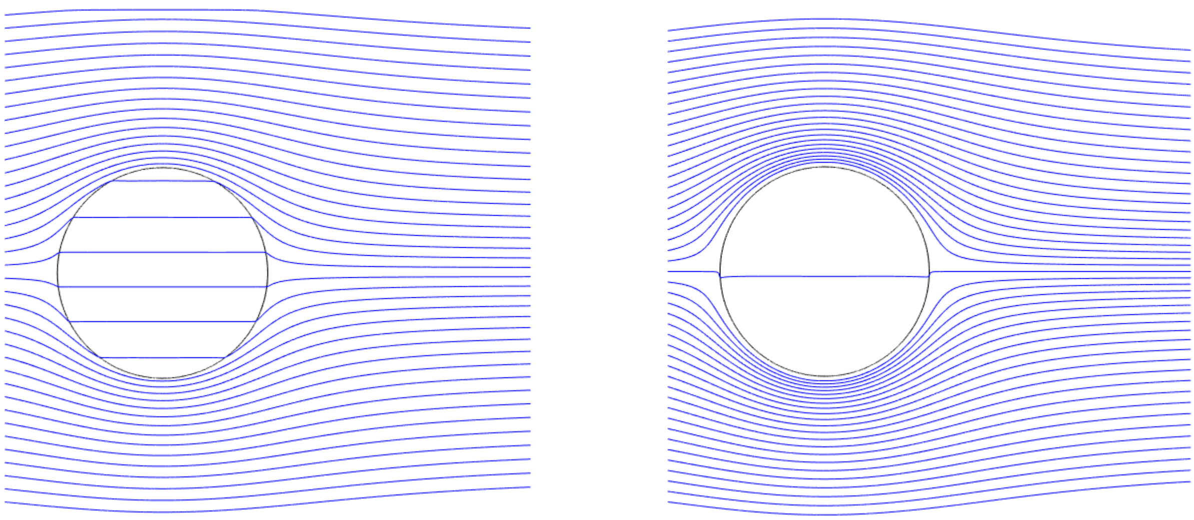

Figure 2.7: Streamlines generated at the first (left) and third (right) iteration for $\lambda\Delta t = 10^3$

Nevertheless, the value of the Lagrangian multiplier $\lambda\Delta t = 1000$ seems to be a reasonable choice since it leads to an error on the velocity inside the body lower than 0.3% compared to the curve with a relaxation factor of 10^5, as testified the FIGURE 2.8. In fact, the FIGURE 2.8 exposes the velocity profile on the vertical surface across the center of the cylinder, on which we can note the same kind of boundary layer inside the obstacle discussed in the previous section.

The FIGURE 2.9, which presents the mean velocity norm on concentric circles to the cylinder, also confirms our choice of $\lambda\Delta t = 1000$ since its curve overlaps the one of $\lambda\Delta t = 10^8$, that corresponding to a difference lower than 0.5%. One can compare this graph with the FIGURE 2.6, representing the same data for the explicit method, and notice that the boundary layer starts at a lower radius (≈55%) and is noisier in the implicit case. This is actually due to the number of iterations performed: the curves of the explicit case have been generated after several hundreds of iterations while the implicit approach reports the third iteration for which the fields are not as much defined.

The velocity profiles in a vertical section across the center of the cylinder for different spatial step, corresponding to the mesh size $n = \{64,\ 128,\ 256,\ 512\}$, are shown on FIGURE 2.10. As one could expect, we get better results for a lower spatial step, that being for the mesh size $n = 512$. The maximum velocity error done with $n = 512$ is actually more than 200% lower than in the case $n = 256$.

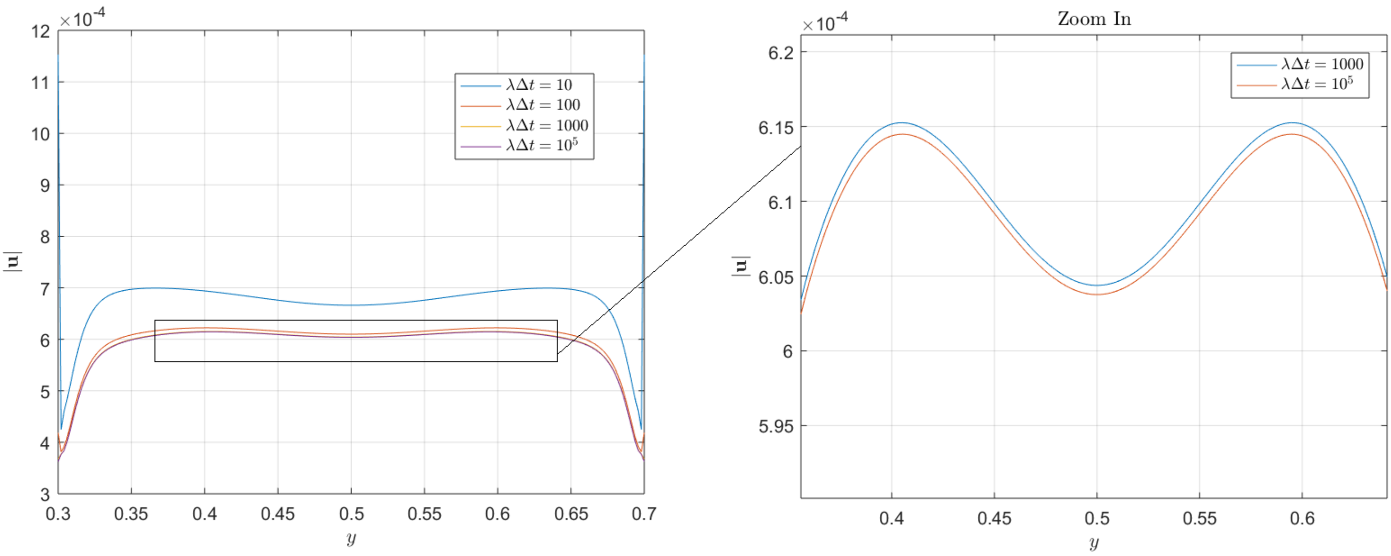

Figure 2.8: Velocity profiles in a section across the center of the cylinder, with the *implicit approach on velocity* for $\lambda\Delta t = \{10,\ 100,\ 1000,\ 10^5\}$

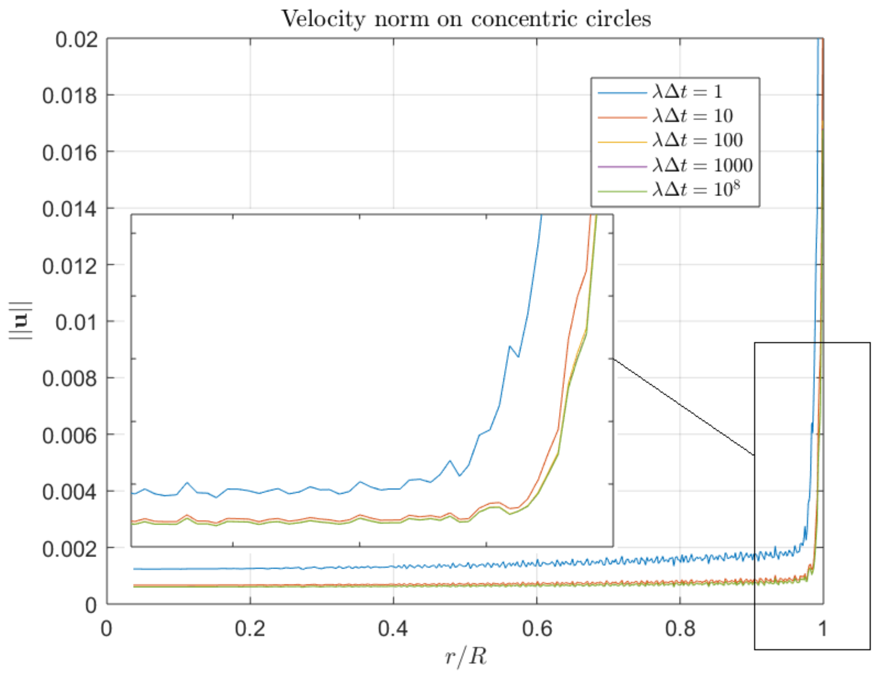

Figure 2.9: Mean velocity norm on concentric circles to the cylinder for $\lambda\Delta t = \{1,\ 10,\ 100,\ 1000,\ 10^8\}$, with the *implicit approach on velocity*

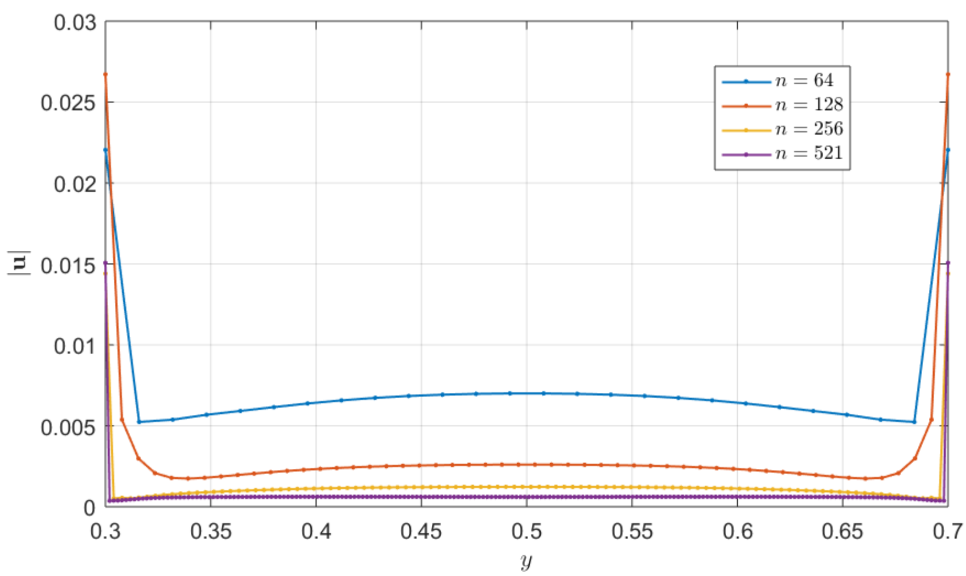

Figure 2.10: Velocity profiles in a section across the center of the cylinder, with the *implicit approach on velocity* for the mesh size $n = \{64,\ 128,\ 256,\ 512\}$

2.3.2 Implicit approach on vorticity

Globally, one can notice the same behavior for the *implicit approaches on vorticity* and *velocity* in terms of sensitivity to the relaxation factor $\lambda\Delta t$ presented by the Figure 2.11. Thus, all the comments done in the previous subsection are also valuable here.

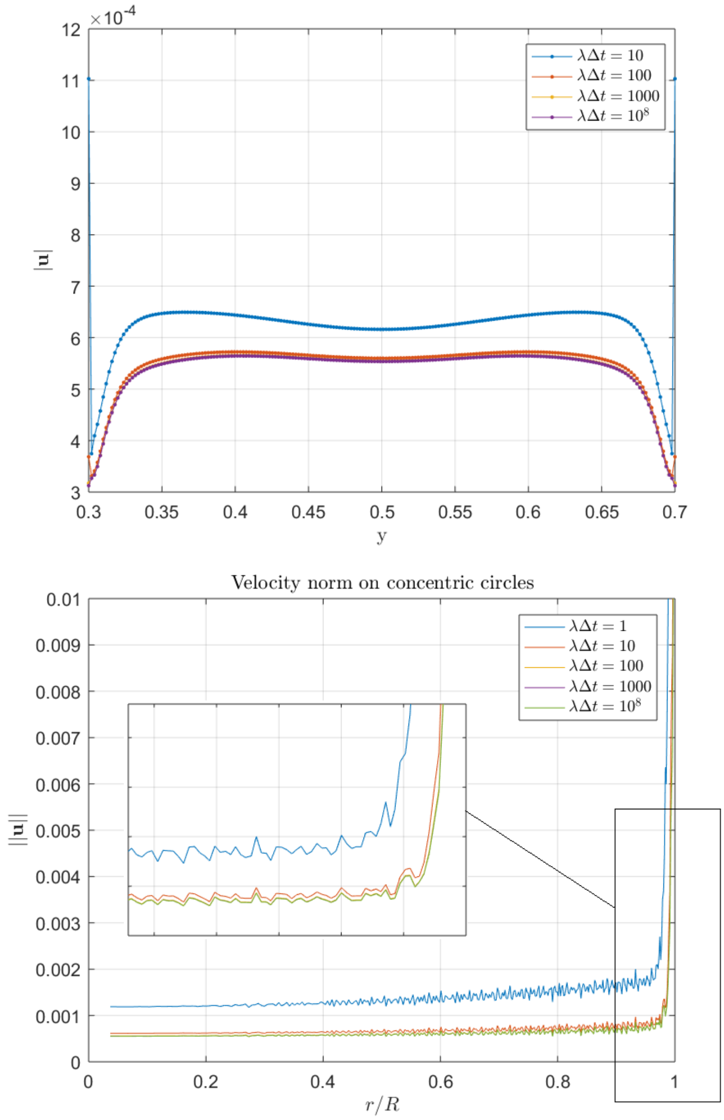

Figure 2.11: Velocity profiles in a section across the center of the cylinder, with the *implicit approach on vorticity* for $\lambda\Delta t = \{10,\ 100,\ 1000,\ 10^8\}$ - Mean velocity norm on concentric circles to the cylinder for $\lambda\Delta t = \{1,\ 10,\ 100,\ 1000,\ 10^8\}$, with the *implicit approach on velocity*

Chapter 3

Impulsively started flow past a cylinder

The previous chapters both provide the basic knowledge about the mechanism of the numerical methods considered in this thesis, and therefore constitute a strong introduction to that last chapter.

This final chapter actually presents the validation of our three considered VPM methods for an impulsively started flow past a 2-dimensional circular cylinder at a moderate Reynolds number $Re = 550$.

After a brief introduction of the settings and conditions of the flow, a study on the convergence of the methods is done with respect to the penalization parameter and the mesh size. Finally, the chapter ends with the flow analysis and its validation, in which a comparison of the used methods is performed.

Contents

3.1 Numerical setup and flow conditions

The computational domain and the geometrical setup are shown on FIGURE 3.1, where D is a rectangle delimited by its boundaries Γ. The radius R of the circular cylinder is equal to 0.2 (its diameter is marked d) and the body is centered at $(x_c,\ y_c) = (0.3,\ 0.5)$, this latter being immersed in a free flow velocity $\mathbf{U}_\infty$ purely horizontal, such that $\mathbf{U}_\infty = (U_\infty, 0)$. The whole computational domain D is covered by an uniform Cartesian mesh and defined as $(x, y) \times (x, y) = [0, 1] \times [0, 1]$.

The control volume on which the procedure of force calculation is done (see Noca et al. [30]; also briefly presented in the first chapter) is marked V.

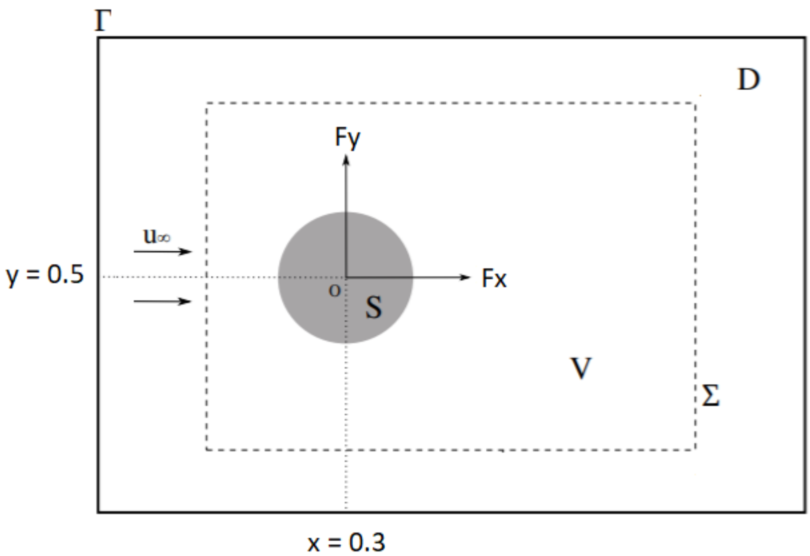

Figure 3.1: Computational domain D and control volume V

The Reynolds number of the performed simulations is fixed to $Re = 550$, that has been stated as moderate since it is a value where the effects of viscosity and advection are quite balanced. The time integration is firstly done with a time step fixed to $\Delta t^* = 10^{-6}$, it is then adapted to our results, as explained in the first chapter.

Here are the dimensionless variables used along this chapter :

$$\omega^* = \frac{d\,\omega}{U_\infty} \qquad t^* = \frac{t\,U_\infty}{d} \qquad Re = \frac{d\,U_\infty}{\nu} \qquad C_D = \frac{\mathbf{F}\cdot\mathbf{e}_x}{\frac{1}{2}\rho U_\infty^2 d} \tag{3.1}$$

3.2 Convergence studies

The purpose of this section is the evaluation of the influence of the Lagrangian multiplier and the mesh size on the simulations generated from the three considered VPM methods.

3.2.1 Lagrangian multiplier

The FIGURES 3.2 and 3.3 show the velocity profile (inside the body) on the vertical section across the center of the cylinder for several values of the relaxation factor. They are actually performed at the time $t^* = 0.5$ for which the velocity field inside the body has been converged.

First of all, we can obviously note a significant difference between those graphs and the same ones provided at the previous chapter, where the Navier-Stokes equations are not taken into account. In fact, it is due to the diffusion operation which is performed and has the effect of increasing the error on velocity for the same value of the Lagrangian multiplier. We also have a different behavior at the solid boundary, the velocity profile being interpreted as the formation of a boundary layer from the inside to the outside of the body by Caltagirone [7].

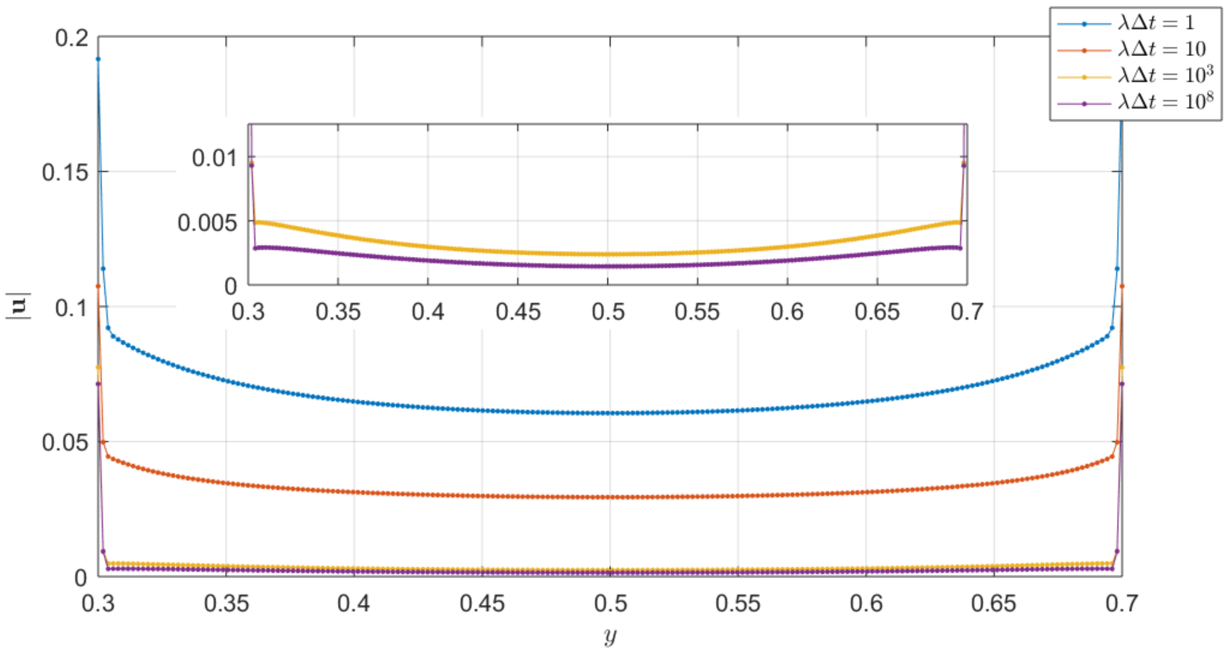

Figure 3.2: Velocity profiles in the vertical section across the center of the cylinder for $\lambda\Delta t = \{1,\ 10,\ 1000,\ 10^8\}$, according to the *implicit approach on velocity*, at time $t^* = 0.5$

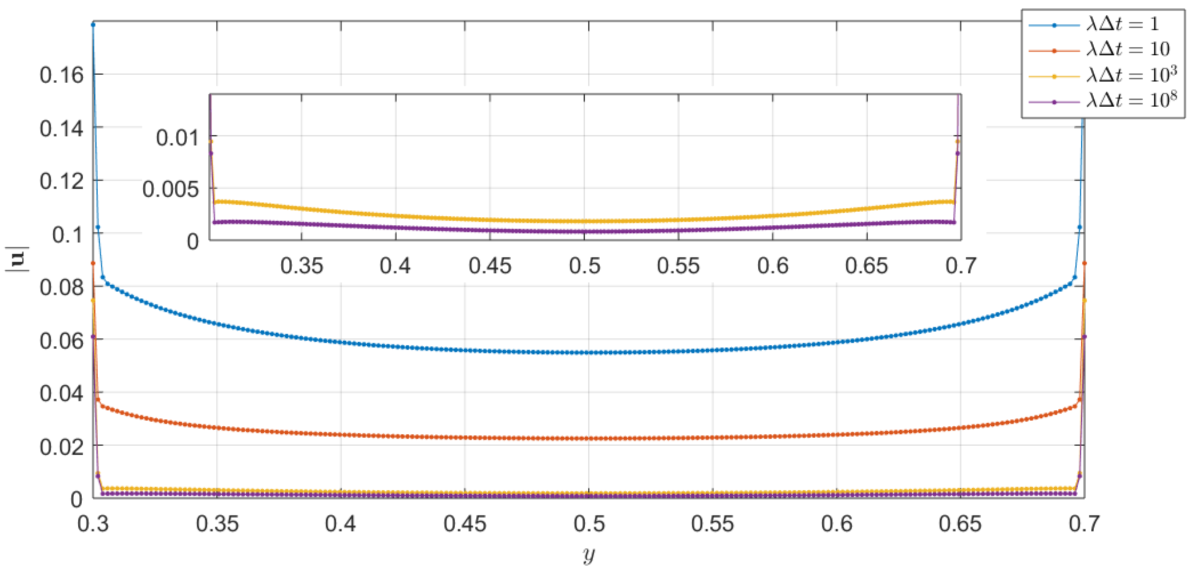

Figure 3.3: Velocity profiles in the vertical section across the center of the cylinder for $\lambda\Delta t = \{1,\ 10,\ 1000,\ 10^8\}$, according to the *implicit approach on vorticity*, at time $t^* = 0.5$

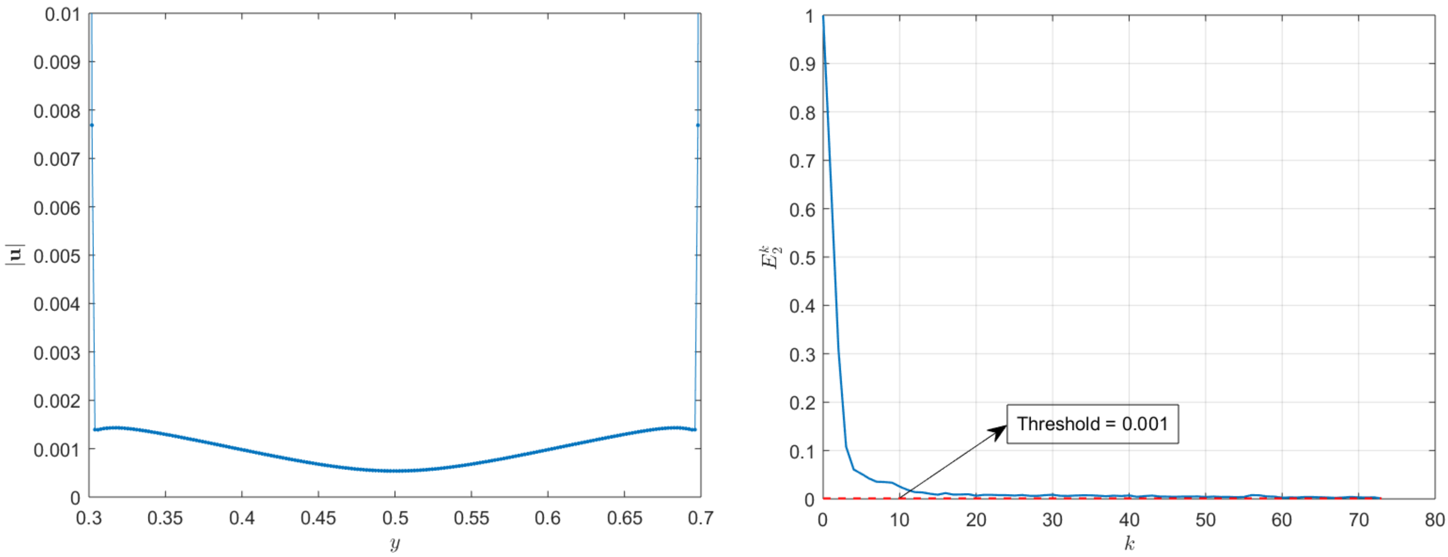

Figure 3.4: Velocity profile in the vertical section across the center of the cylinder (left) and the corresponding 2-norm error E_2^k as function of iterations performed to reach the threshold fixed to 0.001 (right), according to the *Hejlesen approach* with the coefficient $\lambda\Delta t = \alpha = 1.92$, at the first time step $t^* = 0$

.

Still for the FIGURES 3.2 and 3.3, despite the cited differences with the graphs of the previous chapter, one can note a similar behavior in terms of accuracy between the *implicit approaches on velocity* and *vorticity*, this latter being more precise for a same value of $\lambda\Delta t$, as exposed on the FIGURES 3.2 and 3.3.

Obviously, the same statement of facts can also be done here: for higher values of $\lambda\Delta t$, the body is much better taken into account, as shown on the corresponding FIGURES on which the velocity inside the cylinder is decreasing as the relaxation factor is increasing.

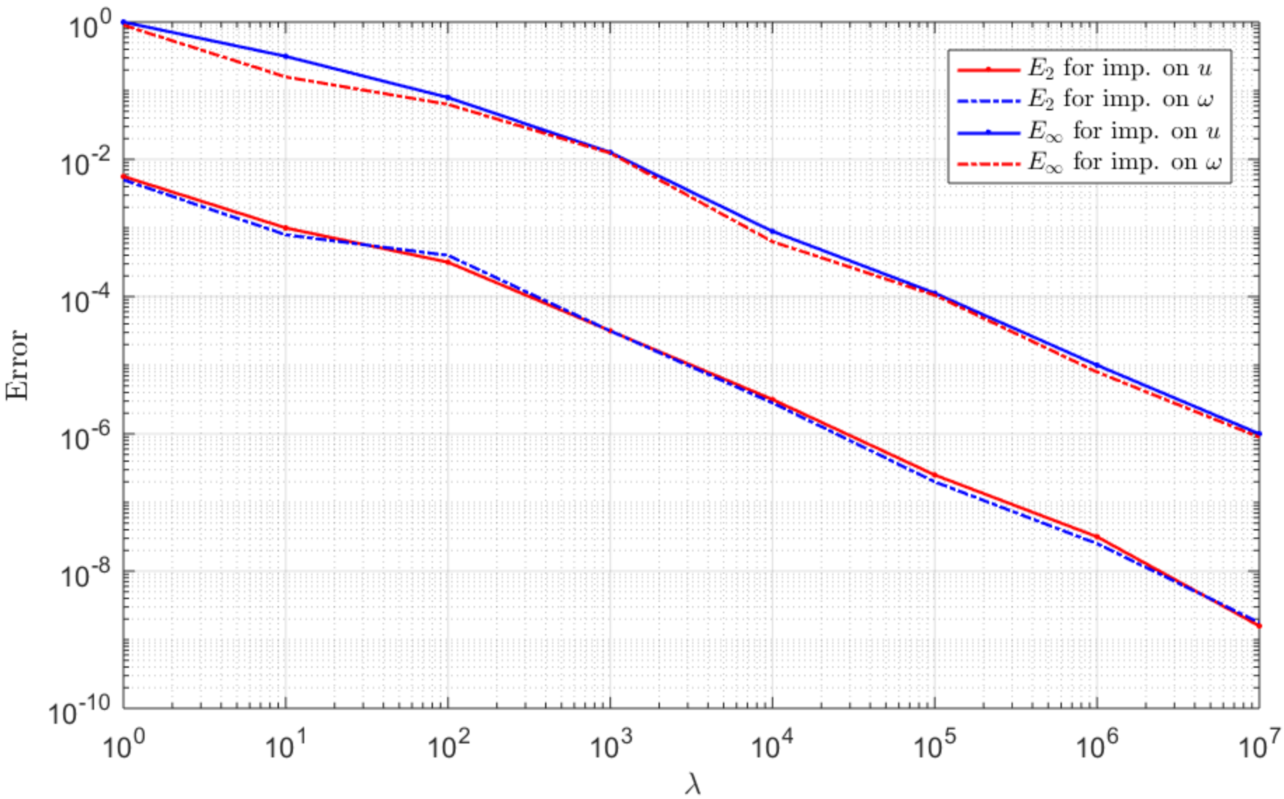

Figure 3.5: 2-norm and infinity-norm errors done inside the body Ω_b as function of the relaxation factor, for the implicit approaches on velocity u and vorticity ω

In order to have a more precise idea of the sensitivity of the results to the penalization parameter $\lambda\Delta t$, a convergence study is exposed on FIGURE 3.5 where the calculated 2-norm and infinity-norm errors in the body, respectively marked E_2 and E_∞, are defined according to the mean velocity magnitude $|\bar{\mathbf{u}}|$ (in the obstacle).

Their definitions are thus given by Eq. 3.2 and 3.3. This convergence study is still carried out at $Re = 550$ with $\Delta t^* = 10^{-6}$ (for the first iteration) and $n = 512$.

$$E_2 = \left\|\mathbf{u}^\lambda - \bar{\mathbf{u}}\right\|_2 = \left(\sum_{ij}(u_{ij}^\lambda - |\bar{\mathbf{u}}|)^2 h^2\right)^{1/2} \tag{3.2}$$

$$E_\infty = \left\|\mathbf{u}^\lambda - \bar{\mathbf{u}}\right\|_\infty = \max_{i,j}\left(\left|u_{ij}^\lambda - |\bar{\mathbf{u}}|\right| h\right) \tag{3.3}$$

We can observe on the FIGURE 3.5 that the numerical error decreasingly evolves as $O((\lambda\Delta t)^{-1/2})$ for $\lambda\Delta t < 10^2$ and as $O((\lambda\Delta t)^{-1})$ for $\lambda\Delta t > 10^2$, which is well in agreement with the results provided by Mimeau et al. [29]. As expected, the error done for the *implicit approach on vorticity* is under the one corresponding to the *implicit approach on velocity.*

Angot et al. [10] show that for an increase of $\lambda\Delta t$ to high values ($\lambda\Delta t > 10^6$), a mesh refinement is necessary to ensure the establishment of a higher accuracy, which is also verified by Mimeau et al. [29]. In fact, this is why our simulations for which the implicit approaches are used take as relaxation factor $\lambda\Delta t = 10^3$, the next section rightly exposes the reason of this choice.

Finally, the same results generated from the *Hejlesen approach* are presented on the FIGURE 3.4. We note that this method needs to perform more than 70 iterations to reach the same accuracy than the implicit approaches at later times. Nevertheless, it allows such an accuracy from the very first iteration, that being lacking to the *implicit approaches.* Obviously, it implies a significant computational cost since the Poisson problem has to be solved at each iteration, as discussed by Hejlesen et al. [22]. In order to give a global view of this cost, here are some values from [22]: for the thresholds $\epsilon = \{10^{-1},\ 10^{-2},\ 10^{-3},\ 10^{-4}\}$, the additional work due to the iterative procedure is such that it takes respectively $\{3\%,\ 6\%,\ 11\%,\ 54\%\}$ of the total cost of a time step at a later time t^*.

T. Gillis et al. [18] propose an alternative technique consisting in an efficient iterative method using recycled Krylov subspaces. In fact, since the numerical results only slightly change between two time steps, previous solutions can be efficiently recycled thanks to the algorithm rBiCGStab (recycled Bi-Conjugate Gradient Stabilized). Its application to a flow past a cylinder in 2D (at $Re = 9500$) actually divides the cost of the iterative penalization by nearly 50. This topic of enhancement of the iterative penalization procedure is currently an ongoing work.

3.2.2 Mesh size

The FIGURE 3.6 presents the impact of the mesh size on the velocity field. Here are only exposed the data from the *implicit approach on velocity* and the *Hejlesen technique* (once again, the *implicit approach on vorticity* generates very similar results to the other implicit technique, it is why it is not shown).

We can observe that the velocity field is converging as the mesh size is increasing (or similarly the spatial step h is decreasing) for a fixed value of $\lambda\Delta t$. In fact, this is the reason why it is not necessary to use too high values of $\lambda\Delta t$: it does not affect the accuracy of the velocity field anymore, without reducing the spatial step, as introduced in the previous subsection. Additional validations are given in the following section.

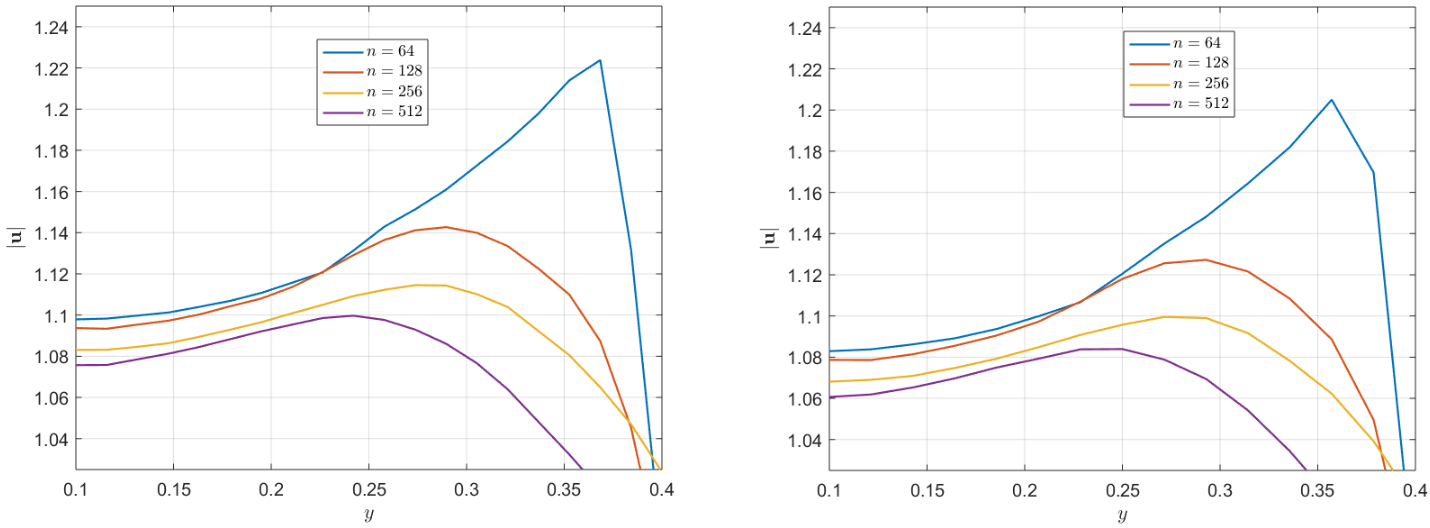

Figure 3.6: Velocity fields in a vertical section at the location $x = 0.8$ behind the cylinder at time $t^* = 0.5$ for the mesh sizes $n = \{64, 128, 256, 512\}$. Data generated from the implicit approach on velocity (left) and the Hejlesen approach (right), respectively for $\lambda\Delta t = 10^3$ and $\lambda\Delta t = \alpha = 1.92$.

3.3 Validations

In this section, we validate our Vortex Particle-Mesh methods by computing the forces exerting on the cylinder (within the meaning of Noca et al. [30], as introduced in the Chapter 1). Then a flow analysis is done through the presentation of the vorticity fields at several times t^*.

3.3.1 Coefficient of drag

Here we use a theoretical curve of the coefficient of drag valid for short times [34] and given by Eq. 3.4 as reference for the comparison of numerical data produced by our VPM methods.

$$C_D = 4\sqrt{\frac{\pi}{Re\ \ t^*}} + 2\pi\left(9 - \frac{15}{\sqrt{\pi}}\right)\frac{1}{Re} \tag{3.4}$$

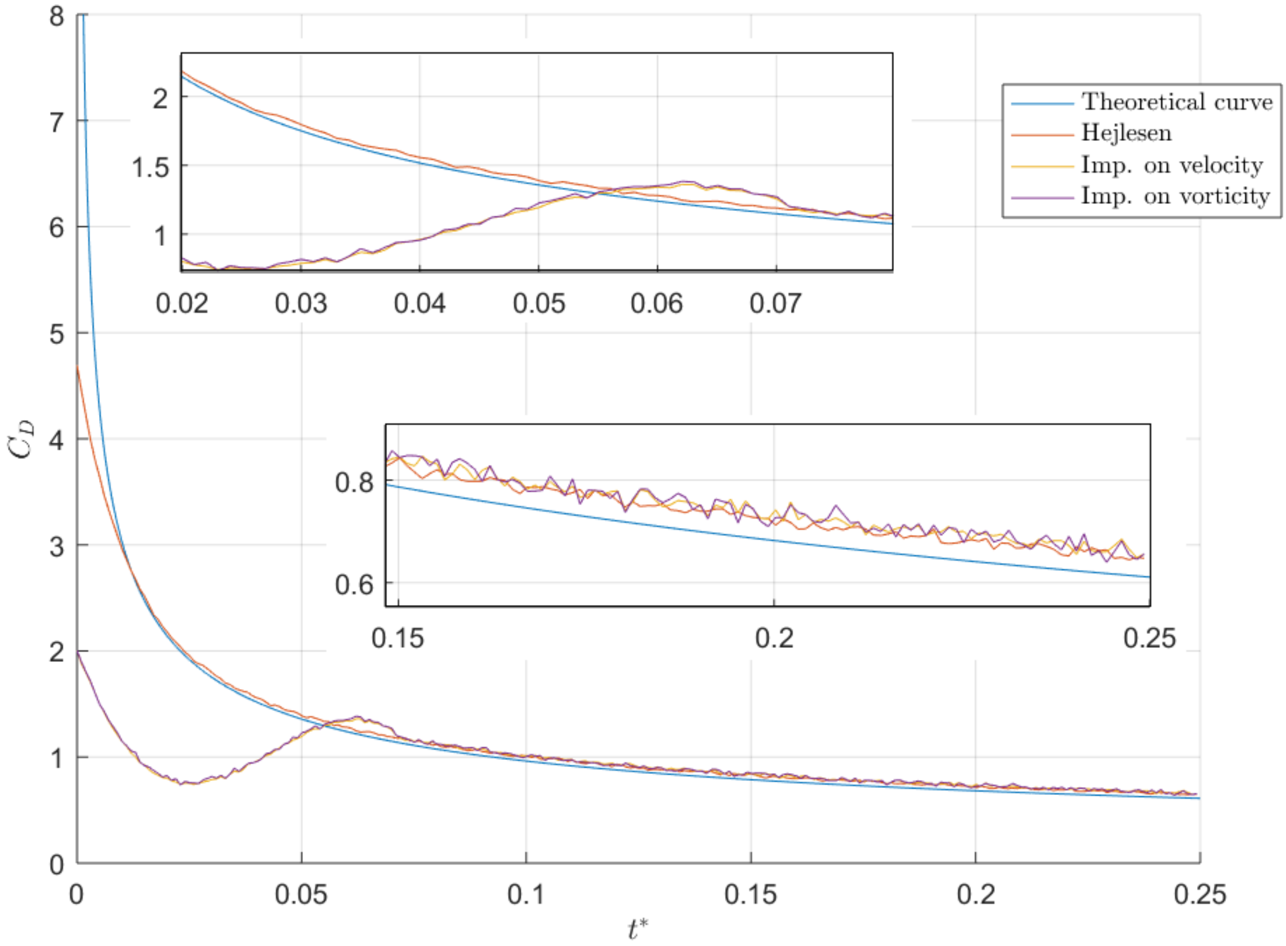

Figure 3.7: Coefficient of drag C_D for the considered methods at $Re = 550$

The FIGURE 3.7 exposes the curves of coefficient of drag C_D generated by our methods. As expected, for all numerical methods, C_D is underestimated at the very early times since the theoretical behavior is actually singular at $t^* = 0^+$, that corresponding to an infinitely thin boundary layer. It is important to note that the coefficient of drag produced at very early times for the implicit approaches is much lower than for the Hejlesen method. In fact, this can be interpreted as follows: since the implicit methods let a more substantial velocity inside the body than the Hejlesen method, the velocity difference between the flow and the solid boundary is lower, that resulting to a obstacle less accelerated, "less pushed" and thus a lower C_D. We can also observe that the implicit techniques take more time to converge to the analytic solution than the Hejlesen approach.

Nevertheless, soon after, all our simulations are well-resolved and their C_D curve matches with the theoretical one (temporarily, as Eq. 3.4 is only valid for short times). Our results are not filtered and thus exhibit a low signal of noise.

3.3.2 Flow analysis

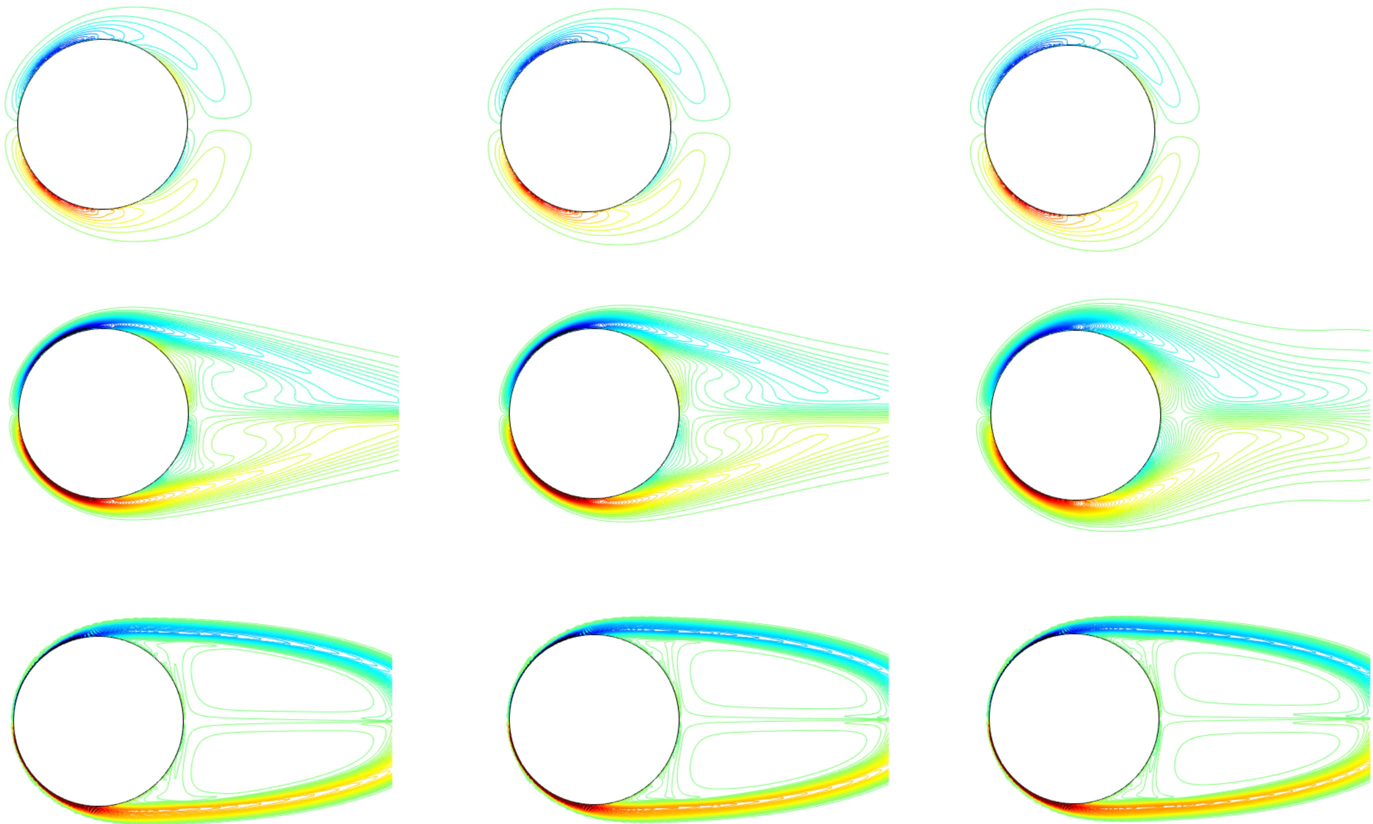

Figure 3.8: Vorticity isocontours for an impulsively started cylinder. Columns from left to right: the *implicit approaches on velocity* and *vorticity*, and the *Hejlesen method*. From top to bottom, the profiles at time $t^* = \{0.05,\ 0.1,\ 0.2\}$

The FIGURE 3.8 compares the vorticity isocontours for the impulsively started flow around a cylinder, produced by our three VPM methods at several times t^*. It is actually quite difficult to show a clear difference between the two implicit methods.

However, a consistent difference can be observed for the location and shape of the vortical structures that are generated on the cylinder boundary for the Hejlesen method and the others. Our interpretation of this is actually the same as the one explained at the previous section. Since the non-iterative implicit penalization methods fail at deflecting the flow around the cylinder and thus lets the flow penetrating the solid boundary and remaining active within the penalized region, the velocity field takes a kind of shortcut through the obstacle, that resulting to further located fields in the implicit approaches than in the Hejlesen approach.

Hejlesen et al. also verify those differences, as we can note on the FIGURE 3.9 for an impulsively started flow around an obstacle at $Re = 9500$. Indeed, the further location of vortices, as well as their shapes more developed, refer to the same phenomenon due to the lack of accuracy of the non-iterative methods at early times.

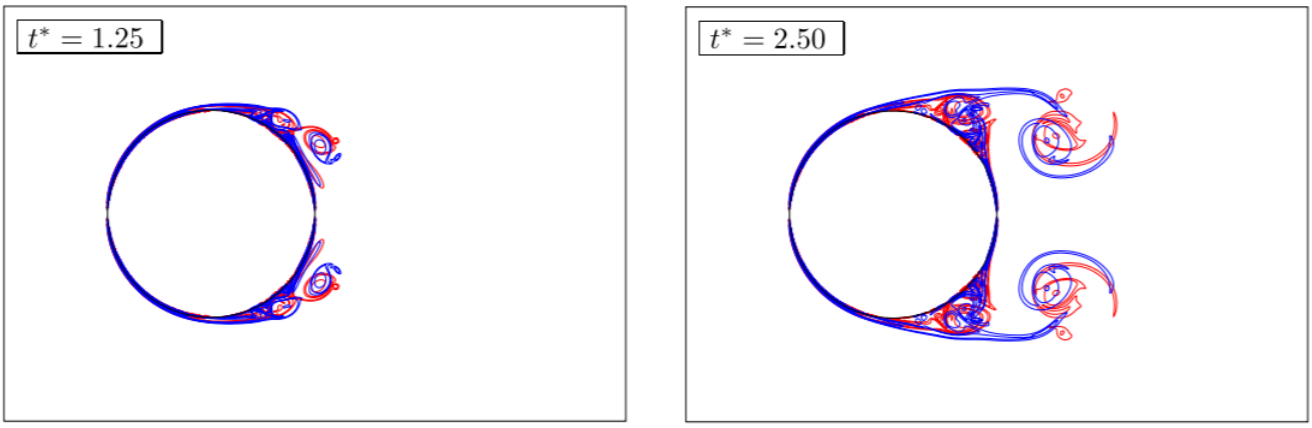

Figure 3.9: Vorticity contours for an impulsively started flow around a cylinder at $Re = 9500$ from Hejlesen et al. [22]

Conclusion

This Master thesis compares the treatment of solid boundaries achieved by three Brinkman penalization methods through the Vortex Particle-Mesh framework for an impulsively started flow around a cylinder at low Reynolds number. Those methods have been named as *the implicit approach on velocity*, *the implicit approach on vorticity* and *the Hejlesen approach* throughout this document.

The starting point of this work is the presentation of the development of the vortex method and its two main approaches, *the continuous and discrete forcing approaches*, as well as the primary interests of this latter for a general use. A current state-of-the art is thus established and brings us to the employment of the Vortex Particle-Mesh framework for which the adaptation of these approaches is far from being straightforward and has only been partially studied.

Then this work focuses on the *continuous forcing approach* and more specifically, on the Brinkman penalization framework applied to VPM method. We have shown that it actually consists in the use of an artefact, an additional term, in the Navier-Stokes equations to take into account an immersed body, and thus, in enforcing the no-slip condition at the surface of the obstacle.

Those introductions lead us to the actual topic of this master thesis: the transposition of those Brinkman penalization techniques to the VPM method, the comparison of three of them through their validations for a 2D flow past a cylinder. To this end, we firstly deal with the additional penalized velocity term which constitutes the Brinkman penalization. It is rightly shown that its perfect integration carries out the corresponding potential flow, and a convergence study is done in order to establish the impact of the involving parameters on the solution.

Finally, this documents concludes on the 2-dimensional simulations of an impulsively started flow around a cylinder. As expected, it is the iterative method proposed by Hejlesen et al. [22] which stands out since it guarantees from the very first time step the desired threshold of error on velocity admitted in the body.

Nevertheless, the considered implicit approaches return very satisfying results to the views of validations and their respective required computational cost.

As it is detailed in this work, the reduction of the computational cost for the iterative penalization methods is an ongoing project. This topic is therefore an opened subject which aims at always increasing the accuracy, reducing the costs,etc. in order to better take into account an immersed body.

Bibliography

[1] Massimiliano Margonari. Scilab finite element solver for stationary and incompressible Navier-Stokes equations. `http://www.openeering.com/node/18/` [Online; accessed 14-July-2019]

[2] D. Goldstein, R. Handler, and L. Sirovich, *Modeling a no-slip flow boundary with an external force field*, J. Comput. Phys, 105 (1993), 354–366

[3] S. Peskin, Charles. (2003). The Fluid Dynamics of Heart Valves: Experimental, Theoretical, and Computational Methods. Annual Review of Fluid Mechanics. 14. 235-259.

[4] Charles S Peskin, Numerical analysis of blood flow in the heart, Journal of Computational Physics, Volume 25, Issue 3, 1977, Pages 220-252.

[5] E.M. Saiki and S. Biringen, Numerical simulation of a cylinder in uniform flow: application of a virtual boundary method, J. Comput. Phys., 123 (1996), 450–465.

[6] E. Arquis and J.P. Caltagirone, Sur les conditions hydrodynamiques au voisinage d'une interface milieu fluide-milieu poreux: application à la convection naturelle, C.R. Acad. Sci Paris, 299, No 1, Série II (1984), 1–4.

[7] J.P. Caltagirone, Sur l'interaction fluide-milieux poreux: application au calcul des efforts exercés sur un obstacle par un fluide visqueux, C. R. Acad. Sci. Paris, 318 Série II (1994), 571–577.

[8] Ph. Angot and J.P. Caltagirone, Natural convection through periodic porous media, Proceeding 9th Int. Heat Transfer Conf., Jerusalem, Hemisphere, 5 (1990), 219–224.

[9] Ph. Angot, Analysis of singular perturbation on the Brinkman problem for fictitious domain models of viscous flow, Math. Methods Appl. Sci., 22 (1999), 1395–1412.

[10] Angot, Philippe Bruneau, Charles-Henri Fabrie, Pierre. (1999). A penalization method to take into account obstacles in viscous flows. Numerische Mathematik. 81. 497-520.

[11] Gilles Carbou, Pierre Fabrie. Boundary layer for a penalization method for viscous incompressible flow. Advances in Differential Equations, Khayyam Publishing, 2003, 8 (12), pp.1453-1480.

[12] Bruneau, Ch.-H., Fabrie, P. (1994): Effective downstream boundary conditions for incompressible Navier-Stokes equations. Int. J. Numer. Methods in Fluids 19, 693– 705.

[13] R. Mittal, G. Iaccarino, Immersed boundary methods, Annual Review of Fluid Mechanics 37 (2005) 239–261.

[14] N. K.-R. Kevlahan, J.-M. Ghidaglia, Computation of turbulent flow past an array of cylinders using a spectral method with Brinkman penalization, European Journal of Mechanics - B/Fluids 20 (3) (2001) 333–350.

[15] P. Koumoutsakos, A. Leonard, High-resolution simulations of the flow around an impulsively started cylinder using vortex methods, J. Fluid Mech. 296 (1) (1995) 1–38.

[16] G.-H. Cottet, P. Koumoutsakos, Vortex Methods, Theory and Practice, Cambridge University Press, 2000.

[17] G.S. Winckelmans, Vortex methods, in: Encyclopedia of Computational Mechanics, John Wiley Sons, Ltd, ISBN 9780470091357, 2004, pp. 129–153.

[18] T. Gillis, G. Winckelmans, P. Chatelain, An efficient iterative penalization method using recycled Krylov subspaces and its application to impulsively started flows, Journal of Computational Physics, Volume 347, 2017, Pages 490-505.

[19] Y. Marichal, P. Chatelain, G. Winckelmans, An immersed interface solver for the 2-D unbounded Poisson equation and its application to potential flow, Comput. Fluids 96 (2014) 76–86.

[20] M. J. Lighthill, Introduction of boundary layer theory, in: R. L. (Ed.), Laminar Boundary Layers, Oxford University Press, New York, 1963

[21] M. Gazzola, P. Chatelain, P. Koumoutsakos, Vortex methods for fluid-structure interaction problems with deforming geometries and their application to swimming, in: Division of Fluid Dynamics, American Physical Society, 2010.

[22] M.M. Hejlesen, P. Koumoutsakos, A. Leonard, J.H. Walther, Iterative Brinkman penalization for remeshed vortex methods, J. Comput. Phys. 280 (2015) 547–562.

[23] *ONERA photograph, Werlé* - Gallon 1972

[24] P. Chatelain, A. Leonard, Isotropic compact interpolation schemes for particle methods, Journal of Computational Physics 227 (6) (2008) 3244 – 3259.

[25] M. Hejlesen, J. Rasmussen, P. Chatelain, J. Walther, A high order solver for the unbounded Poisson equation, Journal of Computational Physics 252 (2013) 458–467.

[26] P. Chatelain, A. Curioni, M. Bergdorf, D. Rossinelli, W. Andreoni, P. Koumoutsakos, Billion vortex particle Direct Numerical Simulations of aircraft wakes, Computer Methods in Applied Mechanics and Engineering 197 (13) (2008) 1296–1304.

[27] P. Chatelain, P. Koumoutsakos, A Fourier-based elliptic solver for vortical flows with periodic and unbounded directions, J. Comput. Phys. 229 (7) (2010) 2425–2431

[28] J. H. Williamson, Low-storage Runge-Kutta schemes, Journal of Computational Physics 35 (1) (1980) 48–56.

[29] C. Mimeau, F. Gallizio, G.-H. Cottet, I. Mortazavi, Vortex penalization method for bluff body flows, Int. J. Numer. Methods Fluids 79 (2) (2015) 55–83, fld.4038.

[30] F. Noca, D. Shiels, D. Jeon, A comparison of methods for evaluating time-dependent fluid dynamic forces on bodies, using only velocity fields and their derivatives, J. Fluids Struct. 13 (5) (1999) 551–578.

[31] J.T. Rasmussen, G.-H. Cottet, J.H. Walther, A multiresolution remeshed Vortex-In-Cell algorithm using patches, J. Comput. Phys. 230 (2011) 6742–6755

[32] PETSc library - `https://www.mcs.anl.gov/petsc/` [Online, accessed 15-Jul-2019]

[33] *Steady, incompressible and irrotational, 2-D flows*, Note for the course LMECA2322, G. Winckelmans.

[34] M. Bar-Lev, H.T. Yang, Initial flow field over an impulsively started circular cylinder, J. Fluid Mech. 72 (1975) 625–647

UNIVERSITÉ CATHOLIQUE DE LOUVAIN
École polytechnique de Louvain
Rue Archimède, 1 bte L6.11.01, 1348 Louvain-la-Neuve, Belgique | **www.uclouvain.be/epl**