



Université catholique de Louvain

Ecole polytechnique de Louvain

Institut de Mécanique, Matériaux et Génie Civil

**Vortex particle-mesh method with combined immersed
boundary and mesh refinement techniques. Application
to bluff-body and wake-vortex flows**

Timothée Lonfils

Mai 2011

Composition du jury:

Pr. G. Winckelmans (UCL, promoteur)

Pr. P. Chatelain (UCL)

Dr. P. Geuzaine (Cenaero)

Pr. P. Koumoutsakos (ETH)

Pr. J.-F. Remacle (UCL)

Pr. T. Pardoën (UCL, président)

Thèse présentée en vue de

l'obtention du grade de docteur

en sciences de l'ingénieur

©2011

Timothée Lonfils

All Rights Reserved

Acknowledgments

Je voudrais avant tout remercier mon promoteur, Grégoire Winckelmans, qui m'a fait confiance et m'a permis de réaliser cette expérience inoubliable. Sa rigueur scientifique et son regard critique m'ont poussé jour après jour à me dépasser. Je remercie également chacun des membres de mon jury, Philippe Chatelain, Petros Koumoutsakos, Philippe Geuzaine, Jean-François Remacle, Thomas Pardoën et Grégoire Winckelmans pour les remarques avisées lors de ma défense privée.

Je remercie le F.R.S.-FNRS (Fonds de la recherche scientifique) qui m'a financé durant ces quatre années en m'accordant une bourse de doctorat F.R.I.A. ainsi qu'en co-financant un modeste cluster parallèle de visualisation.

De plus, je remercie chaleureusement le CENAERO (Center for Excellence in Aeronautics) et dans une moindre mesure le C.I.S.M (Centre de calcul intensif et de stockage de masse) qui m'ont permis l'utilisation de leur cluster de calcul parallèle afin de réaliser les simulations numériques présentées dans ce document. Leur support et leur connaissance technique m'ont été fort bénéfiques.

Ces quatre années n'aurait pas été aussi agréables sans l'ambiance qui régnait au sein du pôle de recherche. Merci à mes collègues : les *Wincky-boys* (Matthieu, Ivan, Laurent, Roger, Goéric, Yves, Nicolas) bien entendu... mais aussi tous les autres: Amel, Xavier, Julien, Francesco, Nicolas, Frank, ...

Je remercie tout particulièrement Matthieu avec qui j'ai partagé mes débuts en tant que "chercheur" et qui, au fil des années, est devenu, par son regard avisé et ses remarques toujours pertinentes, un deuxième promoteur.

Cette thèse aurait été fort différente sans la deuxième vie apportée par Philippe. La collaboration a été malheureusement trop courte mais néanmoins

fructueuse. Il a été, de facto, mon troisième promoteur!

Merci à Dominique pour la relecture de ce document. Merci à celle qui partage ma vie, Audrey, et qui m'a soutenu tout au long de ces années. Merci à toute ma famille et mes amis.

Enfin, je souhaiterais remercier la communauté *open source*. Aujourd'hui, un ensemble performant et gratuit de logiciels professionnels sont développés, et ce toujours dans l'esprit originel! Merci à tous ces anonymes...

Contents

1	Introduction	1
1.1	Background	1
1.2	Some preliminaries	3
1.3	Objectives	5
1.4	Outline	6
2	An IB technique within the combined VPM-PFM method	9
2.1	Introduction	10
2.2	An Immersed Boundary Method Within a Vortex Particle-Mesh Method	12
2.3	Validation: flow past a sphere at $Re = 300$	20
2.4	Flow past a very low aspect-ratio wing	27
2.5	Conclusion	36
3	A mesh refinement technique for the VPM method	37
3.1	Introduction	37
3.2	A basic multi-block generator	39
3.3	A redistribution scheme based on average-interpolation	41
3.3.1	Multi-level redistribution scheme	41
3.3.2	Treatment of varying resolutions	44
3.4	A 1-D multiple resolution example	47
3.5	Vortex ring test case	49
3.6	Multiple resolution simulation of the flow past a sphere	62
3.7	Conclusion	67

4	Investigation of velocity deficit effects on aircraft wake rollup	69
4.1	Introduction	69
4.2	Initial conditions	70
4.3	Numerical setup	78
4.4	Results	81
4.4.1	3D dynamics: a qualitative analysis	81
4.4.2	Quantitative analysis	82
4.4.3	Characterization of the rolled-up vortex sheet	103
4.5	Conclusion	110
5	Numerical investigations of end-effect	113
5.1	Introduction	114
5.2	Numerical method	117
5.3	Analysis and comparison	119
5.4	T-D and S-D simulations of an accelerated wing	132
5.5	Conclusion	159
6	Conclusions & perspectives	161
6.1	Achievements	161
6.2	Perspectives	163
6.2.1	A VPM method using a staggered grid	163
6.2.2	Further improvements of the Poisson solver for hierarchi- cally refined grids	164
6.2.3	Improvement of the handling of no-slip	165
6.2.4	A coupling approach using an Eulerian “near-wall” solver	165
	Bibliography	170
A	S-D simulation of a near-wake rollup using a modeled inflow	181
B	3D filament simulations of Crow-like instabilities IGE	187
B.1	Introduction	187
B.2	Vortex filament method	188
B.3	Definition	190
B.4	Comparison: numerical simulation versus experiment	191
B.5	Analysis of the Crow instability in ground effect	194

Chapter 1

Introduction

1.1 Background

The present work joins a global effort in the development of efficient and accurate numerical tools for the simulation of complex fluid flows. More specifically, we focus on unsteady external flows, a fluid dynamics discipline which actually pervades applied sciences and engineering. These are the flow encountered in aircraft or car aerodynamics, wind energy, biological locomotion, etc. In the problems addressed here, we assume that the flow is incompressible and that the inertial forces dominate the viscous stresses: we are dealing with both moderate and high Reynolds number flows.

The governing equations of fluid mechanics were derived between the 15th and 19th century, culminating with the works of Navier and Stokes some 150 years ago. The resulting Navier-Stokes equations have since then been one of the pillars of fluid mechanics and have been used universally in all fields of science.

The development of computers, since the seventies, has enabled new perspectives for the study of complex fluid flow problems. This discipline, dedicated to the numerical resolution of the fluid mechanics equations, is called computational fluid dynamics (CFD). The straightforward solution approach consists in capturing all scales of the flow through a very fine space-discretization; a direct numerical simulation (DNS) is then performed. This approach however quickly leads in practice to an unaffordable computational time.

Two main approaches are commonly used to tackle this issue: the Large Eddy Simulations (LES) and the Reynolds Averaged Navier-Stokes (RANS) simulations. The former models the effect of the unresolved scales on the resolved scales. The latter directly obtains the time-averaged solution through the modeling of the Reynolds stress tensor. Because we are interested in accurately capturing transient or unsteady phenomena, we here rely on the first approach, LES.

Furthermore, the present work treats the flow equations in a very distinct approach. We use a vortex particle-mesh (VPM) method, which belongs to the broader class of vortex methods. Such methods use the vorticity-velocity formulation of the Navier-Stokes equations, rather than the velocity-pressure formulation: the vorticity field is thus the primary variable. The major advantage comes from the compactness of the vorticity field for external flows and wakes. A limited number of particles is thus required to discretize the entire flow.

In the vortex method used here, the elements of discretization are “vortex particles” that are convected with the local velocity vector. The major advantage comes from the compactness of the vorticity field for some flows of interest (e.g., external flows and wakes). A limited number of particles is then required to discretize the entire flow. Moreover, vortex methods intrinsically take into account unbounded-flow boundary conditions.

Vortex particle methods, however, have drawbacks, some of them fundamental. First, the element of discretization being isotropic, they are not intrinsically adapted to capture physical phenomena such as boundary layers. Second, they require, in general, more computational time. Finally, with time, the particles can cluster or deplete in some regions. Redistribution schemes with high order conservation properties are then required; they allow performing accurate long time simulations avoiding the clustering/depletion of particles inherent to any Lagrangian method.

Vortex methods have been developed for quite some time and are now applied to aerodynamics, fundamental fluid mechanics or biological locomotion (Hieber and Koumoutsakos [36], Zhang and Eldredge [93]). They are used for incompressible and compressible flows (Eldredge [28]), reacting flows (Thiriy [82]), viscous and inviscid flows, free vortical flows or wall-bounded flows

(Koumoutsakos [47], Ploumhans and Winckelmans [67] and Cottet and Poncet [17]).

DNS and high quality LES can also be performed. Cocoli et al. [12] proposed an accurate multi-scale subgrid model for such LES.

1.2 Some preliminaries

Before going further, let us quickly go through the fundamentals of vortex methods.

Formulation equations The Navier-Stokes equations for an incompressible flow, thus $\nabla \cdot \mathbf{u} = 0$, in vorticity-velocity formulation read

$$\frac{D\boldsymbol{\omega}}{Dt} = (\nabla \mathbf{u}) \cdot \boldsymbol{\omega} + \nu \nabla^2 \boldsymbol{\omega}, \quad (1.1)$$

where $\mathbf{u}$ is the velocity field and $\boldsymbol{\omega} \triangleq \nabla \times \mathbf{u}$ is the vorticity field, thus $\nabla \cdot \boldsymbol{\omega} = 0$. $\frac{D}{Dt} \triangleq \frac{\partial}{\partial t} + \mathbf{u} \cdot \nabla$ is the material derivative. Using the Helmholtz decomposition, one can write the velocity field as

$$\mathbf{u} = \nabla \times \boldsymbol{\Psi} + \nabla \phi, \quad (1.2)$$

where $\boldsymbol{\Psi}$ is the streamfunction and ϕ is the scalar potential. We take ϕ such that $\nabla \phi = \mathbf{U}_0$, where $\mathbf{U}_0$ is the upstream velocity. If we set the gauge of $\boldsymbol{\Psi}$ as $\nabla \cdot \boldsymbol{\Psi} = 0$, the vorticity field and the stream function are related by the Poisson equation

$$\nabla^2 \boldsymbol{\Psi} = -\boldsymbol{\omega}. \quad (1.3)$$

Discretization The vorticity field is discretized using vortex particles which are responsible for a volume V_p and carry a vorticity vector quantity ($\boldsymbol{\alpha}_p \triangleq \boldsymbol{\alpha}(\mathbf{x}_p) = \int_{V_p} \boldsymbol{\omega} dV \simeq \boldsymbol{\omega}(\mathbf{x}_p) V_p$). The evolution equations for the position and the strength of those particles read

$$\frac{d\boldsymbol{\omega}_p}{dt} = (\nabla \mathbf{u}(\mathbf{x}_p)) \cdot \boldsymbol{\omega}(\mathbf{x}_p) + \nu \nabla^2 \boldsymbol{\omega}(\mathbf{x}_p) \quad (1.4)$$

$$\frac{d\mathbf{x}_p}{dt} = \mathbf{u}(\mathbf{x}_p) \quad \forall p = [1 \dots N] \quad (1.5)$$

Computation Vortex particle methods require to solve Eq. (1.2) and (1.3) to obtain the velocity field from the particle strengths. For unbounded domains, Eq. (1.2) and (1.3) can be solved using fast summation techniques for the Green’s function, such as the Parallel Fast Multipole (PFM) method to reduce cost complexity from $\mathcal{O}(N^2)$ to $\mathcal{O}(N \log N)$ (Barnes and Hut [1], Greengard [33] and Greengard and Rokhlin [34]). The parallel implementation used here has been developed by Salmon and Warren [74] and further extended for vortex methods by Winckelmans [90]. However these methods are still expensive for the simulation of external flows, even at moderate Reynolds number (e.g. $\mathcal{O}(10^3)$).

The presents approach tackles this problem through the efficient combination of the “Vortex Particle-Mesh” (VPM) method, also called “Vortex-In-Cell” (VIC), originally developed by Christiansen [10], see also Cottet and Poncet [17] and Cocle et al. [13]) method and the PFM method. The VPM methods solve the Poisson equation (Eq. (1.3)) using fast grid solvers that are much faster than the fast summation for the Green’s function based Poisson solvers. The required boundary conditions are here provided by the PFM method. Those boundary conditions being accurate, a compact grid can be used, that is tight to the vorticity field. This is the basis of the hybrid VPM-PFM method. From a parallel computational point of view, we note that there is no need to solve a global Poisson equation. Rather, multiple Poisson equations can be solved locally, on each subgrid, using the proper boundary conditions provided by the PFM method (which has a global view of the whole vorticity field).

These hybrid methods require the source term of Eq. (1.3) to be available on the grid. This entails the translation of particles strength into a grid vorticity through a redistribution. The evaluation of the right hand side (RHS) of Eq. (1.4) and Eq. (1.5) are required to integrate in time the evolving equations. Those are here computed using finite differences. That requires, on the one hand, to interpolate particle strengths onto the grid in order to compute the time-variation of each particle intensity and position. On the other hand, this “grid information” has to be interpolated from the grid to the particles. This method has been successfully developed by Cocle et al. [13, 11] for (half-)unbounded vortical flows.

We also note that the Poisson equation can also be solved with unbounded

boundary conditions using the Hockney algorithm [37] as done for 2-D wall-bounded flows by Morgenthal and Walter [61] and for 3-D free vortical flows by Chatelain and Koumoutsakos [9]. This approach is very efficient because it uses fast Fourier transforms (see [8] for a massively parallel application using a billion grid points), but it requires twice more grid points for each unbounded direction. The introduction of a multi-resolution capability in the latter approach is likely quite complex.

1.3 Objectives

It is now well established that vortex method definitely overcome classical methods (i.e. CFD solvers based on the discretization of the Navier-Stokes equations using the velocity-pressure formulation) when simulating free vortical flows (see Winckelmans et al. [88], Daeninck et al. [22, 21], Cocoli [11], Cottin et al. [19]).

In addition, this is still true whatever the value of the Reynolds number. In spite of the major developments in vortex method in the last decades, those methods rapidly become unaffordable when carrying out the simulation of vortical flows interacting with a viscous ground or the simulation of external flows past bodies. It is still very challenging to perform external flow simulations using vortex methods, especially when the Reynolds number is high. They are intrinsically well adapted to capture varying flow resolutions, yet they still require further improvements in order to benefit from the recent advances:

- Combining a fast grid-based Poisson solver together with the handling of a varying resolutions. External flows are generally characterized by the need for multiple resolutions.
- Carrying out the flow past moving and deforming bodies. Vortex methods require to deal with discontinuous vorticity field, as we here consider the Navier-stokes equations in vorticity-velocity formulation: the velocity tends to zero at the wall whereas the vorticity is maximum at the wall and zero inside (for non rotating bodies), and thus discontinuous.
- Efficiently dealing with boundary layers. Vortex particle methods are intrinsically not adapted to capture boundary layers as the element of discretization (i.e., a particle) is isotropic.

The present research work aims at pursuing the development of the vortex particle-mesh (VPM) method with two outstanding contributions in order to address those problems.

First an immersed boundary technique has been developed and adapted to capture the flow past arbitrary shape bodies. The method is based on the approach first developed by Koumoutsakos [45] and further improved by Ploumhans [67, 68]. The motivation is to benefit from the enhanced efficiency provided by the VPM method in terms of computational cost. The required Poisson equation is solved using an efficient grid-based solver combined with a parallel fast multipole method (which provides the required boundary conditions on each subdomain) as shown in previous works, also within our group, see Cocle et al. [11]. The method is here further developed in the spirit of the immersed boundary method as we model an extra term of the evolving equations, expressed as grid quantities, in order to take into account the presence of a no-slip boundary.

Second, an accurate approach handling hierarchically refined meshes has been developed and validated. The basis of this technique is in the same spirit as the methodology developed by Bergdorf et al. [2] and Hejazialhosseini et al. [35]. We use both grid patches and particles of varying resolution.

The originality of this work is the combination of the handling of multiple flow resolutions together with a full 3-D VPM Navier-Stokes solver to compute incompressible flows. This method also benefits from the versatility of the parallel fast multipole method which enables the efficient solution of the Poisson equation and straightforward domain decomposition. We have also demonstrated the potential of the mesh refinement technique on a Large-Eddy Simulation of a turbulent vortex wake rollup at very high Reynolds number.

1.4 Outline

The first part of the manuscript provides the description of two advances in vortex methods and then their application to the numerical investigations of bluff-body and wake-vortex flows. The two major improvements done in this work are: i) using the VPM-PFM method with the Immersed Boundary (IB) method to compute wall-bounded external flows at a moderate Reynolds num-

ber and ii) including a multi-resolution approach in the VPM-PFM method.

- **Chapter 2** presents the numerical algorithm developed to compute the flow past arbitrarily-shaped geometries (lifting- and bluff-bodies) at a moderate Reynolds number (i.e. $\mathcal{O}(10^2 - 10^3)$). The method is based on an immersed boundary approach to solve the Poisson equation required to obtain the streamfunction. The vorticity sources, corresponding to the cancellation of the residual slip velocity and needed by the Poisson equation, are here computed by means of the vortex panel method. This approach is validated against the benchmark of the flow past a sphere at Reynolds 300. The method is also used for the DNS (at $Re = \frac{U_0 c}{\nu} = 3000$) of an impulsively started flow past a profiled body: i.e. a very low aspect ratio wing.
- **Chapter 3** deals with the integration of a multiple space-resolution technique within the VPM-PFM method. The approach uses the classical redistribution schemes and the average-interpolation wavelets. It provides smooth functions across grids of different resolutions. In practice, it turns out that the linear impulse is quite well conserved. To make the code more efficient in terms of load balancing (computational cost and memory cost), both the parallelization and the mesh management have been redesigned. The method is successfully tested on a vortex ring simulation using various mesh size transitions. The chapter also focuses on the gain resulting from the combination of the immersed boundary approach and refined grids. The simulation of the flow past a sphere at Reynolds 300 is carried out. Multiple resolutions are used to capture the wake up to 38 diameters. The computational gain is highlighted in comparison to the results obtained in Chapter 1.

The second part of the manuscript is devoted to the numerical investigation of various vortex-systems using vortex methods as efficient numerical investigation tools, i.e., the combined VPM-PFM (called VIC-PFM in those chapters as they consist of previous published reports where the term VIC was used instead of VPM) and the regularized vortex filament method. The focus is on the physics of the investigated problems.

- **Chapter 4** investigates the effect of the velocity deficit due to the friction drag on the near-wake rollup of a vortex-sheet at $Re = 10^6$ up to 10 wingspan downstream of a lifting body. Space-developing LES, with and without velocity deficit, have been carried out. First, the space-developing rolled-up vortex pair with friction drag is analyzed and characterized (circulation and tangential velocity profiles, vortex-core radius, residual inner-core velocity, turbulence, etc.). Second, the two simulations are compared in terms of net-result after rollup. The effect of the velocity deficit effect is also analyzed in the case of a time-developing simulation.
- **Chapter 5** investigates the “end-effect” phenomena. The end-effect occurs when the total circulation of the vortex system varies longitudinally. The associated pressure wave induces a longitudinal velocity in the vortex core. The system possibly becomes unstable, leading to the creation of a helical instability. A comparison to the experiment of a counter-rotating vortex-pair (carried out by the CNRS-IRPHE) is done. Moreover the space-developing end-effect is investigated in the case of an accelerated vortex sheet.

Finally, the effect of an inviscid ground on the Crow(-like) instability growth is considered in Appendix B. The regularized vortex filament method was used. Two cases were investigated. First, a comparison to an experiment (carried out by CNR-IRPHE) is done. The simulation replicates as much as possible the experimental setup, that is, the evolution of the long-wave instability of a pair of fat vortices put in ground effect, and as long as viscous effect are negligible. Second, the influence of various initial perturbation levels of a thin vortex pair is analyzed.

Chapter 2

Development of an immersed boundary technique within the combined Vortex Particle-Mesh/Parallel Fast Multipole Method

This chapter was written for the ECCOMAS 2010 conference proceeding as a paper entitled “Development of an immersed boundary method using boundary elements within a vortex-in-cell/parallel fast multipole method”. This article is essentially reproduced as it was, yet with some modifications and with some additional results.

Abstract *This paper presents a method for the simulation of incompressible flows past arbitrary shape bodies. The method couples a combined vortex particle-mesh (VPM) and parallel fast multipole (PFM) method to an Immersed*

Boundary (IB) approach. The Navier-Stokes equations in the vorticity-velocity formulation are solved. The Poisson equation for the streamfunction is solved efficiently in an unbounded domain by means of a direct solver; in each subdomain, the boundary conditions are provided by the PFM method which has a global view of the whole vorticity field.

Because the vorticity field is a compact field, a tight grid can be used. This makes an efficient method for unbounded vortical flows.

An IB method is used in order to deal with complex body geometries. In vorticity-velocity formulation, imposing the no-slip condition requires to solve a vector integral equation. This is done here using a boundary element method. The obtained panel strengths provide the equivalent vorticity flux at the wall.

The method has been successfully validated using a well-referenced test case, which consists in the flow past a sphere at $Re = 300$. It is then also tested on the impulsively started flow past a very low aspect ratio wing at $Re = 3000$.

2.1 Introduction

This paper describes the extension of the VPM-PFM method in order to compute external flows past arbitrary geometries at moderate Reynolds number. The methodology is based on an Immersed Boundary (IB) approach. This was first developed by R. Peskin [65] for velocity-pressure based Navier-Stokes solvers for the simulation of inner-heart flows. See also the review of Mittal et al. [58] for more details on IB methods.

Several previous works have dealt with combining a vortex method to an immersed boundary technique. Morgenthal et al. [61] have also developed a 2-D IB method for VPM algorithms. An influence matrix (Particle-Particle Particle-Mesh algorithm, the P^3M) is used in order to capture the under-resolved field in the near-wall region. Cottet and Poncet [17], and Poncet [70] developed an IB method in a VPM code using potential sources instead of vorticity sources. In their approach, the body surface and the potential source are discretized over a few grid points. These sources are solved at every step.

Brinkman penalization methods have also been used to capture such flows. (See Coquerelle and Cottet [15] and Rossinelli et al. [72])

We here capture the discontinuities of the velocity field, due to slip, by

means of a panel method, thus in a step reminiscent of a sharp interface method (see [85, 54, 52, 92]). In a second step however, this jump, corresponding to a vorticity flux has to be passed to the grid. Our approach is thus indeed an immersed boundary method as the wall discretization is finally mollified.

Vortex methods solve the Navier-Stokes equations in the vorticity-velocity formulation. The vorticity field is discretized using N vortex particles; each carrying a strength vector and moving with the local velocity vector. The convection is thus done in a Lagrangian manner; leading to negligible dispersion error. The evolving equations for the strength and the position of a vortex particle p read

$$\frac{d\boldsymbol{\omega}_p}{dt} = (\nabla \mathbf{u}(\mathbf{x}_p)) \cdot \boldsymbol{\omega}(\mathbf{x}_p) + \nu \nabla^2 \boldsymbol{\omega}(\mathbf{x}_p) \quad (2.1)$$

$$\frac{d\mathbf{x}_p}{dt} = \mathbf{u}(\mathbf{x}_p) \quad \forall p = [1 \dots N]. \quad (2.2)$$

The velocity field $\mathbf{u}$ is computed from the vorticity field $\boldsymbol{\omega}$ through the stream-function Ψ by solving the following Poisson equation

$$\nabla^2 \Psi = -\boldsymbol{\omega}. \quad (2.3)$$

The velocity is then computed using the relation

$$\mathbf{u} = \nabla \times \Psi + \mathbf{U}_0. \quad (2.4)$$

This method entails to “translate” the particle strength onto a grid strength. This is here done using the redistribution procedure. The redistribution is also used to handle Lagrangian distortion in particle methods. With time, particles tend to cluster or deplete some region of the flow. Redistribution proceeds by periodically generating a new set of particles (from the old set) on a regular lattice.

This chapter is structured as followed. First, the algorithm is described in details. Secondly, the proposed methodology is validated against the flow past a sphere at $Re = 300$; a well-referenced test case in the literature. Finally, the method is applied to the impulsively started flow past a profiled-body consisting in a low aspect ratio NACA0012 wing.

2.2 An Immersed Boundary Method Within a Vortex Particle-Mesh Method

The coupling of an IB method is well-suited for codes using the velocity-pressure formulation because velocity components tend to zero at the wall. An IB method in a vorticity-based Navier-Stokes solver however has to handle a discontinuous field. Indeed, the vorticity is non zero on the fluid side and maximum at the wall and it is zero inside the body up to the wall. In the present approach, the wall treatment relies on a symmetrized vorticity field with respect to the wall. The presented algorithm is mainly based on the approach developed by Ploumhans [66, 68] and adapted to the VPM-PFM method [13].

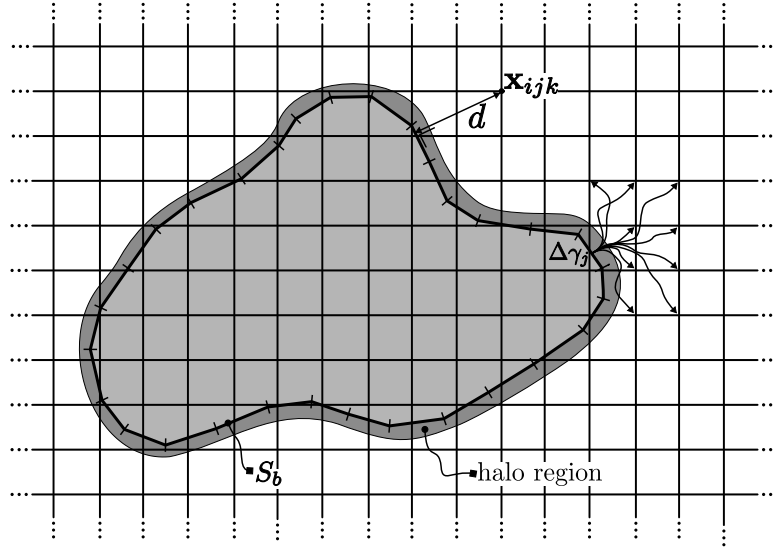


Figure 2.1: General setup of the IB method.

The algorithm reads: At time t^n , the vorticity field is discretized using a set $\mathcal{P}_1$ of particles characterized by their position and their strength: $(\mathbf{x}_p^n, \alpha_p^n)$ for $p \in [1 \cdots N]$ where N is the number of particles. Moreover, the arbitrary body shape is discretized using a set $\mathcal{S}$ of flat triangular panels forming a closed surface and described by the position of the three vertices: $(\mathbf{x}_{1,j}^n, \mathbf{x}_{2,j}^n, \mathbf{x}_{3,j}^n)$ for $j \in [1 \cdots M]$ where M is the number of panels. The following 10 steps are performed to obtain the solution at time t^{n+1} .

1. **Mesh overlay & Random grid shifting.** A Cartesian grid with a

uniform resolution h , is laid over the support of vorticity, i.e. it covers the location of all particles and panels. The particle/panel position are randomly shifted between $-0.5h$ and $0.5h$, in each direction, in order to distribute the IB discretization error over the body surface. Indeed, the body discretization crosses arbitrarily the Cartesian grid.

2. **Wall distance.** The grid points are tagged relative to the wall distance. For each grid point $\mathbf{x}_{ijk} = (x_i, y_j, z_k) = (x_0 + ih, y_0 + jh, z_0 + kh)$, a point $\mathbf{x}_{wall}$ on the wall surface exists such that $d = \min(|\mathbf{x}_{ijk} - \mathbf{x}_{wall}|)$, d is then the distance from $\mathbf{x}_{ijk}$ to the wall. In practice, the distance to the wall d is here computed as done in Daeninck et al. [21]: first the nearest panel to the point $\mathbf{x}_{ijk}$ is computed by means of the nearest neighbor search functionality of the fast multipole panel method. Second, the distance is defined as the normal component in the local reference frame of the nearest panel.

3. **Evaluation of the vorticity field on the grid.** We construct a set $\mathcal{P}_2$ of image vortex particles inside the body such that the spurious vorticity flux at the wall, corresponding to the case without those particles, is cancelled. Of course, there is always a small spurious effect due to the non-zero local curvature, R , of the wall, as compared to h (need $R/h \gg 1$). The methodology is sketched in Fig. 2.2. For each near-wall particle, an equivalent image particle is created inner the body which has the same tangential vorticity components and an opposite normal vorticity component. As we use a “halo” region (see Ploumhans [66, 68]), the union of the sets $\mathcal{P}_1$ and $\mathcal{P}_2$ does not cover the entire flow and a small gap exists around the wall (i.e. $d/h < 0.25$ where d is the distance to the wall). For these near-wall grid points, using the particle redistribution scheme provides an “incomplete” field. The near-wall particle weight thus has to be extrapolated from the flow along the normal to the wall. This

is here done using one-side second-degree polynomial functions $\mathbf{P}$:

$$\left\{ \begin{array}{l} \mathbf{P}_{t,n}(\mathbf{x}_{wall} + h\mathbf{n}) = \boldsymbol{\alpha}_{t,n}(\mathbf{x}_{wall} + h\mathbf{n}) \\ \mathbf{P}_{t,n}(\mathbf{x}_{wall} + 2h\mathbf{n}) = \boldsymbol{\alpha}_{t,n}(\mathbf{x}_{wall} + 2h\mathbf{n}) \\ \frac{d\mathbf{P}_t}{dn}(\mathbf{x}_{wall}) = 0 \text{ for tangential components} \\ \mathbf{P}_n(\mathbf{x}_{wall}) = 0 \text{ for normal component} \end{array} \right. \quad (2.5)$$

where $\mathbf{P}_n = \nabla \mathbf{P} \cdot \mathbf{n}$ and $\mathbf{P}_t = \mathbf{P} - \mathbf{P}_n \mathbf{n}$.

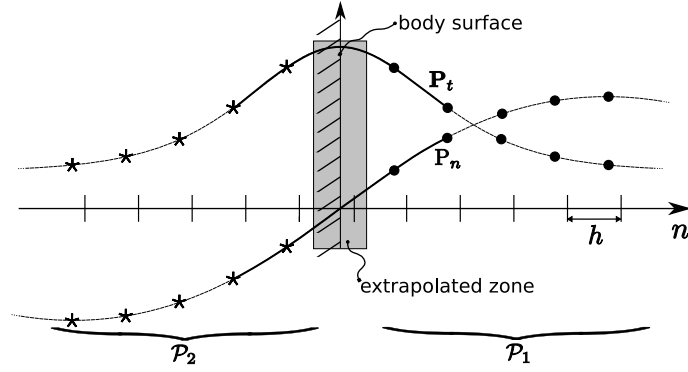


Figure 2.2: General description of the methodology for computing the near-wall vorticity field in the local reference frame. The “bullets” (•) and the “stars” (*) respectively represent the set $\mathcal{P}_1$ of outer-body particles and the set $\mathcal{P}_2$ of inner-body image particles. To compute the vorticity field at the near-wall grid point in the “halo” region (here shaded region), one-sided extrapolating functions are used (differentiating the normal and the tangential components).

Afterward, the weights of the particle sets ($\mathcal{P}_1$ and $\mathcal{P}_2$) are redistributed onto the complete grid (no halo region) using the M'_4 scheme:

$$\boldsymbol{\alpha}_p^n \xrightarrow{M'_4} \boldsymbol{\alpha}_{ijk}^n. \quad (2.6)$$

More precisely,

$$\boldsymbol{\alpha}_{ijk}^n = \sum_{p=0}^N M'_4(x_i - x_p) M'_4(y_j - y_p) M'_4(z_k - z_p) \boldsymbol{\alpha}_p^n. \quad (2.7)$$

where,

$$M_4'(x) = \begin{cases} 1 - \frac{5}{2}|x|^2 + \frac{3}{2}|x|^3, & |x| \leq 1, \\ \frac{1}{2}(2 - |x|)^2(1 - |x|), & 1 < |x| < 2, \\ 0, & |x| \geq 0. \end{cases} \quad (2.8)$$

The M_4' scheme has good conservation properties, see Monaghan [60], Cotted and Koumoutsakos [16] and Winckelmans [87]. The vorticity, the linear impulse and the angular impulse are conserved.

4. **No-slip condition.** The vortex panel vector intensity is computed such that the tangential vector velocity underneath each panel is canceled (see Fig. 2.1).

Imposing the no-slip flow condition at the wall requires to solve a vector integral equation. This is done here using a Boundary Element Method (BEM). For vortex panels of vector strength $\Delta\gamma$:

$$\begin{aligned} \frac{1}{2}\Delta\gamma(\mathbf{x}) \times \mathbf{n} + \frac{1}{4\pi} \int_S \frac{1}{|\mathbf{x} - \mathbf{x}'|^3} \cdot (\mathbf{x} - \mathbf{x}') \times \Delta\gamma \, dS(\mathbf{x}') &= \mathbf{U}_0 + \mathbf{u}_\omega(\mathbf{x}) \\ &= \Delta\mathbf{U}_{slip} \end{aligned} \quad (2.9)$$

where $\mathbf{u}_\omega$ is the solution of Eq. (1.2) and (1.3), the vortex-field induced velocity. This equation is solved using an iterative method (a under-relaxed Jacobi type algorithm). The surface body is discretized using triangles. The matrix-vector multiplications (i.e. particle-panel and panel-panel interactions) in the linear solver are evaluated efficiently by means of the PFM method.

The RHS, i.e. the residual velocity $\Delta\mathbf{U}_{slip}$, is also evaluated using the PFM method. The particles are considered for this step as “point” particles so as to have an entire view of the vorticity field. A “halo” region, where no particle can exist, is defined to avoid any spurious vorticity flux at the wall see Ploumhans et al. [67, 68]. Typically, one fixes $d_{halo}/h = 0.20 \cdots 0.25$. Particles entering this zone are rebounded with respect to the wall.

5. **Vorticity flux.** A Lighthill’s model of vorticity flux at the wall is used [16, 67, 68]. For each grid point $\mathbf{x}_{ijk}$, the net circulation increase

$\Delta\alpha_{ijk,cons}^n$, due to the diffusion of the vortex panels within the flow between t^n and t^{n+1} , is computed using a conservative scheme. Let's consider a panel with an area Ω and an intensity $\Delta\gamma$. Defining an infinitesimal surface $dS = d\xi d\eta$, the net circulation increase over a time Δt is given by the following formula

$$\Delta\alpha_{ijk} = \int_0^1 \frac{d\alpha_{ijk}}{d(t/\Delta t)} d(t/\Delta t),$$

where

$$\begin{aligned} \frac{d\alpha_{ijk}}{d(t/\Delta t)} &= \frac{1}{2}\Delta\gamma \left([\text{erfc}(u)]_{(z_k+h/2)/\sqrt{4\nu t}}^{(z_k-h/2)/\sqrt{4\nu t}} \right. \\ &\quad \left. \int_S [\text{erfc}(u)]_{(x_i+h/2-x(\xi,\eta))/\sqrt{4\nu t}}^{(x_i-h/2-x(\xi,\eta))/\sqrt{4\nu t}} [\text{erfc}(u)]_{(y_j+h/2-y(\xi,\eta))/\sqrt{4\nu t}}^{(y_j-h/2-y(\xi,\eta))/\sqrt{4\nu t}} d\xi d\eta \right). \end{aligned} \quad (2.10)$$

ξ and η are the local integration coordinates. $h_l/2$ is equal to z_k if $z_k < h/2 + d_{halo}$ and equal to $h/2$ otherwise. This formula is here numerically integrated i) in time using four-point Gauss quadrature and ii) over the triangle using the three-point Hammer rule. This formula can be analytically integrated if the panel is a rectangle ($S = f \times b$). Eq. (2.10) can then be rewritten as followed (see Ploumhans [66, 68])

$$\begin{aligned} \frac{d\alpha_{ijk}}{d(t/\Delta t)} &= \nu t \Delta\gamma \left([\text{erfc}(u)]_{(z_k+h/2)/\sqrt{4\nu t}}^{(z_k-h/2)/\sqrt{4\nu t}} \right. \\ &\quad \left([\text{ierfc}(u)]_{(x_i-h/2-b/2)/\sqrt{4\nu t}}^{(x_i+h/2-b/2)/\sqrt{4\nu t}} - [\text{ierfc}(u)]_{(x_i-h/2+b/2)/\sqrt{4\nu t}}^{(x_i+h/2+b/2)/\sqrt{4\nu t}} \right) \\ &\quad \left([\text{ierfc}(u)]_{(y_j-h/2-f/2)/\sqrt{4\nu t}}^{(y_j+h/2-f/2)/\sqrt{4\nu t}} - [\text{ierfc}(u)]_{(y_j-h/2+f/2)/\sqrt{4\nu t}}^{(y_j+h/2+f/2)/\sqrt{4\nu t}} \right). \end{aligned} \quad (2.11)$$

In practice, if the triangular panels are equilateral enough, the panels can be considered as rectangular (i.e. $f \triangleq b \triangleq \sqrt{S_\Delta}$). Otherwise, the integral is done numerically. In the present work, both are available and needed (as triangles are not always “almost equilateral”) such as in the discretization of an airfoil, see later. Moreover, the scheme can be made conservative through a weighted correction as developed by Ploumhans

et al. [68],

$$\Delta\alpha_{ijk,cons}^{wall} = \Delta\alpha_{ijk} + \frac{|\Delta\alpha_{ijk}|^2}{\sum_{r,s,t} |\Delta\alpha_{(r,s,t)}|^2} \left(S\Delta\gamma - \sum_{r,s,t} \Delta\alpha_{(r,s,t)} \right).$$

6. **Poisson equation.** Solving the Poisson equation $\nabla^2 \Psi_{ijk}^n = -(\omega_{ijk}^n + \Delta\omega_{ijk}^n)$ using the mathematical library **FISHPACK** (for more details concerning this library, see [80, 78, 79]). $\Delta\omega_{i,j,k}$ is the “representation” of the panel vorticity onto the grid. It is computed either using the redistribution scheme or using the diffused-panel field. The former provides robust behavior for impulsively started flows and the latter provides better results in regime mode. The required boundary conditions are obtained using the PFM method.
7. **Evolution equation.** The RHS of Eq.(2.1) and (2.2) are computed using Finite Differences (FD)

$$\mathbf{u}_{ijk}^n = \nabla \times \Psi_{ijk}^n$$

and

$$\frac{d\omega_{ijk}^n}{dt} = (\nabla \mathbf{u}_{ijk}^n) \cdot \omega_{ijk}^n + \nu \nabla^2 \omega_{ijk}^n.$$

Notice that the velocity and the stretching terms are computed using centered second-order FD. The diffusion term is computed using a 27 points “cube” scheme instead of the “cross” scheme (see [13]), which provides less sensitivity to the grid orientation.

8. **Vorticity divergence.** With time, the vorticity field does not remain an image of the velocity curl for several reasons. i) The interpolation procedure (particle-mesh, mesh-particle) using the M_4' scheme does not conserve the divergence of the vorticity field. ii) The stream function Ψ obtained by solving the Poisson equation, using **FISHPACK**, is not divergence free. Moreover, a “collocated” lattice is used instead of a “staggered” lattice. The numerical scheme is thus not fully consistent: the numerical Laplacian operator does not match with the numerical double rotational operator ($\nabla_h \times (\nabla_h \times \Psi) \neq -\nabla_h^2 \Psi = \omega$).

The vorticity field thus has to be regularly corrected in order to project it into a divergence free space, following a methodology first developed by Cottet and Poncet [17], and Poncet [69]. Such a methodology was also used by Cocle et al. [13]

The vorticity field $\boldsymbol{\omega}$ can be corrected to a divergence free field $\tilde{\boldsymbol{\omega}}$:

$$\tilde{\boldsymbol{\omega}} = \boldsymbol{\omega} - \nabla D, \quad (2.12)$$

by taking D such that

$$\nabla^2 D = \nabla \cdot \boldsymbol{\omega}. \quad (2.13)$$

Moreover, the presence of a wall entails boundary conditions for the above equations:

$$\tilde{\boldsymbol{\omega}} \cdot \mathbf{n} = 0 \Leftrightarrow \nabla D \cdot \mathbf{n} = \boldsymbol{\omega} \cdot \mathbf{n}. \quad (2.14)$$

The Poisson equation is solved using the same combination of the **FISHPACK** library and the PFM method. For the same reason that the vorticity field is not exactly divergence free (consistency between Laplacian operator versus double rotational operator), the above correction scheme (Eq. (2.13) and (2.12)) does not exactly provide a solenoidal field. However, it turns out that this step provides satisfactory results, if the vorticity field is smooth enough.

However the near-wall vorticity field is often locally weakly-resolved, due to the “halo” region. The evaluation of the divergence sources then leads to spurious corrections of the vorticity field. For this reason, the near-wall divergence sources (i.e. $d < 2h$) and the boundary condition are not taken into account during this correction step.

9. **Interpolation.** Grid quantities (i.e. velocity, time-derivative of strength and net increase of intensity due to panel diffusion) are interpolated onto

the particles using the M'_4 scheme

$$\begin{aligned} \mathbf{u}_{ijk}^n &\xrightarrow{M'_4} \mathbf{u}_p^n \\ \left(\frac{d\boldsymbol{\alpha}}{dt}\right)_{ijk}^n &\xrightarrow{M'_4} \left(\frac{d\boldsymbol{\alpha}}{dt}\right)_p^n \\ \Delta\boldsymbol{\alpha}_{ijk}^{n,wall} &\xrightarrow{M'_4} \Delta\boldsymbol{\alpha}_p^{n,wall} \end{aligned}$$

10. **Time integration.** The particle positions and strengths are integrated in time using respectively the second-order Leap-Frog (LF) scheme and the second-order Adams-Bashford (AB) scheme

$$\begin{aligned} \mathbf{x}_p^{n+1} &= \mathbf{x}_p^{n-1} + 2\Delta t \mathbf{u}_p^n. \\ \boldsymbol{\alpha}_p^{n+1} &= \boldsymbol{\alpha}_p^n + \frac{1}{2}\Delta t \left(3 \left(\frac{d\boldsymbol{\alpha}}{dt}\right)_p^{n-1} - \left(\frac{d\boldsymbol{\alpha}}{dt}\right)_p^n \right) + \Delta\boldsymbol{\alpha}_p^{n,wall} \\ t^{n+1} &= t^n + \Delta t. \end{aligned}$$

After particle redistribution (typically every 5 time steps), the AB integration scheme cannot be used. It is replaced by the second-order Runge-Kutta (RK) scheme. then the first step (predictor) is

$$\begin{aligned} \mathbf{x}_p^* &= \mathbf{x}_p^n + \Delta t \mathbf{u}_p^n \\ \boldsymbol{\alpha}_p^* &= \boldsymbol{\alpha}_p^n + \Delta t \left[\left(\frac{d\boldsymbol{\alpha}}{dt}\right)_p^n + \frac{\Delta\boldsymbol{\alpha}_p^{n,wall}}{\Delta t} \right] \\ t^* &= t^n + \Delta t, \end{aligned}$$

and the second step (corrector) is

$$\begin{aligned} \mathbf{x}_p^{n+1} &= \mathbf{x}_p^n + \frac{1}{2}\Delta t (\mathbf{u}_p^n + \mathbf{u}_p^*) \\ \boldsymbol{\alpha}_p^{n+1} &= \boldsymbol{\alpha}_p^n + \frac{1}{2}\Delta t \left[\left(\frac{d\boldsymbol{\alpha}}{dt}\right)_p^n + \frac{\Delta\boldsymbol{\alpha}_p^{n,wall}}{\Delta t} + \left(\frac{d\boldsymbol{\alpha}}{dt}\right)_p^* + \frac{\Delta\boldsymbol{\alpha}_p^{*,wall}}{\Delta t} \right] \\ t^{n+1} &= t^*. \end{aligned}$$

2.3 Validation: flow past a sphere at $Re = 300$

The present VPM-PFM-IB methodology has been validated on a well-known test case for which experimental and numerical data are available: the flow past a sphere at $Re = \frac{U_0 D}{\nu} = 300$. At such a Reynolds number, the flow is unsteady, thus making it a relevant test case, even though at moderate Reynolds number.

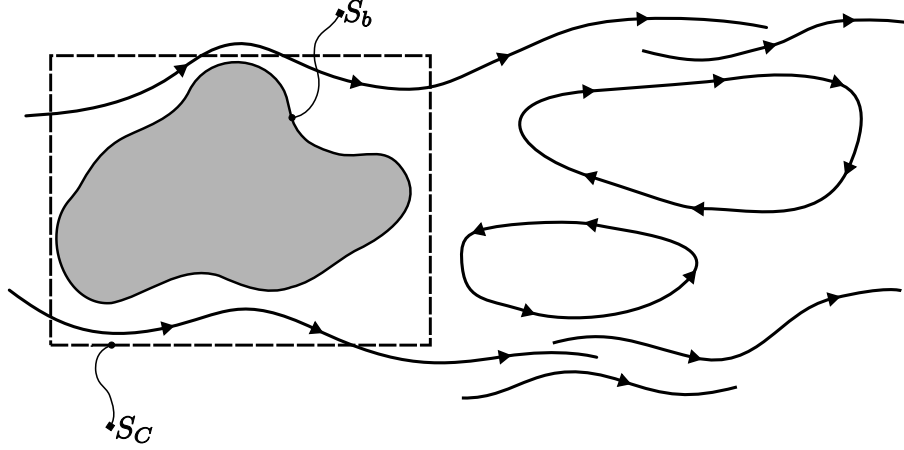


Figure 2.3: General view of the flow past an arbitrarily shape.

The force acting on a body can be computed from an exact formula developed by Noca et al. [62, 63] and applied to PIV (Particle Image Velocimetry) measurements. This formula was also successfully used in the context of vortex particle method by Ploumhans et al. [68] and Daeninck [21]. Let us consider a control volume V_C with a surface S_C which includes the body described by the simply-connected closed surface S_b , see Fig. 2.3. The force acting on S reads

$$\begin{aligned}
 \frac{\mathbf{F}}{\rho} = & -\frac{1}{\mathcal{N}-1} \frac{d}{dt} \int_{S_C+S_b} ((\mathbf{u} \cdot \mathbf{x})\mathbf{n} - \mathbf{u}(\mathbf{x} \cdot \mathbf{n}) + (\mathcal{N}-1)\mathbf{x}(\mathbf{u} \cdot \mathbf{n})) dS \\
 & + \int_{S_C} \left(\frac{1}{2}(\mathbf{u} \cdot \mathbf{u})\mathbf{n} - (\mathbf{n} \cdot \mathbf{u})\mathbf{u} \right) dS \\
 & - \frac{1}{\mathcal{N}-1} \int_{S_C} (\mathbf{n} \cdot \mathbf{u})(\mathbf{x} \times \boldsymbol{\omega}) dS + \frac{1}{\mathcal{N}-1} \int_{S_C} (\mathbf{n} \cdot \boldsymbol{\omega})(\mathbf{x} \times \mathbf{u}) dS \\
 & + \frac{1}{\mathcal{N}-1} \int_{S_C} \mathbf{x} \times (\mathbf{n} \times \nabla \cdot \mathbf{T}) dS + \int_{S_C} \mathbf{n} \cdot \mathbf{T} dS, \quad (2.15)
 \end{aligned}$$

where $\mathbf{T} = \nu (\nabla \mathbf{u} + (\nabla \mathbf{u})^T)$ and $\mathcal{N}$ is the number of dimension (here $\mathcal{N} = 3$).

This formula is particularly well adapted for vortex methods, because it only requires the knowledge of the velocity and its derivatives; the pressure field is not required. Note that the surface integral on S_b vanishes when $\mathbf{u} = \mathbf{0}$ at the wall (as for the flows considered in this thesis). The evaluation of the various fields are thus only needed on the control surface S_C . Here, the control volume is a cube centered at the sphere center with the edge size fixed to $1.6D$. The size of the control volume has no effect, as shown in Daeninck [21].

The mesh resolution is here fixed to $h/D = 1/75$. An adaptive time step is used such that both the maximal CFL criteria are $|\mathbf{u}|_{\max}\Delta t/h \leq 0.90$ and $|\boldsymbol{\omega}|_{\max}\Delta t \leq 0.75$. The resulting time step is $\Delta t U_0/D \simeq 8.4 \cdot 10^{-3}$. The wake is simulated up to 20 diameters downstream. The sphere is discretized using 20 480 quasi-equilateral triangles obtained by recursive subdivision of an icosahedron, see Fig. 2.4. The average surface of the triangles is $\sqrt{S_\Delta}/h = 0.93$. The flow was discretized using $\sim 100 \cdot 10^6$ grid points and $\sim 30 \cdot 10^6$ particles. To perform the complete simulation, 128 processors were required for 3 weeks in order to accumulate the time-history of the force.

Fig. 2.5 shows various views of the wake behind the sphere. The vortical structure, the so-called “hairpin” vortices, are clearly present (here visualized using the λ_2 criterion [41]). Fig. 2.6 shows a 2-D cut of the vorticity field for the entire flow (a) and a zoom in the near-wall region (b).

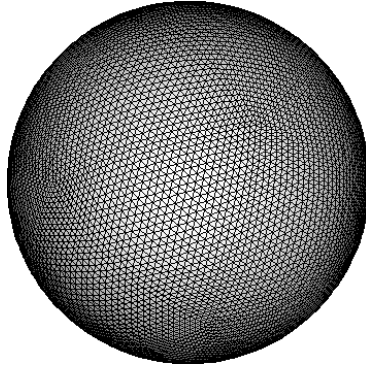


Figure 2.4: Sphere discretization using recursive subdivision of an icosahedron.

One defines the Drag coefficient (C_D)

$$C_D = \frac{\mathbf{F} \cdot \mathbf{e}_y}{\frac{1}{2}\rho U_0^2 (\pi D^2/4)} \quad (2.16)$$

where $\mathbf{e}_y$ is the streamwise unit vector. The Lift coefficient (C_L) is defined as

$$C_L = \frac{\|\mathbf{F} - (\mathbf{F} \cdot \mathbf{e}_y)\mathbf{e}_y\|}{\frac{1}{2}\rho U_0^2(\pi D^2/4)} \quad (2.17)$$

The time-evolution of the diagnostics (drag/lift coefficients) are presented in Fig. 2.7. Table 2.1 shows a comparison with reference results. The time-averaged value of C_D and C_L , as well as the Strouhal number ($Str = f D/U_0$), compare well with those of the literature. As for the oscillation amplitude ($\Delta C_D \triangleq \frac{1}{2}(\max C_D - \min C_D)$ and $\Delta C_L \triangleq \frac{1}{2}(\max C_L - \min C_L)$), they also compare quite well with those of the literature. It is worth noting that those latter values are more difficult to obtain. Moreover, results with higher quality are obtained when using the highest allowed time step. As a matter of fact, the larger the time step, the larger the number of grid points on which the diffusion process from the panels is captured. For a given a resolution, it thus exists a minimum time step. The ratio of effective diffusion length to the grid size can be estimated as

$$\frac{l_{diff}}{h} \simeq n_{diff} \frac{\sqrt{4\nu\Delta t}}{h}.$$

The effective diffusion should be such that it covers at least n_{diff} grid points for the diffusion to be well discretized. For a given Reynolds number ($Re = U_0 D/\nu$) and a set value for n_{diff} (here we use $n_{diff} = 4$), the previous equation leads to

$$\Delta t \frac{U_0}{D} \simeq \frac{1}{4} Re \left(\frac{h}{D} \right)^2.$$

It turns out that using a time step too small leads to oscillation of the force diagnostics due to an inconsistency between the solution of the Eq. (2.11) (that should cancel the residual slip velocity) and what the grid really receives after the diffusion process.

During the simulation, the mesh Reynolds numbers, based on the vorticity field ($Re_{mesh}^\omega = \frac{|\omega|_{\max} h^2}{\nu}$) and based on the velocity field ($Re_{mesh}^{\mathbf{u}} = \frac{|\mathbf{u}|_{\max} h}{\nu}$), were both maintained at a low level, of $\mathcal{O}(3 \cdots 7)$ (this ensuring a good resolution of the flow physics), see Fig. 2.8.(a). The divergence of the vorticity field was also maintained at an acceptable low level, of $\mathcal{O}(5 \cdot 10^{-4})$, see Fig. 2.8.(b).

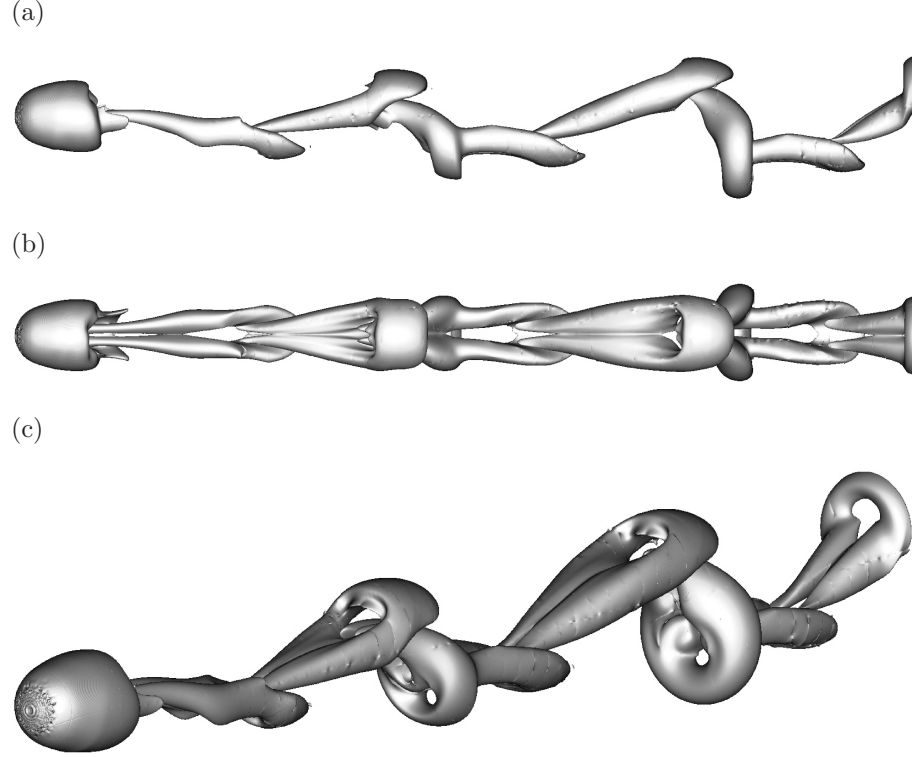


Figure 2.5: Isocontour of the λ_2 criterion: (a) top view, (b) side view and (c) offset view.

Table 2.1: Summary table for quantities of interest compared to the literature for the flow past a sphere at Reynolds 300.

	$\overline{C_D}$	ΔC_D	$\overline{C_L}$	ΔC_L	Str
Present work	0.672	$3.4 \cdot 10^{-3}$	0.065	$1.5 \cdot 10^{-2}$	0.134
Johnson & Patel [42]	0.656	$3.5 \cdot 10^{-3}$	0.069	$1.6 \cdot 10^{-2}$	0.137
Tomboulides & Orszag [83]	0.671	$2.8 \cdot 10^{-3}$	-	-	0.137
Constantinescu et al. [14]	0.655	-	0.065	-	0.136
Ploumhans et al. [68]	0.683	$2.5 \cdot 10^{-3}$	0.061	$1.4 \cdot 10^{-2}$	0.135
Georges [31]	0.661	$2.5 \cdot 10^{-3}$	0.066	$1.3 \cdot 10^{-2}$	0.134

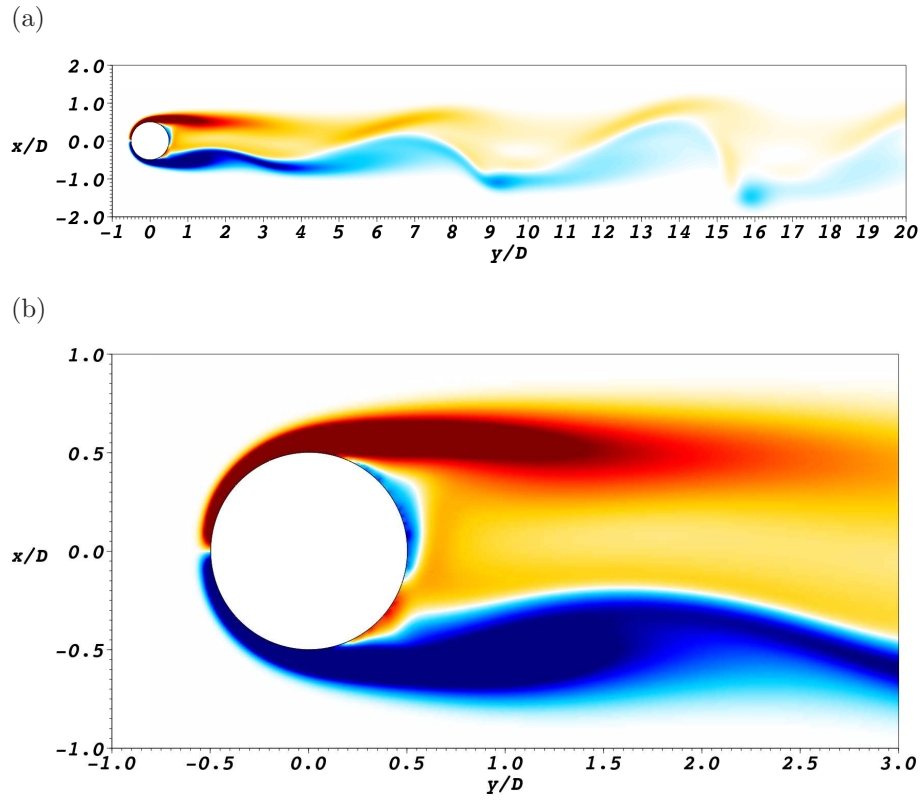


Figure 2.6: 2D slice of the z-component of the vorticity field $\frac{D}{U_0}\omega$: (a) global view and (b) zoom in the near-wall region. Saturation was used in order to see both the near wall vorticity and the wake vorticity.

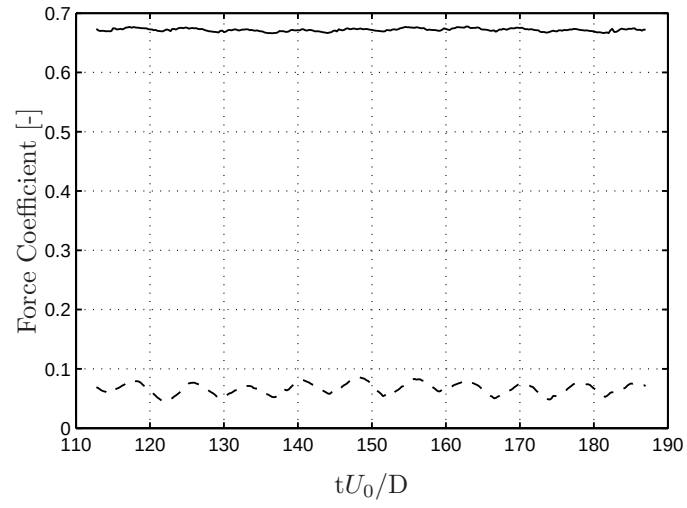


Figure 2.7: Time-evolution of the force acting on the sphere: drag coefficient (solid line) and lift coefficient (dashed line).

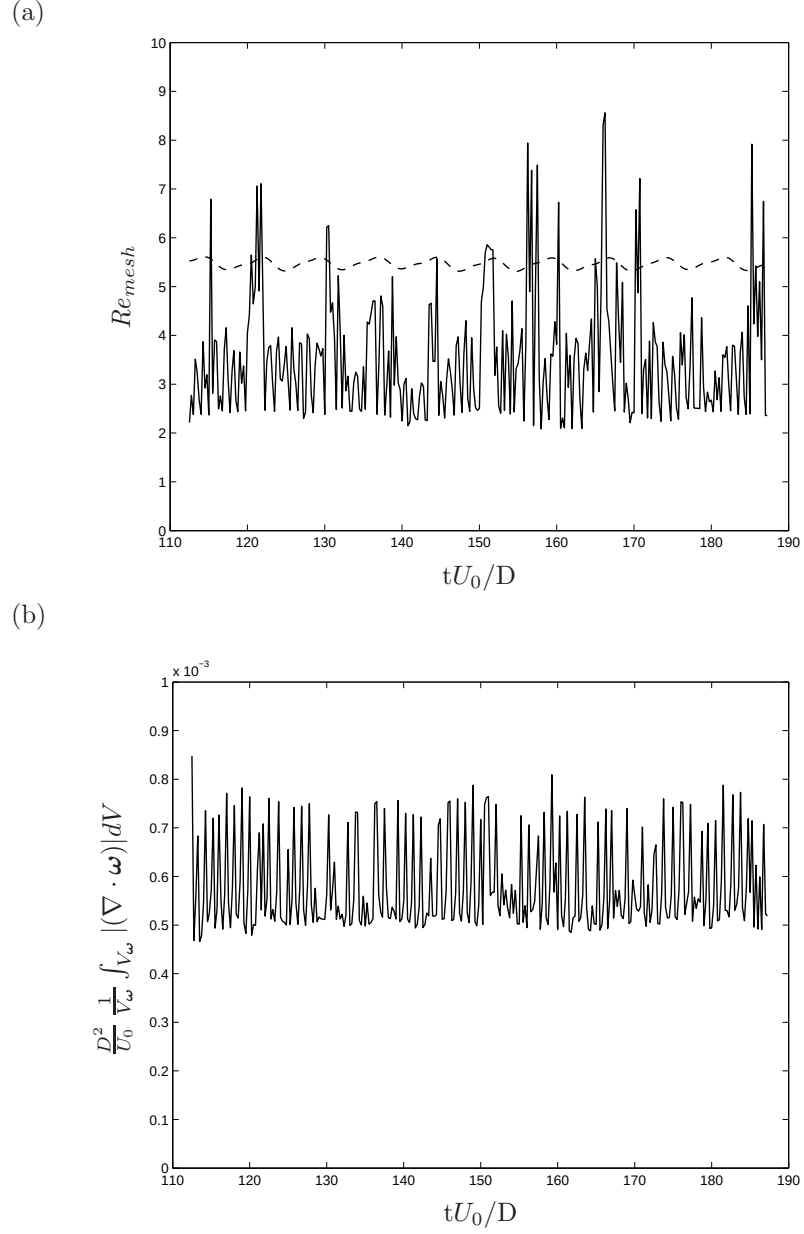


Figure 2.8: (a) Time-evolution of the mesh Reynolds number Re_{mesh}^{ω} (solid line) and Re_{mesh}^u (dashed line). (b) Time-evolution of the divergence of the vorticity field. V_{ω} is the volume where ω is non zero.

2.4 Application: flow past a very low aspect-ratio wing at $Re = 3000$

This section describes the results obtained when computing the flow past a lifting body such as a rectangular wing at moderate Reynolds number.

Initial condition & numerical setup The IB technique within the VPM-PFM method has been also applied to the flow past a profiled body (i.e. a “wing”) at a moderate Reynolds number $Re = cU_0/\nu = 3000$, where c is the wing chord. More precisely, we here consider a body consisting in a very low aspect-ratio rectangular wing ($A_R \triangleq b/c = 1$) using a NACA0012 profile section. The angle of attack is fixed to 5° . The wing surface is discretized using $\sim 152\,000$ triangular panels. For that purpose, the *Gmsh* mesh generator was used (for more details concerning this software, see Geuzaine and Remacle [32]). The panel sizes are twice finer ($\sqrt{S_\Delta}/c \simeq 3.5 \cdot 10^{-3}$) in the wingtip region than in the wing center. Fig. 2.9 shows the top view and the side view of the wing discretization. The flow is captured by a resolution fixed to $h/c = 3.0 \cdot 10^{-3}$. Notice that, in this section the field

$$\text{sign}(\omega_k) \log \left(1 + \frac{U_0}{c} |\omega_k| \right)$$

is used in color plots instead of the vorticity component ω_k itself. It allows to use the same colorbar limits when the vorticity component varies by several orders of magnitude.

Results The wing is impulsively started and the simulation was carried out up to $tU_0/D = 3.0$. At this time, the total number of grid points is roughly $120 \cdot 10^6$. 256 processors were used to compute the flow and ~ 72 hours of computational time were required. Notice that the origin of the coordinate system is centered with respect to the wingspan but shifted at the trailing edge. The coordinate system is such that x , y and z point to the spanwise, streamwise and normal directions respectively.

If we follow the Prandtl lifting line theory which is in fact only valid for high aspect-ratio wings ($A_R \geq 10$) and high Reynolds number flows, the lift

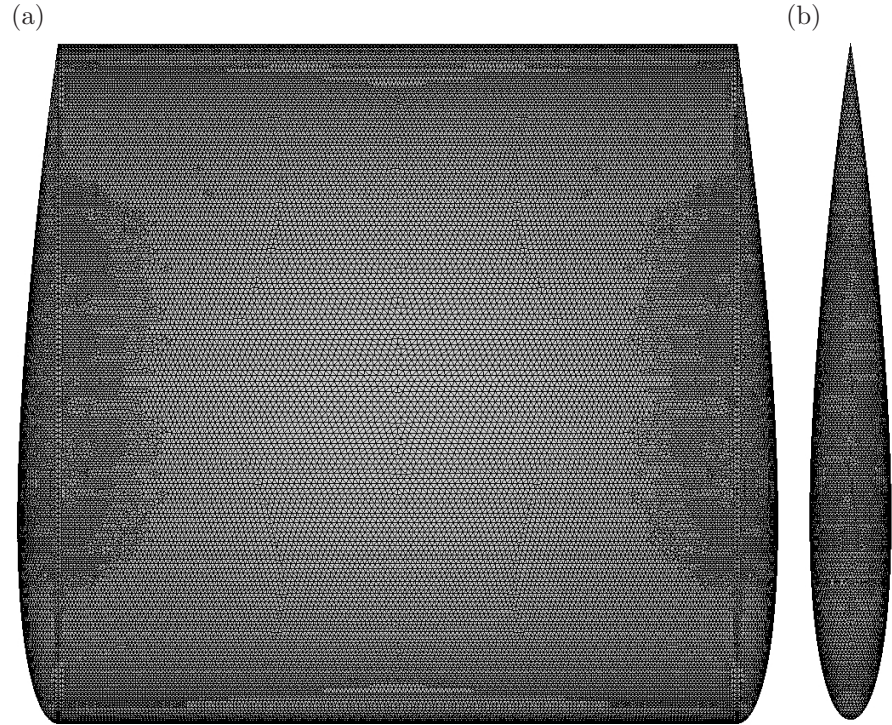


Figure 2.9: Discretization of the NACA0012 very low aspect ratio wing using $\sim 152\,000$ triangles. The resolution is twice coarser at the center. (a) top view and (b) side view.

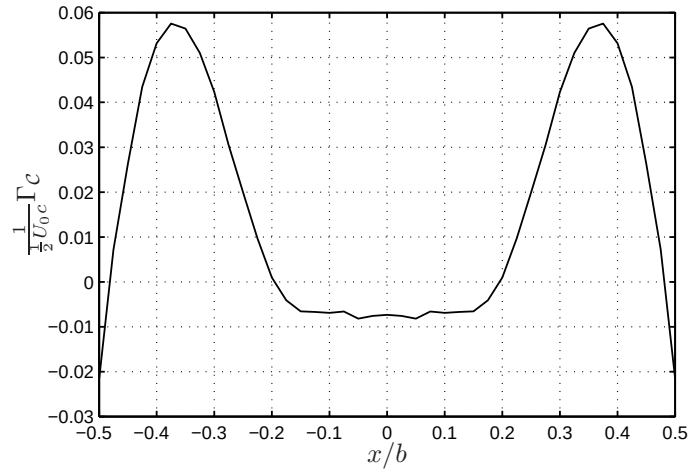


Figure 2.10: Flow past a very low aspect ratio wing: circulation around each profile at $t \frac{U_0}{b} = 3$.

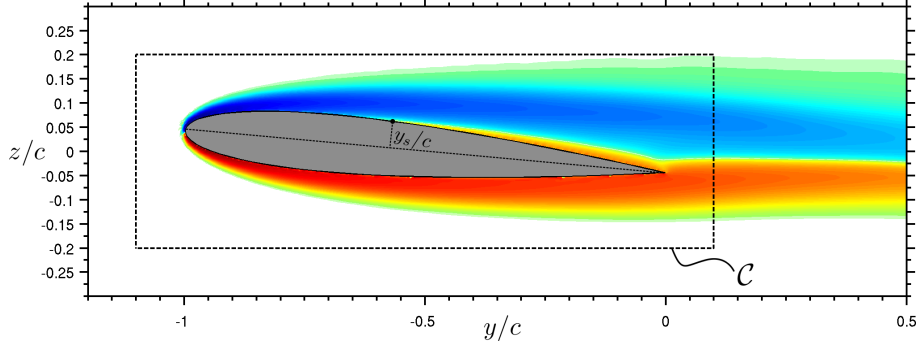


Figure 2.11: Flow past a very low aspect ratio wing: illustration of the contour $\mathcal{C}$ used for the circulation evaluation and of the separation point y_s/c .

would then be given by:

$$\frac{L_{tot}}{\rho} = U_0 \int_{-b/2}^{b/2} \Gamma_C(x) dx$$

where $\Gamma_C(x)$ is the circulation around a contour $\mathcal{C}$ enclosing the wing section at x :

$$\Gamma_C(x) = - \int_{\mathcal{C}} \mathbf{u} \cdot d\mathbf{l} = - \int_{S_C} \omega_x dydz.$$

Fig. 2.10.(b) and Fig. 2.11 show the obtained circulation and the contour chosen for the calculation respectively. One sees the separation point of the boundary layer (found at the position $y_s/c = 0.42$). Surprisingly, we observe that the circulation around the wing for the section at the center is negative. If one estimates the lift coefficient (reminding the severe assumptions associated to the Prandtl theory) through the integration of the circulation, one obtains $C_L \simeq 0.016$. This value has to be compared to the 2D lift coefficient: $C_l \simeq 2\pi \left(\frac{5\pi}{180} \right) \simeq 0.55$. The performance of the present wing is thus, as expected, very low (even if it is likely better than $C_L \simeq 0.016$ in reality). The preponderant factor is indeed the aspect ratio. The wingtip effect appears to be of high importance as the streamlines have to go around the leading edge extremities. However, the efficiency is even worse because the Reynolds number is too low; the boundary layer is laminar and separates sooner.

The slice of the spanwise vorticity component, at the center line, is plotted in Fig. 2.12.(a). The boundary layer detachment is observed as well as the starting

vortex. Fig. 2.12.(b) shows a 3D perspective view of 6 streamlines and 6 cuts of the streamwise vorticity component at $tU_0/D = 3.0$. The global formation of the trailing vortices is visualized. The counter-rotating vortices, due to the wing lift, are surrounded by opposite vorticity. This is generated by the flow circumventing the leading edge extremities. Fig. 2.13 and 2.14 respectively present various slices of the streamwise vorticity and the streamwise velocity at various downstream distances. Close to the wing, different recirculation zones are created. The vorticity field appears to be well captured and is smooth up to the wall in spite of the immersed boundary. In the wake region, the complex formation of the counter-rotating vortex pair is observed. The opposite vorticity remains quite strong. A residual trace of velocity deficit, created by the boundary layers, is still observed in the vortex-core. The inner-core velocity deficit is roughly $u_y/U_0 - 1 \simeq -0.1$.

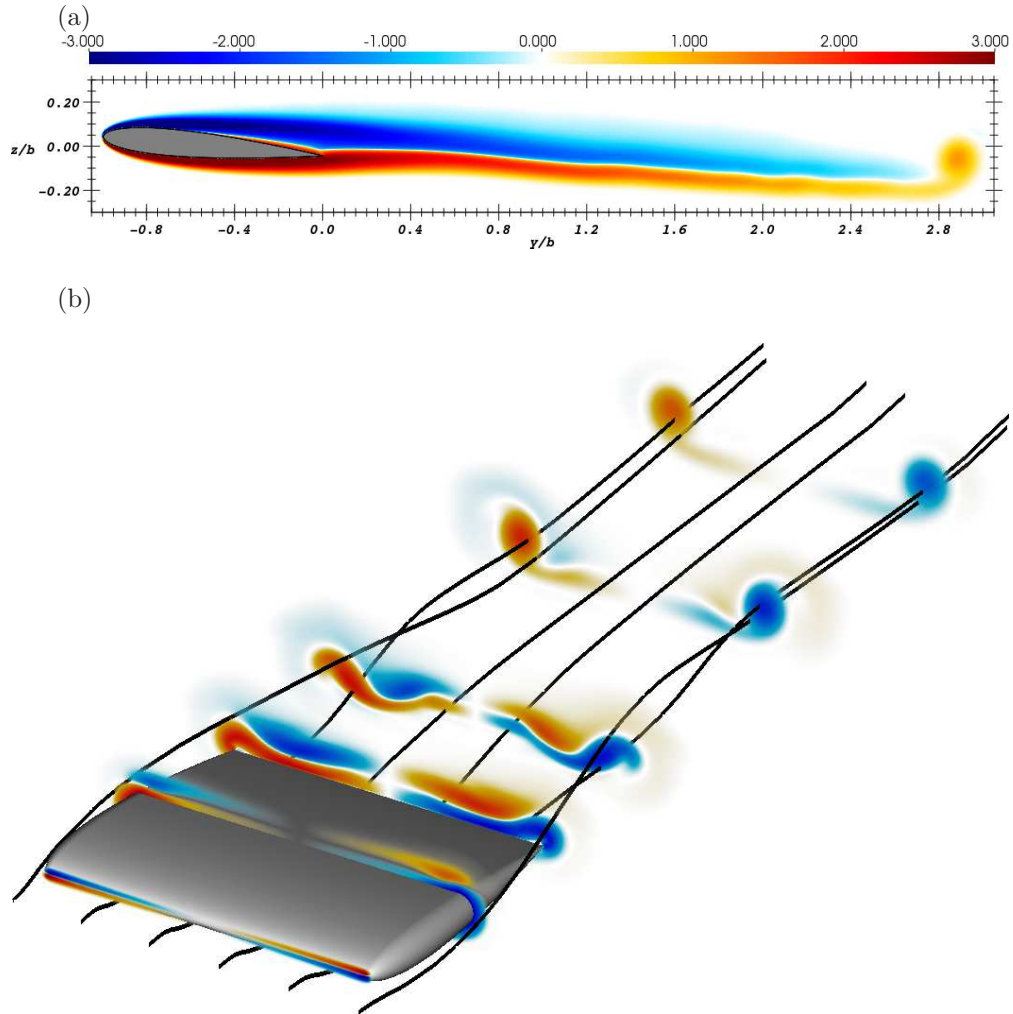


Figure 2.12: Flow past a very low aspect ratio wing. (a) Slice of the spanwise vorticity field at the center: $\text{sign}(\omega_x) \log(1 + \frac{c}{U_0} |\omega_x|)$. (b) 3D perspective view of 6 streamlines and 6 cuts of the streamwise vorticity component, located at $x/b = [-1.0, -0.5, 0.0, 0.5, 1.5, 2.5]$.

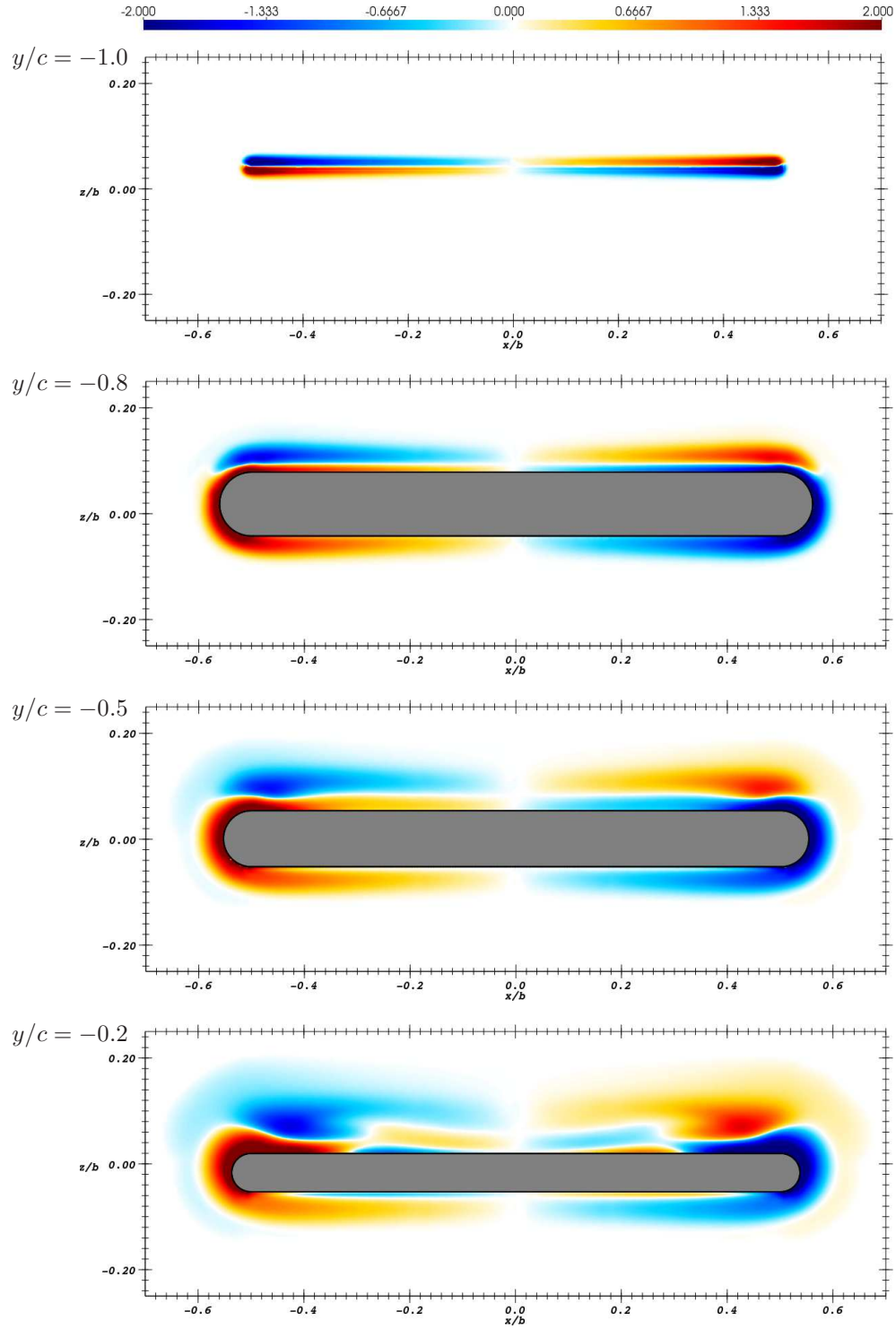


Figure 2.13: Flow past a very low aspect ratio wing. Downstream slices of the streamwise vorticity field $\text{sign}(\omega_y) \log(1 + \frac{c}{U_0} |\omega_y|)$.

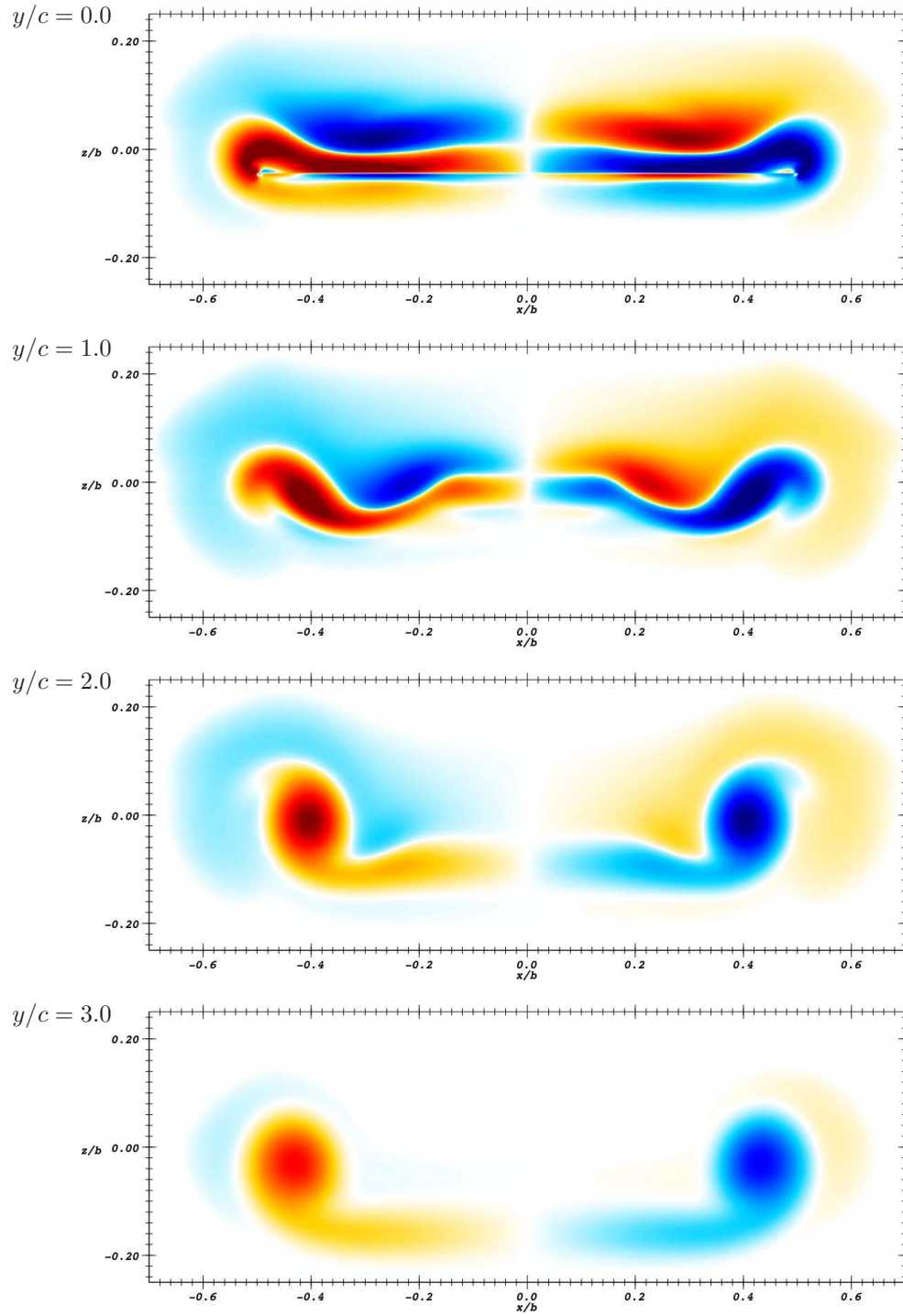


Figure 2.13: Cont'd.

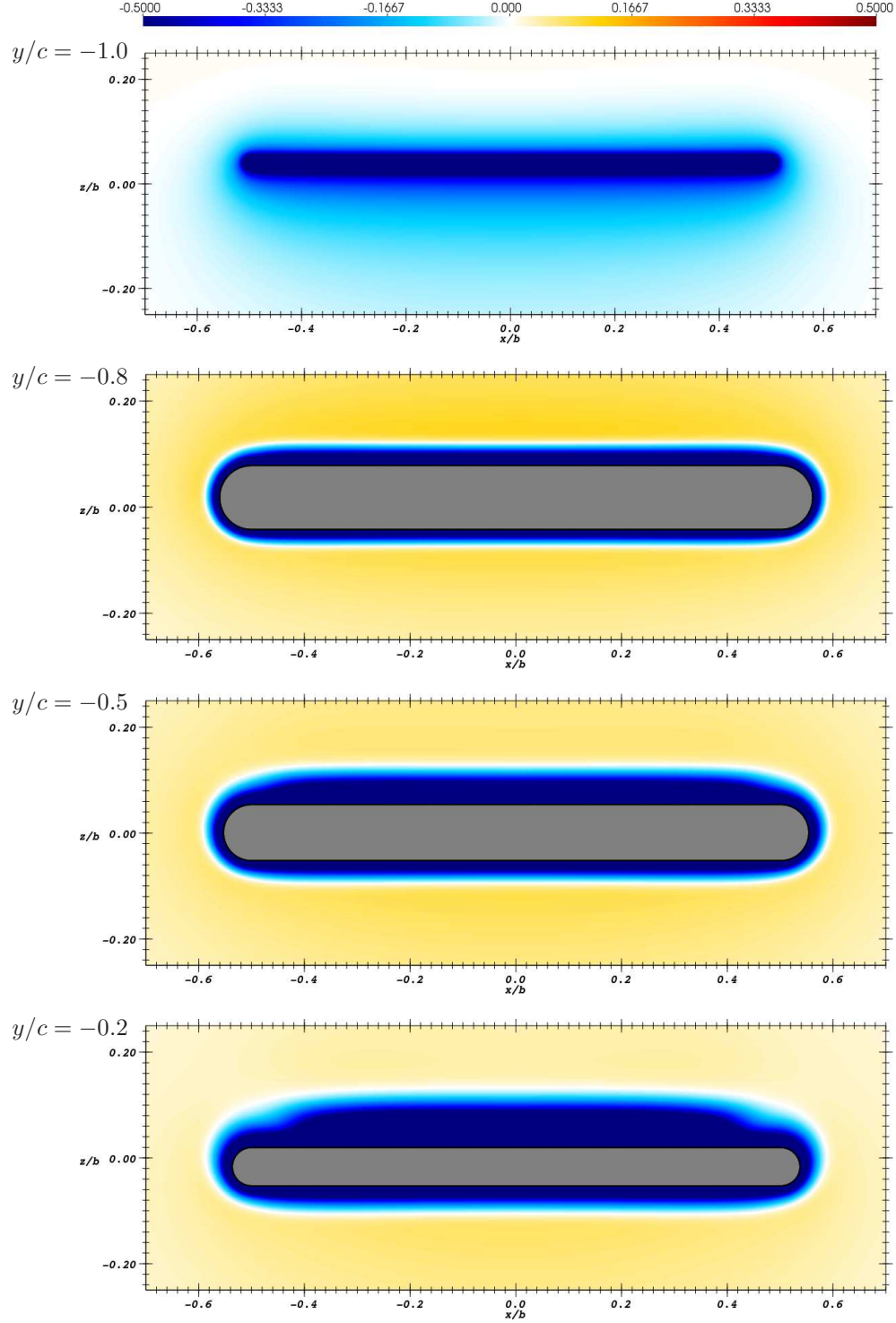


Figure 2.14: Flow past a very low aspect ratio wing. Downstream slices of the streamwise velocity field $u_y/U_0 - 1$.

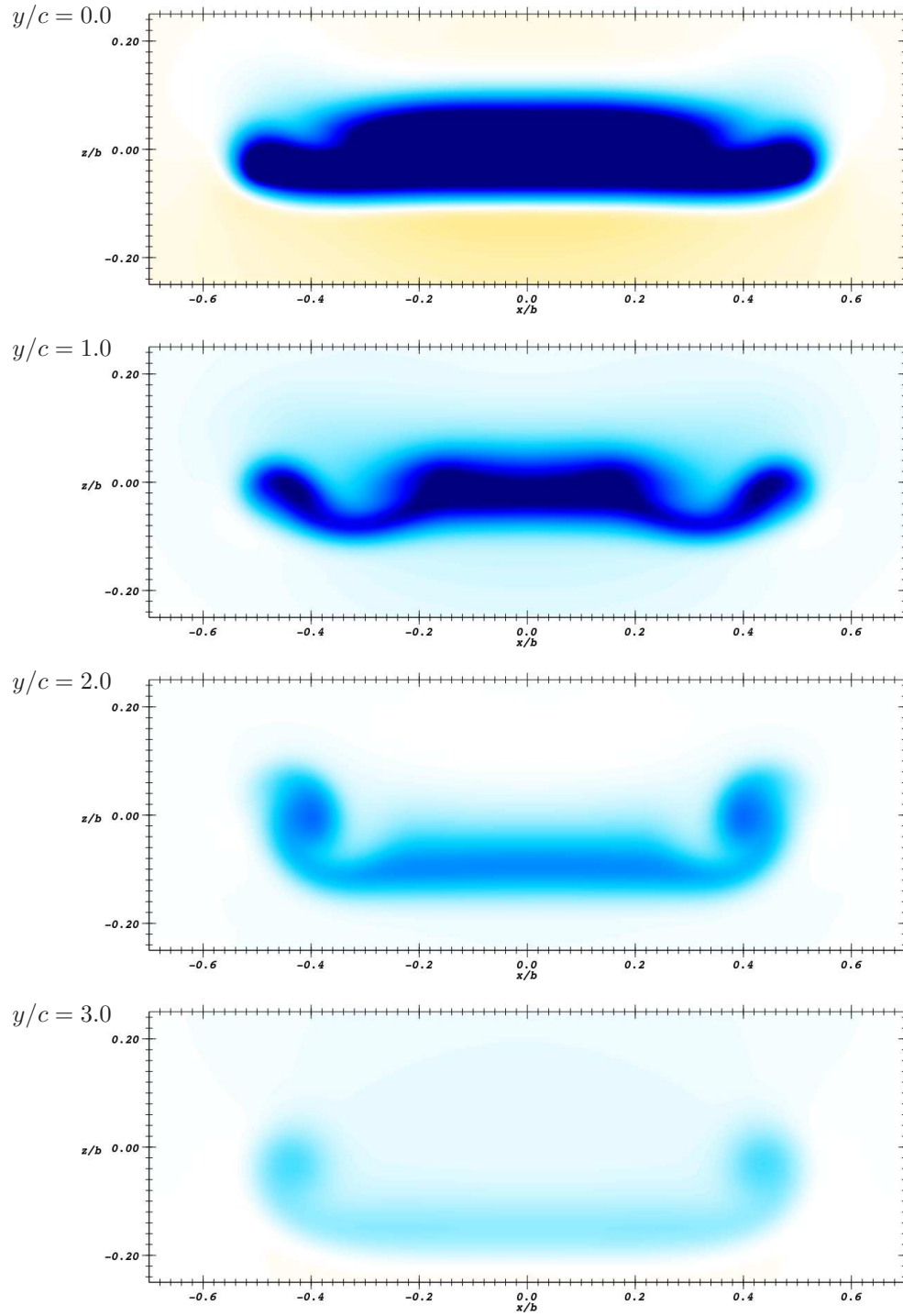


Figure 2.14: Cont'd.

2.5 Conclusion

An IB method within the VIC-PFM method has been exhaustively described. This approach provides satisfactory results for the simulation of the flows past arbitrary body shape, at moderate Reynolds number, in terms of flow dynamics and forces acting on the body.

The IB approach conserves the main advantages of the VIC-PFM method[13], i.e., i) the easy and efficient parallelization of any problem by locally solving the Poisson equation on each subdomain and ii) the unbounded boundary conditions are implicitly taken into account.

The method has been successfully validated against the test case of the flow past a sphere at $Re = 300$. Satisfactory results were obtained in terms of averaged values, amplitude- and frequency-oscillations.

The method has also been successfully applied to the simulation of the impulsively started flow past a very low aspect-ratio wing (NACA0012 profile, $AR = 1$) at moderate Reynolds number. The complete formation of lift-induced vortices is captured.

So far, the method uses uniform isotropic grids. The extension to a multi-resolution framework is described in the next chapter.

Chapter 3

A mesh refinement technique for the vortex particle-mesh method

3.1 Introduction

Numerical methods for external wall-bounded flows, that use a uniform resolution, rapidly lead to an unaffordable number of grid points due to the constraint of capturing the boundary layers (for which the thickness scales with $Re^{-1/2}$) as well as the wake. “Classical” vortex particle (VP) methods (i.e. using the Biot-Savart fast evaluation to compute the velocity field from the particle strength, as opposed to the hybrid vortex particle-mesh method, that uses a grid-based Poisson solver) are well-suited to automatically adapt to the local resolution. However, an appropriate redistribution is performed since a reset of the particle on a regular lattice has to be done in order to tackle the clustering/depletion issue. Three main approaches exist to handle varying resolution. i) A global mapping is used in order to apply the classical redistribution. ii) The mesh is composed of multiple uniform blocks which have different resolutions. The redistribution procedure is then adapted to this particular mesh. iii) Wavelet-particles (see Bergdorf and Koumoutsakos [3]) are used. That allows very fine local refinement.

Ould Salihi [64] (see also Cottet and Koumoutsakos [16]) used a mapping of the redistribution lattice to perform non-uniform simulations. Ploumhans [68] extended this approach in 3-D to achieve the simulation of the flow past a sphere for various moderate Reynolds numbers.

Daeninck [21] developed a redistribution technique on hierarchically refined grids. Classical redistribution scheme are used in order to redistribute a particle of size $h_l/h = 1/2, 1$ and 2 on a grid h . The length scale, h_l , and the strength of this particle are modified to remain conservative on the grid h . Moreover, a local and conservative smoothing of the solution is performed. The method was successfully applied to the flow past a hemisphere at $Re = 3000$. This method provides good results for “classical” vortex particle methods. The two first derivatives, however, oscillate too much when they are evaluated using finite differences. Another idea is based on average-interpolating wavelets (see Bergdorf et al. [2, 3] and Rossinelli et al. [73]). This approach was successfully applied to 2-D and 3-D convection-diffusion equations. In this chapter, a mesh refinement technique, based on the latter, is presented for the resolution of the 3-D Navier-Stokes equations using either Direct Numerical Simulations (DNS) or Large Eddy Simulations (LES). It is also worth mentioning that, in the context of bluff-body flows, Rossinelli et al. [72] used the GPU-enhanced performance to significantly reduce the computational cost.

We here use hierarchically refined grids to handle the multiple resolution. The methodology for solving the Poisson equation remains the same; i.e. the Poisson equation is solved using a fast grid solver (here using the *FISHPACK* library) and the required boundary conditions are provided using the Parallel Fast Multipole (PFM) method. However, the mesh management (generation & parallelization) has to be redesigned in order to have a better load balancing. The current VPM implementation only allows one block per cpu.

The present chapter is structured as follows. A mesh-block generator is first described. Second, a redistribution scheme with multiple resolution handling capability is depicted. And, finally, the overall methodology is illustrated and validated.

3.2 A basic multi-block generator

The parallel efficiency of a refined mesh technique lies in load balancing. In the first version of the VPM-PFM code, when using uniform resolution in the entire computational domain, the mesh was simply split over each processor by equal subdivisions. This approach leads to important drawbacks and limitations. A new multi-block mesh generator has thus been developed.

The mesh generator described hereinafter is based on hierarchically refined grids in the spirit of Adaptive Mesh Refinement (AMR, first developed by Berger and Olinger [4]). However, we here only present results using a prescribed mesh size function; it is thus not “adaptive”. The algorithm reads as follow:

1. **choice of a mesh size function:** the user provides a function $\delta(\mathbf{x})$ that defines the maximal mesh size to be respected over the whole grid.
2. **multi-block initialization:** a mother-box (level 0) is first created using the user-provided bounding box.
This mother-box is then equally split in $[n_x^0, n_y^0, n_z^0]$ in each direction. The $n_x^0 \times n_y^0 \times n_z^0$ boxes are set with a (user-provided) initial resolution h^0 .
3. **refinement strategy:** if the mesh resolution of a box (at a level l) is too coarse with respect to the prescribed function $\delta(\mathbf{x})$, this box is successively split into 8 child boxes with a resolution $h^{l+1} = \frac{1}{2}h^l$. This process is continued until the resolution constraint is satisfied for every point in the box.
4. **coarsening strategy:** if the mesh resolution is twice too fine for every point in a box (typically when the initial mesh size h^0 is too constraining), the mesh size of the box is doubled.
5. **smoothing:** the ensemble of boxes are smoothed to ensure that the mesh size of any box is never twice higher/lower than that of its neighbors.
6. **block agglomeration:** in order to reduce the computational cost associated with the evaluation of the boundary conditions (required for solving the Poisson equations and evaluated using the parallel fast multi-

pole PFM method), adjacent equal-resolution boxes are merged such as to form a bigger parallelipedic block with the same resolution.

7. **parallel implementation:** the boxes are finally distributed over each processor using the *METIS* Library[44, 43]. It optimizes the load balancing, by both equally sharing the computation and/or memory costs and by minimizing the communication cost. The former, associated to a block i , is computed as follows:

$$T_i = C_1 (C_2 N_{Surf}^i \log(N_{Part}^{tot}) + (1 - C_2) N_{Part}^i) + (1 - C_1) N_{Grid}^i,$$

N_{Surf}^i , N_{Part}^i , N_{grid}^i , N_{Part}^{tot} are the number of grid points located on the surfaces of the block i , the number of particles located in the block i , the number of grid points located on the block i and the total number of particles respectively. C_1 and C_2 lie between $[0 \ 1]$. C_1 determines the relative importance of the computational cost with respect to the memory cost. C_2 is intrinsic to the numerical code implementation and provides the relative computational time in the various steps of the algorithm. The latter, associated to the pair of blocks (i, j) , is simply defined to be proportional to the number of grid points on the common block surfaces:

$$T_{(i,j)} \triangleq T_{(j,i)} = C_3 N_{surf}^{(i,j)},$$

C_3 fixes the relative importance of the communication cost and depends on the hardware architecture. The values of C_1 , C_2 and C_3 are empirically found and also depend on the considered applications.

This new mesh generator library leads to major improvements:

- the number of subdivisions is independent of the number of processors,
- the blocks, when they become useless (no vortex particle in the block), are disabled “on-the-fly”,
- the blocks can be non-contiguous,
- the parallelization of the multi-block mesh over each processor is versatile and performed using the *METIS* [44, 43] library that allows to take into account both the computational cost and the communication cost.

3.3 A redistribution scheme based on average-interpolation

This section aims at describing the methodology to redistribute the strength of a particle, which has a size h_l , on a uniform grid which has a different resolution h . The basis of the following approach is an extension of the well-known redistribution schemes (Λ_3 , M'_4 , etc.). Those schemes redistribute on a uniform grid h a particle p of strength α_p and of size $h_l = h$ located at the position x_p using a compact kernel $\mathcal{K}$. In 1-D,

$$W(x_i) = \sum_p \alpha_p \mathcal{K} \left(\frac{x_i - x_p}{h} \right)$$

is the fraction of the particle strength α_p set to the grid location x_i . Depending on the choice of the kernel, the representation of the particle strength on the grid h conserves a certain number of moments; e.g., moments up to the third order are conserved when using the M'_4 scheme. The 3-D extension is straightforward, using tensor products:

$$W(\mathbf{x}_{ijk}) = \sum_p \alpha_p \mathcal{K} \left(\frac{x_i - x_p}{h} \right) \mathcal{K} \left(\frac{y_j - y_p}{h} \right) \mathcal{K} \left(\frac{z_k - z_p}{h} \right).$$

We here extend this methodology to handle multiple sets of particles which have various sizes $h_l/h = 1/2, 1, 2$. In particular, patches of particles of different resolutions can lie over a grid of parameter h . One thus has to i) “detect” what is the size of the particle that lives above every grid points in order to ii) redistribute the sets of particles with respect to which particle size is seen by the grid at those locations.

3.3.1 Multi-level redistribution scheme

One defines (α_p^l, x_p) , the strength and the position of a particle p at a level l and living on a grid for which the mesh size is h . At a level l , the particle size is, by definition, $2^l h$. The value of the level l takes either the value 0 or ± 1 . Three cases can occur: the particle is redistributed over a grid which has either the same resolution, or a resolution twice as fine, or a resolution twice

as coarse.

The basic idea is to combine two methodologies: 1) the classical redistribution procedure and 2) the average-interpolating wavelet (for more information concerning the wavelets, see Sweldens and Schröder [81]). First generation average-interpolating wavelets constitute a very efficient mathematical toolbox if one wants to interpolate uniformly distributed data with an accurate control over the resolution. Second generation wavelets deal with unevenly distributed data. It basically constructs the primitive function of the variable of interest (taken as the averaged value over its area) in order to compute the average on another area by evaluating this primitive function. The scheme is presented in 1-D; it can be readily extended to three dimensions through the use of tensor products, just as for uniform redistribution.

1. classical redistribution: Each particle of size h_l is first redistributed onto a grid of the same resolution h_l . For that purpose, classical redistribution schemes (such as M'_4 , Λ_3 schemes) are used, see Fig. 3.1.(a). These grid points are virtual, meaning that they are not created, unless they are at the same level as the particle.

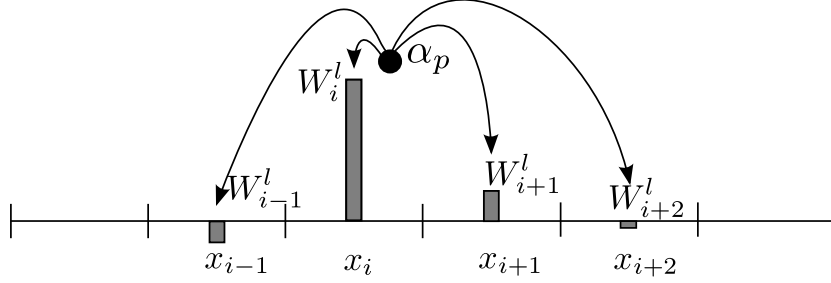
2. average-interpolating wavelet: Consider a particle not aligned with the grid. Its strength is redistributed on four points (x_{i-1} , x_i , x_{i+1} , x_{i+2}) with four associated strengths (W_{i-1}^l , W_i^l , W_{i+1}^l , W_{i+2}^0). We define W_i^l as the redistributed strength at the position x_i at the resolution level l ,

$$W_i^l = \int_{x_i - h_l/2}^{x_i + h_l/2} \omega(x') dx'.$$

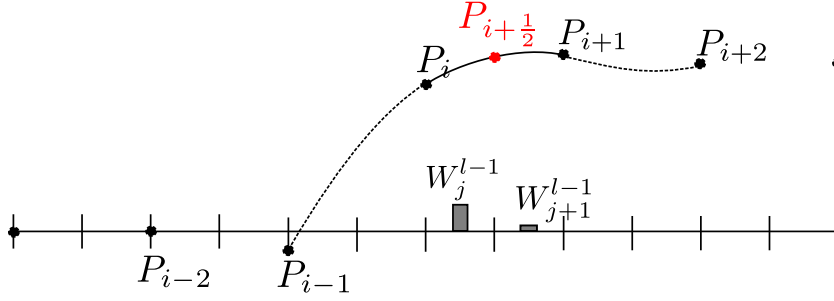
The primitive function $P(x) \triangleq \int_{-\infty}^x \omega(x') dx'$ is thus known between the redistribution points x_i , i.e. at $x_i + h_l/2$, since

$$\begin{aligned} P(x_i + h_l/2) \triangleq P_i &= \int_{-\infty}^{x_i + h_l/2} \omega(x') dx' \\ &= \sum_{n=0}^i \int_{x_i - h_l/2}^{x_i + h_l/2} \omega(x') dx' \\ &= \sum_{n=0}^i W_n^l. \end{aligned}$$

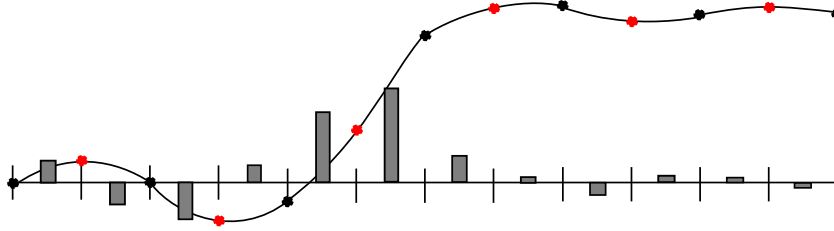
(a)



(b)



(c)



(d)

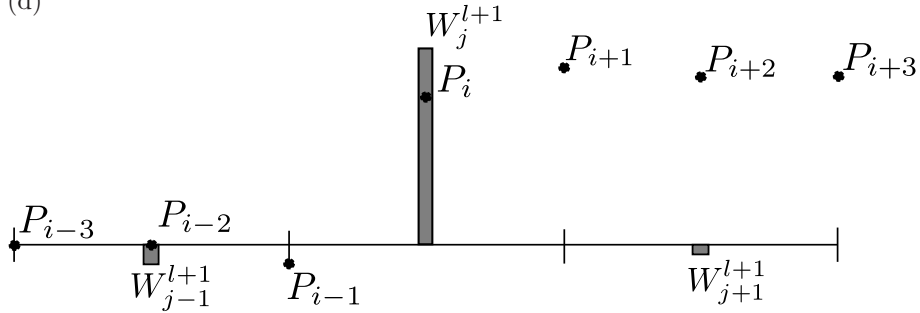


Figure 3.1: Sketch of the redistribution procedure on refined grids. (a) Redistribution of the strength of the particle of size h_l onto a mesh of size $h = h_l$, (b) Piece-wise interpolation of the primitive $P(x)$ to provide 2 new strengths on the mesh of size $h = \frac{1}{2}h_l$, (c) Entire redistribution of the particle strengths of size h_l onto a mesh of size $h = \frac{1}{2}h_l$ and (d) redistribution of a particle strength of size h_l onto a mesh of size $h = 2h_l$.

The function $P(x)$ can then be used to compute the strengths at various locations for different resolution levels, l' , using

$$\begin{aligned} W_{i'}^l &= \int_{x_{i'} - h_{l'}/2}^{x_{i'} + h_{l'}/2} \omega(x') dx' \\ &= P(x_{i'} + h_{l'}/2) - P(x_{i'} - h_{l'}/2). \end{aligned}$$

- **Lifting scheme:** We here consider the case with coarse-to-fine average-interpolation between level l and $l-1$. Two weights (W_j^{l-1} and W_{j+1}^{l-1}) are created between the points x_i and x_{i+1}

$$\begin{aligned} W_j^{l-1} &= P_{i+\frac{1}{2}} - P_i \\ W_{j+1}^{l-1} &= P_{i+1} - P_{i+\frac{1}{2}} \end{aligned}$$

where, $P_{i+\frac{1}{2}} = P\left(\frac{x_i + x_{i+1}}{2}\right) = \frac{1}{16}(-P_{i-1} + 9P_i + 9P_{i+1} - P_{i+2})$ is obtained using piecewise third-order polynomial interpolation. This step is illustrated in Fig. 3.1.(b). The procedure can be done for each interval $[x_{k-1}, x_k]$ in order to construct the complete piecewise redistribution as shown in Fig. 3.1.(c). This lifting procedure provides a smooth redistribution for which the two first moments are conserved. Concerning the numerical implementation, no intermediate grid h_l is created; the particles of size h_l are directly redistributed on the grid h_{l-1} without using a grid h_l .

- **Restriction scheme:** the fine-to-coarse average-interpolation simply consists in concatenating multiple blocks in order to create a coarser one

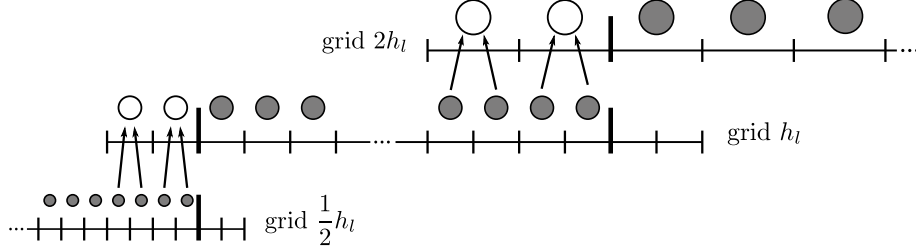
$$W_j^{l+1} = P_{i+1} - P_{i-1}.$$

This step is illustrated in Fig. 3.1.(d). Only the zero order moment is conserved.

3.3.2 Treatment of varying resolutions

We have just introduced a scheme which is able to redistribute a particle of size h_l on a grid of size h_l , $h_l/2$ or $2h_l$. The direct application of this scheme to a set of particles with different resolutions would lead to oscillations. In-

(a)



(b)

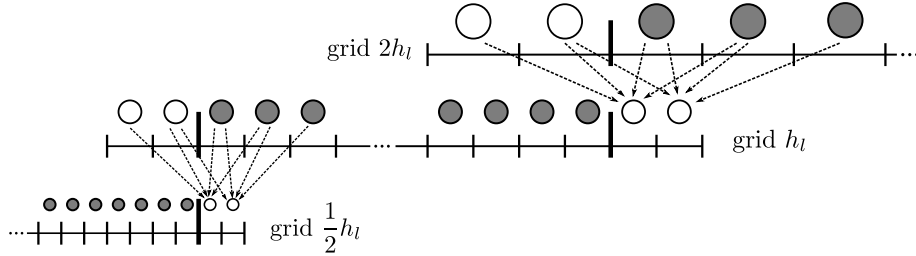


Figure 3.2: Sketch of the ghost particle (empty circles) creation from refined grids using “real” particles (filled circles). (a) Creation of coarse ghost particles (b) creation of fine ghost particles.

deed, the smooth interpolation of particle quantities requires a continuous and overlapping particle arrangement.

To tackle this problem, Daeninck [21] applied a global smoothing step which conserves the first three moments of the solution. This approach unfortunately proved unadapted to the use of grid-based methods (as it is the case here). We indeed need to preserve the smoothness of the solution. The present approach will rather rely on ghost particles. This buffer region maintains the overlap of particles up to the resolution interface and preserves the smoothness of the solution. We illustrate the procedure on a 1-D example (see Fig.3.3.(a) for the illustration). The physical or “real” particles of size $\frac{h}{2}$, h and $2h$ lie over a grid of size h .

First, we generate our ghost particles in a buffer region to extend the “real” ones (as described in Fig. 3.2). Knowing the complete solution on the “real” domain, we initialize the “real” and the “ghost” particles. Notice that both “real” and “ghost” particles are initialized using the same characteristic length as that of the generating grid. The procedure in order to create those ghost

particles depends on the resolution of the neighbor grid. The procedure can be summarized as follows:

1. When the neighbor grid uses the same resolution, “ghost” particles are created as the “real” particles; i.e. by identifying the particle strength with the grid strength located at the same position.
2. When the neighbor grid is twice finer (see Fig.3.2.(a)), neighbor “real” particles (e.g. 2 in 1-D and 8 in 3-D) are concatenated into a “ghost” particle.
3. When the neighbor grid is twice coarser (see Fig.3.2.(b)), neighbor “real” particles together with previously created neighbor “ghost” particles are used for the interpolation to create fine “ghost” particles. It is worth noting that this step should be necessarily done at the end as it uses “ghost” particles created during the step 2.

Each of these ghost particles therefore carries either the same vorticity as several finer real particles, or a fraction of the vorticity carried by a coarser particle. This can be visualized in Fig.3.3.(a) where we represent the handling of the interface between resolutions h and $\frac{h}{2}$: ghost particles of size h are created above the real particles of size $\frac{h}{2}$. In the same way, ghost particles of size $\frac{h}{2}$ are created above the real particles of size h .

Notice that the coarse-to-fine redistribution (i.e. redistribution of particles with size $2h$ onto grid h) require a larger support. The number of created ghost particles thus also depends on the support of the redistribution scheme. In the previous section, we have seen that a consistent M_4' interpolation will require two neighbor particles at the same level, assuming a moderately distorted particle set. When carrying out a coarse-to-fine interpolation, this number of required neighbors is increased by the support of the average-interpolation as the support of M_4' is broader than that of the third order cubic spline. For the fine-to-coarse interpolation, it is however not the case.

In a second step, the size of the real particles are redistributed onto the grid; one basically tags each grid points with the size of the real particles living above. The third step then consists in a selective particle-to-mesh interpolation. A mesh node with a tag level l can only receive the redistributed information

of particles at a level l . This step plays the same role as the indicator function introduced by Bergdorf.

The extension in two- and three-dimension is readily done by a straightforward application of the procedure described here-above. Fig.3.3.(b) illustrates the easy 2D extension of the procedure. One determines region of the grid in order to only use a unique particle size for the redistribution at a given grid point.

Real and ghost particles are convected, ensuring a proper overlap of particles with the same size up to the interface between two resolutions. Because the sets of real and ghost particles are a self-consistent representation of the vorticity field, no limitation on CFL is required, e.g. a particle can venture arbitrarily far into a patch of a different resolution. This is actually still a condition for this self-consistency. The thickness of the ghost particle buffer must be sufficient for the interpolation wavelet procedure. Those ghost particles can indeed get deformed by the flow, requiring instead a Lagrangian CFL condition, just as for the real particles.

3.4 A 1-D multiple resolution example

As an illustration, let us consider a 1-D Gaussian function $f(x)$ with $\sigma^2/h^2 = 30$ and discretized using three sets of particles over the interval $x/h = [-30, 30]$. Particles of size $2h, h, h/2$ are discretizing the sub-intervals $[-30, -10]$, $[-10, 10]$ and $[10, 30]$, respectively. These particles are then redistributed on a uniform grid h using the methodology described before. Fig. 3.4 shows the three sets of particles (and their respective strength $\alpha_p = f(x_p)h$), the Gaussian function and the discrete function $\tilde{f}(x_i) = \alpha_i/h$ obtained after redistribution: the functions f and $\tilde{f}$ are superposed. The absolute errors made on the function f and on its two first derivatives,

$$\begin{aligned} E_f(x) &= |f - \tilde{f}|, \\ E_{f'}(x) &= \left| \frac{df}{dx} - \frac{d\tilde{f}}{dx} \right|, \\ E_{f''}(x) &= \left| \frac{d^2f}{dx^2} - \frac{d^2\tilde{f}}{dx^2} \right|, \end{aligned}$$

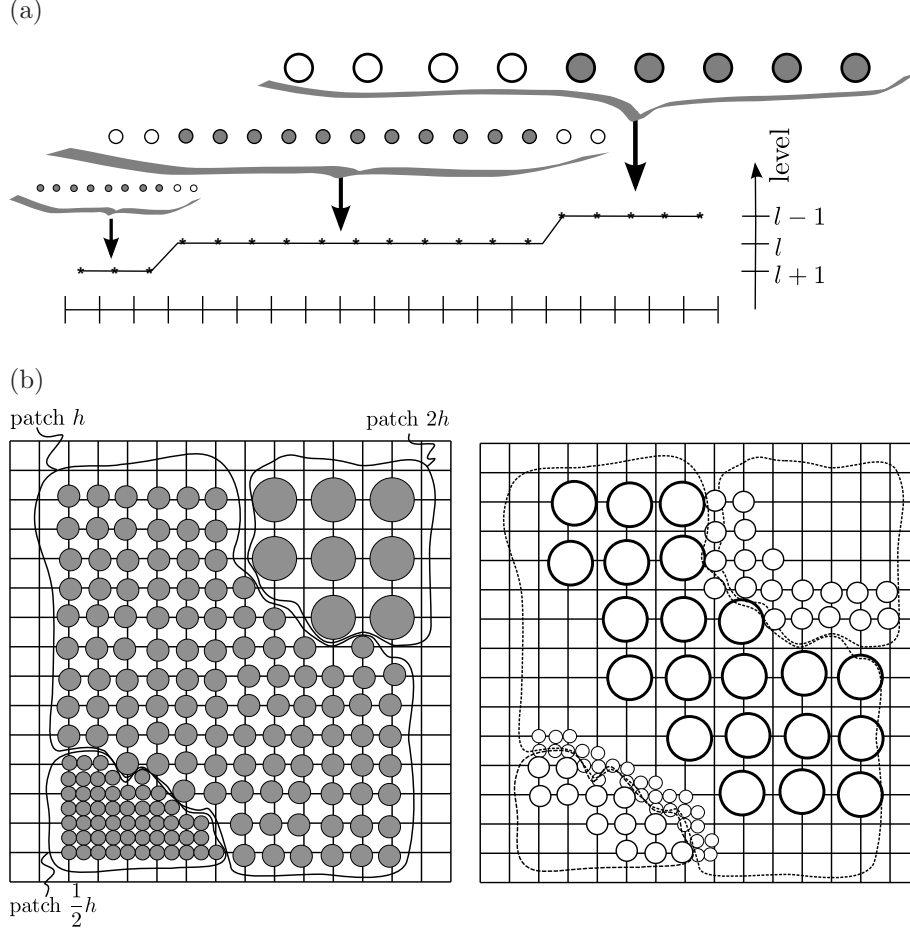


Figure 3.3: Sketch of the grid point tagging and the use of the ghost particles: (a) in 1D and (b) in 2D. Filled in circles “real” particles and circles “overlapping” particles.

are plotted in Fig. 3.5. The errors made on f and its first derivative (see Fig. 3.5.(a)) follow the same evolution and are seen to be of the same order of magnitude. However, the errors made by the redistribution of a particle on a grid of a different size are at least 10 times higher than those made by the redistribution on a grid of the same size. Concerning the second derivative (see Fig. 3.5.(b)), the error made for each redistribution case is roughly continuous and remains relatively low, but yet ten times higher than those made on the function and its first derivative. Concerning the interpolation of coarse particles on a finer grid, the errors made are indeed higher, yet are better than what we could expect as we started with twice less information. Regarding the

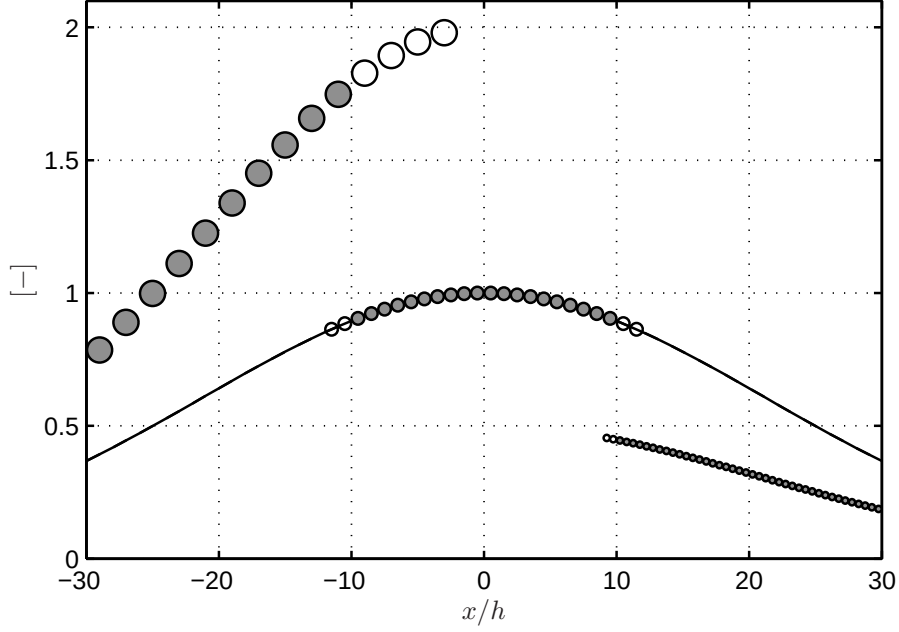


Figure 3.4: 1-D redistribution of a Gaussian function. Set of particle with various size and their strength α_p : the filled circles correspond to the “real” particles and the empty circles to the ghost particles. The Gaussian function $f(x) = \exp\left(-\frac{x^2}{V}\right)$, (solid line) and the redistributed function $\tilde{f}(x)$ (superposed dashed line).

interpolation of fine particles on a coarser grid, we can definitely conclude that the added average interpolation error does not constitute an issue.

The errors, made on the function f and its two first derivatives, are low enough. This is essential as, in our case, they correspond to the discretization of velocity, the vortex-stretching term and the diffusion term.

3.5 Vortex ring test case

In order to assess the quality of the refined mesh technique, the simulation of a vortex ring has been carried out. The vortex ring is first discretized using a uniform resolution, then it crosses various resolution regions.

Initial condition The initial azimuthal component of the vorticity field is taken as Gaussian,

$$\omega_\theta = \frac{\Gamma_0}{\pi\sigma_0^2} \exp\left(-\frac{s^2}{\sigma_0^2}\right),$$

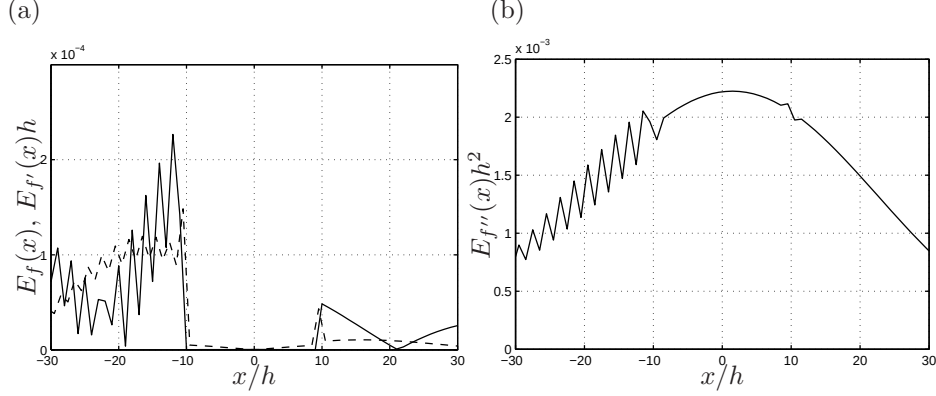


Figure 3.5: 1-D redistribution of a Gaussian function. (a) error on the function f (solid), on its first derivative (dash) and (b) on its second derivative.

where $s^2 = z^2 + (r - R)^2$, Γ_0 is the initial circulation, σ_0 the initial core size and R the ring radius. One defines the time-scale as $t_0 = R^2/\Gamma_0$. The Reynolds number based on the circulation is $Re = \Gamma_0/\nu = 5500$. The initial core size is fixed to $\sigma_0/R = 0.4131$ (same case as Shariff et al. [75, 76] and Cogle [11]). The time step used for the simulation is $\Delta t/t_0 = 0.05$. Three simulations have been done: i) one reference simulation using uniform resolution (fixed to $h/R = 0.04$) in the entire domain, ii) a simulation where the vortex ring starts from a resolution h then goes through a region of resolution $2h$ and then finally back to a region with the initial resolution h , and iii) a simulation where the vortex ring starts from a resolution h then goes through a region of resolution $h/2$ and then finally back to a region with the initial resolution h . Fig. 3.6 shows a restricted view of a cut through the mesh for each investigated case.

Results Fig. 3.7 presents a cut of the vorticity field for each case at various time: $t/t_0 = [0.0, 4.0, 6.0, 9.0, 13.0, 20.0]$. The vortical structures smoothly cross the different resolution areas, from a finer to a coarser (FTC) resolution as well as from a coarser to a finer resolution (CTF). No reflection or oscillations are observed at the transitions. Notice that the FTC transition is the easiest case because the redistribution acts like a smoother. The structures still remain accurate, even if the vortex-core is captured using only $\sigma_0/(2h) \simeq 5$ grid points. This can be especially observed at time $t/t_0 = 9.0$; at that time the vortex-core is clearly under-resolved. Concerning the CTF transition, the lifting scheme

works properly: the interpolation does not create spurious small scales. At time $t/t_0 = 20.0$, the vortices reach the initial resolution region, their core is finally convected at the same position for each of the three cases. That indicates that the linear impulse is qualitatively well conserved (as seen below). No major difference is seen in terms of the physics; even the thin wake is still captured.

Fig. 3.8, 3.9 and 3.10 respectively show the time-history of the three components of the total vorticity $\mathbf{\Omega} \triangleq \int_V \boldsymbol{\omega} dV$, of the linear impulse $\mathbf{I} \triangleq \int_V \mathbf{x} \times \boldsymbol{\omega} dV$ and of the angular impulse $\mathbf{A} \triangleq \frac{1}{3} \int_V \mathbf{x} \times (\mathbf{x} \times \boldsymbol{\omega}) dV$. Since a free vortical ring is considered, those quantities have to be conserved.

However, because the grid-to-particle interpolation procedure is not conservative (as we use the M'_4 scheme to interpolate information from the mesh to deformed scattered particles) the first three moments are then not exactly conserved. This is particularly due to the particle-set distortion (the so-called “clustering/depletion” effect); the particles do not remain on a regular lattice. Those moments are nevertheless qualitatively well conserved. More precisely, concerning the time-evolution of the total vorticity ($\mathbf{\Omega}$) and of the linear impulse ($\mathbf{I}$), the global trend is quite similar to each case. Concerning the angular impulse ($\mathbf{A}$), the FTC case provides better conservation. It is worth noting that the present code works in simple-precision: those quantities are at least 10^{-8} . Moreover, it is worth noting that no re-projection of $\boldsymbol{\omega}$ into a divergence free field was done in the case with uniform grid, because the divergence measure did not reach the user-prescribed divergence threshold. That also explains why the total vorticity is not mathematically conserved to zero.

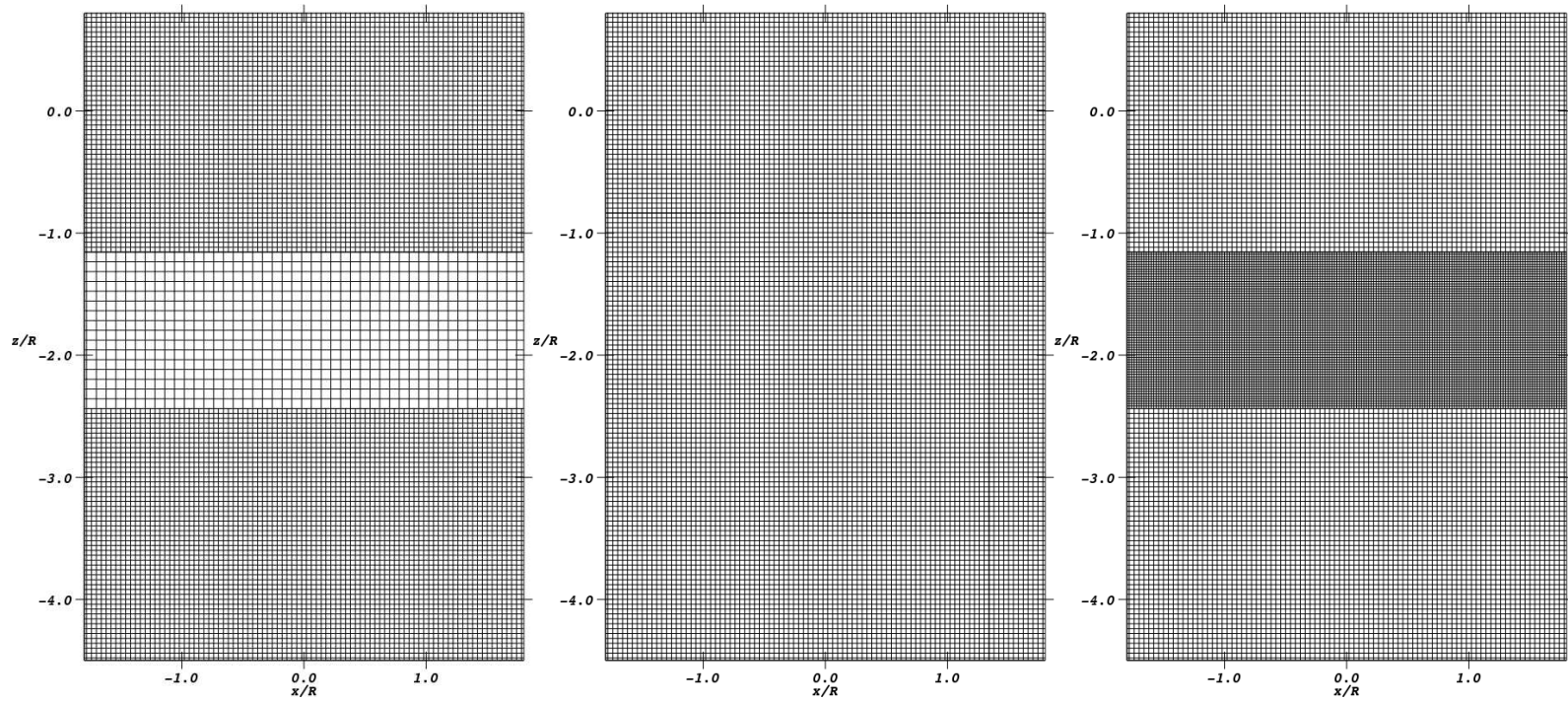


Figure 3.6: Vortex ring at $Re = 5500$. Slice of the mesh at $y/R = 0$: case using uniform resolution (center), case using locally a coarser resolution (left) and case using locally a finer resolution (right).

$$t/t_0 = 0.0$$

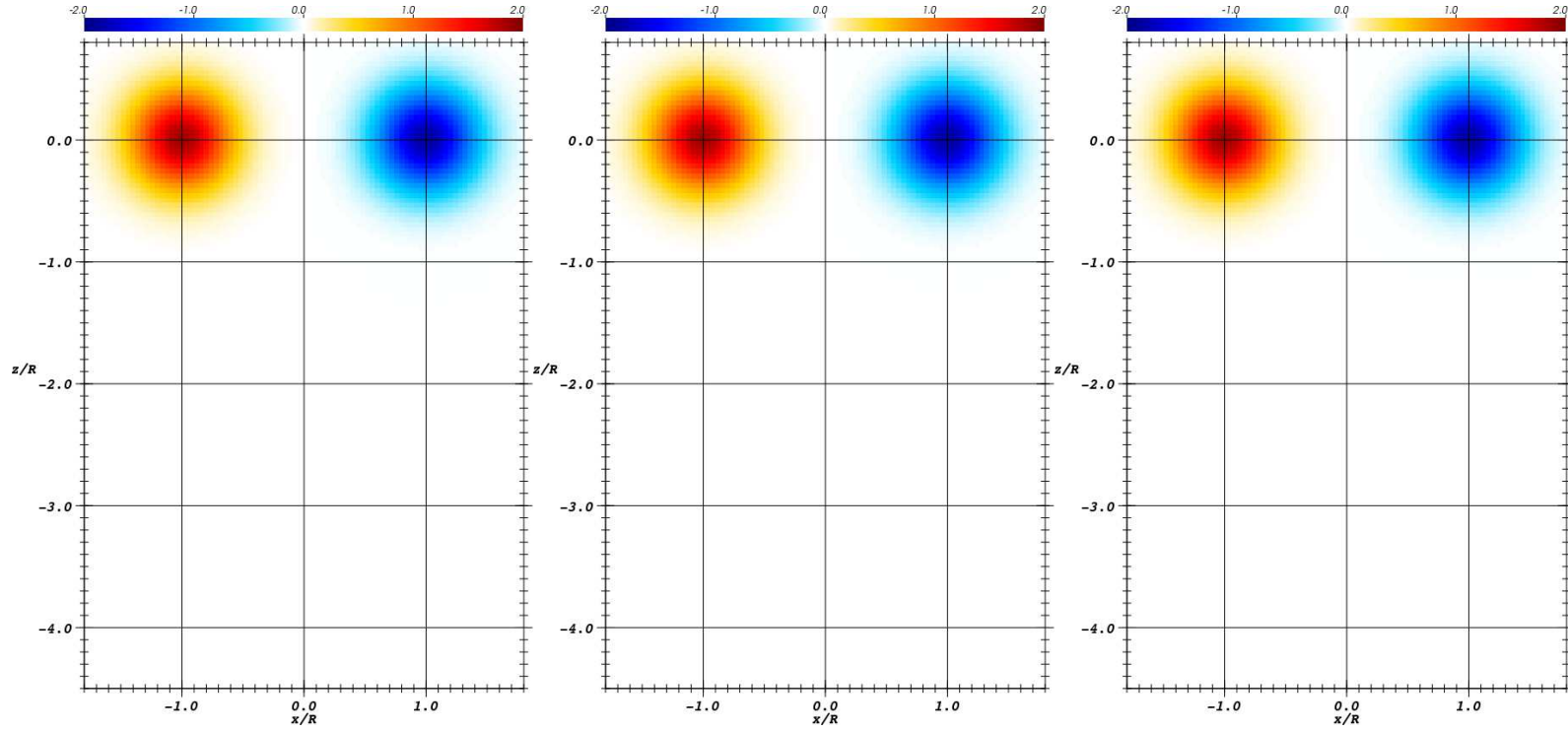
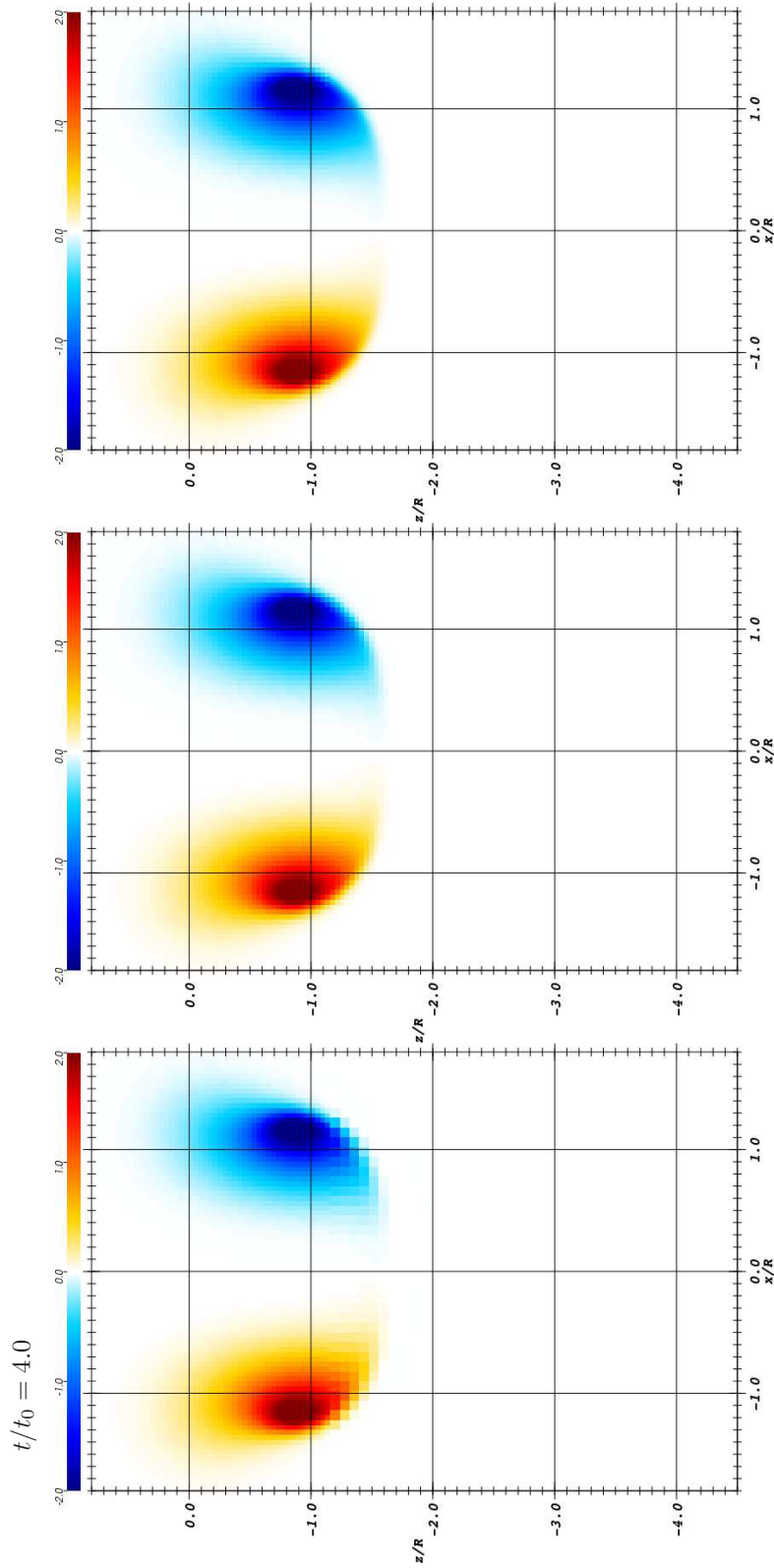
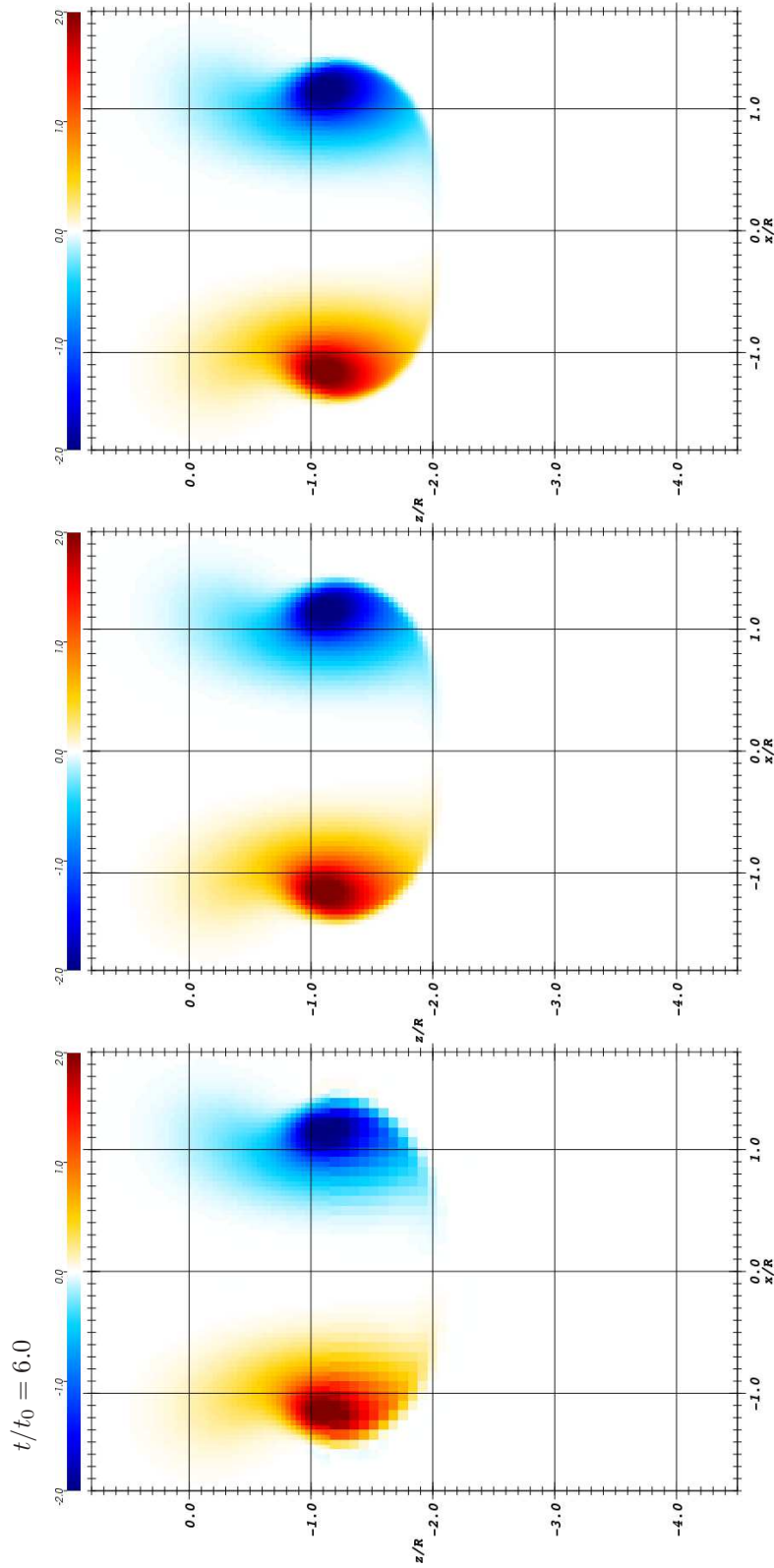
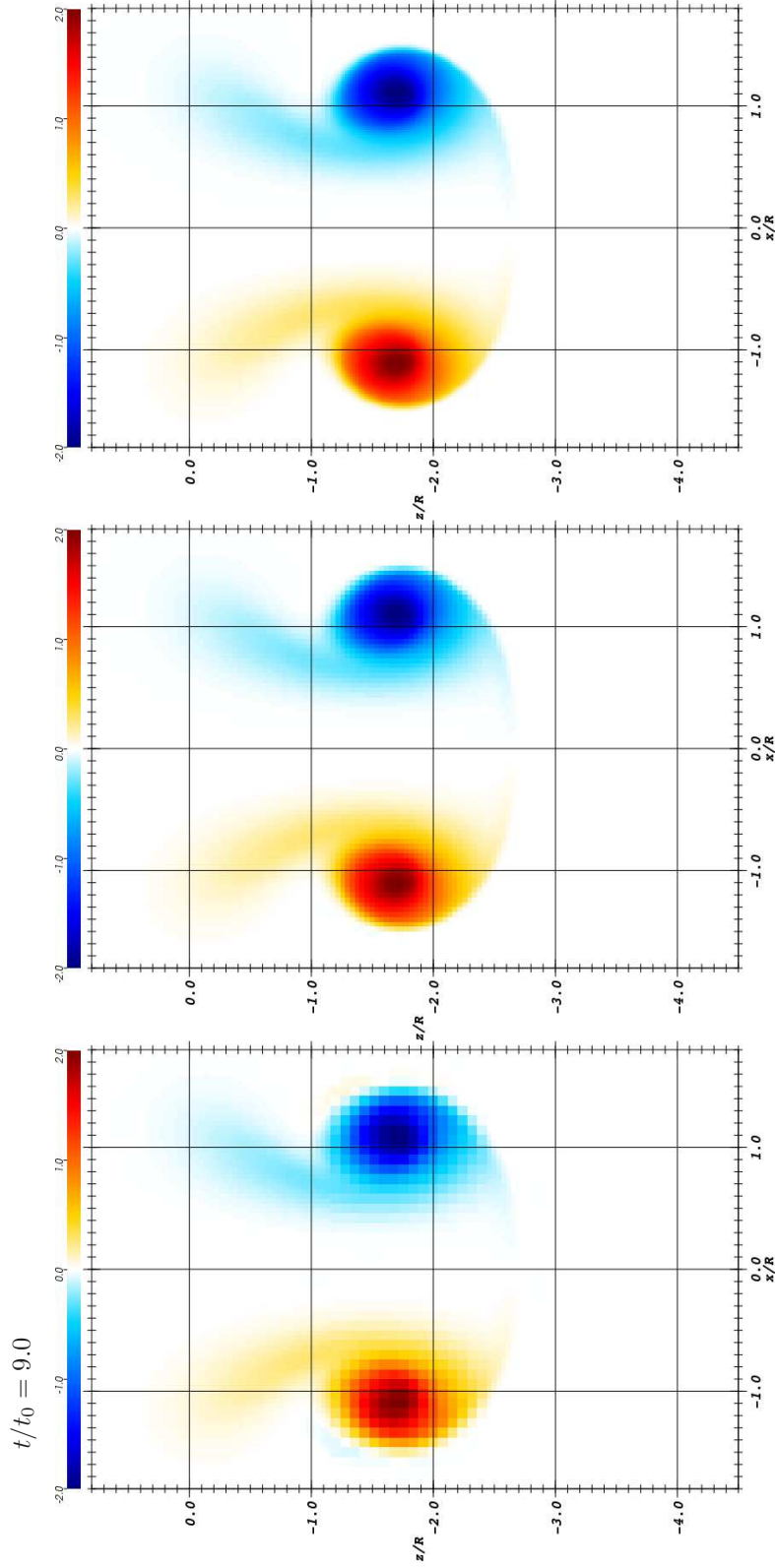


Figure 3.7: Vortex ring at $Re = 5500$. Slice of $\omega_y t_0$ at $y = 0$ for the simulation using a uniform resolution (center), using locally a coarser resolution (left), and using locally a finer resolution (right).

Figure 3.7: *cont'd*

Figure 3.7: *cont'd*

Figure 3.7: *cont'd*

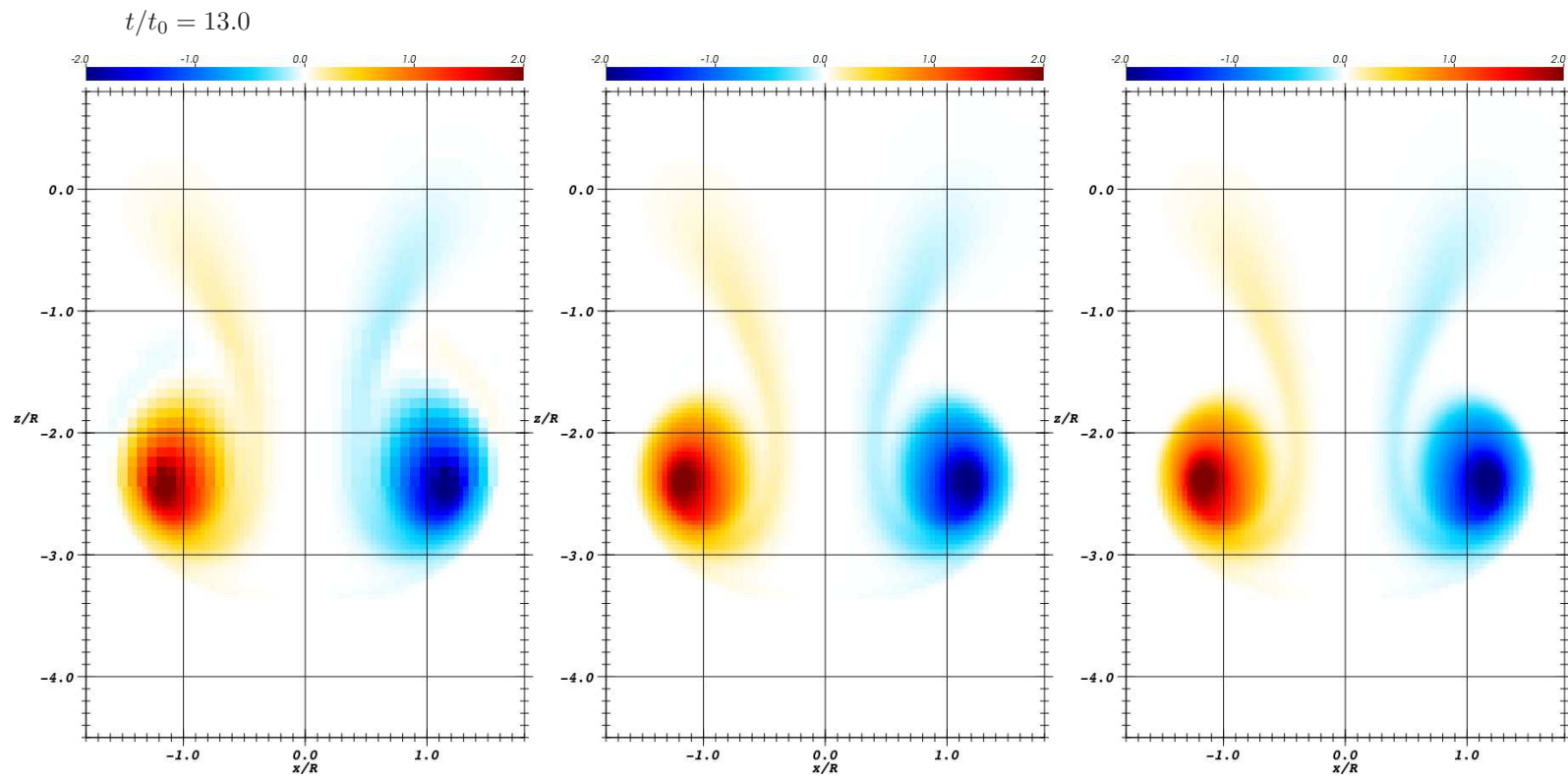
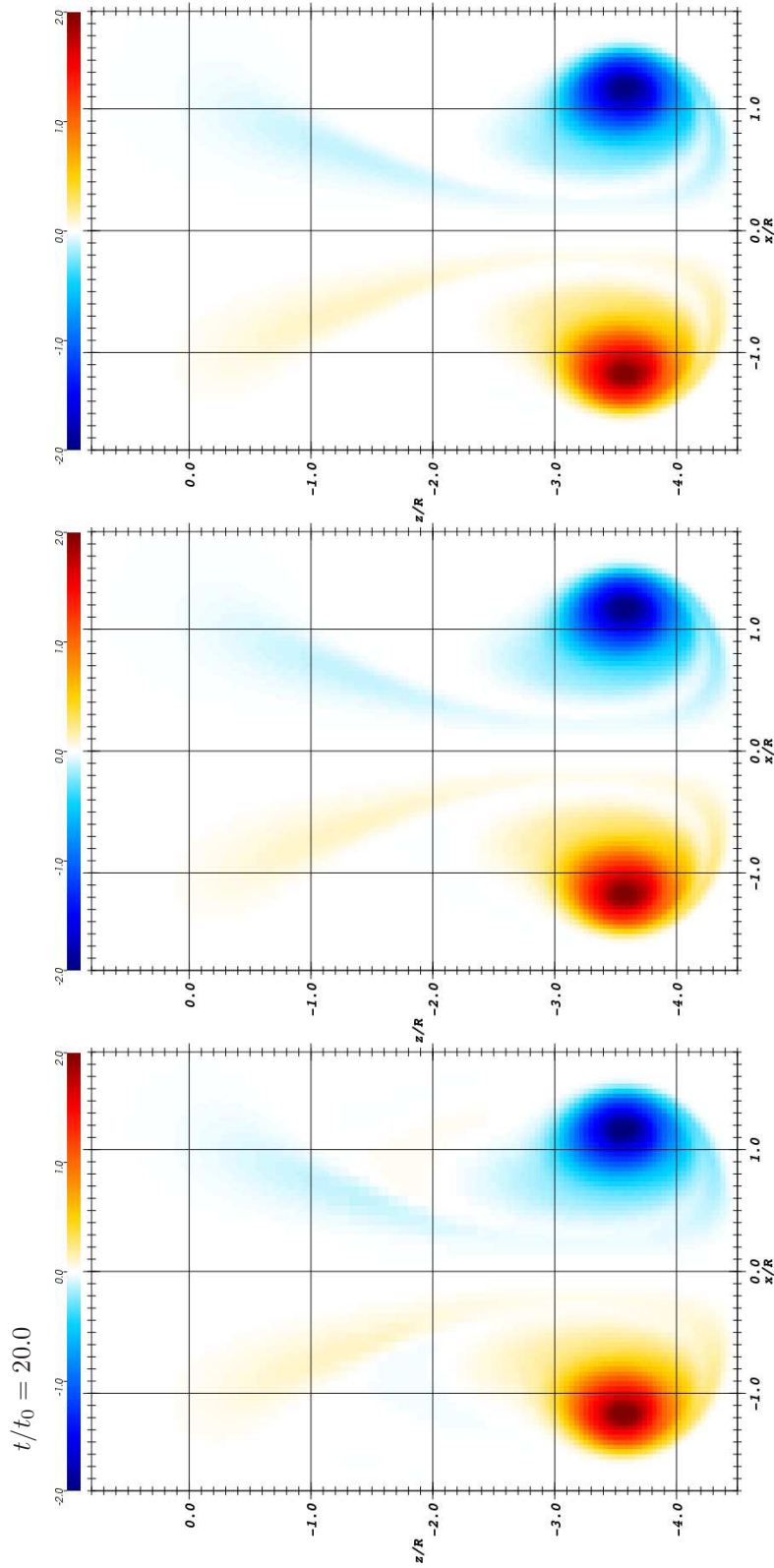


Figure 3.7: cont'd

Figure 3.7: *cont'd*

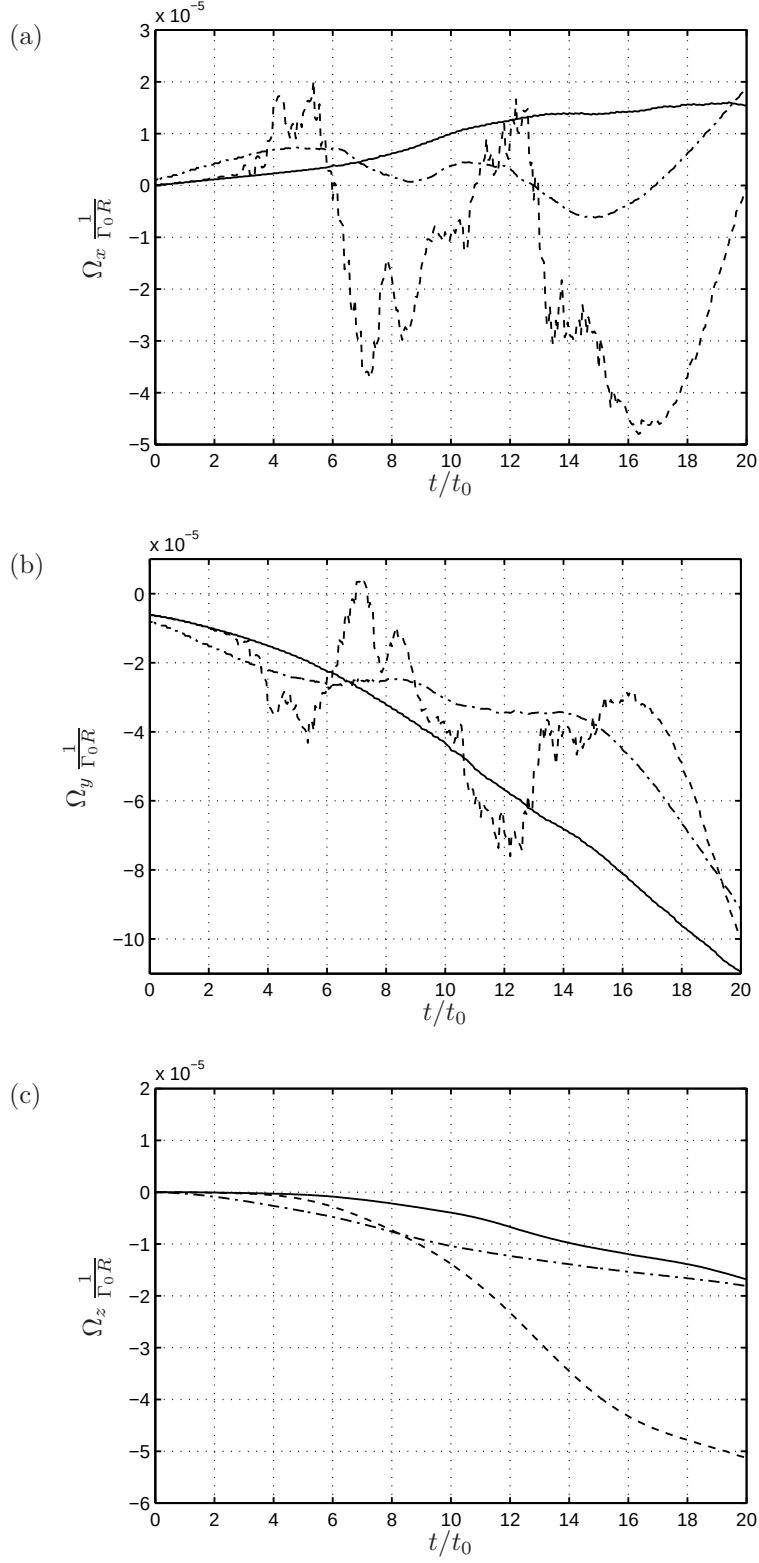
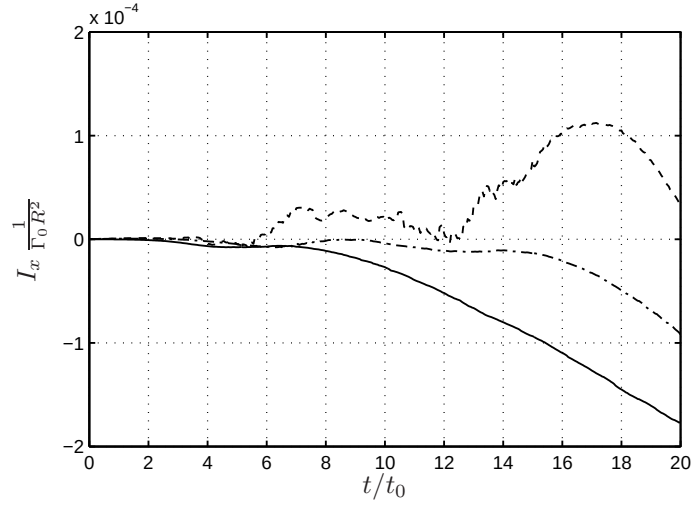
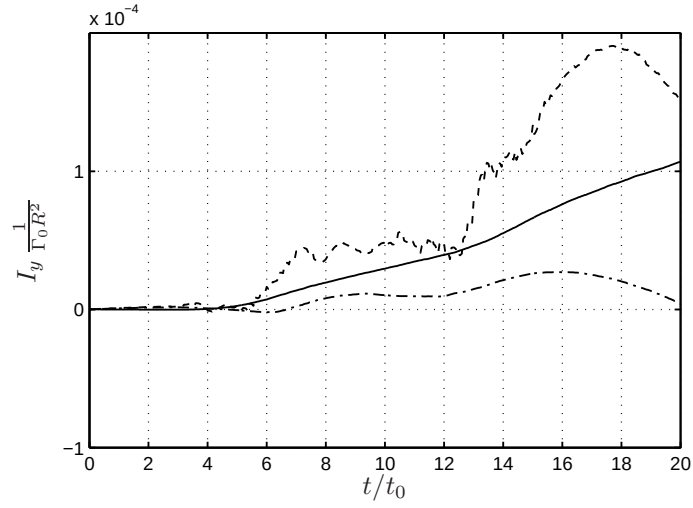


Figure 3.8: Time-evolution of the three components of the total vorticity: i) using a uniform resolution (solid line) ii) using locally a coarser resolution (dashed line) and iii) using locally a finer resolution (dash-dotted line).

(a)



(b)



(c)

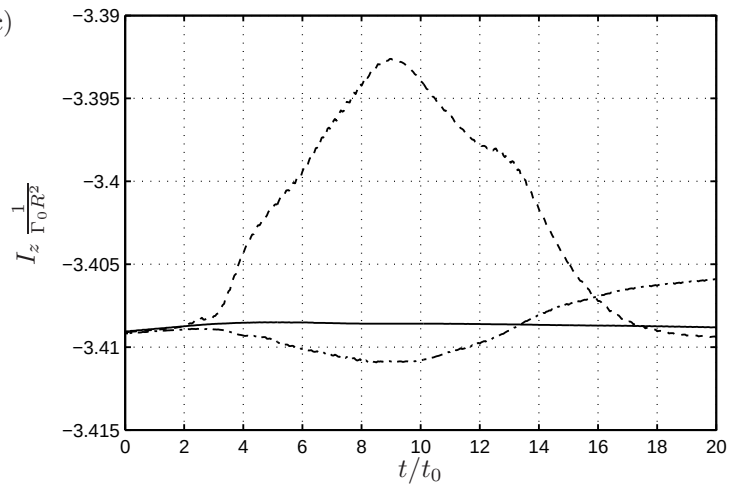


Figure 3.9: Time-evolution of the three components of the linear impulse: i) using a uniform resolution (solid line) ii) using locally a coarser resolution (dashed line) and iii) using locally a finer resolution (dash-dotted line).

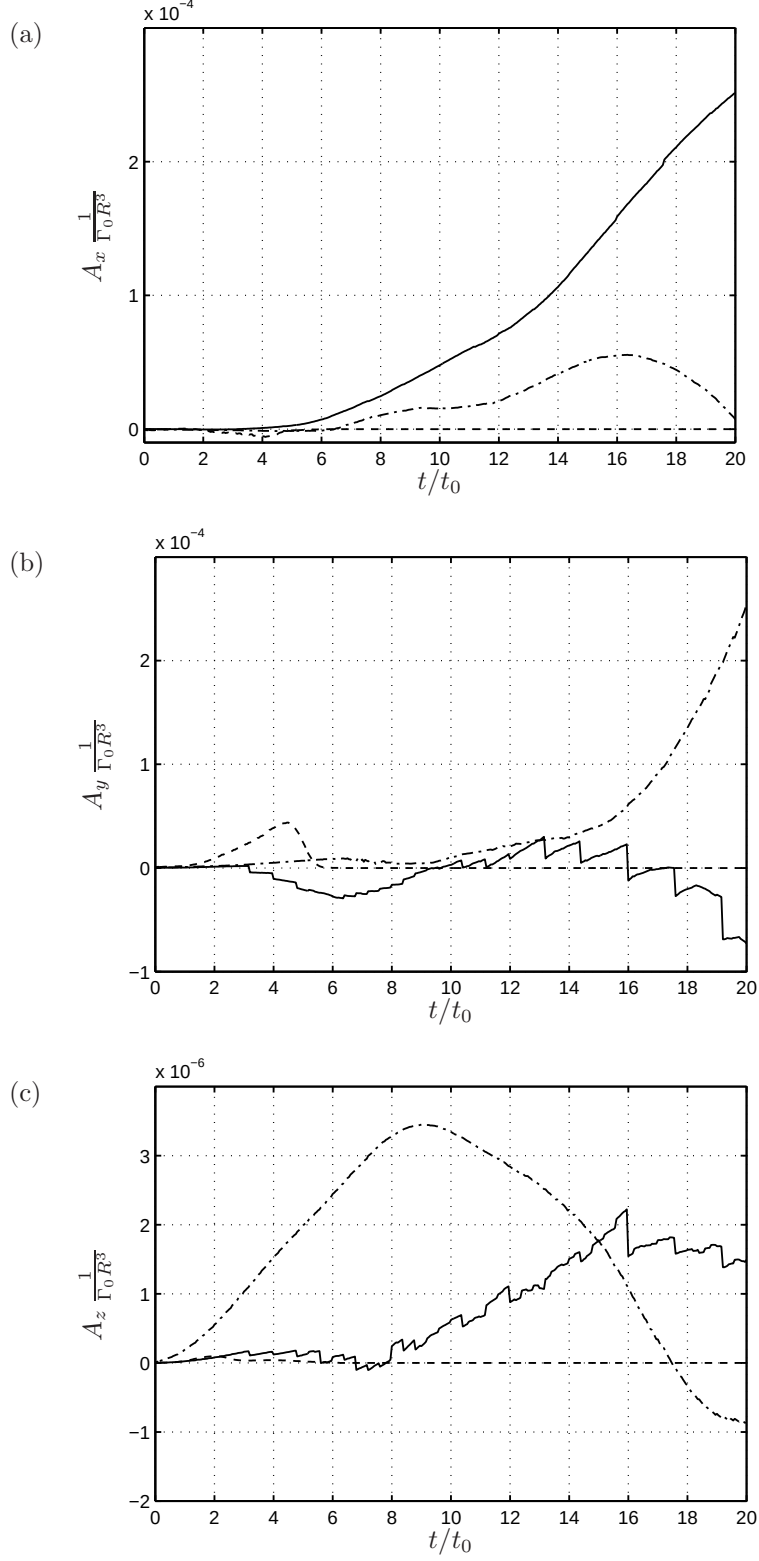


Figure 3.10: Time-evolution of the three components of the angular impulse: i) using a uniform resolution (solid line) ii) using locally a coarser resolution (dashed line) and iii) using locally a finer resolution (dash-dotted line).

3.6 Multiple resolution simulation of the flow past the sphere at $Re = 300$

The DNS of the flow past a sphere at $Re = \frac{U_\infty D}{\nu} = 300$ has also been carried out using the refined mesh technique described before. The immersed boundary technique is also used in order to perform this simulation. This is thus the VPM-PFM-IB code with mesh refinement added. The present validation case is basically the same as that using the immersed boundary technique (described in Chap. 2, Sect. 3) with uniform resolution. The latter simulation is definitely too resolved in the wake region, and led to too many grid points ($\sim 120 \cdot 10^6$).

Setup The multi-resolution simulation uses the same resolution in the near-wall region $h^0/D = 1/75$. The wake is captured up to 38 diameters downstream, thus comparable to the case with uniform resolution. The time-step is taken as before $\Delta t U_\infty / D = 8.5 \cdot 10^{-3}$. The maximal CFL criterion were seen to be $CFL = \left(\frac{|\mathbf{u}|}{h} \right)_{\max} \Delta t \leq 0.89$. The resolution function $\delta(\mathbf{x})$ is taken as follows:

$$\delta(\mathbf{x}) = \begin{cases} \frac{h^0}{D} & \text{if } \frac{r}{D} \leq 0.8 \\ \frac{h^0}{D} (25 \frac{r}{D} - 19) & \text{if } 0.8 < \frac{r}{D} \leq 1.0 \\ 6 \frac{h^0}{D} \left(\frac{10}{3} \right) \left(\frac{r}{19} \right)^{-1} & \text{otherwise} \end{cases}$$

where $r = |\mathbf{x} - \mathbf{x}_c|$ is the distance of a point $\mathbf{x}$ to the sphere center $\mathbf{x}_c$. Vortex particles, that go out of the mesh, are simply removed. In opposition to the case using uniform resolution (see Chap.2, Sect. 3, or in [21, 11]), no specific outflow condition is used. In fact, the varying resolution method can itself be used as a way to properly do wake flow with outflow boundary condition: simply running a very long domain with resolution getting coarser and coarser provides a natural way of having a good outflow condition.

Results Fig. 3.11.(a), 3.11.(b) and 3.11.(c) show a slice of the vorticity field for each component. Typical results are obtained. They are also comparable to those obtained by Ploumhans [68]. A cut of the mesh is presented in Fig. 3.11.(d). The blank squares located in the up and down corners correspond to disabled block because there are particle free. The ratio between the coarsest

and the finest mesh size is 32. The total number of grid points needed for this simulation is $\sim 11 \cdot 10^6$, to be compared to the case using uniform resolution (i.e. $\sim 120 \cdot 10^6$). The gain is thus substantial. A zoom cut of the vorticity field close to the surface of the sphere together with the mesh is shown in Fig. 3.12. It is seen that the vorticity field is smooth up to the surface, indicating that the simulation is well-resolved. Fig. 3.13 presents 3 views of a λ_2 surface. The typical hairpin vortices shed behind the sphere are visualized; four of them are observed. The time-history of the lift and drag coefficients are shown in Fig. 3.14. Table. 3.14 summarizes the global values of interest. Those values are computed as for the case using uniform resolution; i.e. using the Noca's formula over a control volume. The control volume is also taken as done for the case with uniform resolution (i.e. a cube centered in the sphere center and with a edge length fixed to $1.6D$). Those are compared with our simulation using uniform resolution and with those of literature. The values are seen to be in fair agreement. The averaged lift is a bit too high; but one must recall that it is quite difficult to evaluate. The values could also be affected by a maybe too coarse level of the resolution after $17.5D$

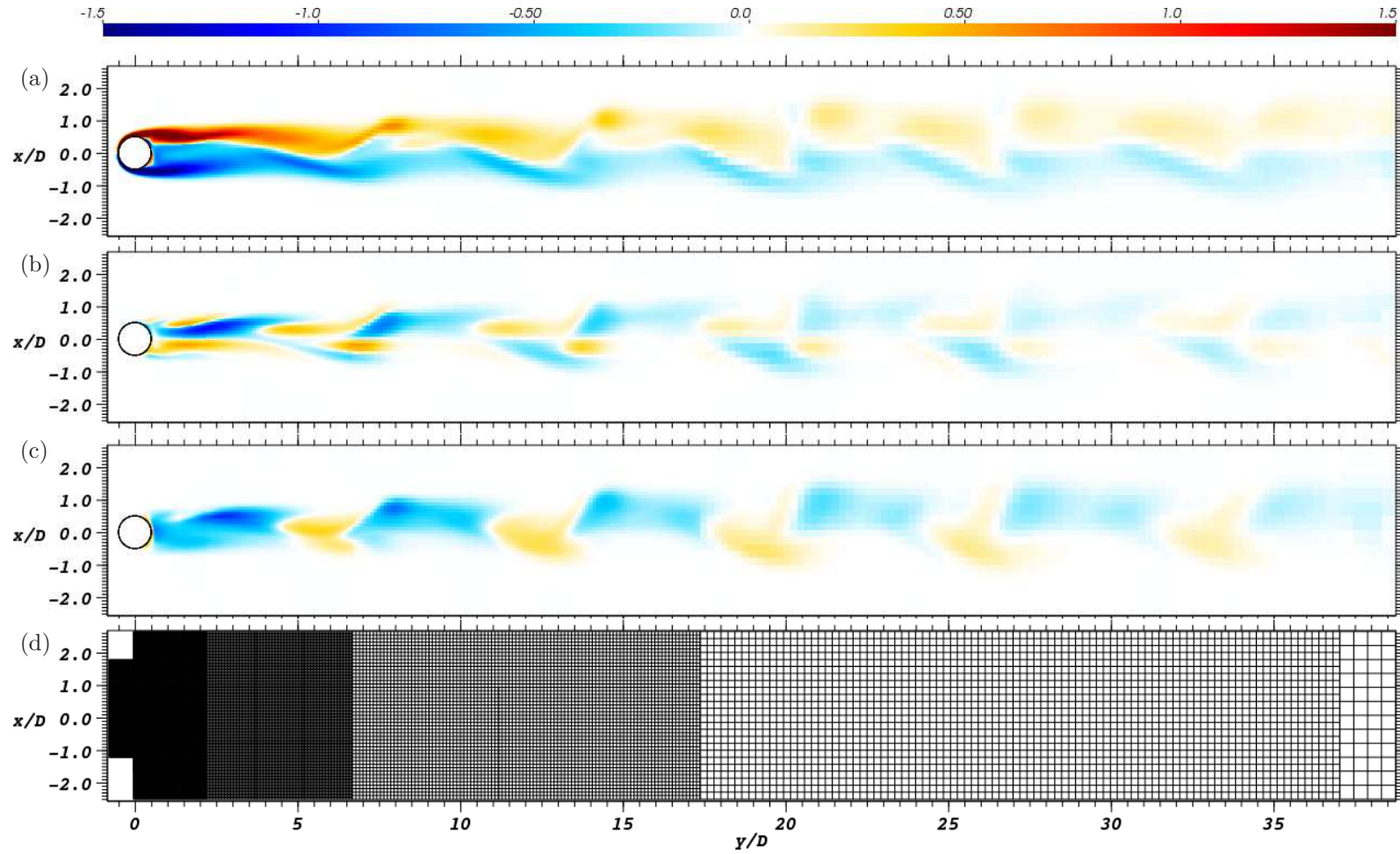


Figure 3.11: Multi-resolution simulation of the flow past a sphere at $Re = 300$: global view of a slice of the three vorticity components: (a) $sign(\omega_z) \log \left(1 + \frac{D}{U_0} |\omega_z| \right)$, (b) $sign(\omega_y) \log \left(1 + \frac{D}{U_0} |\omega_y| \right)$ and (c) $sign(\omega_x) \log \left(1 + \frac{D}{U_0} |\omega_x| \right)$. (d) cut of the refined mesh.

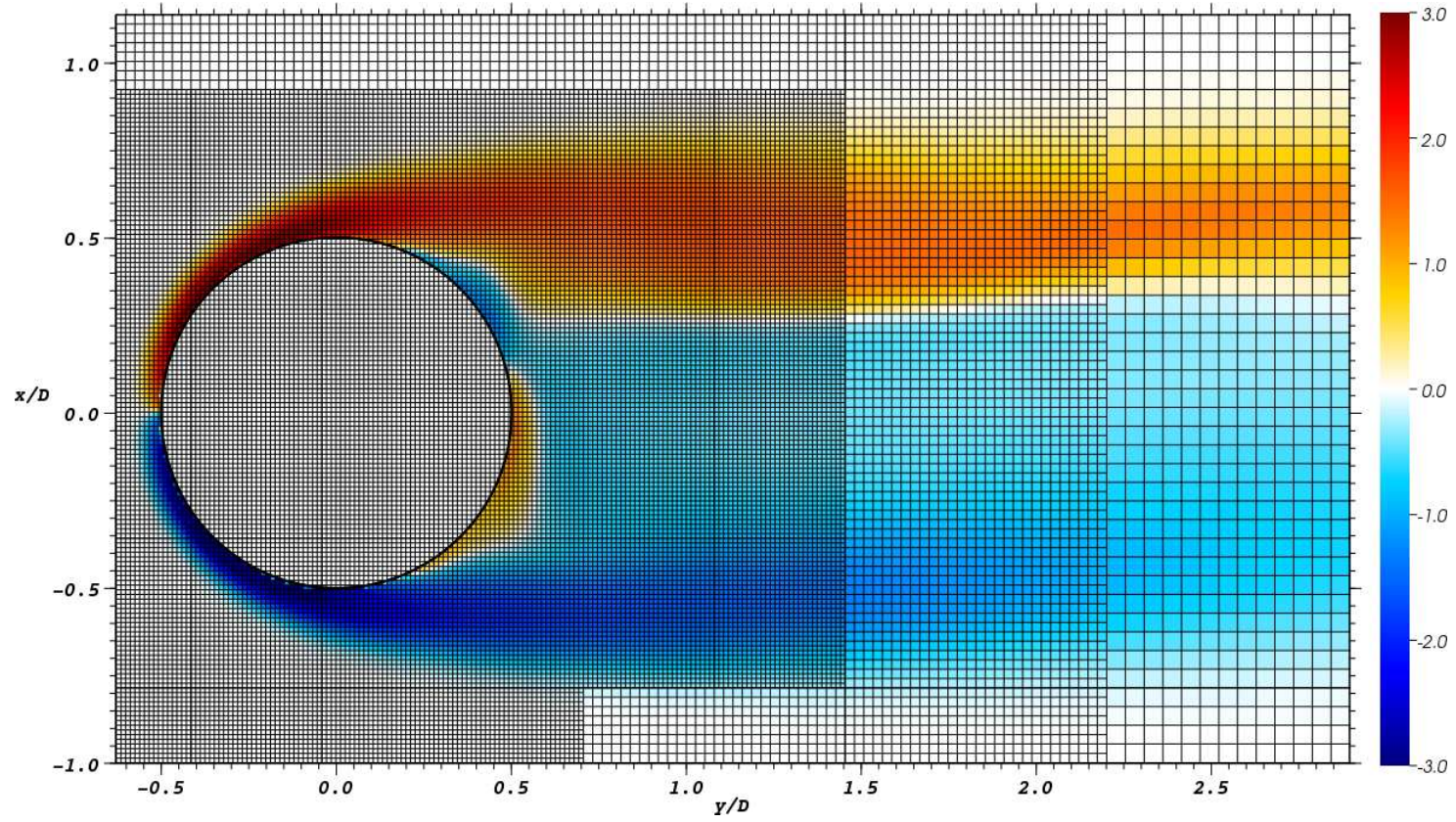


Figure 3.12: Multi-resolution simulation of the flow past a sphere at $Re = 300$: zoom of the vorticity component $\text{sign}(\omega_z) \log \left(1 + \frac{D}{U_0} |\omega_z| \right)$. A cut of the mesh used for the simulation is also shown.

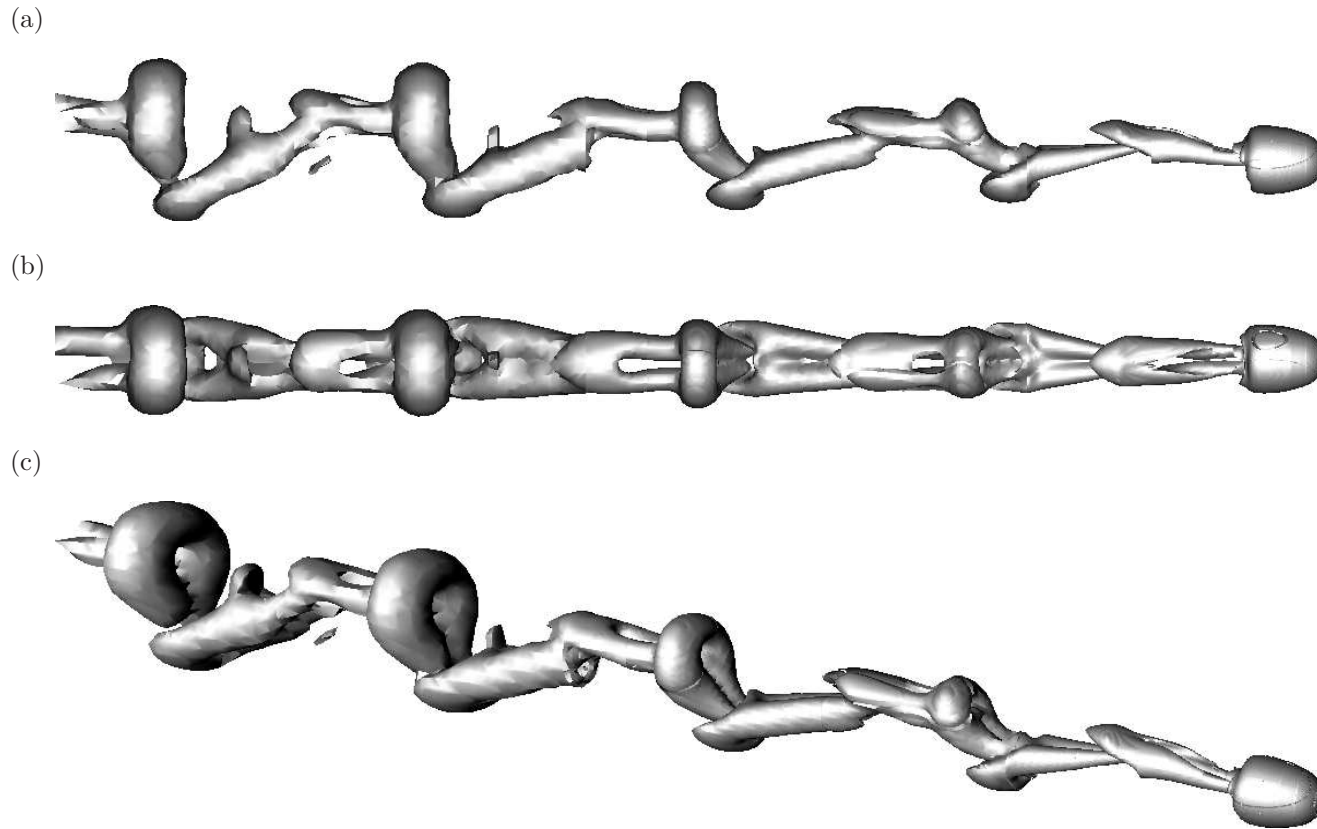


Figure 3.13: Multi-resolution simulation of the flow past a sphere at $Re = 300$: surface of the λ_2 criterion: (a) top view, (b) side view and (c) offset view.

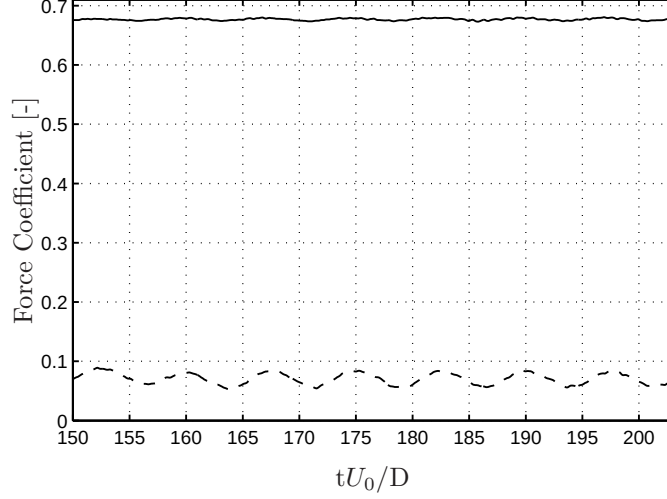


Figure 3.14: Time-evolution of the drag coefficient (solid line) and lift coefficient (dashed line).

Table 3.1: Flow past a sphere at Reynolds 300: summary table of global quantities compared to the literature

	$\overline{C_D}$	ΔC_D	$\overline{C_L}$	ΔC_L	Str
Multi-resolution	0.677	$3.3 \cdot 10^{-3}$	0.070	$1.4 \cdot 10^{-2}$	0.134
Uniform resolution	0.672	$3.4 \cdot 10^{-3}$	0.065	$1.5 \cdot 10^{-2}$	0.134
Johnson & Patel [42]	0.656	$3.5 \cdot 10^{-3}$	0.069	$1.6 \cdot 10^{-2}$	0.137
Tomboulides & Orszag [83]	0.671	$2.8 \cdot 10^{-3}$	-	-	0.137
Constantinescu et al. [14]	0.655	-	0.065	-	0.136
Ploumhans et al. [68]	0.683	$2.5 \cdot 10^{-3}$	0.061	$1.4 \cdot 10^{-2}$	0.135
Georges [31]	0.661	$2.5 \cdot 10^{-3}$	0.066	$1.3 \cdot 10^{-2}$	0.134

3.7 Conclusion

We have developed a mesh refinement technique for particle-mesh methods. It combines the approach of Bergdorf et al. [2, 3] and Rossinelli et al. [73] to achieve, at the same time, block-wise grid refinement, high order accuracy, and CFL relaxation.

We have extended this approach for the simulation of 3D incompressible

flows, possibly with solid bodies. A parallel Poisson solver has thus been integrated. We here successfully applied the approach of Cocle et al. [11] in the mesh refinement context. That is, the required Poisson equation is here solved using an efficient combination of a fast grid-based Poisson solver with a parallel fast multipole method. The method has been validated for 3D incompressible flows with solid bodies. A simulation of a vortex ring crossing different resolution regions has been done. It was seen to qualitatively conserve well the three first moments for each transition.

In order to handle refined meshes, a mesh generator based on hierarchically refined grids was also developed. It enables to have a good load balancing taking into account both the computational cost and the memory cost, using the *METIS* library.

The multi-resolution simulation of the flow past a sphere at Reynolds 300 has also been carried out. The results have been compared to the equivalent simulation using uniform resolution and to those of literature; a good adequacy has been obtained in terms of the time-evolution of the forces acting on the body.

Chapter 4

Investigation of velocity deficit effects on aircraft wake rollup

This chapter is the follow-up of a technical report of the FAR-Wake project entitled “Rollup of a temporally-evolving wing wake with velocity deficit” [19]. European project AST4-CT-2005-012238, Report TR2.2.2-3, January, 2007. The authors are C. Cottin, G. Winckelmans, T. Lonfils, R. Cocle.

4.1 Introduction

Vortex wake (WV) dynamics have been extensively investigated for decades. It is now well-established that the wake behind a lifting-body essentially consists, after rollup, in a pair of counter-rotating vortices. When created by an aircraft, those vortices can be potentially hazardous for a following aircraft. The WV intensity scales with the lift and thus with the weight. The WV persistence is at the origin of the separation standards between two aircraft for landing and takeoff phases. This limits the capacity of the busiest airports. Much research focuses on the way to diminish their intensity by means of active/passive control systems. Another avenue of research concerns the reduction of those standards with respect to the meteorological data, measured and/or forecast

(wind, turbulence, temperature). Indeed, the standard separation from ICAO (International Civil Aircraft Organization) are often too conservative. Experimental and numerical investigations are thus required to further understand the mechanism of formation and of destruction of wake vortices. See the general review of Spalart [77], for more information concerning aircraft trailing vortices, including the formation, motion and persistence of wake vortices and relevance to air travel applications.

This work is dedicated to the investigation of the effect of a velocity deficit, due to the friction drag, on the near-wake rollup dynamics on the net-result after rollup. It is the follow-up of the research carried out by Cottin et al. [19] in the framework of the EC funded European FAR-Wake project. Three-dimensional simulations were then performed. Although they were interesting, they lacked relevance since only time-developing (T-D) simulations were considered. A T-D simulation evolves in-time an initial condition using periodic condition in the axial direction and thus assumes a streamwise homogeneous direction. The reality is much different; the flow evolves in-time as well as spatially. In this chapter, space-developing simulations (S-D) were performed, using the refined mesh handling capability as described in Chapter 3, in order to obtain more realistic results. A comparison to the T-D case is also presented for completeness.

A characterization of the net result of wake rollup, in terms of tangential velocity profile, axial velocity, and their evolution in the wake, is provided, using the results of the S-D simulation. The fits of these profiles with the most usual vortex models is also done. In particular, we show that the two length-scale models of Proctor-Winckelmans and of Jacquin provide a very good fit (both in terms of maximum tangential velocity and effective core size).

4.2 Initial conditions

This chapter aims at investigating the effect of the axial velocity deficit due to boundary layers (friction drag) on the roll-up phase and on the resulting vortex system. Two S-D simulations have been carried out with and without velocity deficit. The former uses a modeled external-force corresponding to the wing presence, and the latter uses an inflow modeled vorticity field. A further

T-D simulation with velocity deficit has also been performed using the modeled vorticity field as an initial condition.

The first approach consists in directly adding an extra force-term in the RHS of the Navier-Stokes equations, corresponding to the force exerted by the wing on the fluid. That entails to model the force distribution. The second approach consists in modeling the vorticity field shed by a regularized lifting line at the trailing edge of a generating wing in order to use it as an inflow condition.

Concerning the SD simulation with velocity deficit, we noticed that non-physical results were obtained when using a modeled inflow. Hence, we modified our approach and developed, instead, a modeled external-force. Indeed, when using the modeled vorticity field with velocity as an inflow, the vortex-pair circulation is found to significantly increase with respect to the downstream distance, by $\sim 20\%$. This is due to a net spanwise vorticity flux at the inflow. The imposed wake vorticity is up-and-down symmetric but the streamwise velocity is not (because of the lifting-line). As a consequence, there is a spurious vorticity into the domain. The injected vorticity is not divergence-free, the successive reprojection redirects the spanwise vorticity to the streamwise direction. Those results (diagnostics and visualizations) are provided in Appendix A. It is worth noting that the problem does not occur for the S-D simulation without velocity deficit as the spanwise vorticity is zero; indeed there is only streamwise vorticity resulting from the regularization of the lifting line.

The two approaches use the Prandtl's lifting line theory. This theory consists in replacing the lifting wing of lift distribution, $l(y)$, by a vortex line defined by its span loading, $\Gamma(y)$, using the relation: $l(y) = \rho U_0 \Gamma(y)$ (ρ being the fluid density and U_0 the flight speed). For an elliptically loaded wing, one has:

$$\Gamma(y) = \Gamma_0 \sqrt{1 - \left(\frac{y}{b/2}\right)^2}$$

where Γ_0 is the half plane total circulation. The vortex spacing after roll-up, b_0 is also obtained from

$$\int_{-b/2}^{b/2} \Gamma(y) dy = \Gamma_0 b_0,$$

with $b_0 = \frac{\pi}{4} b$ for elliptical loading. Considering the definition of the total wing

lift, and the lift coefficient, respectively:

$$\begin{aligned} L &= \int_{-b/2}^{b/2} l(y) dy = \rho U_0 \int_{-b/2}^{b/2} \Gamma(y) dy = \rho U_0 \Gamma_0 b_0 \\ C_L &= \frac{L}{\frac{1}{2} \rho U_0^2 \mathcal{S}} = \frac{2 \Gamma_0 b_0}{U_0 \mathcal{S}} \end{aligned}$$

with $\mathcal{S} = \frac{b^2}{A_R}$ the wing surface (A_R being the wing aspect ratio), the characteristics of the wake can be expressed as a function of the wing characteristics as:

$$\Gamma_0 b_0 = \frac{1}{2} U_0 b^2 \frac{C_L}{A_R}$$

One also obtains, for the downwash velocity of the vortex wake of after rollup:

$$V_0 = \frac{\Gamma_0}{2 \pi b_0} = \frac{U_0}{4 \pi} \left(\frac{b}{b_0} \right)^2 \frac{C_L}{A_R}$$

and thus, for an elliptical loading,

$$\frac{V_0}{U_0} = \frac{4}{\pi^3} \frac{C_L}{A_R}$$

In the present work, the wing characteristics are fixed such that $\frac{C_L}{A_R} = 0.20$ and thus, $\frac{V_0}{U_0} = 2.58 \cdot 10^{-2}$. The Reynolds number is taken as very high (i.e. nearly realistic):

$$Re_\Gamma = \frac{\Gamma_0}{\nu} = 10^6.$$

Hence, subgrid-scale modeling will be used.

Description of the vorticity field inflow model For a wing of the circulation $\Gamma(y) = \Gamma_0 \sqrt{1 - \left(\frac{y}{b/2} \right)^2}$, the corresponding shed vorticity field is a vortex sheet with circulation per unit length :

$$\gamma(y) = -\frac{d\Gamma}{dy}(y)$$

This singular vortex sheet is here regularized using a Gaussian kernel in order to produce a regular vorticity field:

$$\omega_\sigma(y, z) = \frac{1}{\pi\sigma^2} \int_{-b/2}^{b/2} \exp\left(-\frac{(y-y')^2 + z^2}{\sigma^2}\right) \gamma(y') dy'$$

where σ is a regularization parameter, set in the present work to $\sigma/b = 1/75$, hence still giving a fairly thin vortex sheet.

The resulting vorticity field is presented in Fig. 4.1. It corresponds to the vorticity field imposed at the inflow for the S-D simulation without velocity deficit. The time-developing simulation (with velocity deficit) uses this axial vorticity field and adds a spanwise vorticity field that corresponds to an added wake component.

Since the axial velocity deficit wake is a model for the net result of the boundary layers developed along the chord of the wing surface, the same elliptic distribution (as for the chord and lift) is here used :

$$u_w(y, 0) = u_{w0} \sqrt{1 - \left(\frac{y}{b/2}\right)^2}$$

where $u_w = u_x - U_0$ is the velocity deficit, with u_y the axial velocity component and U_0 the freestream velocity. The parameter u_{w0} determines the magnitude of the velocity deficit. The axial direction, x , is defined positively when going downstream of the wing: the velocity deficit, u_w , is thus negative. It is also further regularized to obtain a regular axial velocity field $u_{w,\sigma}(y, z)$. The velocity deficit is thus defined by two parameters: the magnitude, u_{w0} , and the thickness (set by the regularization parameter σ).

The approach used to find realistic simulation parameters consists in considering the different contributions to the total drag. As mentioned in [24], the induced drag contributes to 40 to 50% of the total drag in cruise. Since the induced drag is well approximated by the cross-flow kinetic energy in a close plane behind the wing [25], it is obtained by computing the cross-flow kinetic energy per unit length of the initial vorticity field described above:

$$D_i \simeq E_{c0} = \frac{1}{2}\rho \int_S (u_y^2 + u_z^2) dS = \frac{1}{2}\rho \int_S \psi_x \omega_x dS$$

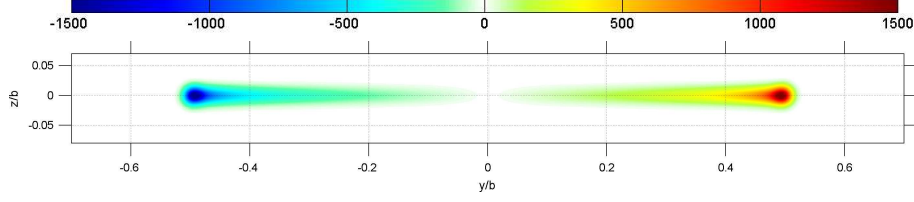


Figure 4.1: Initial axial vorticity field, $\omega_x t_0$, for the reference case (wake without velocity deficit).

where u_y and u_z are the velocity components in y - and z -direction, S is the cross plane (orthogonal to the flight direction x), ω_x is the x -component of vorticity and ψ_x is the stream-function satisfying the Poisson equation, $\nabla^2 \psi_x = \frac{\partial^2 \psi_x}{\partial y^2} + \frac{\partial^2 \psi_x}{\partial z^2} = -\omega_x$.

The total drag is the sum of the induced drag and of the “profile” (viscous) drag. In the configuration investigated here, the velocity deficit is a direct consequence of the development of the boundary layers on the airfoil. The velocity deficit must thus be defined in a such way that the corresponding “profile” drag contributes to 50 to 60% of the total drag. For the spatially-evolving wake, the “profile” drag is obtained as:

$$D_p = \rho \int_S (U_0 - u_x) u_x dS$$

with U_0 the freestream velocity and u_x axial velocity. In the present case, when a time-developing simulation is considered, the relative velocity deficit is as $u_w = -(U_0 - u_x)$, and thus, the profile drag is obtained as :

$$D_p = -\rho \int_S (U_0 + u_w) u_w dS$$

In order to determine u_{w0} , the fact that the initial flow field corresponds to that shed at the trailing edge of the generating wing is considered. Concerning the axial velocity field, the no-slip condition imposes the velocity to decrease to zero at the trailing edge (centerline of the initial vortex sheet). Thus, from this point of view, the natural value for u_{w0} is $-U_0$, corresponding to $u = 0$.

Concerning the regularization parameter: the same value of σ is used in

both models. Thus, the boundary layers have roughly the same thickness as the vortex sheet shed by the wing (from lifting line theory). This set of parameters leads to an induced drag, D_i , and a profile drag, D_p , contributing respectively to 42.4% and 57.6% of the total drag, which is realistic for cruise conditions.

The initial velocity profiles, and the corresponding vorticity and velocity fields obtained are presented in Figs. 4.2 and 4.3, respectively. The initial condition presented in figure 4.3 corresponds to the velocity deficit configuration. The combination of the initial conditions presented in figures 4.1 and 4.3 corresponds to the wake with velocity deficit.

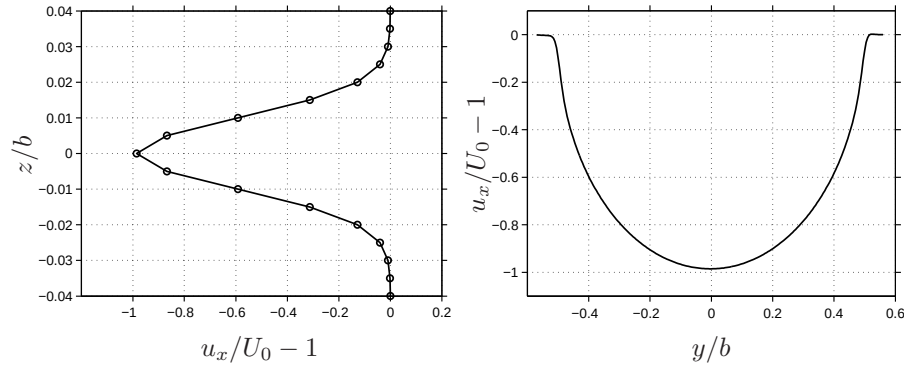


Figure 4.2: Initial axial velocity profiles: (a) $(u_x - U_0)/U_0$ at $y/b = 0$ (the symbols represent the numerical field as discretized with the grid spacing h); (b) $u_x/U_0 - 1$ at $z/b = 0$.

Description of the force distribution model As explained before, the lift per unit length, $l(y)$, is related to the circulation through the relation $l(y) = \rho U_0 \Gamma(y)$. For an elliptically loaded wing,

$$\begin{aligned} l(y) &= \rho U_0 \Gamma_0 \sqrt{1 - \left(\frac{y}{b/2}\right)^2}, \\ &= \rho \frac{2}{\pi} U_0^2 b \left(\frac{C_L}{A_R}\right) \sqrt{1 - \left(\frac{y}{b/2}\right)^2}. \end{aligned}$$

The lift force is further smoothed using a Gaussian kernel. The lift force per unit of volume is taken as:

$$f_L(x, y, z) = l(y) \frac{1}{\pi \sigma_x \sigma_z} \exp\left(-\frac{x^2}{\sigma_x^2} - \frac{z^2}{\sigma_z^2}\right),$$

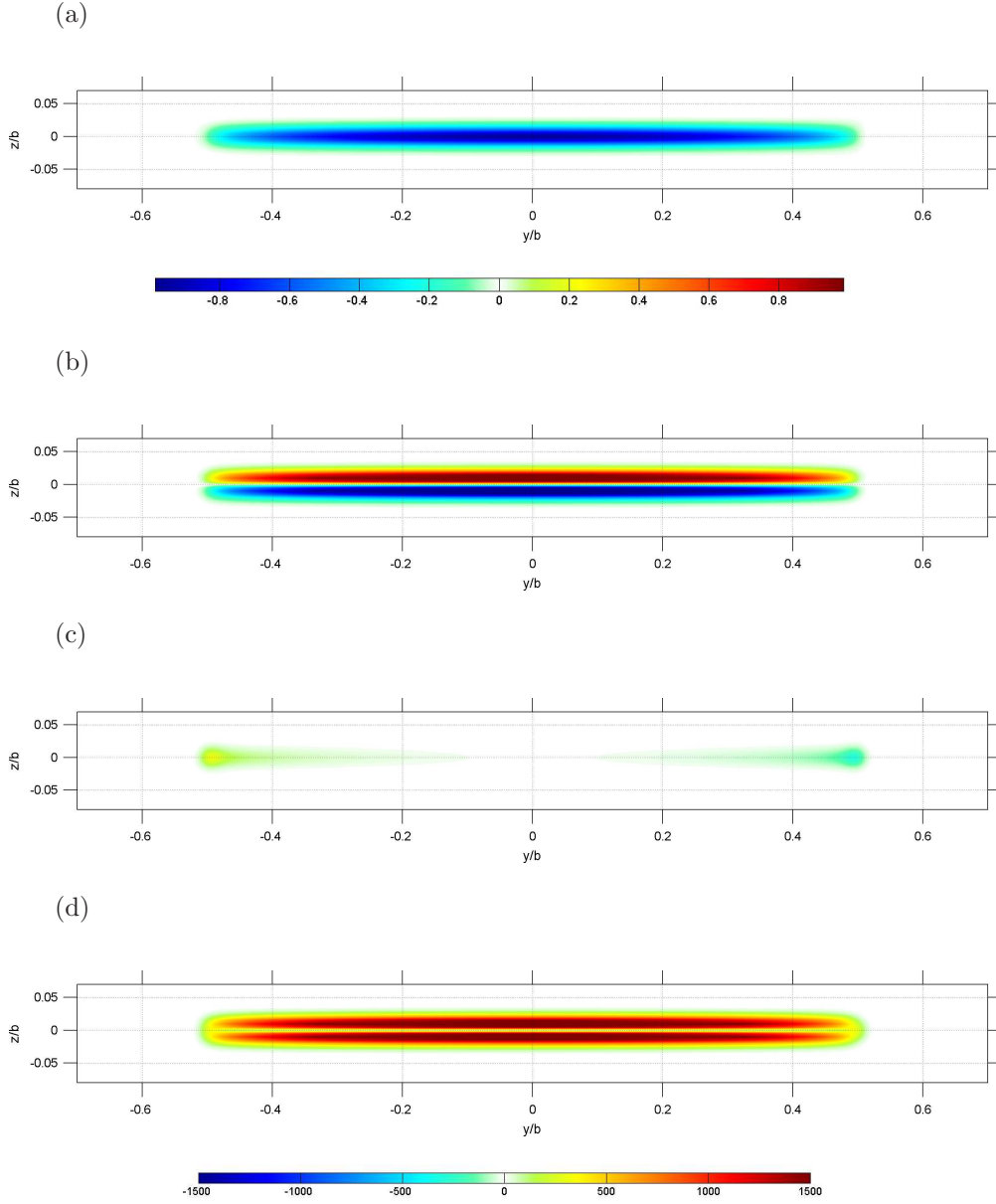


Figure 4.3: Initial velocity and vorticity fields for the velocity deficit configuration: (a) Axial velocity, $u_x/U_0 - 1$; (b) Spanwise vorticity, $\omega_y t_0$; (c) Vertical vorticity, $\omega_z t_0$; (d) transverse vorticity norm, $\omega_{yz} t_0 = (\omega_y^2 + \omega_z^2)^{1/2} t_0$.

where σ_x and σ_z are regularization lengths. We here set $\sigma_x/b = 1/15$ and $\sigma_z = 1/75$ (same value of the vertical regularization parameter as that used for the inflow model). The regularized induced-drag force per unit volume is computed as a “fraction” of the lift using the downwash $\epsilon(y) = \epsilon$; here taken as uniform since an elliptically loaded wing is considered which, according to the Prand'l theory, gives a uniform downwash:

$$\begin{aligned} f_i(x, y, z) &= \epsilon(y) f_L(x, y, z) \\ &= \left(\frac{C_L}{\pi A_R} \right) f_L(x, y, z) \end{aligned}$$

It further makes sense to take into account the velocity deficit effect by adding a model for the friction drag force $f_f(x, y, z)$ proportional to the chord, and thus here also elliptically loaded. The friction drag force per unit volume is thus taken of the same form as that for the induced drag:

$$f_f(x, y, z) = \xi f_i(x, y, z)$$

ξ is here chosen to be unity, consistent with the fact that, for cruise condition, the induced drag contributes to 40...50% of the total drag (see de Bruin [24]). That also leads to a “lift-to-drag ratio” (or “glide-ratio”) equal to:

$$\frac{L}{D_{tot}} = \frac{1}{(1 + \xi) \left(\frac{C_L}{\pi A_R} \right)} \simeq 7.9$$

We also connect the 2-D rollup simulations and the 3-D S-D simulations using:

$$x = U_0 t = U_0 t_0 \tau,$$

with $\tau = t/t_0$ the dimensionless time for the 2-D simulation, with $t_0 = b_0/V_0$ (where $V_0 = \Gamma_0/(2\pi b_0)$ is the wake descent velocity). A station at x/b in the 3-D S-D simulation thus has an equivalent dimensionless time equal to

$$\tau = \frac{x/b}{\left(\frac{U_0}{b} \frac{2\pi b_0^2}{\Gamma_0} \right)} = 3.29 \cdot 10^{-2} x/b.$$

4.3 Numerical setup

The numerical setup of the three simulations is herein described. We first present the two S-D simulations, performed with and without modeling of a velocity deficit (due to the boundary layers). Second, the parameters used for the time-developing simulation are depicted.

S-D simulation without velocity deficit The simulation uses an inflow which is the corresponding axial vorticity field (i.e. vortex sheet) shed by an elliptically loaded wing without the influence of the friction drag. This vortex sheet is regularized using the value $\sigma/h = 1/75$. An outflow condition is also used for which one imposes a “no-through flow” at the outgoing plane, located at $\frac{x}{b} = 10$; the resulting constraints on the vorticity field read (for a $y - z$ plane located at $x = x_o$):

$$\begin{aligned}\omega_n(x_o - x, y, z) &= \omega_n(x_o + x, y, z) \\ \omega_t(x_o - x, y, z) &= -\omega_t(x_o + x, y, z)\end{aligned}$$

where the subscripts n and t stand for normal and tangential. This leads to $u_n(x_o, y, z) = U_0$. A uniform resolution is used and fixed to $h/b = 1/160$. The time-step is chosen as $\Delta t/t_0 = 6.45 \cdot 10^{-5}$. The total number of grid point is roughly $\sim 45 \cdot 10^6$.

S-D simulation with velocity deficit An external force effect is added to take into account the force exerted by the wing on the fluid; i.e. the lift and the total drag (friction and induced). In practice, as the vorticity-velocity formulation of the Navier-Stokes equations is used, ones adds the curl of this external force, computed on the mesh.

An adaptive time step is used (see Koumoutsakos [46]). The time constraint is based on the strain tensor $\mathbf{S} \triangleq \frac{1}{2} (\nabla \mathbf{u} + (\nabla \mathbf{u})^T)$:

$$\Delta t |\mathbf{S}|_{max} < 0.1,$$

where $|\mathbf{S}|_{max}$ is the maximum of the sum of the absolute values of the strain

tensor lines:

$$|\mathbf{S}|_{max} \triangleq \max_i \left(\sum_j |S_{ij}| \right)$$

For this LES, we here use a “complete-small” subgrid-scale model (referred as RVM model in Cogle et al. [12]). The turbulent viscosity scales with the complete strain tensor field while it only acts on small scales of the LES field. For this simulation, one benefits from the multi-resolution capability as described in Chapter 3. The mesh size function $\delta(\mathbf{x})$ is defined as follows

$$\delta(\mathbf{x}) = \begin{cases} \frac{h^0}{b} & \text{if } \frac{x}{b} \leq 2.0 \\ 2\frac{h^0}{b} & \text{if } 2 \leq \frac{x}{b} \leq 6 \\ 4\frac{h^0}{b} & \text{if } 6 \leq \frac{x}{b} \leq 10 \\ 10(x - 10) + 4\frac{h^0}{b} & \text{if } \frac{x}{b} \geq 10 \end{cases}$$

where $h^0/b = 1/250$. Notice that the multi-resolution methodology permits to use a much better resolution in the near-field; here defined as being the region up to two wingspans. The entire flow is captured up to 16 wingspans. From $x/b = 10$, the resolution is forced to deteriorate, constituting an outflow condition. No further outflow condition is used: the vortical structures are simply coarsened, convected and, finally, removed. One defines the total energy, E , and the enstrophy, ε , of the system

$$\begin{aligned} E &= \frac{1}{2} \int_V \boldsymbol{\omega} \cdot \boldsymbol{\Psi} dV, \\ \varepsilon &= \frac{1}{2} \int_V \boldsymbol{\omega} \cdot \boldsymbol{\omega} dV. \end{aligned}$$

The time-history of the energy and of the enstrophy are presented in Fig. 4.4.(a) and 4.4.(b) respectively. At the beginning, the energy and the enstrophy rise. They eventually reach a statistically stationary regime from the time $tU_0/b \simeq 17$. Yet the simulation was run up to $tU_0/b \simeq 47$. The average and the fluctuation average of the flow are post-processed using 91 instantaneous fields in the time frame [17 – 47]. The divergence level is maintained at a low level $\lesssim 4 \cdot 10^{-4}$ (see Fig. 4.4.(c)). The history of the time step is shown in Fig. 4.4.(d). For information, the time step, scaled with the maximum vortic-

ity, is also presented and was found to be roughly $\Delta t \max(|\omega|) \simeq 0.20 \dots 0.35$. The simulation was carried out using 128 cpu's during roughly 2 weeks. The total number of grid point was $\sim 90 \cdot 10^6$.

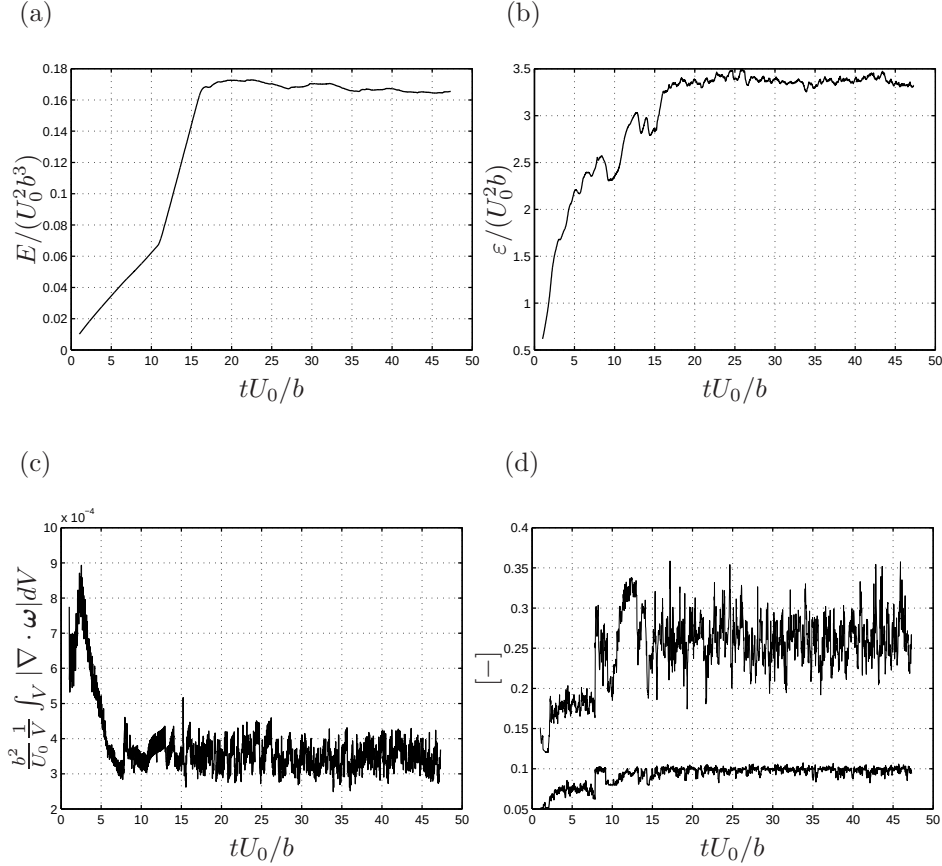


Figure 4.4: S-D simulation of the wake rollup with velocity deficit: (a) time-history of the energy, (b) of the enstrophy, (c) the divergence error and (d) the time constraints $\Delta t \max(|\omega|)$ (thin line) and $\Delta t |S|_{max}$ (thick line).

T-D simulation with velocity deficit The simulation is initialized using the vorticity field model consisting in the vorticity sheet (due to the lift) combined with transverse vorticity (due to the friction drag). A periodic condition is used in the axial (i.e. streamwise) direction. The periodic length is fixed to $0.5b$; it has been shown that it is sufficient to capture the phenomena of interest, see Cottin [19]. The time step is set to $\Delta t/t_0 = 3.23 \cdot 10^{-3}$. A uniform mesh size was used: $h/b = 1/160$. The flow was computed up to the

time $\tau = t/t_0 = 0.3$. Those results have been extensively investigated in the FAR-Wake technical report [19]. One should refer to this report for the related results presented in this chapter.

4.4 Results

First, the vortex dynamics are qualitatively investigated. For each case, instantaneous 3-D surfaces and averaged 2-D slices are shown. Second, a complete characterization of the resulting rolled-up wake is carried out; circulation profiles as well as diagnostics are provided.

4.4.1 3D dynamics: a qualitative analysis

In Fig. 4.5, one presents a 3-D isocontour of the vorticity field for the S-D simulation without velocity deficit. The “classical” simple vortex-sheet rollup is observed; i.e. without generation of turbulence by instabilities. For a $C_L/A_R = 0.2$, the rollup of the vortex system is almost complete at $8 \cdots 10$ wingspans downstream; i.e. when the vortex spacing is equal to the vortex centroid $b_0 = \frac{\pi}{4}b$. The flow is fully stationary.

Fig. 4.6 shows a 3D view of the λ_2 iso-surface (see Jeong and Hussain [41]) of the S-D simulation with velocity deficit. Fig. 4.8.(a) and 4.8.(b) show the related top-view and side-view. The λ_2 criterion is a better vortical-structure tracer than the vorticity field, especially for multi-resolution simulations. A zoom in the near field is also presented in Fig. 4.7. A fast transition to turbulence is observed within the first two wingspans downstream. In the central part of the flow (i.e., in the low vortex-stretching region), jet-like instabilities develop: very-short longitudinal wavelength instabilities and streamwise Kelvin-Helmoltz instabilities. Those instabilities quickly grow, saturate and eventually generate turbulence. In the meanwhile, a supplementary helical instability develops in the primary vortex region; associated with a high vortex-stretching. In this region, the vortex sheet, containing high transverse vorticity, becomes unstable and is wrapped around the vortex core. Eventually, the flow becomes fully turbulent. Moreover, the multi-resolution transition is qualitatively well performed: the small vortical structures are properly convected

through the common interface of two blocks with different resolution. Concerning the T-D rollup with velocity deficit, Fig. 4.9 presents the 3-D vortex wake using a vorticity iso-surface. The same global dynamics as the S-D case are observed. For an exhaustive investigation of this case, refer to Cottin et al. [19].

Concerning the S-D simulation using a modeled wing force, it is worth noting that the “wing” is half-formed at the station $x/b = 0$ because of the symmetry of the external force with respect to $x = 0$. The initial time $\tau = 0$ is defined at the “trailing edge” arbitrarily fixed at $x/b = 2\sigma_x$.

4.4.2 Quantitative analysis

Here, a quantitative analysis of the vortex-wake rollup is presented for each case. Various slices at different downstream distances of the averaged stream-wise vorticity are shown in Fig. 4.10, 4.11, 4.12 for the S-D simulation with and without velocity deficit and for the T-D simulation with velocity deficit respectively. The spirally rolling up of the vortex-sheet is clearly observed especially for the case without velocity deficit. The physics, associated to both cases with velocity deficit, are similar. However, the vortex structures in the central part decay much faster in the S-D simulation. Fig. 4.13 shows the related transverse vorticity. We see the double shear layer emanating from the wing and corresponding to the effect of two modeled boundary layers. Those are at the origin of the transition to turbulence in the central region.

The averaged velocity deficit, $\overline{u}_w \triangleq \overline{u}_x/U_0 - 1$, is presented in Fig. 4.14, 4.15 and 4.16 for the SD simulation with and without velocity deficit effect and for the T-D simulation with velocity deficit. The velocity deficit, occurring in the S-D simulation without velocity deficit, is due to the combination of the S-D effect associated to the spirally rolling up of the wake. This leads to a velocity deficit in the vortex inner-core, which is relatively low $u_x(\mathbf{x}_c)/U_0 - 1 \simeq -0.1$ (where $\mathbf{x}_c$ is center of the primary vortices) surrounded by a velocity surplus. The global physics of the S-D case with velocity deficit are very different. As the primary vortices form, a strong velocity deficit is entrained inside the vortices. A velocity surplus (i.e. a “jet”) shortly occurs, between $\tau = 0.1 \cdots 0.2$, in the vortex center. This “jet” component quickly diffuses, eventually vanishes and

definitely gives way to a net significant deficit component $\bar{u}_x/U_0 - 1 \simeq -1.8$.

A helical instability inside the vortices can occur in the condition to have a low swirl number $S_{wirl} \triangleq \max(u_\theta)/\max(u_w)$, where u_θ is the tangential velocity (see therein for more details). For a q-vortex model, the vortex becomes unstable if $S_{wirl} \lesssim 1.8$, consistent with the significant wake drag. Here, the value is much higher $\sim 4 \cdots 5$, the vortices are thus stable.

The deficit in the vortex inner-core for the equivalent T-D simulation is much different, see Fig. 4.16. The velocity deficit, in the vortex core as well as in the central part, remains quite high $\bar{u}_x/U_0 - 1 \simeq -5$. The related swirl number is $S_{wirl} \simeq 1.3 \cdots 1.8$; a helical instability then develops, see Fig. 4.9 and Cottin et al. [19].

We define the turbulent kinetic energy (TKE), $\bar{k}$, as follows

$$\bar{k} \triangleq \frac{1}{2} \left(\overline{(u'_x)^2} + \overline{(u'_y)^2} + \overline{(u'_z)^2} \right),$$

where $u'_\alpha \triangleq u_\alpha - \bar{u}_\alpha$, see Fig. 4.17 and Fig. 4.18 show the TKE-planes for both cases. The turbulence level is found to be of the same order of magnitude in both cases.

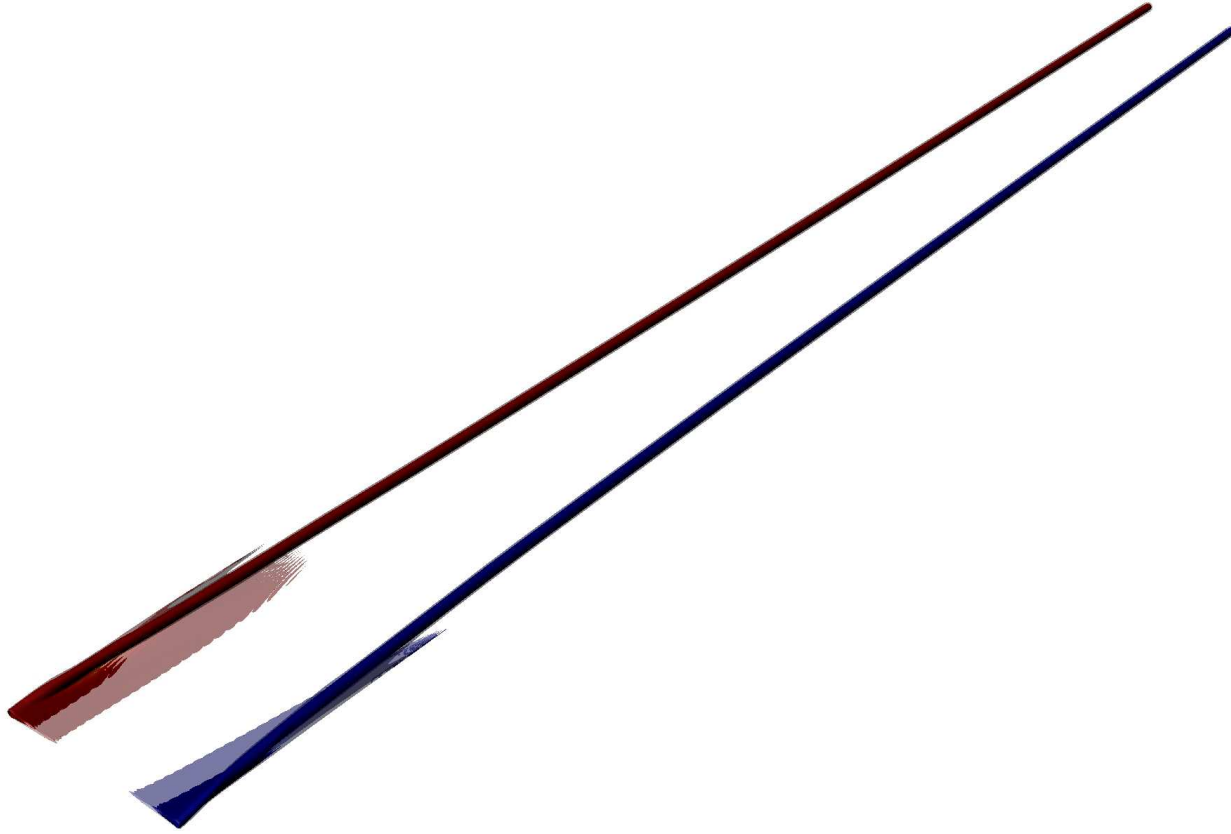


Figure 4.5: Space-developing simulation of the vortex-sheet rollup without velocity deficit: 3D perspective view of two isocontours of the vorticity norm $\|\boldsymbol{\omega}\|_{t_0} = [200\ 400]$ colored with the axial vorticity component.

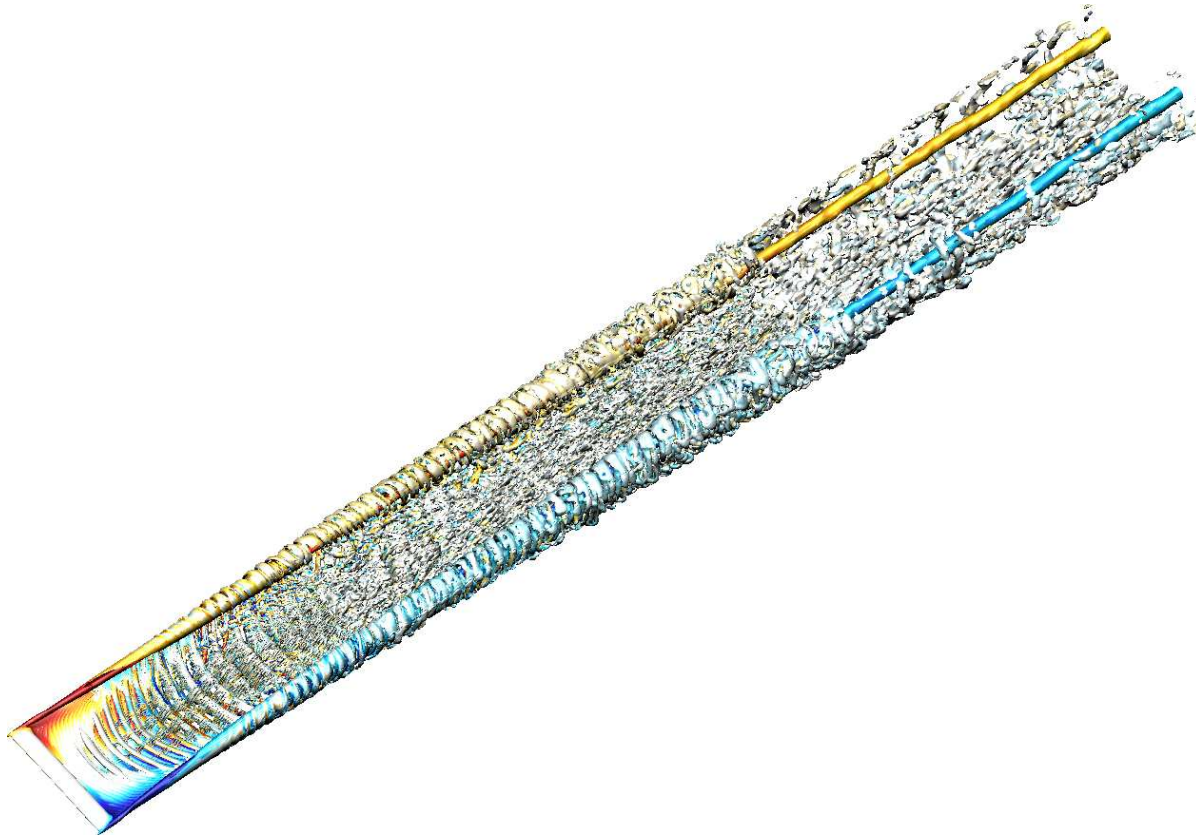


Figure 4.6: Space-developing simulation of the vortex-sheet rollup with velocity deficit: 3D perspective view of the λ_2 surface colored with the streamwise vorticity component.

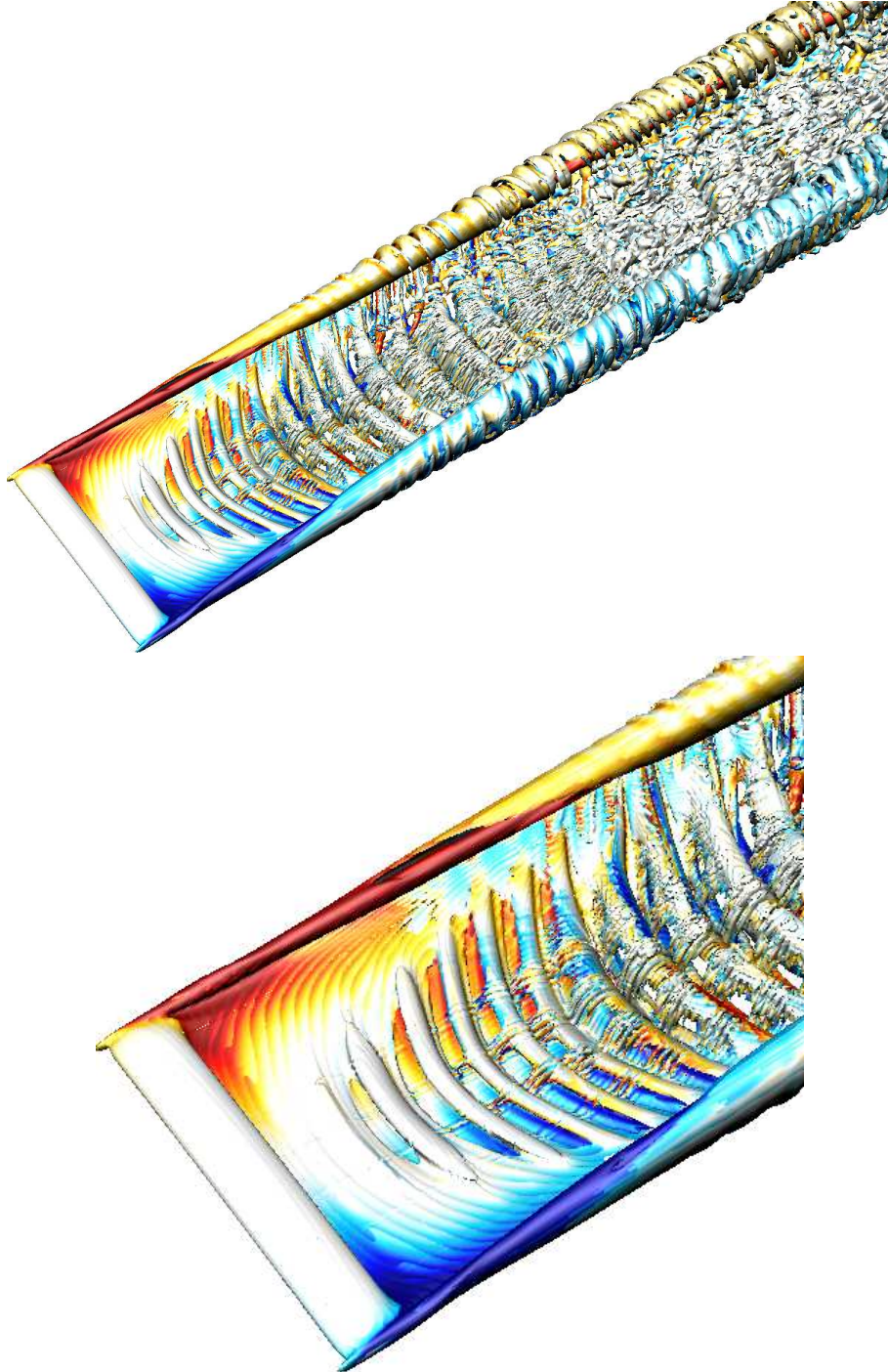


Figure 4.7: Space-developing simulation of the vortex-sheet rollup with velocity deficit: 2 zooms in the near-field region of the 3D perspective view of the λ_2 surface colored with the streamwise vorticity component.

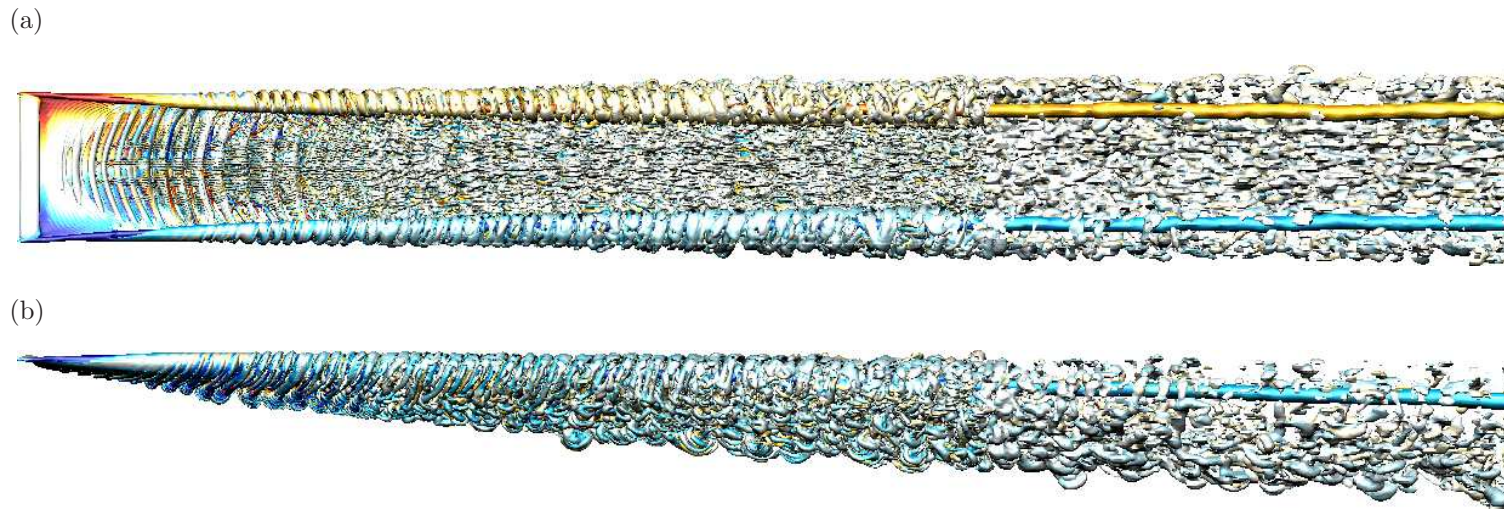
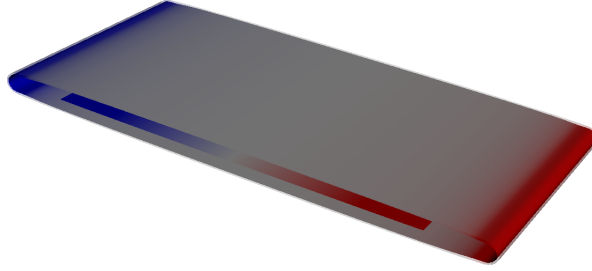
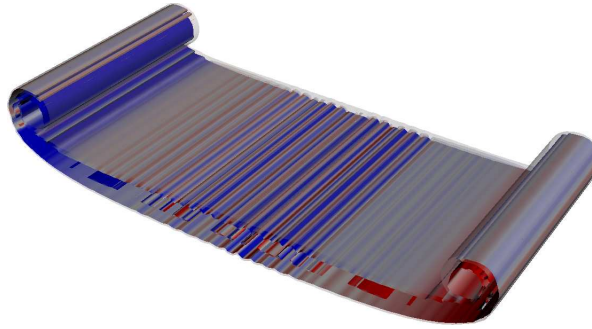


Figure 4.8: Space-developing simulation of the vortex-sheet rollup with velocity deficit: top view (a) and side view (b) of the λ_2 surface colored with the streamwise vorticity component.

$\tau = 0.0$



$\tau = 0.023$



$\tau = 0.065$

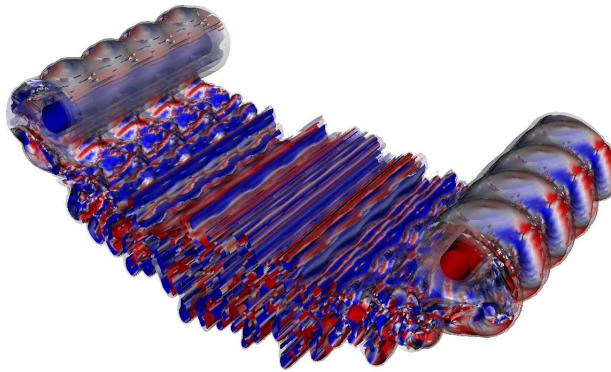
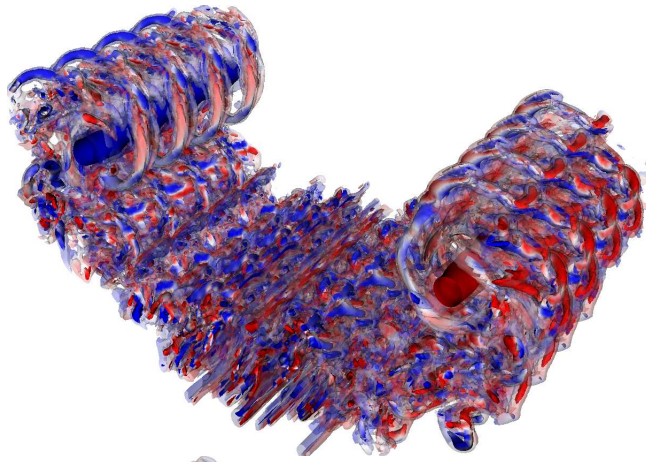
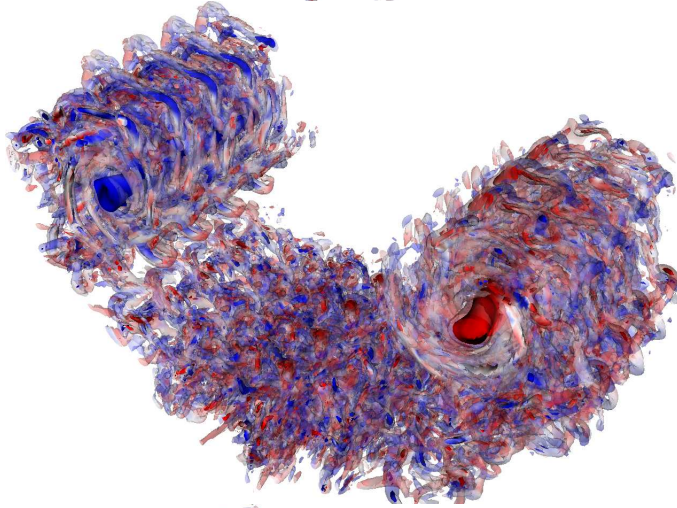
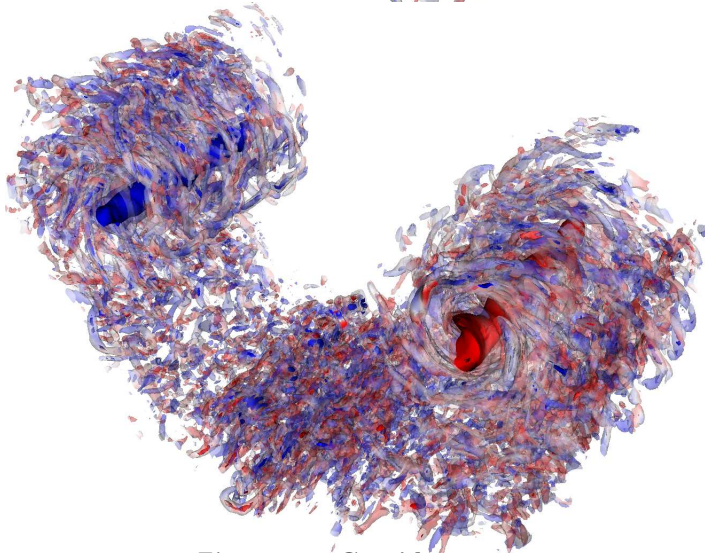


Figure 4.9: Time-developing simulation with velocity deficit: 3D view of two isocontours of the vorticity norm $\|\omega\|_{t_0} = [200, 400]$ colored by the axial vorticity.

$\tau = 0.13$  $\tau = 0.20$  $\tau = 0.26$ *Figure 4.9: Cont'd*

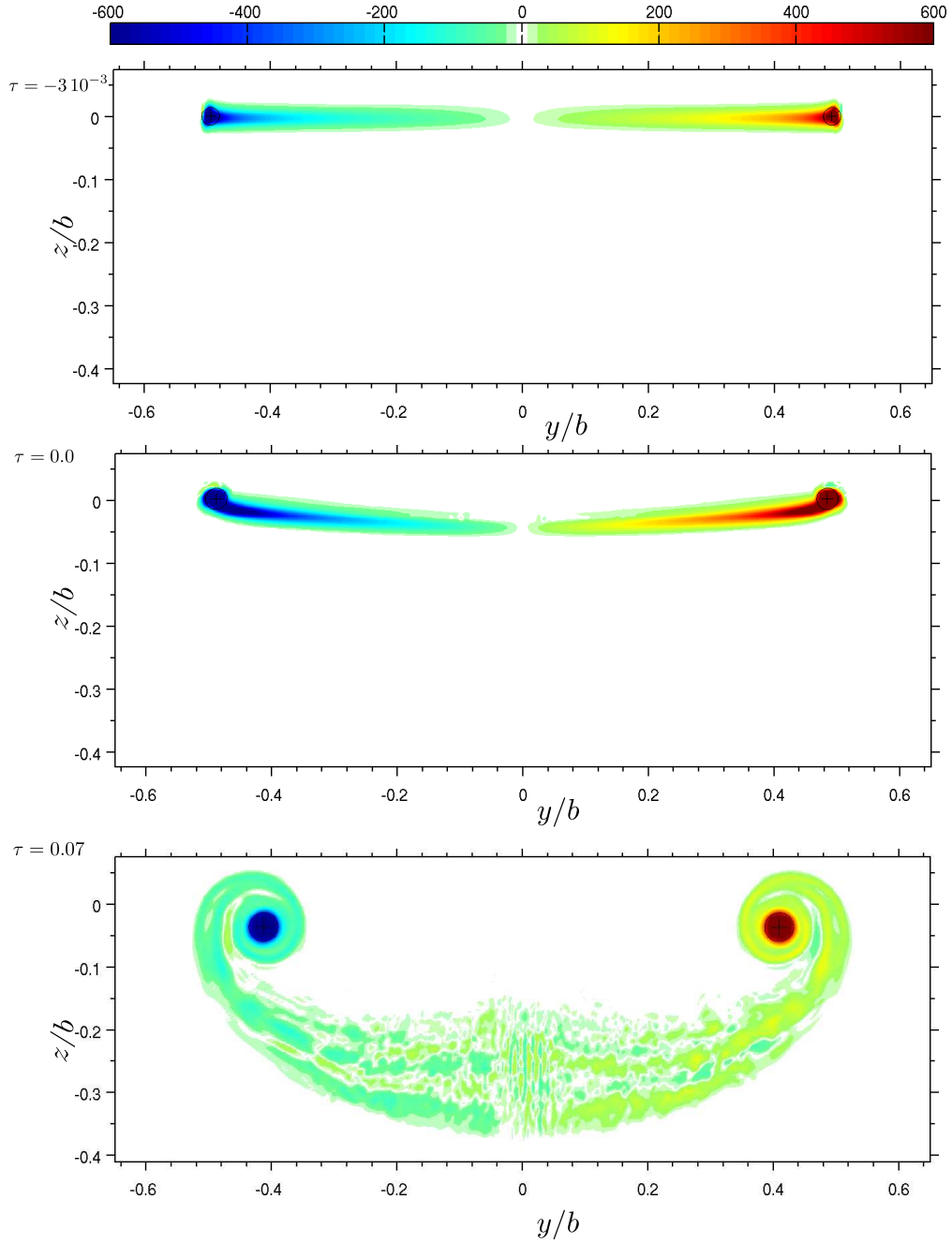


Figure 4.10: Space-developing simulation with velocity deficit: slices of the time-averaged streamwise vorticity $\bar{\omega}_x t_0$; from top to bottom, $\frac{x}{b} = [0, 0.1, 2, 4, 8]$, equivalent time $\tau = [-0.003, 0.0, 0.07, 0.1, 0.3]$.

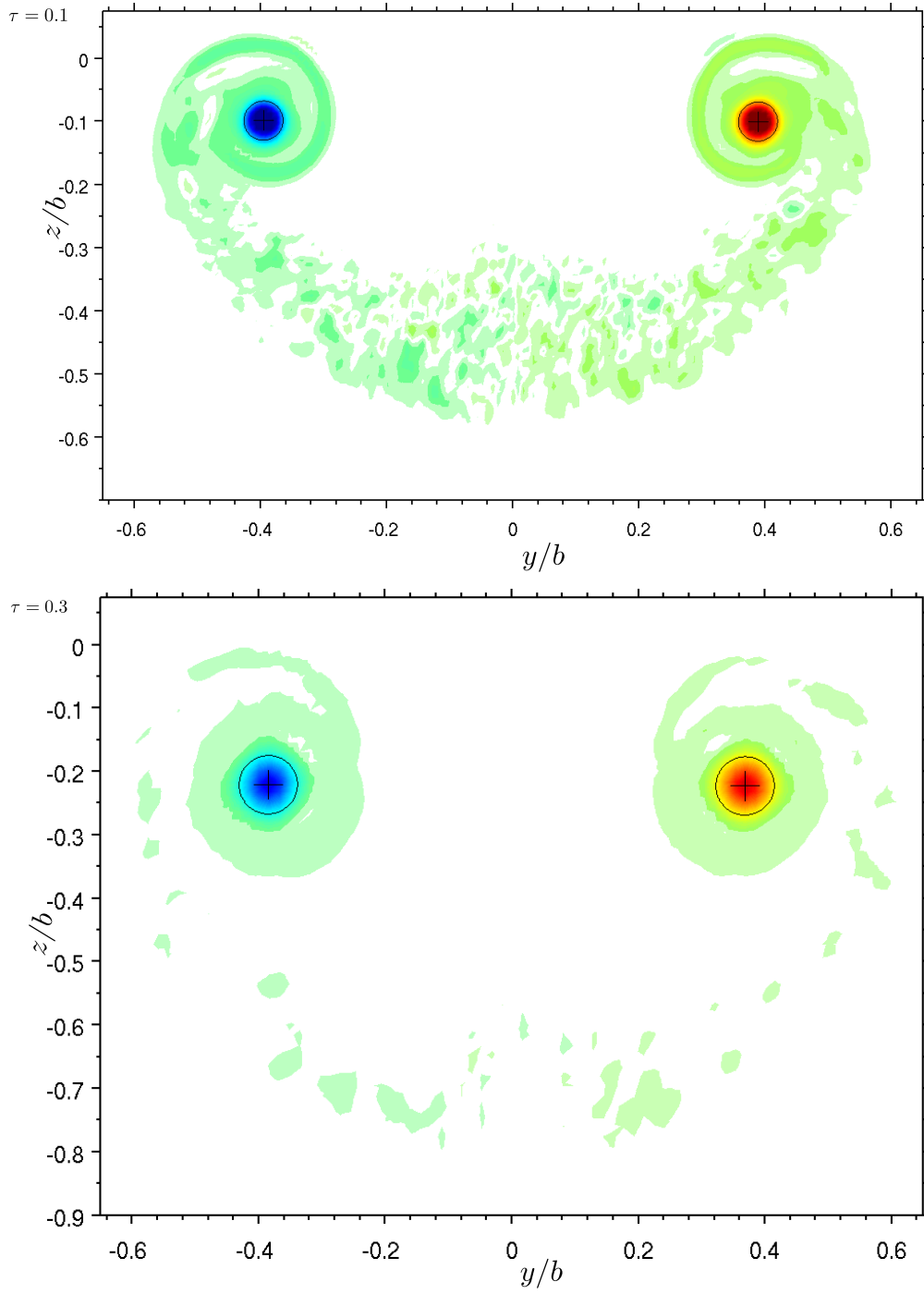


Figure 4.10: cont'd.

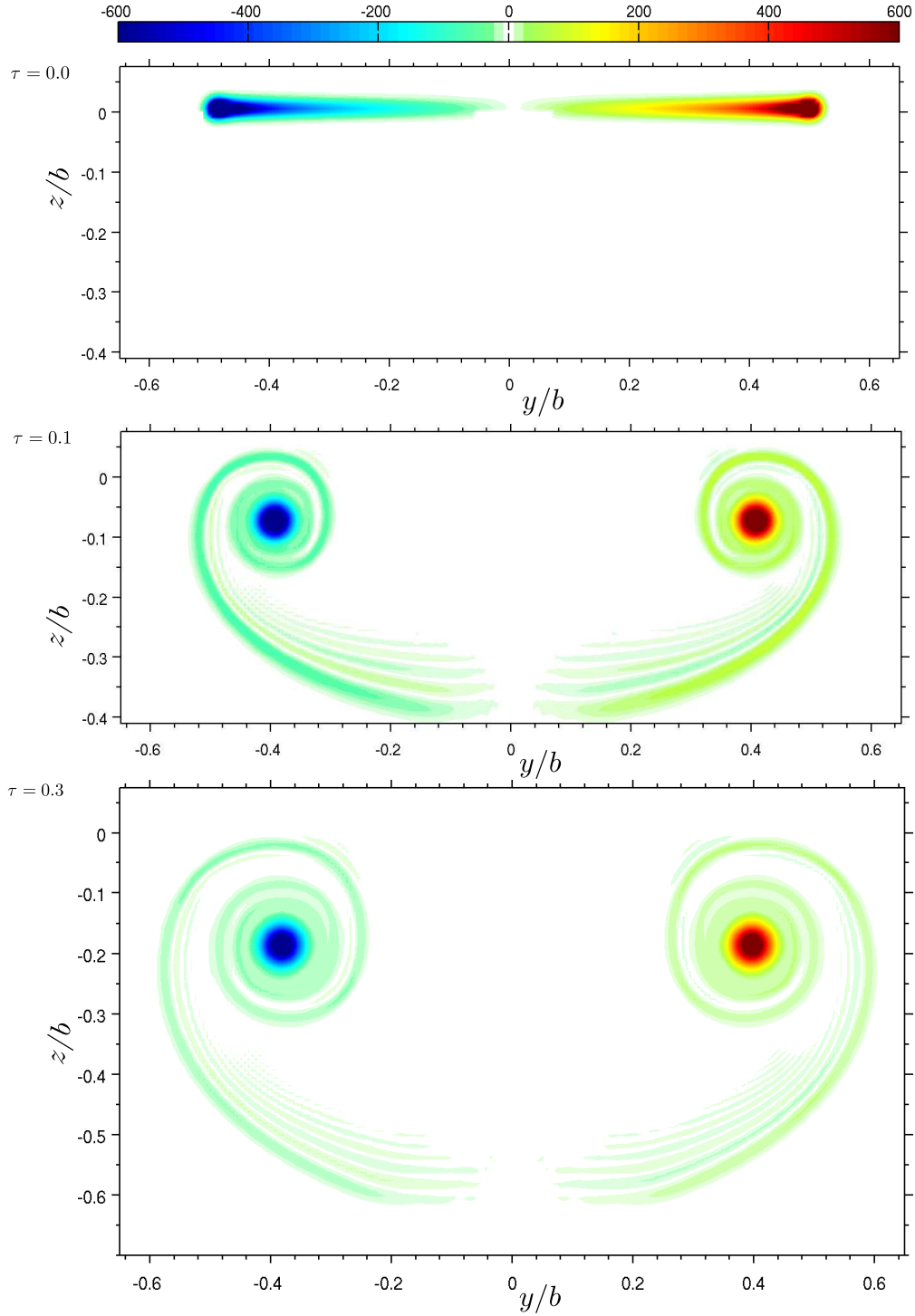


Figure 4.11: Space-developing simulation without velocity deficit: slices of the streamwise vorticity $\bar{\omega}_x t_0$; from top to bottom, $\frac{x}{b} = [0, 4, 8]$, equivalent time $\tau = [0.0, 0.1, 0.3]$.

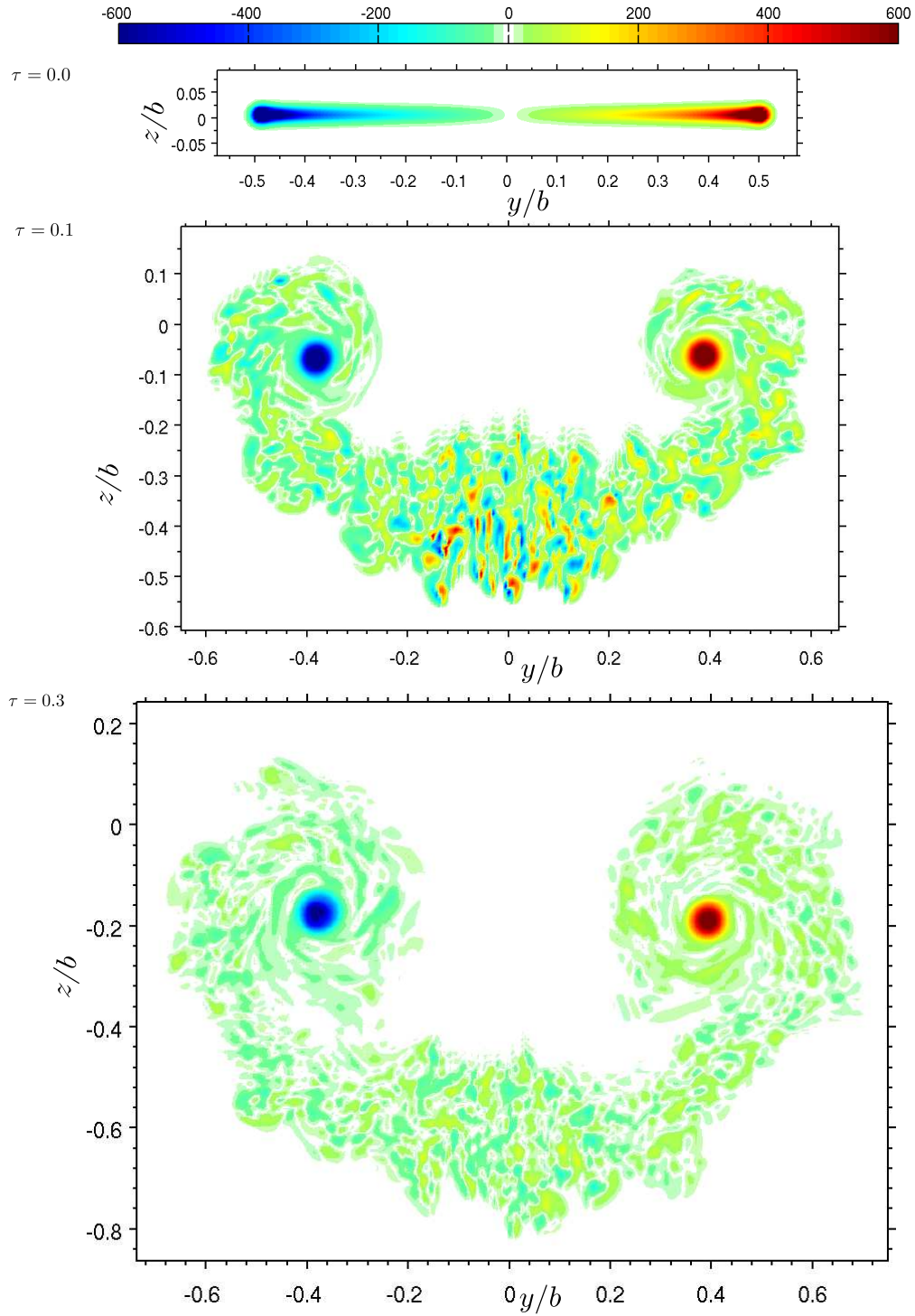


Figure 4.12: Time-developing simulation with velocity deficit: longitudinally-averaged streamwise vorticity $\overline{\omega_x} t_0$; from top to bottom, $\frac{x}{b} = [0, 4, 8]$, equivalent time $\tau = [0.0, 0.1, 0.3]$.

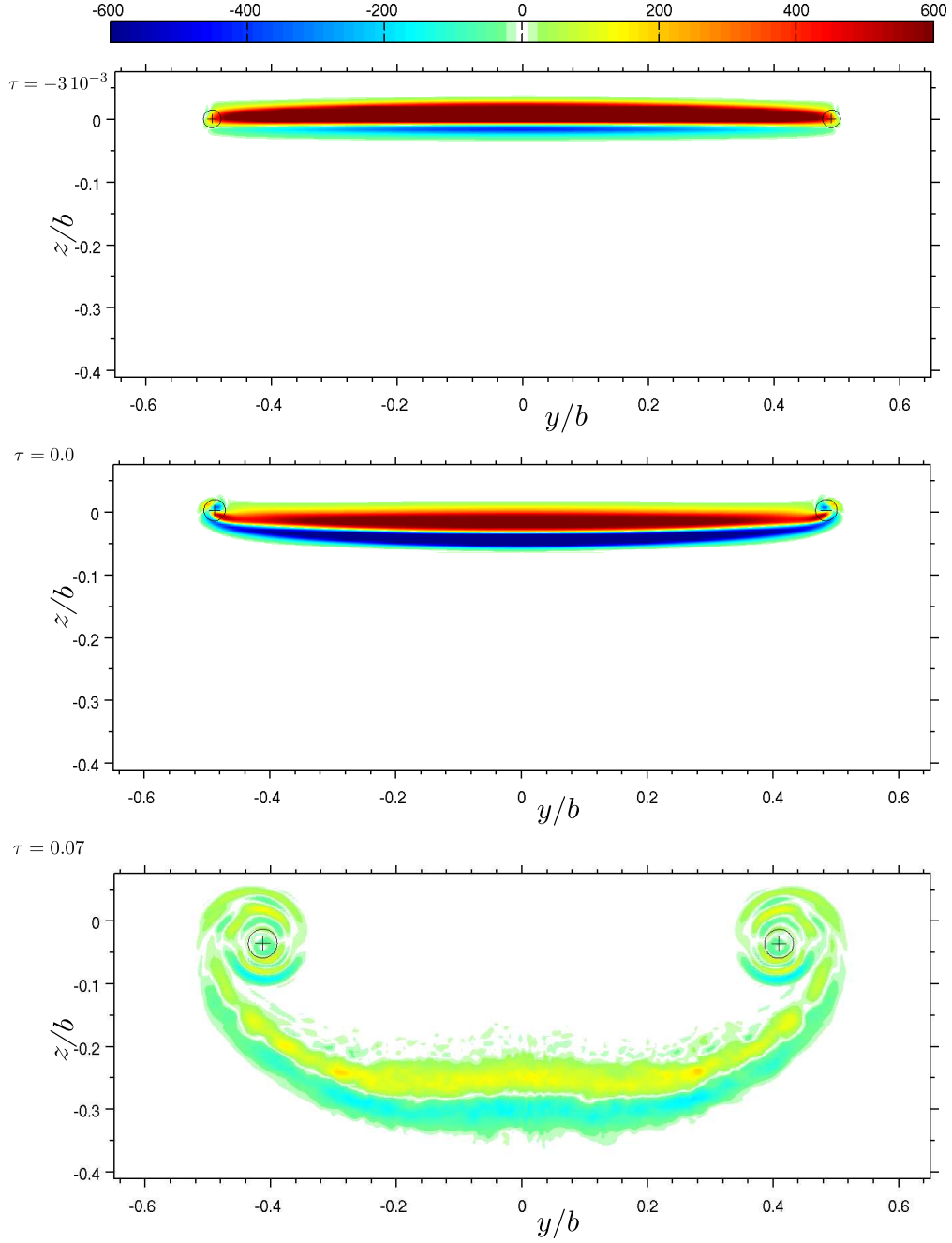


Figure 4.13: Space-developing simulation with velocity deficit: slices of the time-averaged spanwise vorticity $\overline{\omega_y} t_0$; from top to bottom, $\frac{x}{b} = [0, 0.1, 2, 4, 8]$, equivalent time $t = [-0.003, 0.0, 0.07, 0.1, 0.3]$.

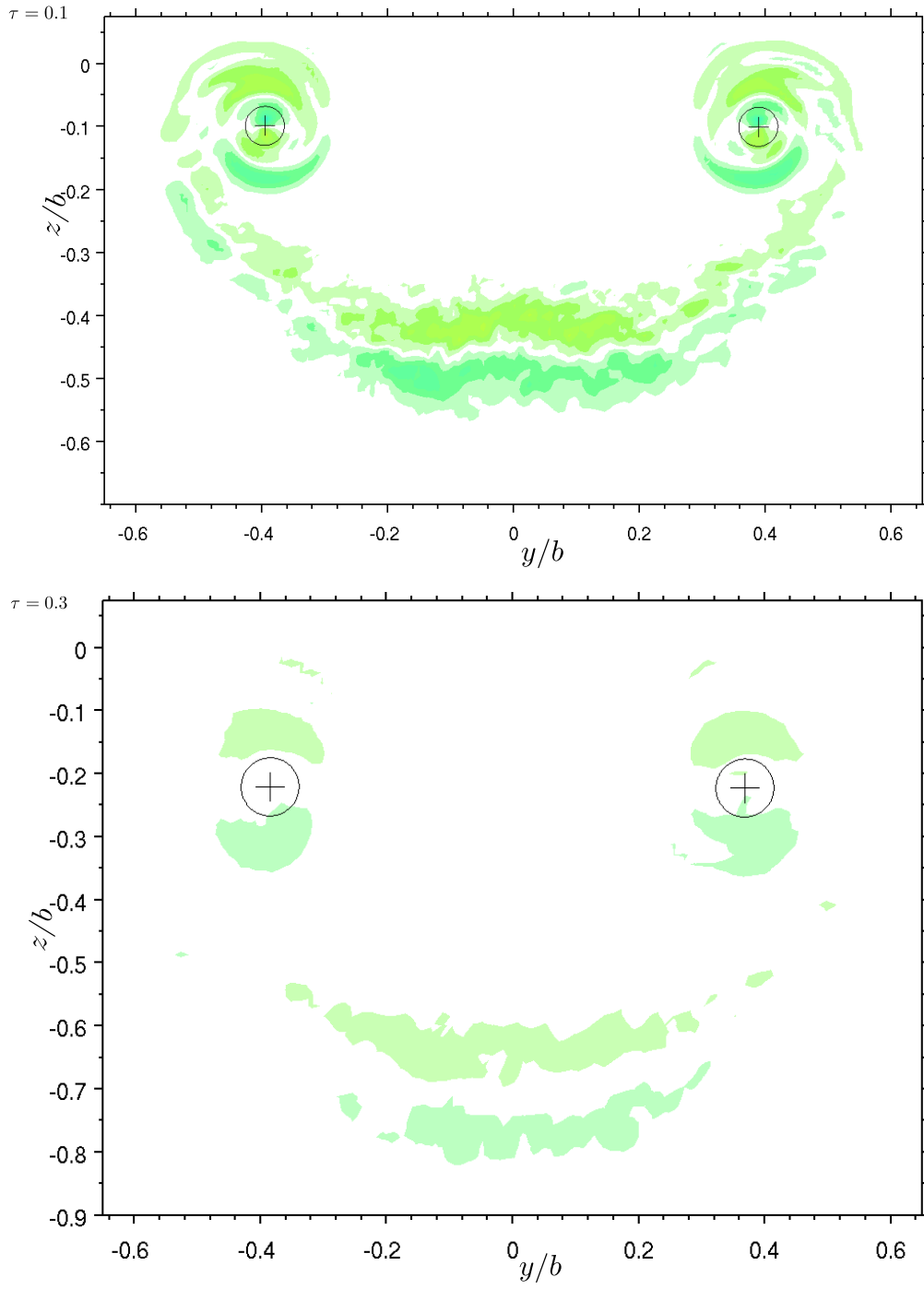


Figure 4.13: cont'd.

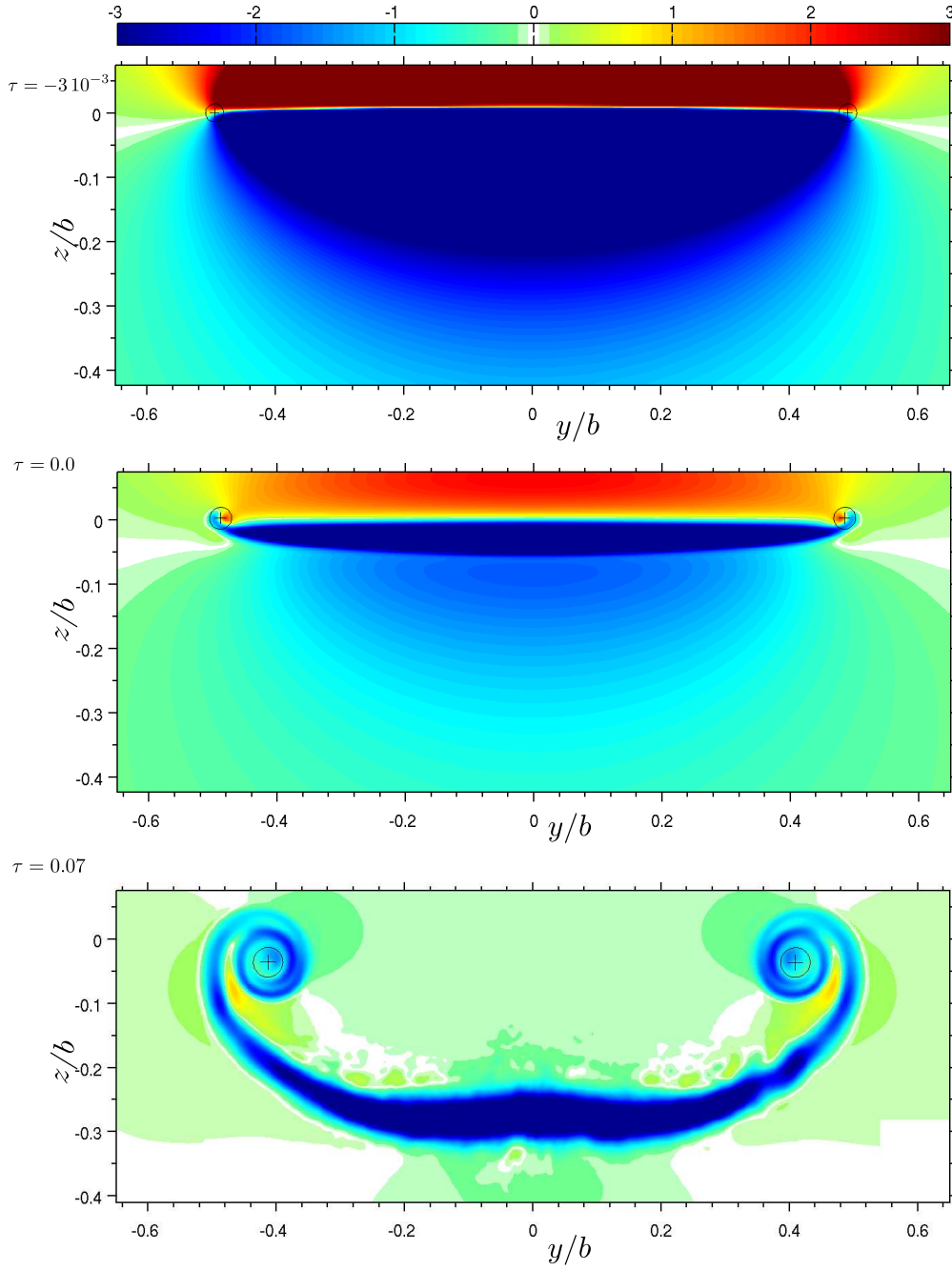


Figure 4.14: Space-developing simulation with velocity deficit: slices of the time-averaged streamwise velocity $\overline{u_w}/V_0$; from top to bottom, $\frac{x}{b} = [0, 0.1, 2, 4, 8]$, equivalent time $\tau = [-0.003, 0.0, 0.07, 0.1, 0.3]$.

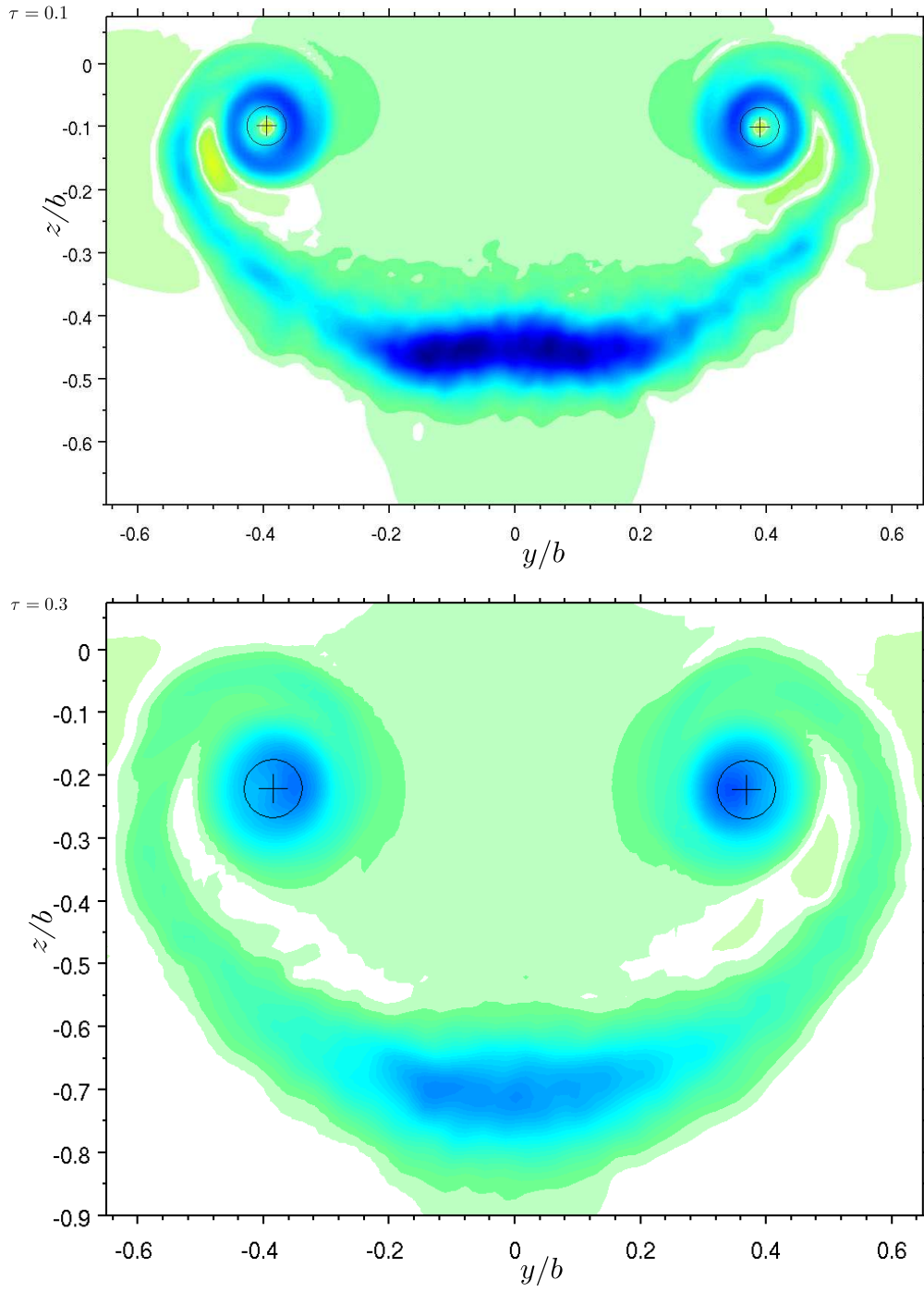


Figure 4.14: cont'd.

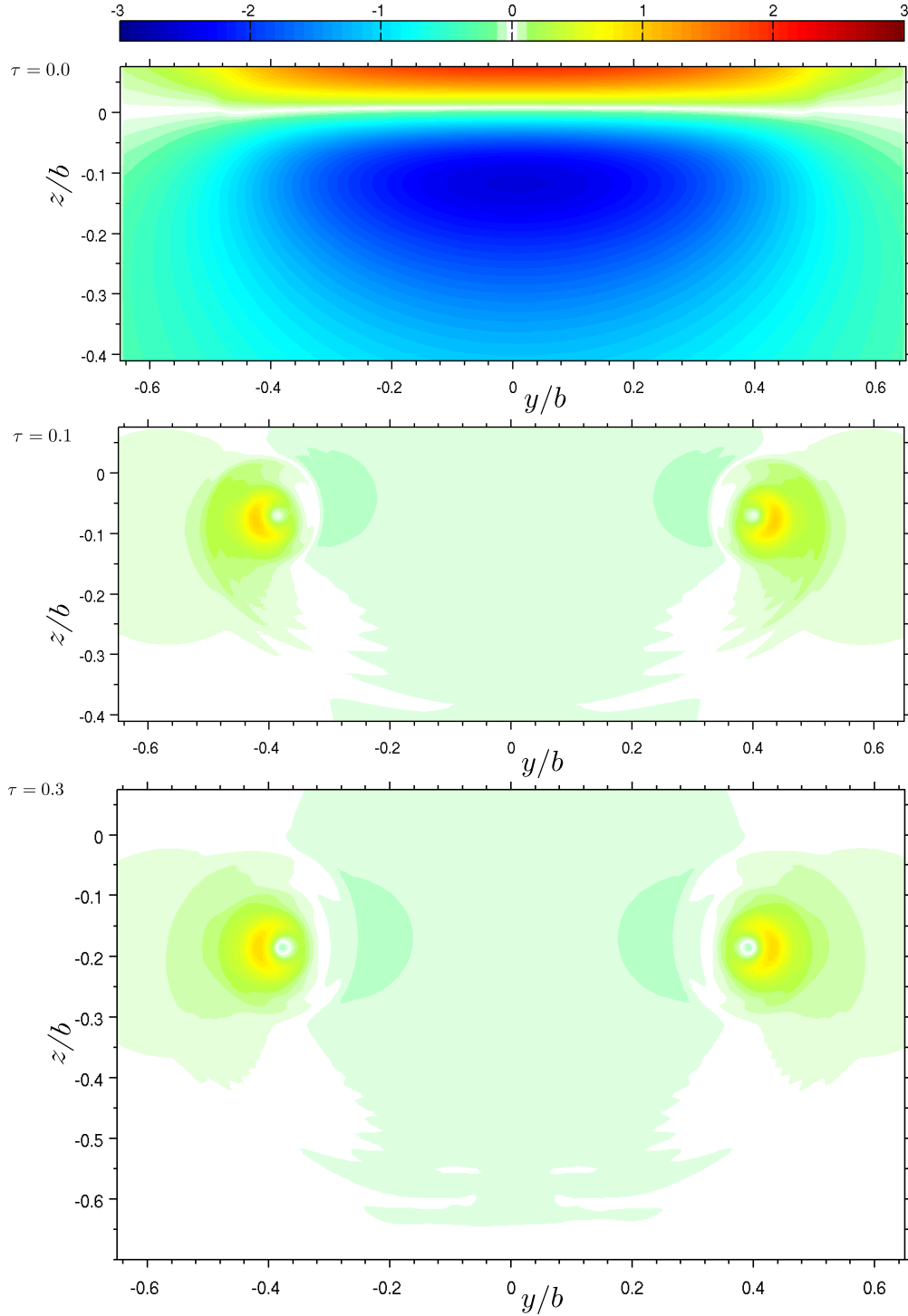


Figure 4.15: Space-developing simulation without velocity deficit: slices of the time-averaged streamwise velocity $\overline{u_w}/V_0$; from top to bottom, $\frac{x}{b} = [0, 4, 8]$, equivalent time $t = [0.0, 0.1, 0.3]$.

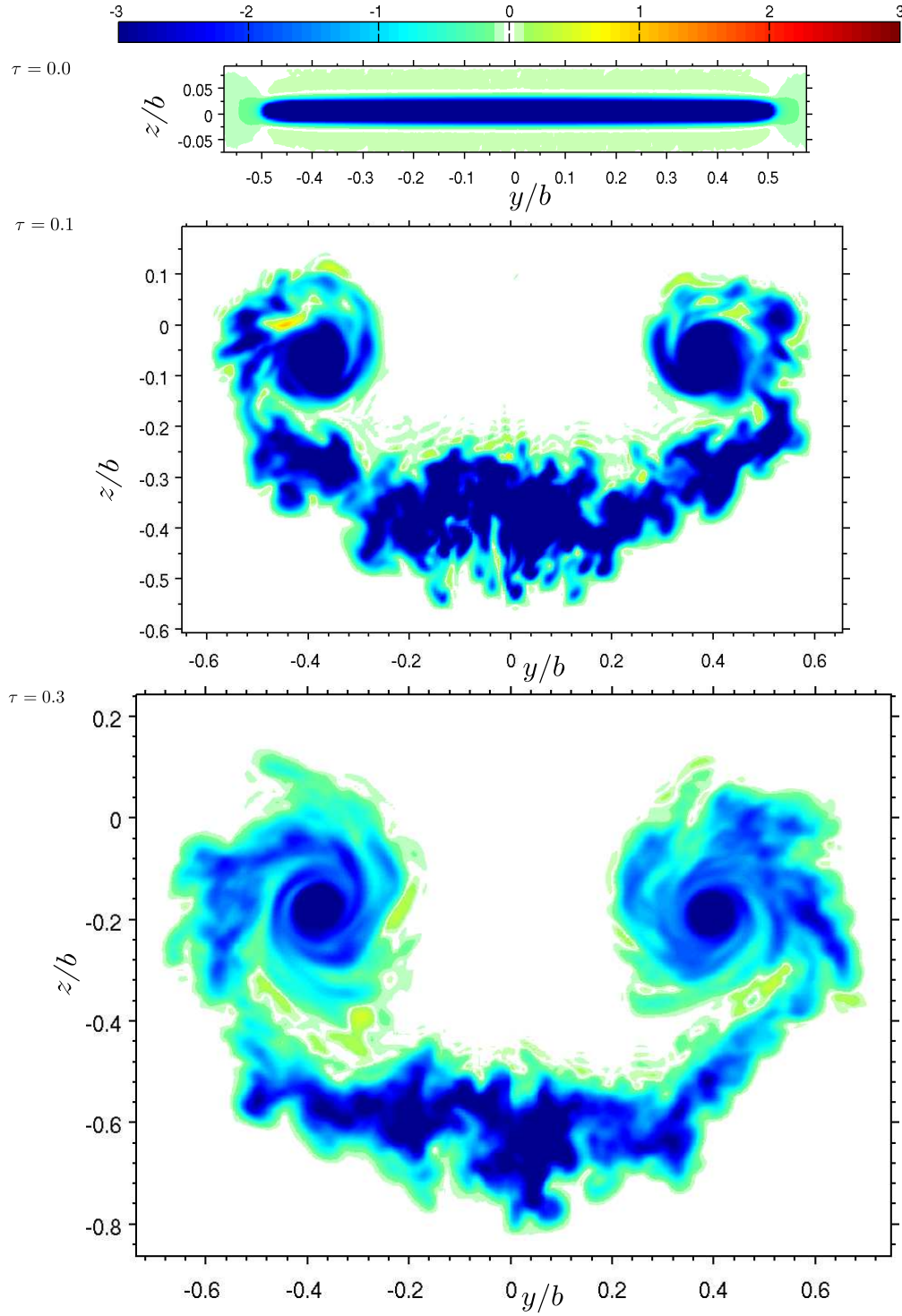


Figure 4.16: Time-developing simulation with velocity deficit: longitudinally-averaged streamwise velocity $\overline{u_w}/V_0$; from top to bottom, time $\tau = [0.0, 0.1, 0.3]$.

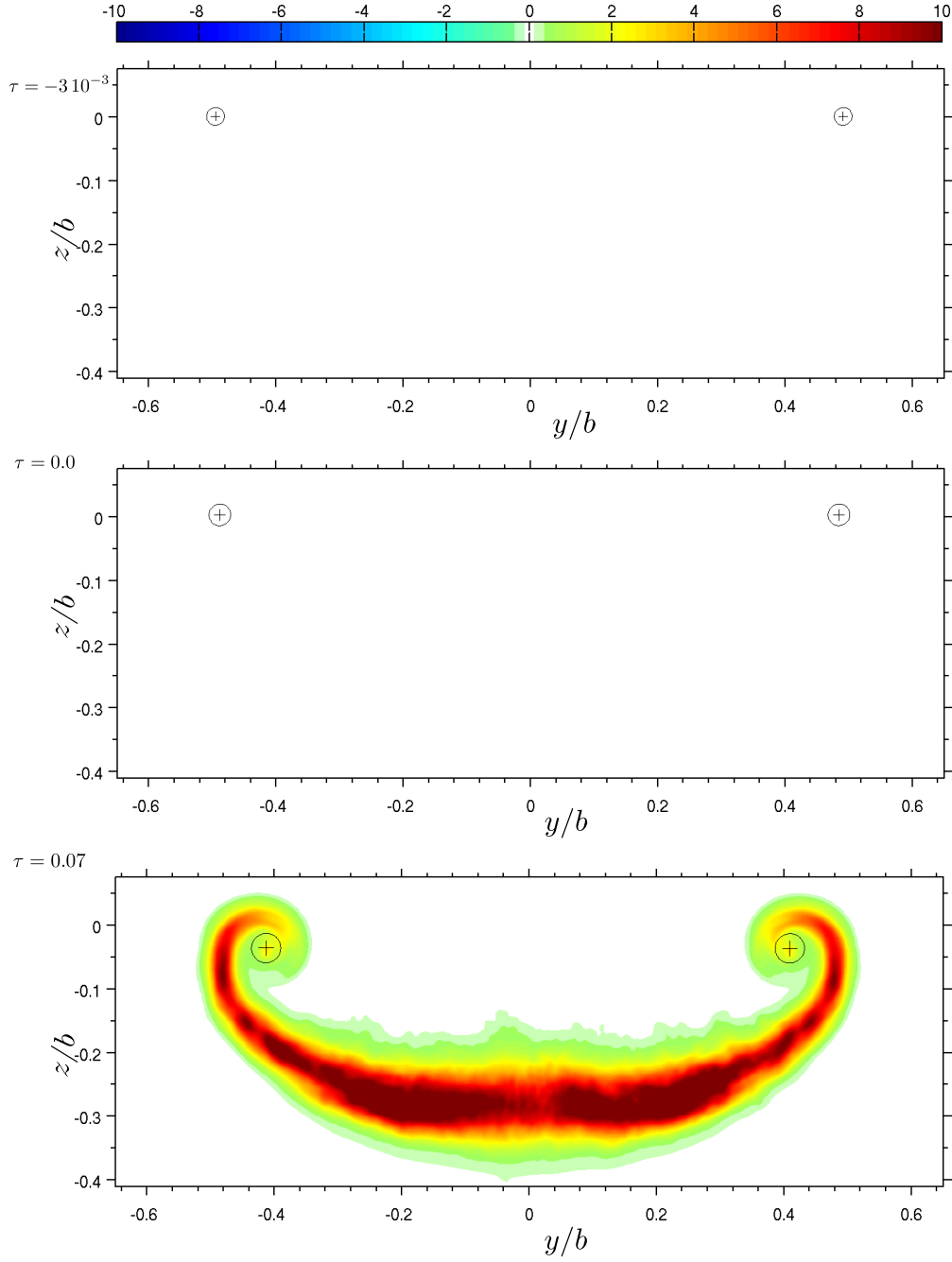


Figure 4.17: Space-developing simulation with velocity deficit: slices of the turbulent kinetic energy $\bar{k}/V_0^2$; from top to bottom, $\frac{x}{b} = [0, 4, 8]$, equivalent time $\tau = [-0.003, 0.0, 0.07, 0.1, 0.3]$.

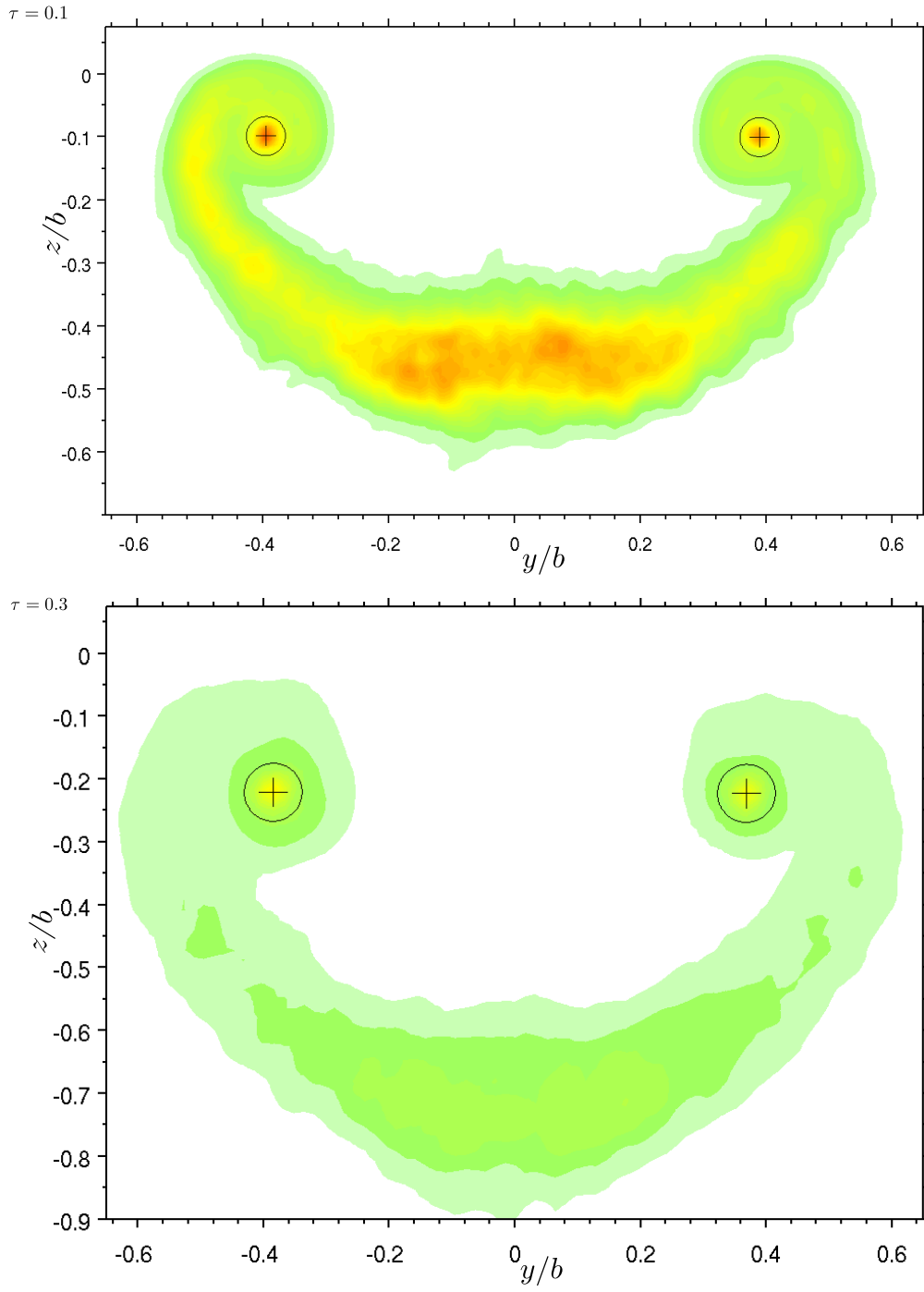


Figure 4.17: cont'd.

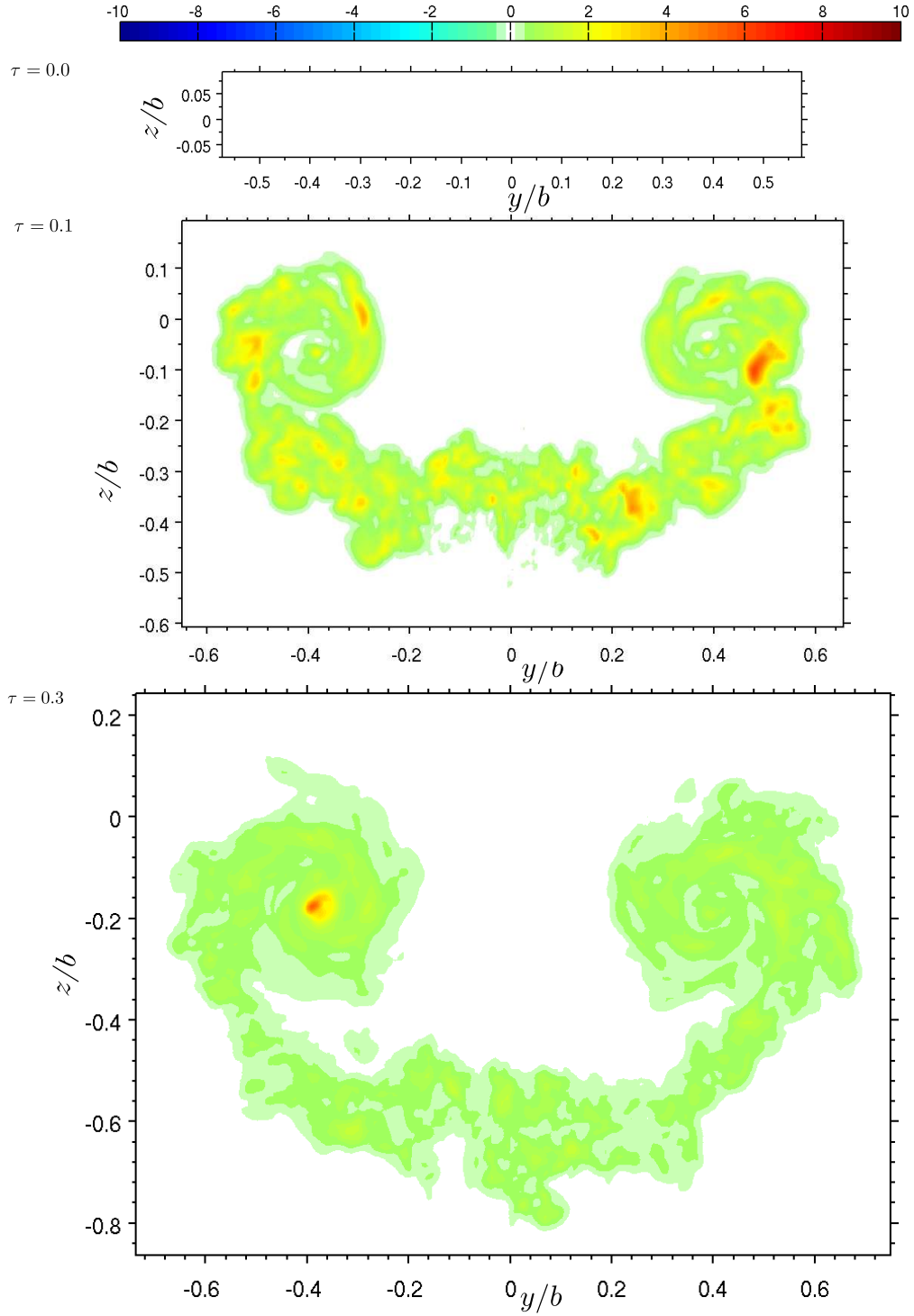


Figure 4.18: Time-developing simulation with velocity deficit: turbulent kinetic energy $\bar{k}/V_0^2$; from top to bottom, time $\tau = [0.0, 0.1, 0.3]$.

4.4.3 Characterization of the rolled-up vortex sheet including the friction- and induced-drag effects

In this section, we characterize the net-result after rollup of the space-developing vortex wake taking into account the friction drag. The circulation profile $\Gamma(r)$:

$$\Gamma(r) = \int_0^{2\pi} \int_0^r \omega_x(r', \theta) r' dr' d\theta$$

and the azimuthal velocity profile $u_\theta(r) = \Gamma(r)/(2\pi r)$ are defined. The “effective vortex core size, r_c , is the radius where the azimuthal velocity is maximum,

$$u_\theta(r_c) = \max_r (u_\theta(r)) = u_{\theta, max}.$$

The total vorticity of the half-plane, Γ^+ , and the vortex-centroid spacing, y_c , at a cross-plane x are also defined:

$$\Gamma^+(x) = \int_{-\infty}^{\infty} \int_0^{\infty} \omega_x(x, y, z) dy dz$$

and

$$y_c(x) = \frac{2}{\Gamma^+(x)} \int_{-\infty}^{\infty} \int_0^{\infty} y \omega_x(x, y, z) dy dz$$

respectively. The circulation profiles $\Gamma(r)$ and the azimuthal velocity profiles $u_\theta(r)$ for the stations $x/b = [2, 4, 6, 8]$ are presented in Fig. 4.19.(a) and Fig. 4.19.(b). The maximum tangential velocity slightly decreases by diffusion from $8.7V_0$ to $8.1V_0$. Fig. 4.20 and Fig. 4.21 show the tangential velocity profiles and the related circulation profiles for various fitted models at $x/b = 4.0$ and $x/b = 8.0$, where the rollup is almost completed. All profiles have been fitted on the tangential velocity profiles in the least-squared sense, without weight function. Table 4.1 provides the model fitted parameters. We observe that the one-length scale models (i.e. the Low Order Algebraic, the High Order Algebraic and the Gaussian) are not able to correctly capture either the circulation profile or the tangential velocity profile. Nevertheless, the best one-length scale model is the Low Order Algebraic one. The two-length scale models (i.e. the Jacquin model and the Proctor-Winckelmans model) are able to match properly both the maximum tangential velocity and the core radius size r_c . It has to

be stressed that the “global parameter” of the Jacquin model, which is usually assumed to be “universal” (for wake at high Reynolds number and with thin vortex sheet before rollup), was here modified to have a better fit. The parameter α has been taken as $\alpha = 0.5$. The exponent p in the Protor-Winckelmans model has been taken as $p = 3$, which is in the usual range $p = 3 - 4$. Finally, the Regularized Kaden-Winckelmans model fits better than all one-length scale models but doesn’t quite reach a high enough maximum tangential velocity. For a long time, it seems to be close to the Low Order Algebraic model.

Fig. 4.22.(a) presents the evolution of the core size, r_c . It grows from $r_c/b = 1.5\%$ to 5% at time $t \simeq 3$. One clearly observes the influence of the resolution that is twice coarsened from $\frac{x}{b} = 2.2$, i.e. $\tau = 0.075$. At this location, $r_c/b = 2.4\%$: the vortex core is then captured using only 6 points, which is still “acceptable” yet at the limit. The captured vortex-core is still fine enough for the results to be of interest. In particular, it still compares well to those of real aircraft system. Fig. 4.22.(b) shows the maximum velocity deficit inside the vortex core. It confirms that the swirl number is ~ 4.5 .

One now recalls some of the assumptions made to end up with the classical relation (originally developed using the inviscid theory) between the lift and the wake circulation at a station x . It basically consists in realizing a momentum balance around a control volume; as done in Chapter 2 using the formula by Noca et al. [62, 63]. This formula determines the force, $\mathbf{F}$, exerted by a fluid on a body knowing the velocity field (and its derivatives) on the surface of a control volume, S_C .

Starting from that formula, if one assumes that the flow is stationary and that the diffusive flux is negligible (as the Reynolds number is very high), this formula reads

$$\begin{aligned} \frac{\mathbf{F}}{\rho} = & \int_{S_C} \left(\frac{1}{2}(\mathbf{u} \cdot \mathbf{u})\mathbf{n} - (\mathbf{n} \cdot \mathbf{u})\mathbf{u} \right) dS \\ & - \frac{1}{2} \int_{S_C} (\mathbf{n} \cdot \mathbf{u})(\mathbf{x} \times \boldsymbol{\omega}) dS + \frac{1}{2} \int_{S_C} (\mathbf{n} \cdot \boldsymbol{\omega})(\mathbf{x} \times \mathbf{u}) dS \end{aligned} \quad (4.1)$$

we neglect the turbulent contributions. We also choose the control volume such that the surface perpendicular to the flight path is fixed at the downstream position x and all the other surfaces tend to infinity. Consequently, only one

term, lying in this perpendicular surface, contributes to the lift (neglecting the pressure contribution to the lift). Simplifying Eq.4.1 leads to

$$\begin{aligned}\frac{L}{\rho} &\simeq \int_{S(x)} u_x u_z dS \\ &= \int_{S(x)} (u_w + U_0) u_z dS \\ &= U_0 \int_{S(x)} u_z dS + \int_{S(x)} u_w u_z dS.\end{aligned}$$

The second integral of the right hand side (RHS) corresponds to the contribution to the lift of the velocity deficit due to friction drag. Integrating by part the first integral, leads to:

$$\frac{L}{\rho} \simeq -U_0 \int_{S(x)} y \frac{\partial u_z}{\partial y} dS + \int_{S(x)} u_w u_z dS.$$

Given that $\omega_x = \frac{\partial u_z}{\partial y} - \frac{\partial u_y}{\partial z}$:

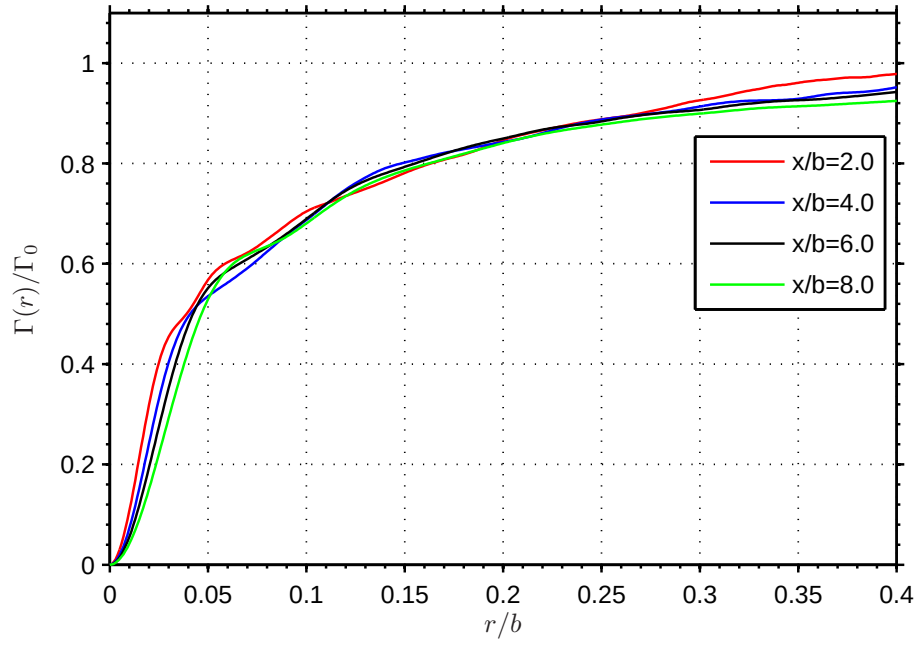
$$\frac{L}{\rho} \simeq U_0 \int_{S(x)} y \omega_x dS - U_0 \int_{S(x)} y \frac{\partial u_y}{\partial z} dS + \int_{S(x)} u_w u_z dS.$$

The second term vanishes as $\frac{\partial u_y}{\partial z}$ is left-right symmetric. Thus we finally obtain:

$$\begin{aligned}\frac{L}{\rho} &\simeq U_0 \int_{S(x)} y \omega_x dS + \int_{S(x)} u_w u_z dS \\ &= U_0 y_c \Gamma^+ + \int_{S(x)} u_w u_z dS.\end{aligned}$$

Fig. 4.22.(c) shows the evolution of those two terms. The former, $\frac{\Gamma^+}{\Gamma_0} \frac{y_c}{b}$, constitutes most of the total lift since $L/(\rho \Gamma_0 b U_0) = \pi/4 \simeq 0.785$. The latter $(\frac{1}{\Gamma_0 b U_0} \int u_w u_z dS)$ contributes to 0.5%. The turbulent terms of the momentum balance, $\int \overline{u'_x u'_z} dS$ is also verified to be negligible. It is worth mentioning that the way the forces (i.e. lift and drag) was imposed could drive the results. Given the imposed lift force was stationary, the averaged field remains a good image of a time-averaged quantity. Finally, the total circulation of the half-plane, Γ^+/Γ_0 , (defined above) is also presented in Fig. 4.22.(c) and is very well conserved.

(a)



(b)

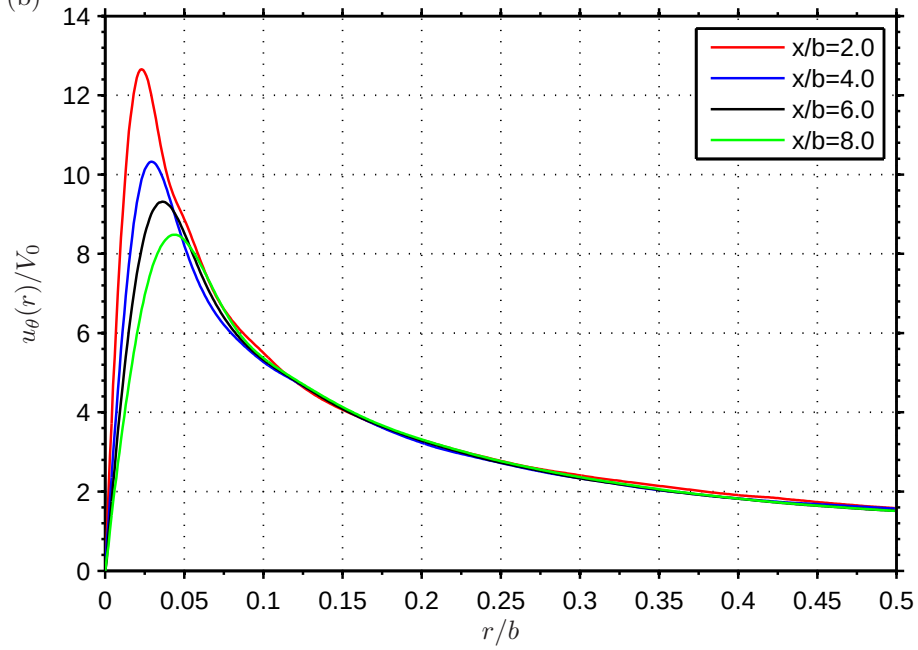
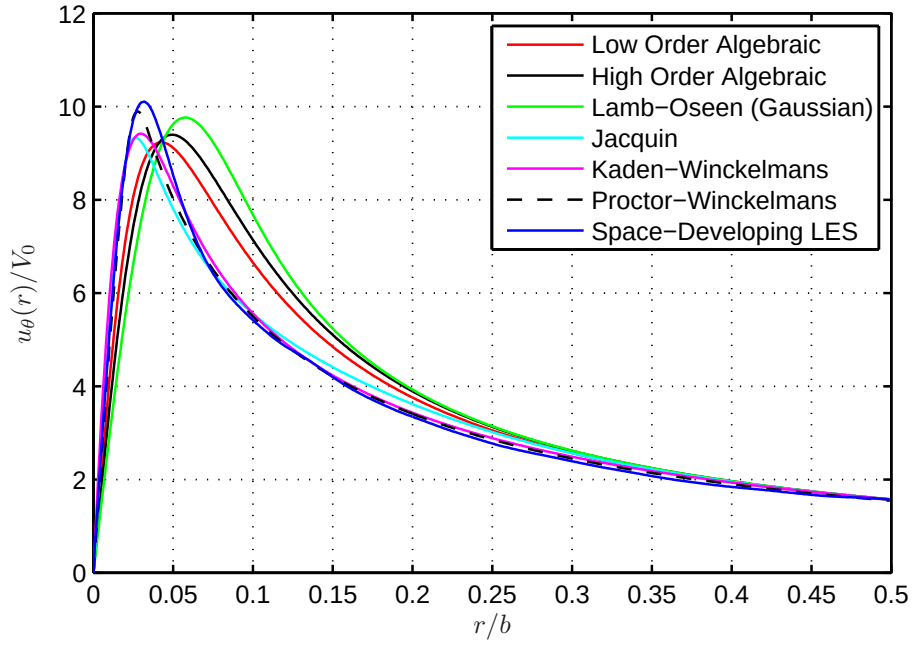


Figure 4.19: Circulation profiles (a) and tangential velocity profiles (b) at various distances in the wake.

(a)



(b)

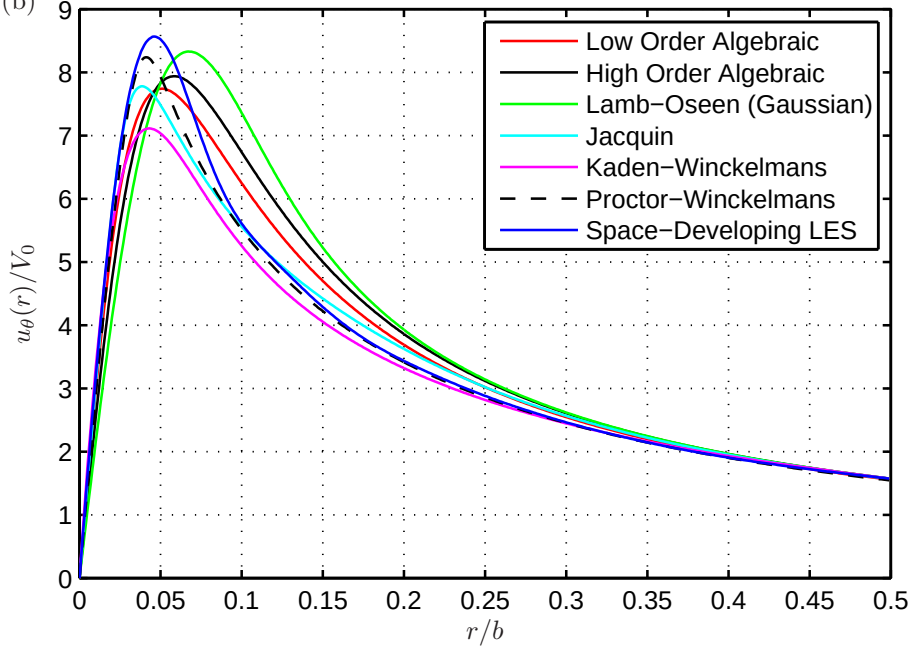


Figure 4.20: Tangential velocity profiles for various vortex models (a) at $x/b = 4.0$ ($\tau = 0.13$) and (b) at $x/b = 8.0$ ($\tau = 0.26$).

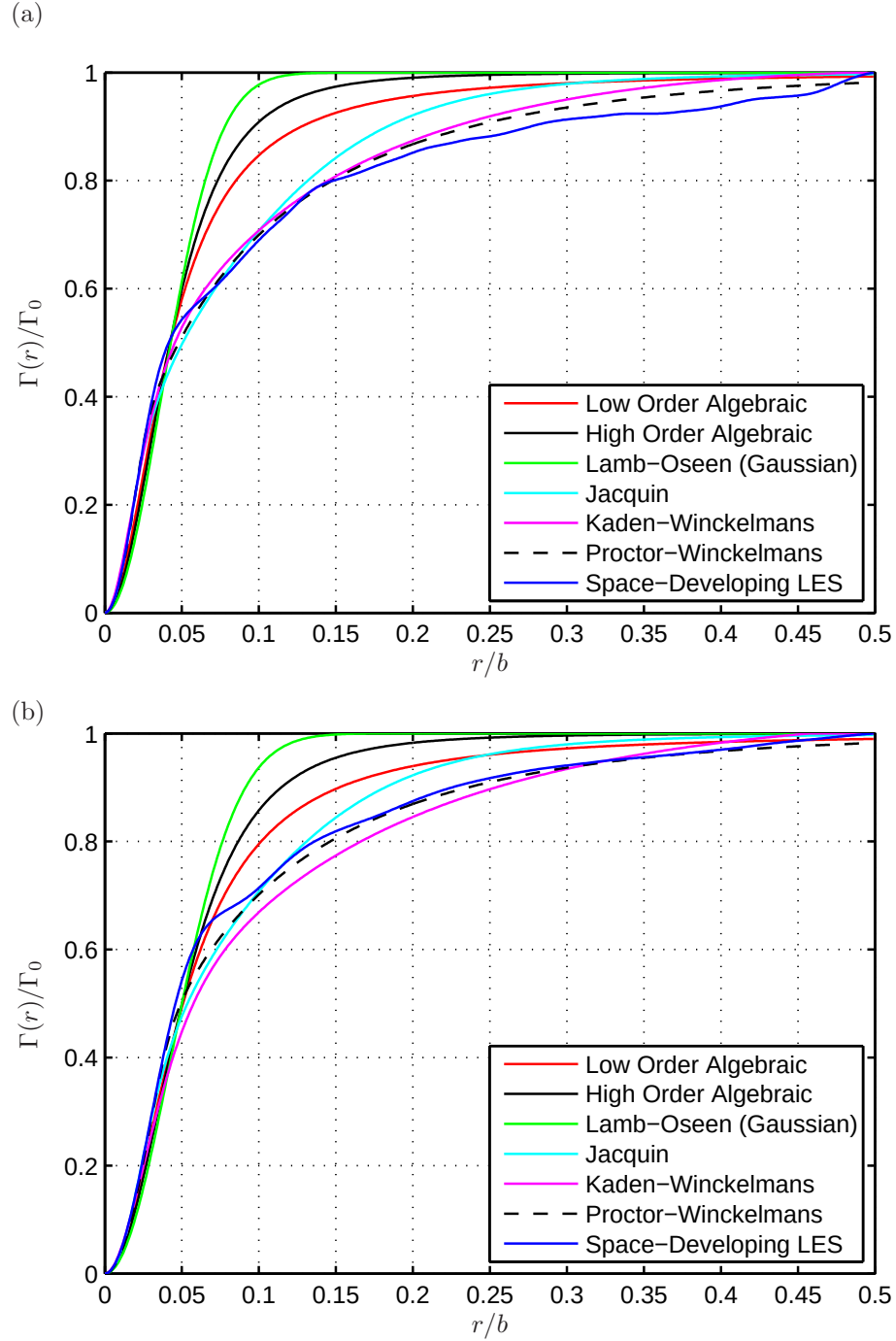


Figure 4.21: Circulation profiles for various vortex models (a) at $x/b = 4.0$ ($\tau = 0.13$) and (b) at $x/b = 8.0$ ($\tau = 0.26$).

Table 4.1: List of usual vortex models. The values of the model parameter(s) fitted on the space-developing rollup simulation at $x/b = 4.0$ and $x/b = 8.0$ are provided

	$\Gamma(r)/\Gamma_0$	a_1/b	a_2/b	other
Low Order Algebraic	$\frac{(r/a_1)^2}{(r/a_1)^2+1}$	0.042	/	$r_c = a_1$
		0.048	/	
High Order Algebraic	$\frac{(r/a_1)^2((r/a_1)^2+2\gamma)}{((r/a_1)^2+\gamma)^2}$	0.048	/	$r_c = a_1$
		0.056	/	$\gamma = 1.781$
Gaussian	$1 - \exp(-\beta(r/a_1)^2)$	0.056	/	$r_c = a_1$
		0.064	/	$\beta = 1.256$
Jacquin	$\frac{(r/a_2)^2}{((a_1/a_2)^4+(r/a_2)^4)^{(1+\alpha)/4} (1+(r/a_2)^4)^{(1-\alpha)/4}}$	0.022	0.19	$\alpha = 0.5$
		0.075	0.17	
Proctor-Winckelmans	$1 - \exp\left(-\frac{(r/a_1)^2}{\left(1+\left(\frac{(r/a_1)^2}{(r/a_2)^{3/4}}\right)^p\right)^{1/p}}\right)$	0.037	0.072	$r_c \simeq (a_1^8/a_2^3)^{1/5}$
		0.049	0.065	$p = 3$
Regularized Kaden-Winckelmans	$\left(1 - \exp(-(r/a_1)^{3/2})\right) \frac{\alpha(r/a_2)^{1/2}}{1+(\alpha-1)(r/a_2)}$	0.020	0.46	$\alpha = 1.78$
		0.030	0.40	

4.5 Conclusion

This chapter presents the investigation of a velocity deficit effect (due to the friction drag) on the rollup of a thin vortex-sheet at very high Reynolds number ($Re = 10^6$). Large eddy simulations were carried out and three cases were investigated: two space-developing (S-D) simulations, with and without this velocity deficit, and one time-developing (T-D) simulation with velocity deficit. The first S-D simulation uses the multi-resolution ability developed and described in Chapter 3 while the other two simulations use a uniform resolution. A “in cruise condition” is here considered. This chapter is the complementary follow-up of a Far-WAKE technical report (Cottin et al. [19]).

First, it is worth noting that the refined mesh handling technique has been applied using large eddy simulation at high Reynolds number and using multiscale subgrid modeling.

Second, it has been shown that the velocity has a major influence on the resulting vortex-system. The flow is intrinsically unstable and rapidly becomes turbulent. A strong velocity deficit remains inside the vortex-core of order $-[1.5 \cdots 1.9]V_0$ (towards the wing).

The wake-rollup with velocity deficit using the T-D and the S-D simulations have been also compared. The global dynamics are similar. In the T-D case, the velocity deficit inside the vortex-core is higher, leading to the development of a helical instability.

Finally, the net result of the S-D wake rollup with velocity deficit has been characterized. Most of vortex models have been fitted for two station $x/b = [4, 8]$. It appears that the two-scales models (Proctor-Winckelmans and Jacquin) are both good candidates to model wake vortices. Indeed, two length scales are required to both fit the vortex-core size and the maximal azimuthal velocity.

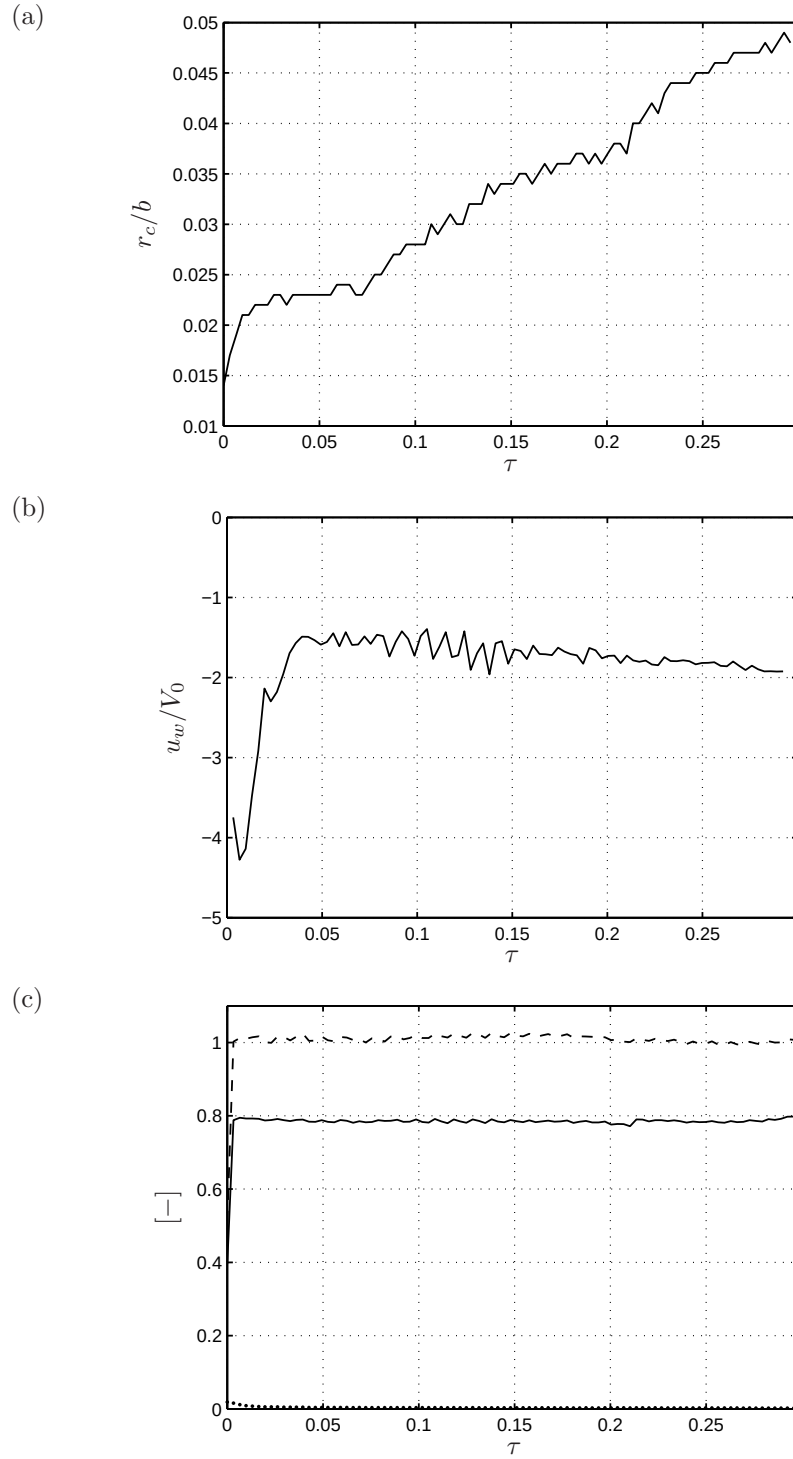


Figure 4.22: (a) Evolution of the vortex effective core size r_c . (b) Evolution of the maximum velocity deficit inside the vortex core and (c) Evolution of the linear-impulse $y_c \Gamma^+ / (\Gamma_0 b)$ (solid line), of the total circulation of the half-plane Γ^+ / Γ_0 (dashed line) and of the integral $\frac{1}{\Gamma_0 b U_0} \int u_w u_z dS$ (dotted line).

Chapter 5

Numerical investigations of end-effects associated with accelerated/decelerated wings : time-developing and space-developing simulations

This chapter was written within the FAR-Wake project as a technical report. European project AST4-CT-2005-012238, Report TR1.1.2-6, February, 2007. The authors are T. Lonfils, R. Cocle, G. Daeninck, C. Cottin and G. Winckelmans.

Abstract *In this report, we investigate the end-effect phenomenon occurring due to the variation of the circulation in a vortex wake. We first focus on comparing a numerical simulation to an experiment made by CNRS/IRPHE[57]. Two types of instabilities occur : an axisymmetric one produced by the circu-*

lation variation, and generating an axial velocity which can lead to a helical one, as it is presently the case, if this velocity is strong enough. We also compare some results to the theory derived for the Lamb-Oseen vortex. We then investigate a more realistic case of end-effects associated with an accelerated/decelerated wing. Analyzing the corresponding time-developing and space-developing (S-D) simulations, an explanation of the different behavior in the accelerated and in the decelerated cases is proposed.

5.1 Introduction

Due to the aircraft lift, wake vortices are inevitably generated. They can be hazardous for a follower aircraft and are then an important issue to ensure a high security level in air traffic. The purpose of this work is to characterize the typical vortex instabilities occurring when aircraft accelerate/decelerate, which could lead to the disappearance of the vortex coherence : this is why such instabilities are of interest. Theoretical and numerical investigations have already been carried out, but they remain academic as they are applied to the Lamb-Oseen vortex (see [40, 30, 59, 29, 39]). This phenomenon has been extensively investigated by Leibovich [48, 49]. See also the review done by Lucca-Negro and O'Doherty [53].

In the first part of this report, we perform a numerical simulation to be compared to an experiment also carried in the framework of FAR-Wake [57] in which the axial propagation of vortex perturbations was investigated.

Two other simulations are also carried out. The first one is time-developing, and thus assumes periodicity. The second one simulates the complete spatial evolution of the wake, which evolves using an inflow condition. The latter constitutes a better representation of the reality, but it requires much more computational resources. Both cases are analyzed and compared.

Experimental investigations of end-effect generated by a accelerated/decelerated wing have also been carried out using the UCL/TERM facilities ([27], [26]). The influence of the lift coefficient, the flight speed and the acceleration/deceleration intensity have also been studied. They confirm that the propagation speed is of the order the maximal tangential velocity. Fig. 5.1 shows a helical instability propagation for a deceleration case.

Because vortex lines have to be closed, the increase of the wake circulation is associated with the creation of spanwise vorticity component. The general description of the corresponding wake configuration is depicted in Fig. 5.2. The global dynamics are essentially dependent on the ratio of the net increase of circulation (i.e. $\Delta\Gamma = (\Gamma_1 - \Gamma_0)/\Gamma_1$) over the acceleration distance E/b_0 ([27]).

The so-called “end-effect” phenomenon (relative to a sudden acceleration/deceleration of the wing) corresponds to the particular case with $\Gamma_0 = 0$; but the term will here be also used to qualify the general case (i.e. $\Gamma_0 \neq 0$).

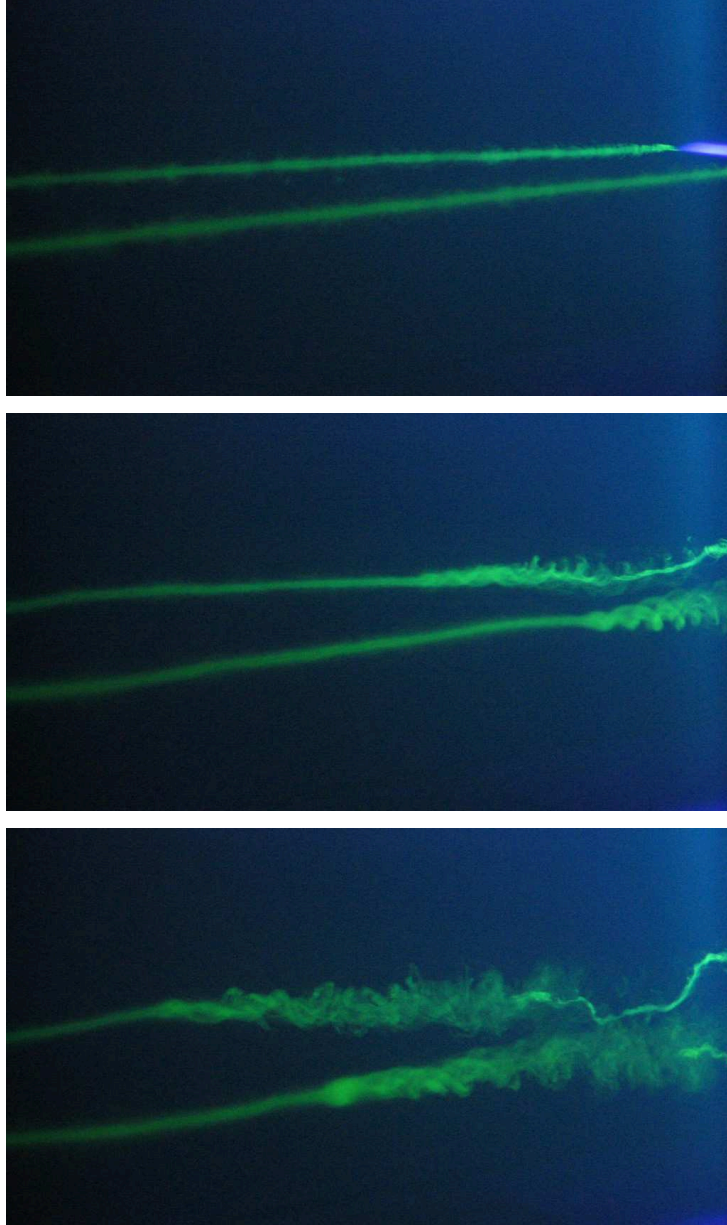


Figure 5.1: Dye visualization of instabilities generated by a decelerated wing. The fluorescein is injected at the wingtips (for more details, see [27]).

5.2 Numerical method

For the simulations, we use the Vortex-in-Cell method combined with the Parallel Fast Multipole Method (called VIC-PFM, for short). It is based on a combination of Lagrangian and finite difference methodologies, and it solves the vorticity form of the incompressible Navier-Stokes equations

$$\frac{D\boldsymbol{\omega}}{Dt} = \nabla \cdot (\mathbf{u}\boldsymbol{\omega}) + \nabla \cdot \boldsymbol{\mathcal{W}}, \quad (5.1)$$

in which $D/Dt = \partial/\partial t + (\mathbf{u} \cdot \nabla)$. The divergence of the diffusive vorticity flux tensor $\boldsymbol{\mathcal{W}}$ is written here

$$\nabla \cdot \boldsymbol{\mathcal{W}} = \nu \nabla^2 \boldsymbol{\omega} + \nabla \cdot \left(\nu_{\text{sgs}} \left(\nabla \boldsymbol{\omega}^s + (\nabla \boldsymbol{\omega}^s)^T \right) \right) \quad (5.2)$$

where ν_{sgs} is the effective subgrid-scale viscosity and $\boldsymbol{\omega}^s$ is the “small-scale” part of the LES field (since we here use the multiscale subgrid-scale approach).

The numerical solution of Equation (5.1) is sought in a two-fold approach. First, the discrete vorticity field is constructed from a sum of regularized vortex particles of strength α such that

$$\boldsymbol{\omega}_\sigma(\mathbf{x}, t) = \sum_p \alpha_p(t) \frac{1}{(\sqrt{2\pi} \sigma_p)^3} \exp \left(-\frac{1}{2} \frac{|\mathbf{x} - \mathbf{x}_p(t)|^2}{\sigma_p^2} \right), \quad (5.3)$$

in which $\alpha_p = \int \boldsymbol{\omega} d\mathbf{x} = \boldsymbol{\omega}_p h^3$, $\mathbf{x}_p$ is the particle position and σ_p its regularization parameter [16, 87]. The convective part, or lhs of Equation (5.1), is evaluated using a Lagrangian approach with $d\mathbf{x}_p/dt = \mathbf{u}(\mathbf{x}_p)$. For the second part, the time variation of the vortex particle strength, i.e., the rhs of Equation (5.1) that includes both the vortex stretching and dissipation terms, it is solved on a regular grid using 2nd order finite differences. Interpolation from the Lagrangian particles to the Eulerian grid is done using the M'_4 scheme [16, 87]. Once on the Eulerian grid, the stream function vector $\boldsymbol{\psi}$ is evaluated by solving the Poisson equation

$$\nabla^2 \boldsymbol{\psi} = -\boldsymbol{\omega}. \quad (5.4)$$

The velocity field, needed for convection and stretching, is then obtained by

evaluating $\mathbf{u} = \nabla \times \boldsymbol{\psi}$ using again 2nd order finite differences. The global time marching procedure is carried out using a 2nd order Leap-frog scheme for the convection and the 2nd order Adams-Bashford scheme for the stretching and dissipation terms. Finally, the divergence-free character of the vorticity vector field is insured by a proper reprojection of the vorticity field (which also requires solving a Poisson equation).

Particular to the present implementation of this procedure is the treatment of the boundary conditions. The idea is here to work on a domain that contains tightly the vorticity field and that can be decomposed in several subdomains on which the exact boundary conditions are obtained using a Parallel Fast Multipole (PFM) method. This amounts solving a 3-D Poisson equation without any iteration between the subdomains (e.g., no Schwarz iteration required): this is so because the PFM method has a global view of the entire vorticity field. So, at any given time step, the boundary conditions for $\boldsymbol{\psi}$ are obtained using the Green's function approach via the PFM method. This allows for a significant reduction of the computational cost for solving Equation (5.4)—as it retains proper open domain conditions and the capability to make parallel subdomain decomposition simulations— when comparing with more classical VIC methodologies [89, 13].

The LES modeling is here done using a multiscale subgrid-scale model: the model acts on the small scale part of LES field, $\boldsymbol{\omega}^s$, only. It is obtained from the complete LES field, $\boldsymbol{\omega}$, using a compact discrete filter (applied iteratively). We here use the Regularized Variational Multiscale (RVM) model [12]. The advantage of such model is that it preserves the inertial range while providing dissipation at the high wave numbers: it is only active during the complex phases of the flow, while remaining inactive during the phases when the vortices are coherent and well-resolved.

5.3 Numerical simulation of end-effects within vortex pairs : analysis and comparison with the IRPHE experiments.

Experimental setup

In this section, we present the numerical investigation carried out in order to compare with one of the CNRS/IRPHE experiments. In their experiments, the vortex is generated in a water tank using a plate which is rotated (for more details, see [57]). We here focused on the case at moderate Reynolds number, i.e. $Re_\Gamma = 3960$, in order to be able to afford an equivalent Direct Numerical Simulation (DNS).

Numerical setup & initial condition

We investigated the case of a vortex pair spaced by b_0 and linked at the extremities as depicted in Fig. 5.2 and Fig. 5.3.

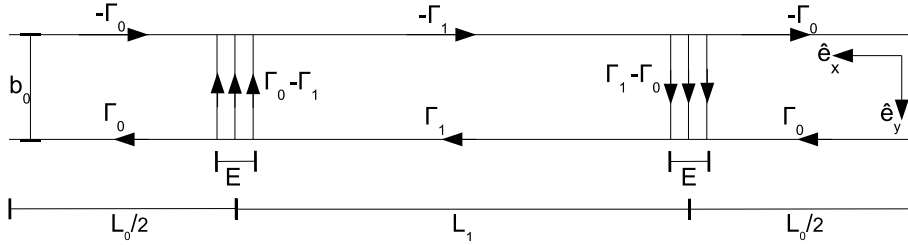


Figure 5.2: Schematic of the wake vortex lines corresponding to an accelerated and decelerated wing. Flow periodicity is here assumed over the length $L_0 + L_1$.

This case is also quite similar to the wake generated by a wing suddenly accelerated and decelerated (i.e. $\Gamma_0 = 0$), assuming that the vortex pair is immediately formed. The vortices are initialized using the Low Order Algebraic (LOA) vortex model :

$$\frac{\Gamma(r)}{\Gamma_1} = \frac{r^2}{r^2 + r_c^2} \quad u_\theta(r) = \frac{\Gamma(r)}{2\pi r}, \quad (5.5)$$

where r_c is the effective vortex core size (defined as the radius where the tangential velocity $u_\theta(r)$ is maximum). Here we use $r_c = 0.05 b_0$. We thus initially

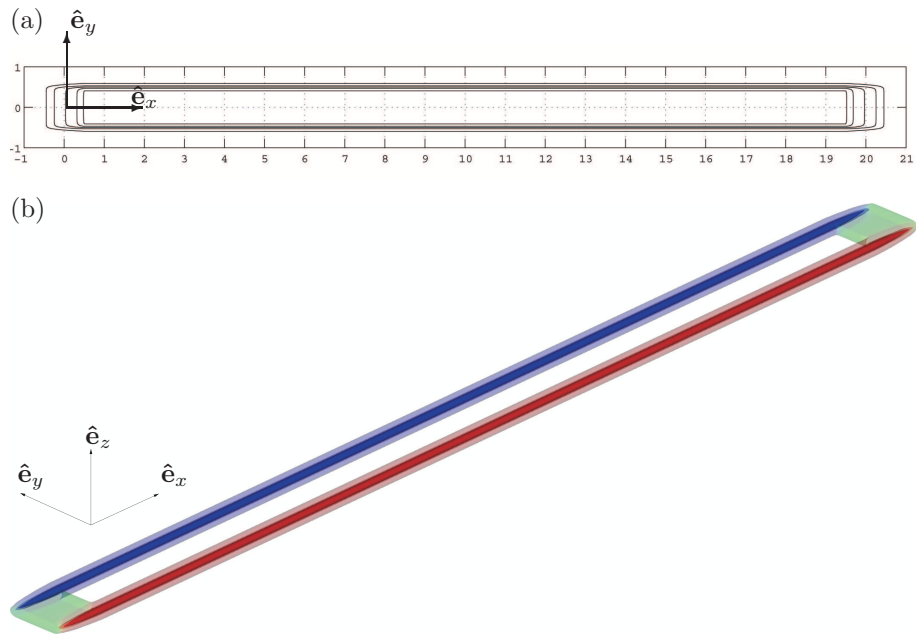


Figure 5.3: (a) 2-D cross section of the initial vorticity field (vortex lines) at $z/b_0 = 0.0$. (b) 3-D view of the vorticity norm isocontour $|\omega|b_0^2/\Gamma_1 = 1.0$ and $|\omega|b_0^2/\Gamma_1 = 5.0$.

have

$$u_{\theta, max} = \frac{\Gamma_1}{4\pi r_c} = \frac{1}{2} \left(\frac{\Gamma_1}{2\pi b_0} \right) / \left(\frac{r_c}{b_0} \right) = \frac{1}{2} V_0 / \left(\frac{r_c}{b_0} \right) = 10V_0, \quad (5.6)$$

where V_0 is the wake descent velocity. The Reynolds number, based on the circulation, is here chosen as $Re = \Gamma_1/\nu = 5000$. The distance of the reconnection E and the vortex length L are respectively equal to $1.0 b_0$ and $20 b_0$. The time step and the mesh size are fixed to $\Delta t \frac{V_0}{b_0} = 3.98 \cdot 10^{-4}$ and $h/b_0 = 1/64$. The simulation was carried out using 60 million grid points. It ran on 16 processors during 240 hours.

Analysis of the results

Since the problem is left/right symmetric and begin/end symmetric, we focus on only one vortex, without loss of generality.

Evolution of the vortex core size The instantaneous circulation profile $\Gamma(r)$ is defined by

$$\Gamma(r) = \int_0^{2\pi} \int_0^r \omega_x(r', \theta) r' dr' d\theta. \quad (5.7)$$

It is well known that the end-effect phenomenon much increases the effective vortex core. Moreover, in the current simulation, the Reynolds number is moderate and thus the increase of the vortex core by diffusion is also not negligible. Using the analogy of the Gaussian vortex diffusion, one characterizes the evolution of the LOA vortex core by diffusion as

$$r_c^2(t) \simeq r_c^2(0) + \alpha \nu t, \quad (5.8)$$

where α is found to be equal to roughly 7.9 (using the current simulation, we analyzed the evolution of the vortex core at the cross plane $x/b_0 = 10$ when it is not yet affected by the end-effect) as shown in Fig. 5.4.(a), instead of 4.0 for the Gaussian vortex. We can also estimate a priori the evolution of the maximum tangential velocity, assuming that the vortex remains close to the LOA model :

$$u_{\theta, max}^{LOA}(t) = \frac{\Gamma_1}{4\pi r_c(t)}. \quad (5.9)$$

Fig. 5.4.(b) shows the evolution of the measured maximum tangential velocity

as based on the circulation profile assuming a LOA model. It turns out that the maximum tangential velocity decreases significantly during the simulation, and that both cases are quite different. The case based on the LOA model underestimates the velocity during the whole simulation : this means that the vortex structure doesn't remain like the LOA one.

Fig. 5.5 presents the evolution of the vortex core size r_c for three cross planes : $x/b_0 = 5.0$, $x/b_0 = 7.5$ and $x/b_0 = 10.0$. Each shows the same global evolution. At first, the vortex core increases purely by diffusion and then suddenly magnifies when the perturbations cross the plane. We can then calculate the propagation speed of the instability as the ratio of the distance between two cross planes, to the time difference when the vortex core increases :

$$u_c = \Delta x / \Delta t.$$

Here we find $u_c/u_{\theta,max}(0) \simeq 2.5/2.8 = 0.89$.

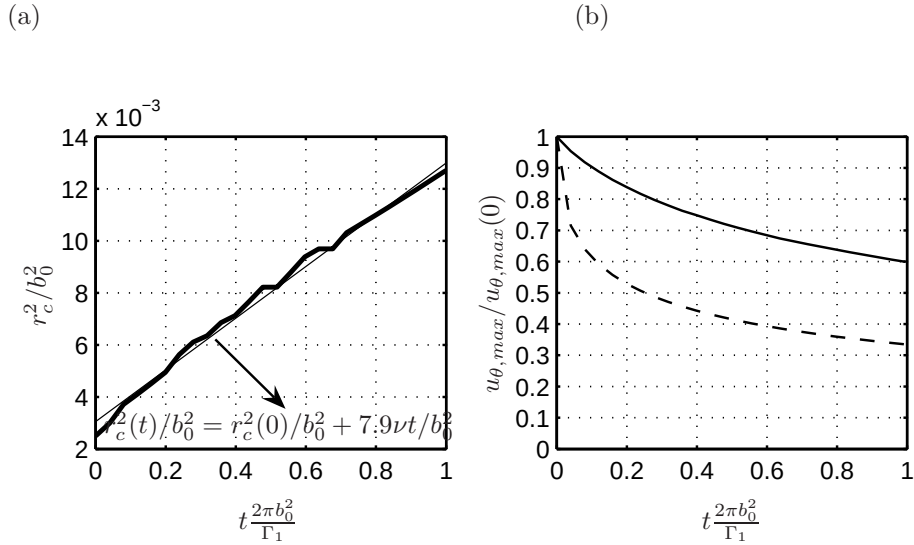


Figure 5.4: (a) Time evolution of the effective core size r_c (thick line) and its least-square fitted line (thin line). (b) Time evolution of the maximum tangential velocity $u_{\theta,max}(t)$ (solid line), and also that assuming that the LOA model remains valid (dashed line).

We also analyze the potential hazard of the vortex. The total circulation of the vortex Γ_1 is not a good estimation since it is conserved. We use Γ_{5-15} : the mean of the vortex circulation between 5 and 15 meter for an aircraft with

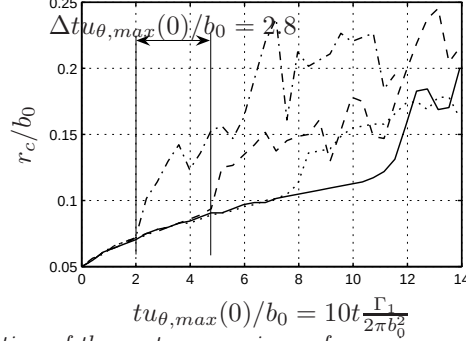


Figure 5.5: Evolution of the vortex core size r_c for various cross plane : $x/b_0 = 2.5$ (dash-dotted line), $x/b_0 = 5.0$ (dashed), $x/b_0 = 7.5$ (dotted line) and $x/b_0 = 10.0$ (solid line).

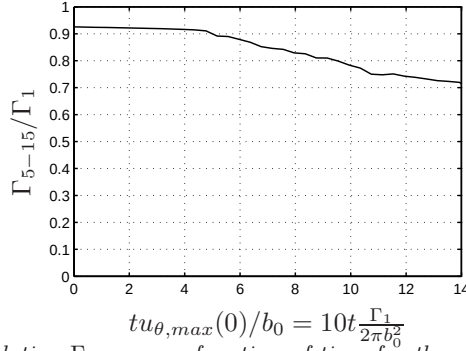


Figure 5.6: Circulation Γ_{5-15} as a function of time for the cross plane $x/b_0 = 5.0$.

a 60 meter wingspan. It reads

$$\Gamma_{5-15} = \frac{1}{b/6} \int_{b/12}^{b/4} \Gamma(r) dr, \quad (5.10)$$

where b is here taken as $b = (4/\pi)b_0$ (assuming a wing with elliptical loading). Fig. 5.6 shows its evolution : we clearly observe a net diminution at $tu_{\theta,max}/b_0 \simeq 4.8$, due to the passage of the perturbations.

Propagation of the perturbations There are several manners to measure the speed of the perturbation. The conclusion of the observations of Meunier in its experiment[57] is that there would be two main instabilities. The first one is supposed to be a “front wave” with a strong axial velocity within the vortex core (as shown in Fig. 5.7.a); the second would be a helical instability which strongly deforms the vortex (as shown in Fig. 5.7.b) and would propagate at a

lower speed .

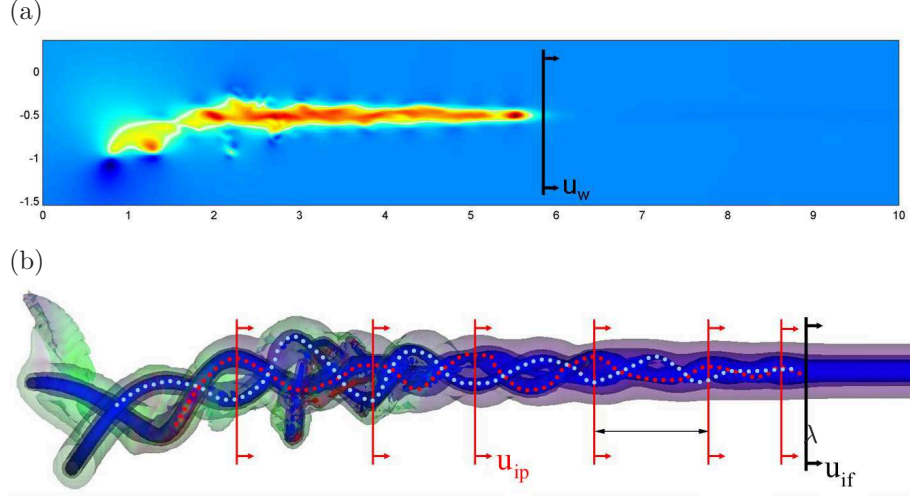


Figure 5.7: (a) Longitudinal section of the vortex at $y = b_0/2$. Axial component of the velocity field and front velocity the non-zero axial velocity (u_w). (b) Side view of the vorticity norm isocontours $||\omega||$ and description of the isophase point speed (u_{ip}) and its front speed (u_{if}).

Fig. 5.8 shows the time evolution of the front position of the non-zero axial velocity. For the nondimensionalization, we used the characteristic length scale b_0 and the maximum tangential velocity as the velocity scale. We compared the propagation speed using three different tangential velocity : (a) the time evolving maximum tangential velocity $u_{\theta,max}(t)$ and (b) the initial maximum tangential velocity $u_{\theta,max}(0)$. As foreseen, the propagation speed is of the order of the maximal tangential velocity in each case : 1.5 and 0.86 respectively. The viscous diffusion of the vortex core here plays a significant role in slowing down the propagation, as the tangential velocity also diminishes. We can also compare the propagation speed to the theoretical one predicting the propagation of the group velocity of an axisymmetric mode for a Gaussian vortex (see Fabre et al.[29] and Jacquin et al.[40]) :

$$u_w \simeq 0.63 \frac{\Gamma_1}{2\pi\sigma} \quad (5.11)$$

where σ is the Gaussian vortex core size. To compare to that, we here consider the Gaussian vortex which has the same long wavelength dynamics as the LOA vortex : this give $\sigma_{eq} = 1.34 r_c$. Here we find $u_w / (\frac{\Gamma_1}{2\pi\sigma_{eq}}) \simeq 0.57$ indeed close

to the theoretical value. In the experiment, the wave perturbation speed was above : roughly 0.9.

Another important information is the swirl parameter S which is the ratio between the maximum tangential velocity and the maximum axial velocity in the vortex. Here we find S being nearly equal to $0.9 \dots 1.2$. A value included in the interval $[0, 1.5]$ means that the helical Kelvin mode of the vortex is unstable.

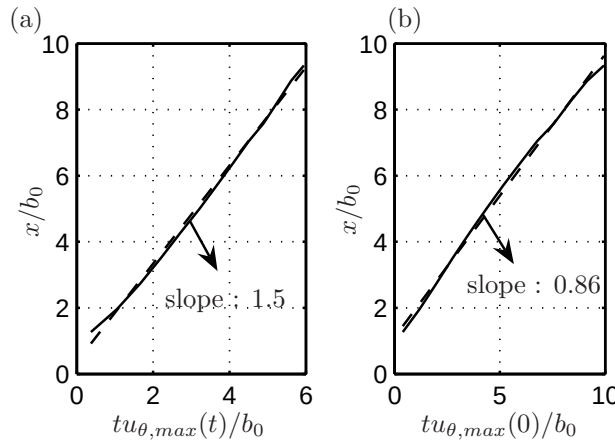


Figure 5.8: Position of the non-zero axial velocity front (solid) and its least-squared fitted line (dashed) for various time scaling based on : (a) the time evolving maximum tangential velocity $u_{\theta,max}(t)$, (b) the initial maximum tangential velocity $u_{\theta,max}(0)$.

We then analyze the propagation of this helical perturbation. The time evolution of the isophase points (as described in Fig. 5.7.b with the red lines) and the position of their front are presented in Fig. 5.9. We find a group velocity equal to $u_{if}/u_{\theta,max}(0) = 0.85$. This velocity is the same as the wave speed u_w (which creates the axial velocity). The reason could be the fact that the axial velocity generated by the wave is strong enough to permit the helical instability to immediately occur. This is the main difference with the experimental results for which the wave speed was twice the speed of the isophase front. Meunier suggests that the speed of the isophase point compares with the phase velocity of the helical Kelvin mode ($m = 1$). We find in our case a perturbation wavelength of $\lambda/r_c = 16$ (taken where the instability is inside the vortex core) and thus a wavenumber $kr_c = 0.39$. The theoretical phase

velocity of the helical mode of a Lamb-Oseen vortex with this wavenumber is

$$0.33 \frac{\Gamma_1}{2\pi\sigma}. \quad (5.12)$$

In our case we find $u_{ip} \simeq 0.23 \Gamma_1 / (2\pi\sigma_{eq})$.

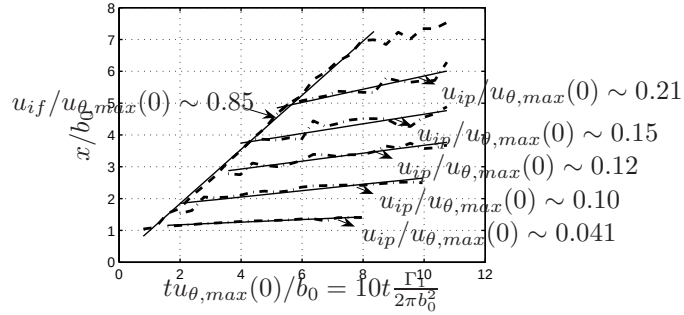


Figure 5.9: Evolution of the deformation front (dash) and of the isophase points (dash-dot). The solid lines correspond to the least-squared fitted lines.

Visualization of the perturbation We define the vorticity perturbation field as

$$\omega = \omega^{(0)} + \omega', \quad (5.13)$$

where $\omega^{(0)}$ is the “base” field (i.e. the vorticity field that would exist without the perturbation). Here, $\omega^{(0)}$ is taken as the field at the center slice, i.e. at $x/b_0 = 10$. ω' is thus the vorticity perturbation field. Isocontours of ω'_x for two different times are presented in Fig. 5.10.

Fig. 5.11 shows the comparison between the numerical simulation and the experiment for the axial vorticity component at a fixed slice for various times. The global dynamics is roughly the same as in the experiment. The first phase consists in the simple laminar diffusion of the vortex. The effective vortex core then grows suddenly when the wave crosses the slice, with loss of coherence after that.

3-D view of vorticity norm isocontours are also presented in Fig. 5.12, clearly slowing the helical instabilities.

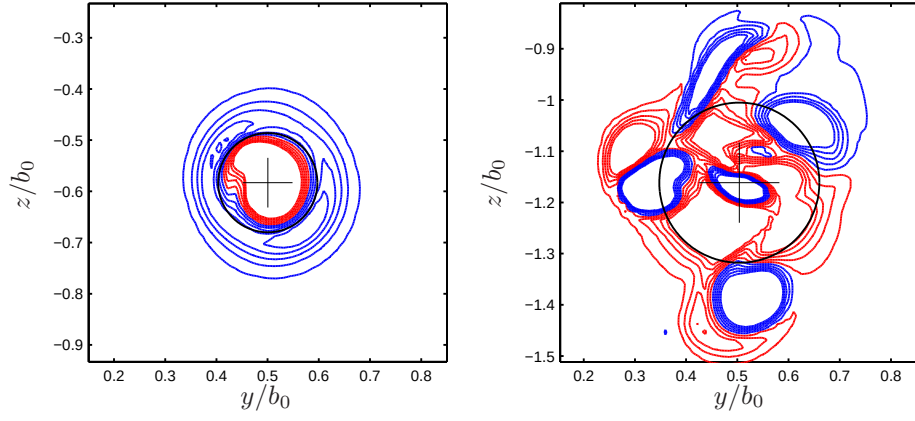


Figure 5.10: Isocontours of the vorticity perturbation axial component $\omega'_x b_0^2 / \Gamma_1$ at $x/b_0 = 5.0$ at $tu_{\theta, max}/b_0 = 6.0$ (left) and 12 (right) : ten equally spaced levels from -4.5 to 4.5 ; positive levels are in red and negative levels are in blue. The vortex center and the effective core radius r_c are also shown.

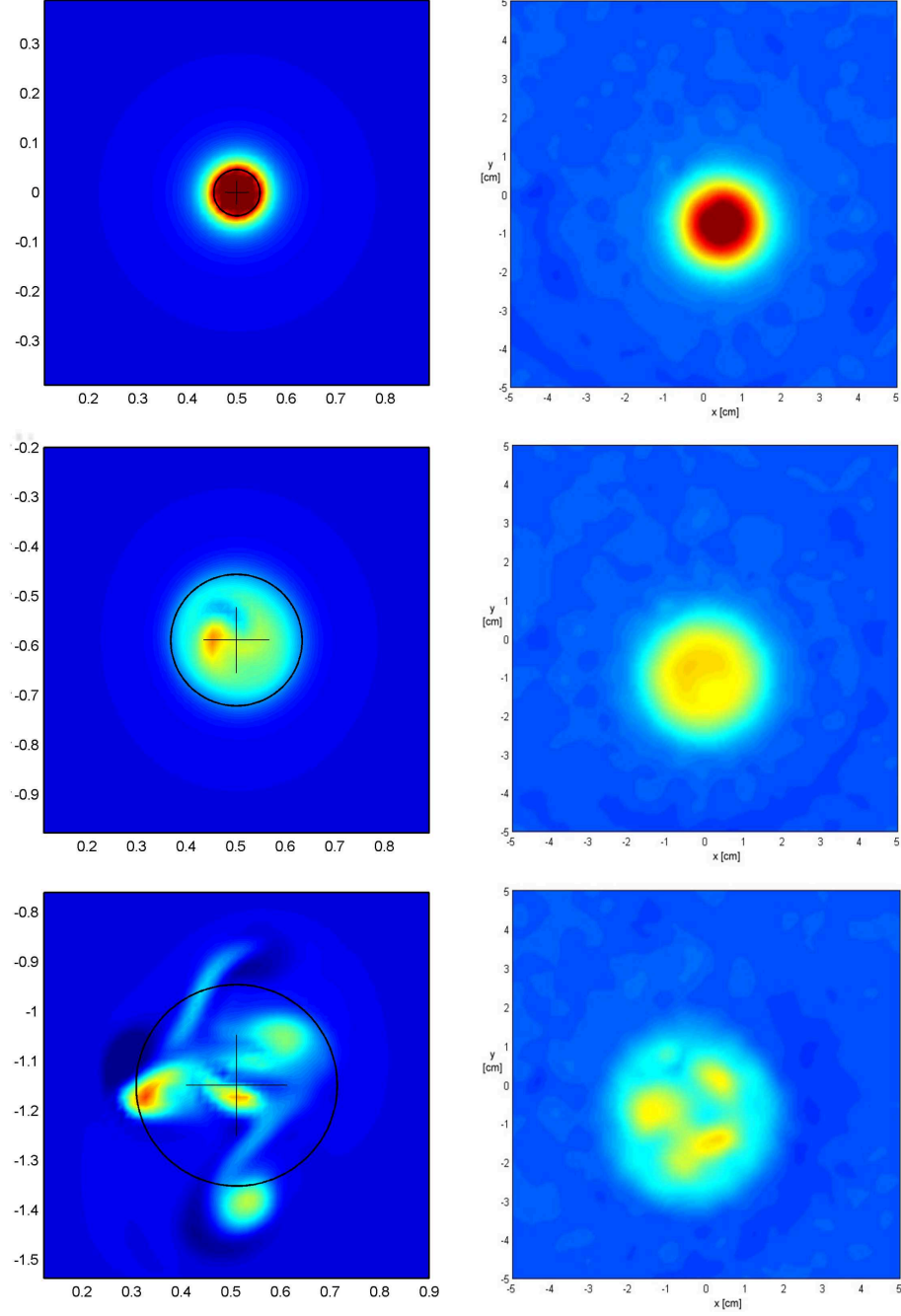
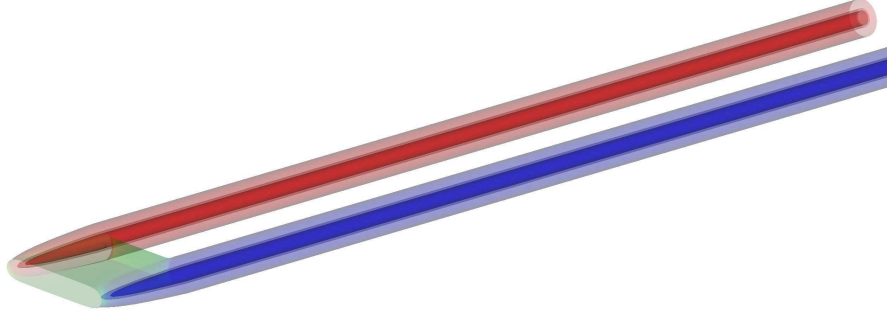
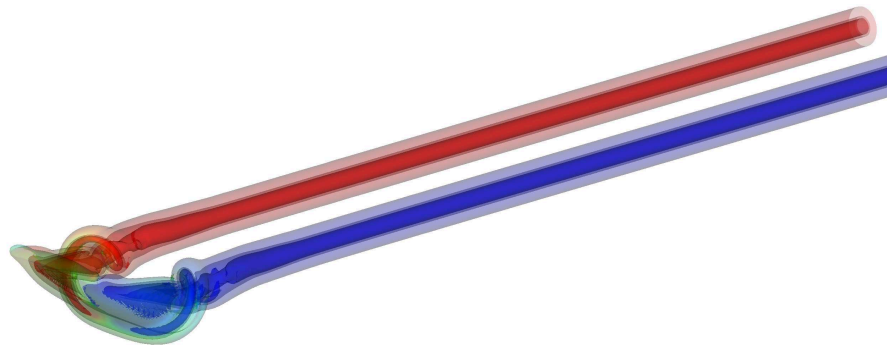


Figure 5.11: Visualization of the vorticity field axial component for the cross plane $x/b_0 = 5.0$. Numerical simulation (left) and IRPHE experiment (right). The size of the visualization boxes are $15.5 r_c$ and $11.1 r_c$ respectively. The time between two plots is $\Delta t u_{\theta, \max} / r_c \simeq 24$. For the numerical simulation, a circle using a radius $r_c(t)$ centered on the vorticity centroid is also plotted.

$$tu_{\theta, \max}(0)/b_0 = 0.0 \left(t \frac{\Gamma_1}{2\pi b_0^2} = 0 \right)$$



$$tu_{\theta, \max}(0)/b_0 = 1.6 \left(t \frac{\Gamma_1}{2\pi b_0^2} = 0.16 \right)$$



$$tu_{\theta, \max}(0)/b_0 = 3.2 \left(t \frac{\Gamma_1}{2\pi b_0^2} = 0.32 \right)$$

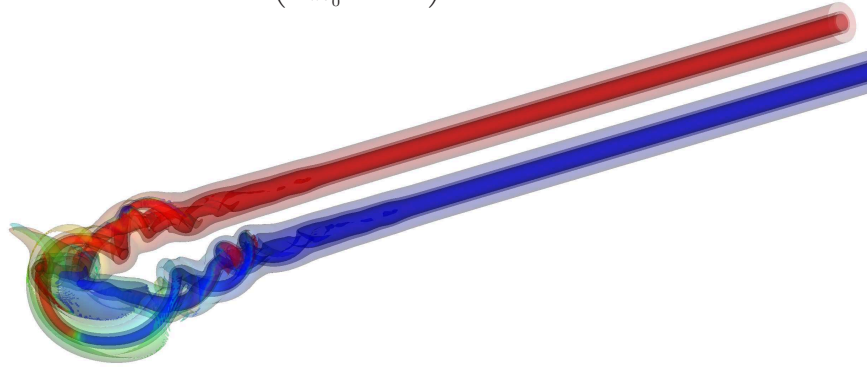
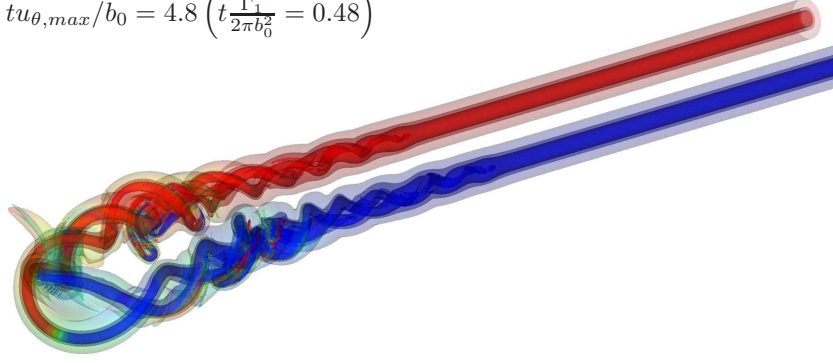
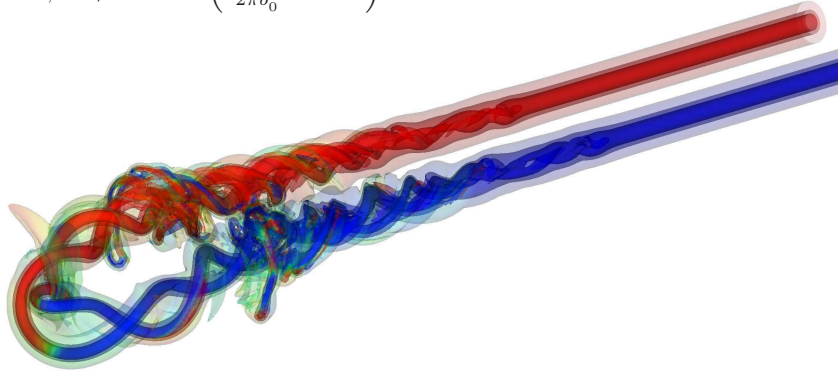


Figure 5.12: Visualization of the vorticity norm isosurfaces for $\|\omega\|b_0^2/\Gamma_1 = 1.0$ (low opacity), 5.0 (medium opacity) and 10 (high opacity), colored with the axial vorticity component.

$$tu_{\theta, \max}/b_0 = 4.8 \left(t \frac{\Gamma_1}{2\pi b_0^2} = 0.48 \right)$$



$$tu_{\theta, \max}/b_0 = 6.4 \left(t \frac{\Gamma_1}{2\pi b_0^2} = 0.64 \right)$$



$$tu_{\theta, \max}/b_0 = 8.0 \left(t \frac{\Gamma_1}{2\pi b_0^2} = 0.080 \right)$$

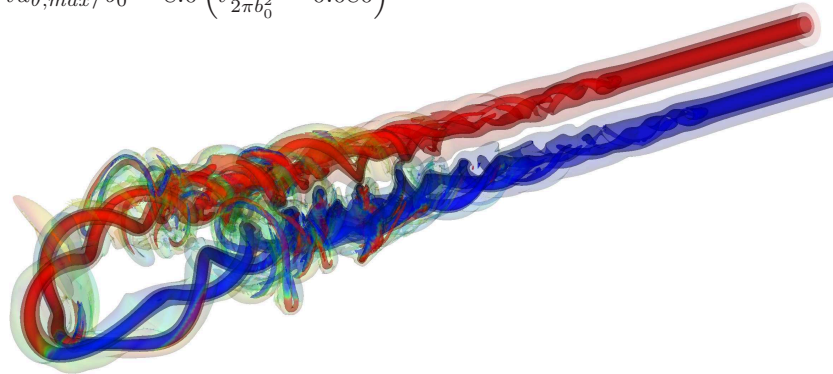
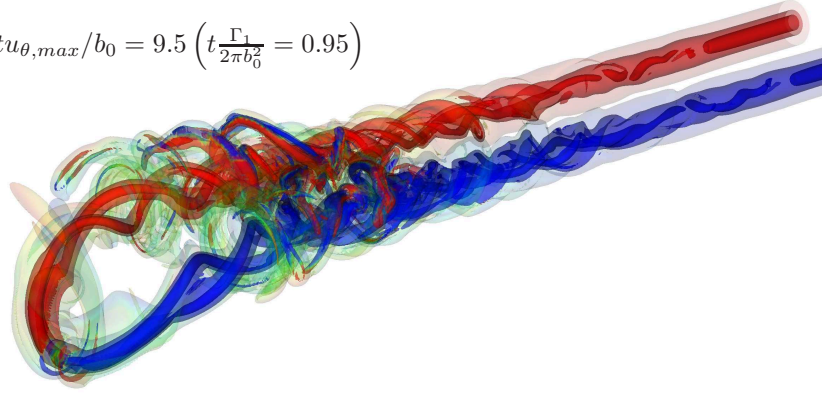
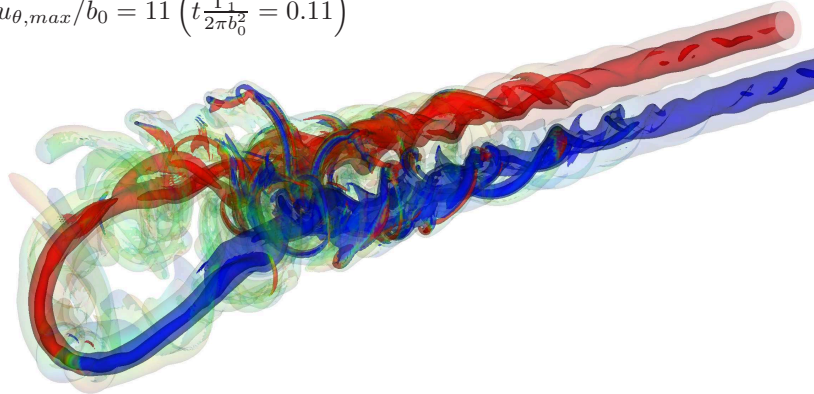


Figure 5.12: [Cont'd]

$$tu_{\theta,max}/b_0 = 9.5 \left(t \frac{\Gamma_1}{2\pi b_0^2} = 0.95 \right)$$



$$tu_{\theta,max}/b_0 = 11 \left(t \frac{\Gamma_1}{2\pi b_0^2} = 0.11 \right)$$



$$tu_{\theta,max}/b_0 = 13 \left(t \frac{\Gamma_1}{2\pi b_0^2} = 0.13 \right)$$

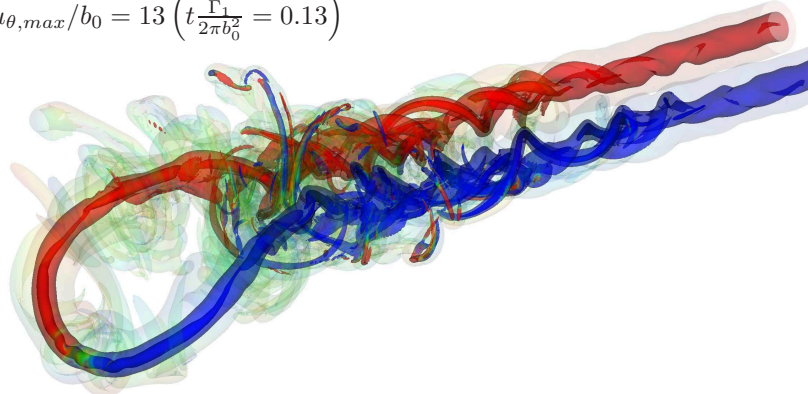


Figure 5.12: [Cont'd]

5.4 Time-developing and space-developing simulations of an accelerated/decelerated wing.

We now investigate the phenomenon of the acceleration/deceleration of a wing. The behavior of the flow over the wing will not be captured by the simulations but modeled using the lifting line approach. This theory (i) neglects the effects of the boundary layers and (ii) “replaces” the wing by a vortex line which has a certain circulation profile $\Gamma(y)$ along the spanwise direction. The lift profile and the circulation profile are related by $l(y) = \rho U \Gamma(y)$ (where U is the flight speed and ρ the fluid density). Since the vorticity field is divergence free, a vortex sheet of circulation per unit length $\gamma(y)$ has to be shed such that

$$\gamma(y) = -\frac{d\Gamma}{dy}(y) \quad (5.14)$$

We can then express the total lift of the wing as

$$L = \int_{-b/2}^{b/2} l(y) dy = \rho U \int_{-b/2}^{b/2} \Gamma(y) dy, \quad (5.15)$$

We define Γ_0 and b_0 , respectively the total circulation of the half plane and the spacing between the vorticity centroids (and thus also the vortex spacing after rollup). Then

$$\int_{-b/2}^{b/2} \Gamma(y) dy = \Gamma_0 b_0 \quad (5.16)$$

and

$$L = \rho U \Gamma_0 b_0. \quad (5.17)$$

Let's now consider an elliptic wing which has the following characteristics :

- wingspan b
- Surface S
- Aspect ratio $A_r \triangleq b^2/S$

The lift can also be related to the aerodynamic characteristic of the wing $C_L \triangleq L/(\frac{1}{2}\rho U^2 S)$. We can thus express the characteristics of the wake (Γ_0 and b_0)

as a function of the wing characteristics :

$$\Gamma_0 b_0 = \frac{1}{2} U S C_L = \frac{1}{2} U b^2 \frac{C_L}{A_r} \quad (5.18)$$

Here we use

$$C_L = 1.5, \quad A_R = 7.5, \quad \text{thus } \frac{C_L}{A_R} = 0.20, \quad Re = \Gamma_0 / \nu = 10^4.$$

The generated vortex sheet rolls up and eventually forms two counter-rotating wake vortices. In the previous section the initial condition was two vortex tubes reconnected at the extremities. We don't do it anymore. Indeed, we have to compare the relative characteristic time of each phenomenon to be sure that the time/space-evolution of the wake is properly investigated. The characteristic time of the vortex wake $t_0 = b_0/V_0$ has to be compared to the characteristic time of the perturbation propagation $t_p = b_0/V_p$. Therefore,

$$\begin{aligned} t_0/t_p &= \frac{V_p}{V_0} \\ &\simeq \frac{u_{\theta, max}}{V_0} \end{aligned}$$

where V_p is the propagation speed of the perturbation which is of the order of the maximum tangential velocity $u_{\theta, max}$ of the vortex; a typical value is $u_{\theta, max}/V_0 \simeq 10$. It means that we have to take into account the phenomenon of rollup.

Moreover, the third phenomenon is the flight itself which has a characteristic time $t_f = \frac{b_0}{U}$. The ratio of the latter characteristic t_f and t_p is

$$\begin{aligned} t_f/t_p &= \frac{V_p}{U} = \frac{V_p}{V_0} \frac{V_0}{U} \\ &\simeq \frac{u_{\theta, max}}{V_0} \frac{4}{\pi^3} \frac{C_L}{A_r} \\ &\sim 0.2 \dots 0.3 \end{aligned}$$

It means that, a priori, the end-effect of the vortex will clearly act in the wake region and won't be affected by the space-developing evolution of the wake. However, this "global" analysis neglected the influence of an axial velocity in the vortex core region due to the 3-D space-developing rollup (see [59] and

[30]). Hence, we also investigate the space-developing case.

So, we propose to investigate and to compare the two cases : (i) the space-developing (S-D) simulation and (ii) the time-developing(T-D) simulation. Fig. 5.13 presents the comparison between those two concepts.

- (i) The S-D simulation consists in three phases. We first simulate the complete space evolving rollup of the vortex sheet shed by a wing flying at a constant speed U_0 . This is the initial condition. During the second phase, the wing accelerates to the speed U_1 . And, finally at the time of the third phase, the wing flies at the constant speed U_1 . The instability evolves timely and is also convected in the wake moving towards the higher circulation region. Since there are an inflow and an outflow, the period of validity, and thus of analysis, is limited by the time needed to convect the instability out of the simulation domain.
- (ii) The T-D simulation assumes that the whole space evolution of the wake, including the three phases, is initially generated. We then use periodic boundary conditions in the flight direction. The initial condition is thus a vortex sheet over a distance L_0 (corresponding to a wing flying at a speed U_0), an acceleration to the speed U_1 over a distance E , and another piece of vortex sheet over the distance L_1 corresponding to the one of the flight speed U_1 . To match with the periodic boundary conditions, we also add a deceleration phase (also over distance a E).

The vortex sheet generated by the wing has to be regularized in order to produce a regular vorticity field to be provided to the VIC-PFM code. For that, we use convolution with a Gaussian :

$$\omega_x(y, z) = \frac{1}{\pi\sigma^2} \int_{-b/2}^{b/2} \exp\left(-\frac{(y-y')^2 + z^2}{\sigma^2}\right) \gamma(y') dy', \quad (5.19)$$

where σ is the regularization parameter of the vortex sheet. For an elliptic wing, the span loading (i.e. circulation profile) also has an elliptic profile

$$\Gamma(y) = \underbrace{\frac{2}{\pi} \frac{U C_L b}{A_r}}_{\Gamma_0} \left(1 - \left(y/(b/2)\right)^2\right)^{1/2} \quad (5.20)$$

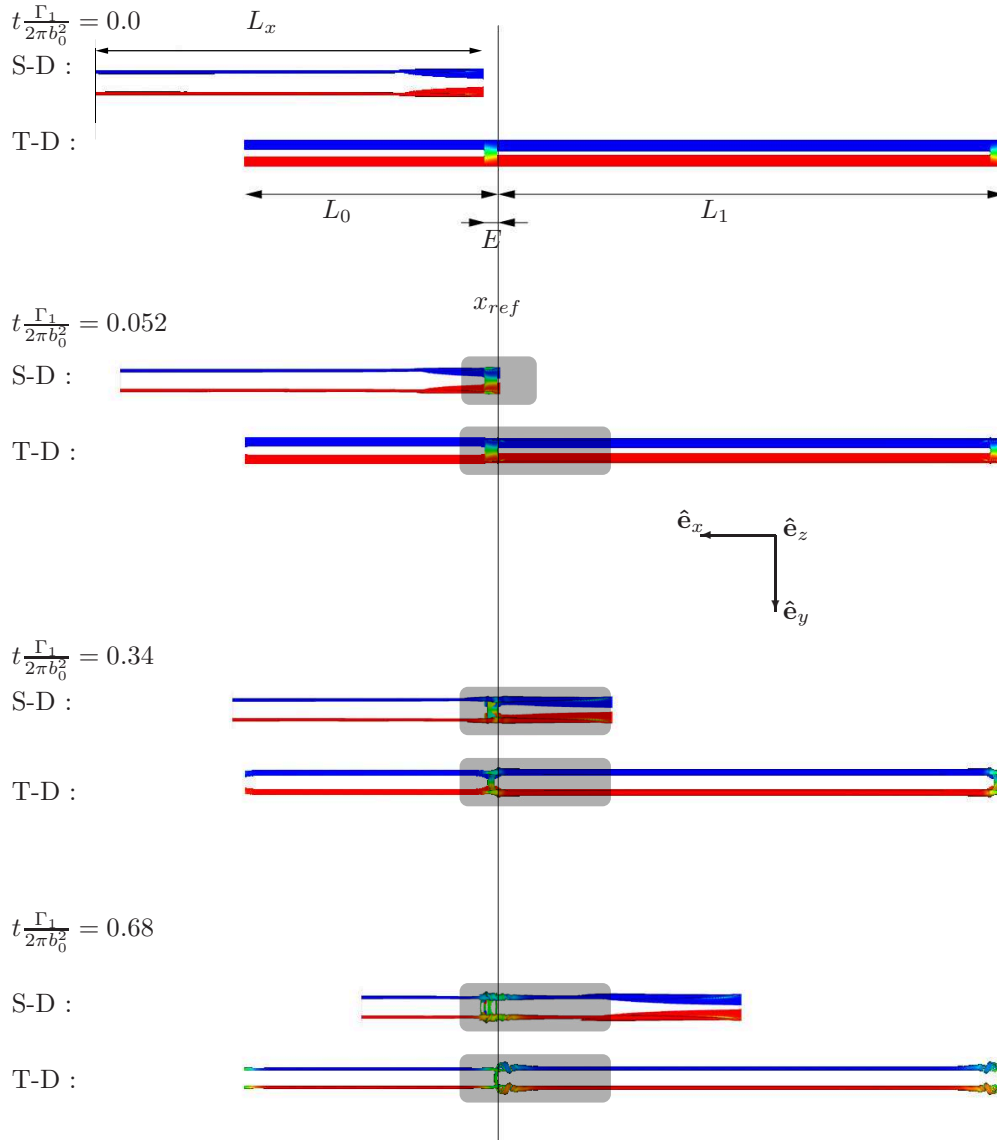


Figure 5.13: Comparison of time-developing (T-D) and space-developing (S-D) simulations: top view of the vorticity field at various times. The zone in gray corresponds to an example of zoom boxes, moving in the S-D case and fixed in the T-D case, used for the 3-D visualizations in Fig. 5.22, 5.23 and Fig. 5.24.

Notice that $\gamma(y)$ becomes infinite at $y = \pm b/2$, but not $\Gamma(y)$. To avoid numerical problems, one rewrites Eq.5.19 using integration by part:

$$\omega_x(y, z) = \frac{2}{\pi\sigma^4} \int_{-b/2}^{b/2} \exp\left(-\frac{(y-y')^2 + z^2}{\sigma^2}\right) (y-y')\Gamma(y')dy'. \quad (5.21)$$

Numerical setup & initial condition for the space-developing simulation

We investigate the S-D behaviour of the wing acceleration/deceleration with a constant lift coefficient C_L . It means that the lift increases/decreases: for instance, it is usually the case for an experiment in a towing tank. The velocity of the wing will follow the expression:

$$U(t) = U_0 + a \Delta t_a g\left(\frac{t-t_a}{\Delta t_a}\right) = U_0 + \Delta U g\left(\frac{t-t_a}{\Delta t_a}\right), \quad (5.22)$$

where U_0 , a , t_a , Δt_a and $\Delta U = U_1 - U_0$ are respectively the wing speed before the acceleration, the acceleration itself, the time when the acceleration starts, the duration of the acceleration and the total velocity variation. The function $g(\zeta)$ is defined as followed :

$$g(\zeta) = \begin{cases} 0 & \text{if } \zeta \leq 0 \\ \zeta & \text{if } 0 < \zeta \leq 1 \\ 1 & \text{if } \zeta > 1 \end{cases} \quad (5.23)$$

The wing motion is composed of three phases. The first phase consists in the flight at a constant velocity U_0 , then followed by an acceleration or an deceleration phase achieved in a time Δt_a and the third one is the flight at a higher/lower velocity U_1 .

The wake circulation increases following $U(t)$:

$$\Gamma(y, t) = \Gamma(y) \left(1 + \frac{\Delta U}{U_0} g\left(\frac{t-t_a}{\Delta t_a}\right)\right). \quad (5.24)$$

Here we use an acceleration since the instability propagates from the low circulation Γ_0 region to the higher circulation Γ_1 region: we can thus observe the end-effect longer due to the inflow/outflow.

The speed of the wing varies from U_0 to U_1 over a distance E . We thus have:

$$\Delta t_a = \frac{2E}{2U_0 + \Delta U} \quad (5.25)$$

The S-D simulation considered here is, at first, a complete rollup of the vortex sheet shed by a wing having an elliptic span loading and flying at velocity U_0 . The wing then accelerates to the velocity U_1 over a distance $E = b/2$. The component of the vorticity field in the flight direction ($\hat{\mathbf{e}}_x$, the axial direction) also increases :

$$\omega_x(y, z, t) = \omega_x(y, z) \left(1 + \frac{\Delta U}{U_0} g\left(\frac{t - T_a}{\Delta T_a}\right) \right).$$

A component of the vorticity field in the spanwise direction ($\hat{\mathbf{e}}_y$) is then also produced corresponding to the temporal variation of the circulation. Starting using the total derivative of $\omega_y(y, z, t)$, we can write

$$\omega_y(y, z, t) = \int \frac{\partial \omega_y}{\partial y} dy + \int \frac{\partial \omega_y}{\partial z} dz + \int \frac{\partial \omega_y}{\partial t} dt. \quad (5.26)$$

Since the vorticity field is divergence free and since the vertical vorticity component is zero, this gives :

$$\omega_y(y, z, t) = - \int \frac{\partial \omega_x}{\partial x} dy + \int \frac{\partial \omega_y}{\partial z} dz + \int \frac{\partial \omega_y}{\partial t} dt. \quad (5.27)$$

One further assumes that the spanwise component of the vorticity due to the acceleration is uniform in the vertical direction ($\frac{\partial \omega_y}{\partial z} = 0$) and over the time step dt ($\frac{\partial \omega_y}{\partial t} = 0$). One then obtains:

$$\begin{aligned} \omega_y(y, z, t) &\simeq - \int \frac{\partial \omega_x}{\partial x} dy \\ &= - \frac{dt}{dx} \int \frac{\partial \omega_x}{\partial t} dy \\ &= - \frac{1}{U(t)} \frac{\Delta U}{\Delta t_a U_0} g'\left(\frac{t - t_a}{\Delta t_a}\right) \int_0^y \omega_x(y', z) dy'. \end{aligned}$$

We define the dimensionless time with the characteristic time of the vortex pair during the high velocity regime: $t_0 = \frac{2\pi b_0^2}{\Gamma_1}$ (region towards which the instability propagates).

A computational region of interest for the T-D simulation corresponds to a region of the S-D simulation moving at U_1 as sketched in Fig.5.13. To make their comparison, we follow a reference cross plane x_{ref} : moving in the S-D case and fixed in the T-D case.

The time step of the simulation is fixed to $\Delta t/t_0 = 6.45 \cdot 10^{-5}$. The mesh size is $h/b = 1/100$ and the regularization parameter is $\sigma = 2h$. The length of the computational box has been chosen long enough to observe the phenomenon : $L_x/b = 15$. The simulation required 20 processors for 13,500 time steps (run for ~ 125 hours). We carried out the simulation using $\sim 40M$ grid points.

Numerical setup & initial condition for the time-developing simulation

The initial condition of the T-D simulation is in the same spirit of the S-D simulation. Over the distance E (also fixed to $b/2$), the circulation of the half vortex sheet increases from Γ_0 to Γ_1 in space (and not in time as in the space-developing simulation) :

$$\omega_x(x, y, z) = \omega_x(y, z) \left(1 + \frac{\Delta U}{U_0} g\left(\frac{x}{E}\right) \right) \quad (5.28)$$

To make the problem periodic, we finally use :

$$\omega_x(x, y, z) = \omega_\sigma(y, z) \left(1 + \frac{\Delta U}{U_0} g\left(\frac{x}{E}\right) g\left(\frac{L_1 - x}{E}\right) \right). \quad (5.29)$$

Since the vorticity field is divergence free and since the vertical component of the vorticity is zero, the spanwise vorticity component is obtained from :

$$\omega_y(x, y, z) = - \int \frac{\partial \omega_x}{\partial x} dy, \quad (5.30)$$

leading to

$$\omega_y(x, y, z) = - \frac{\Delta U}{U_0} \left(g\left(\frac{x}{E}\right) g\left(\frac{L_1 - x}{E}\right) \right)' \int_0^y \omega_x(y', z) dy'. \quad (5.31)$$

Fig. 5.14 presents the different time-evolution of the wake circulation for the S-D and for the T-D simulations which corresponds to the wake circulation of

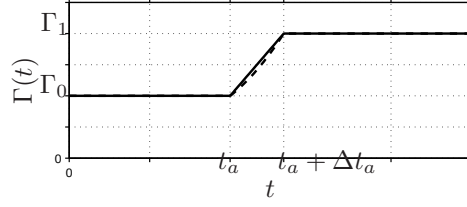


Figure 5.14: Schematic of the wake circulation evolution for the S-D simulation (solid) and for the T-D simulation (dash).

a plane moving in the initial condition at the wing speed. The initial condition for the T-D case is thus a good candidate to approximate the S-D case. The distances corresponding to the two regimes have been fixed to $L_0 = 10b$ and $L_1 = 20b$. The other numerical parameters are the same as in the S-D simulation, i.e. the mesh size $h/b = 1/100$, the time step $\Delta t/t_0 = 6.45 \cdot 10^{-5}$ and the regularization parameter $\sigma = 2h$. The simulation required $\sim 65M$ of grid points and was carried out using 40 processors for 4,800 time steps (it ran 170 hours).

Comparison and analysis of the space-developing and of the time-developing simulations

Fig. 5.15 shows a general 3D view of the S-D vortex sheet rollup at $Re = \Gamma_0/\nu = 10^4$. Information about the circulation profile is an important issue in order to model trailing vortices. Many analytical circulation profile formula, also called “vortex models”, and their related velocity profile, have been developed. The most usual models are itemized in details in [25] (see also Table 4.1 for a summary). We also connect the 2-D rollup simulations and the 3-D S-D simulations using:

$$x = U_0 t = U_0 t_0 \tau,$$

with $\tau = t/t_0$ the dimensionless time for the 2-D simulation, with $t_0 = b_0/V_0$ (where $V_0 = \Gamma_0/(2\pi b_0)$ is the wake descent velocity. A station at x/b in the 3-D S-D simulation thus has an equivalent dimensionless time equal to

$$\tau = \frac{x/b}{\left(\frac{U_0}{b} \frac{2\pi b_0^2}{\Gamma_0}\right)}$$

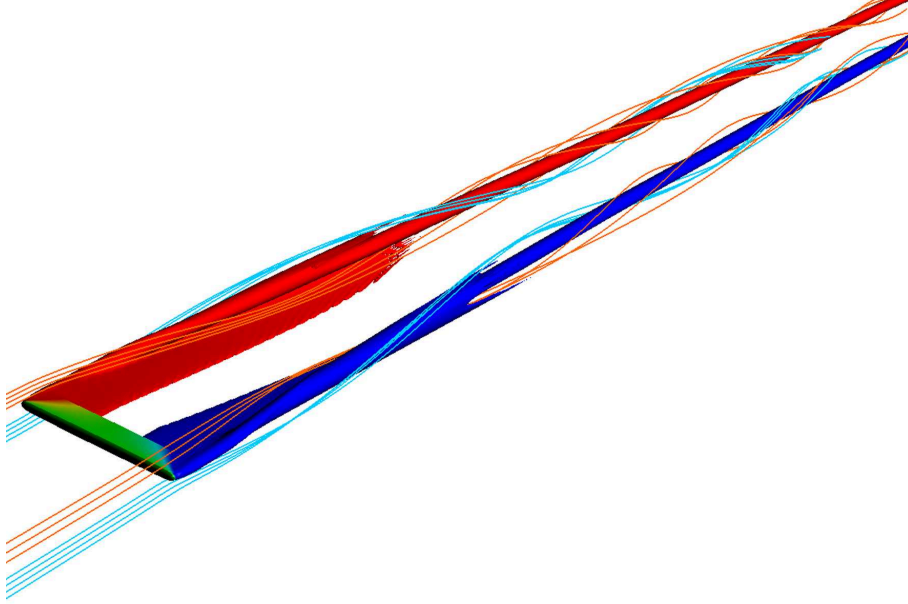
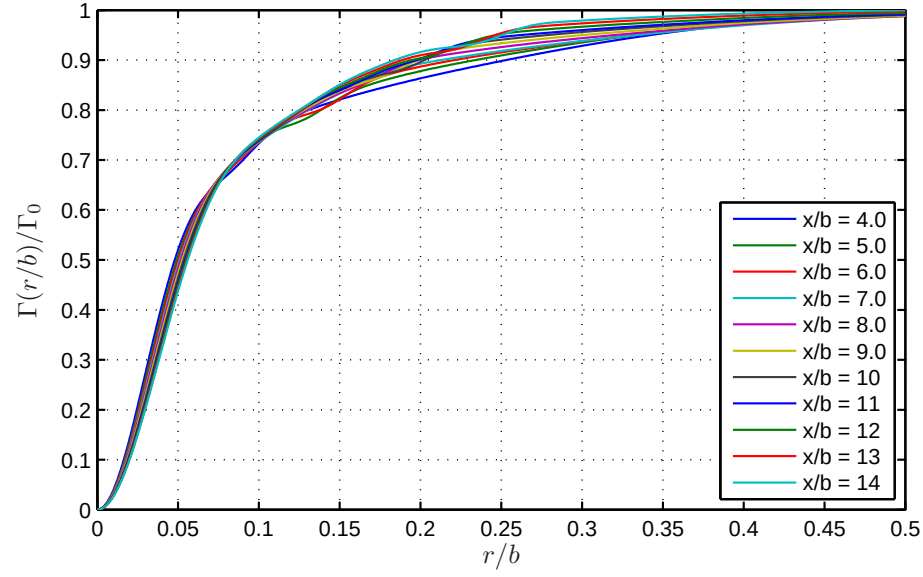


Figure 5.15: 3-D visualization of a S-D vortex sheet rollup. Isocontour of the vorticity field norm $\|\omega\|b^2/\Gamma_0 = 8.0$ colored by the axial vorticity component. A set of streamlines are also plotted in the wingtip region, going above/under the wing (orange/cyan lines).

Characterization of the wake Fig. 5.17 presents the evolution of the vortex core radius r_c for the S-D case and the T-D case. During the formation of the vortex pair, the vortex core quickly increases until $x/b \simeq 5$ ($\tau = 0.16$). As foreseen, the S-D simulation and the T-D simulation are very similar in terms of r_c and $u_{\theta,max}$.

Fig. 5.16 shows the circulation profiles $\Gamma(r)$ and the tangential velocity profiles $u_\theta(r) = \Gamma(r)/(2\pi r)$ that we have obtained for various cross planes. The maximum tangential velocity slightly decreases by diffusion from $(8.2V_0$ to $7.0V_0)$. Fig. 5.18 and Fig. 5.19 show the axial velocity. We analyze the value of the maximum axial velocity (positive and negative) in Fig. 5.18 and provide cross-views in Fig. 5.19. It is interesting to note that both are of the same order of magnitude. We find a peak of axial velocity deficit (i.e. towards the wing) in the vortex core roughly valued to $0.5V_0$ (when the rollup is completed and increasing shortly after). Moreover, an axial velocity in the opposite direction (away from the wing) surrounds the vortex core at the peak value of roughly $0.4V_0$.

(a)



(b)

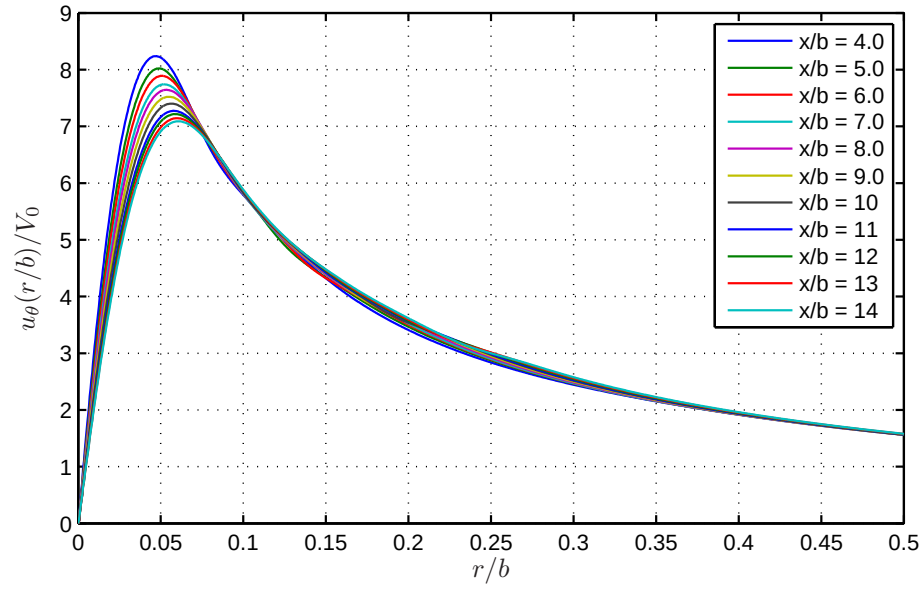


Figure 5.16: Circulation profiles (a) and tangential velocity profiles (b) at various distances in the wake.

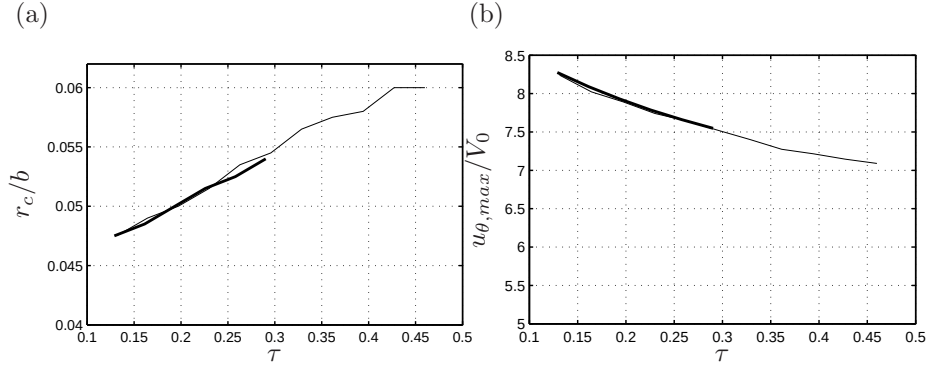


Figure 5.17: Evolution of the vortex effective core size r_c (a) and of the maximum tangential velocity $u_{\theta, max}$ (b) for the T-D simulation (tick line) and for the S-D simulation (thin line).

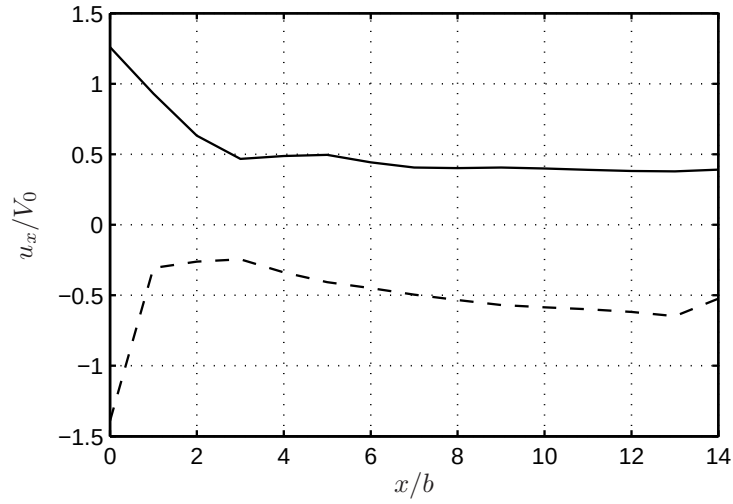
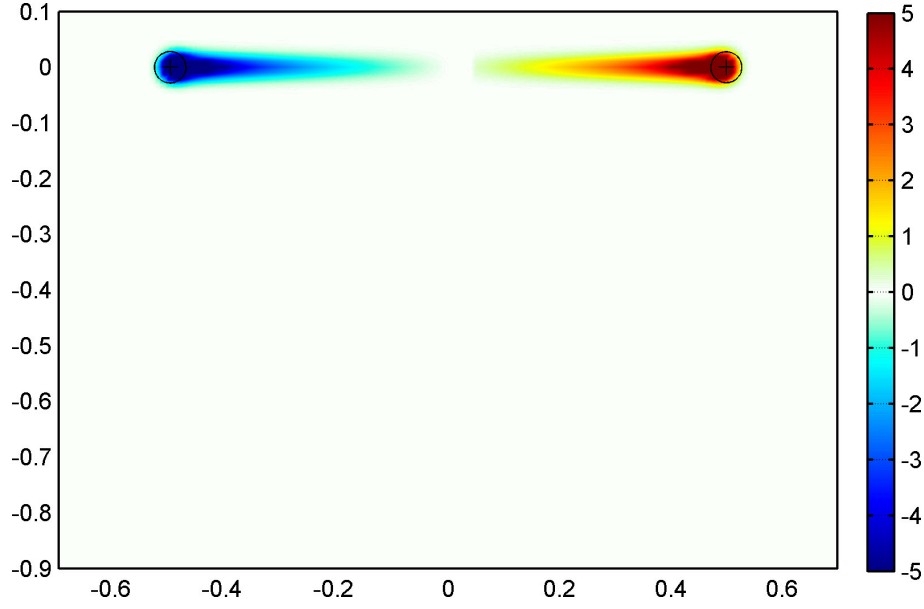


Figure 5.18: Evolution of the maximum axial velocity (solid) and the minimum axial velocity (dash) with respect to the distance in the wake.

(a) $x/b = 0.0$ ($\tau = 0.0$)

(b)

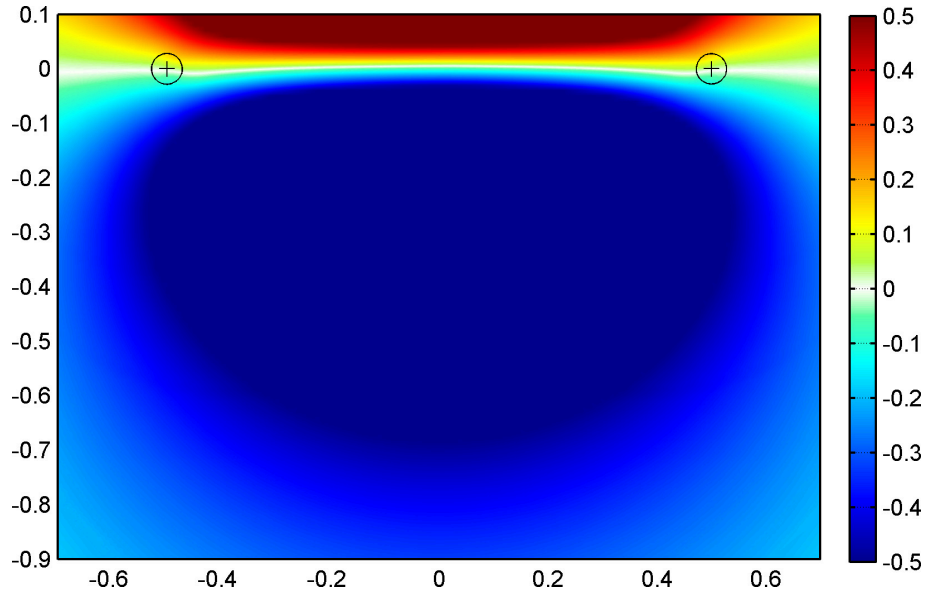
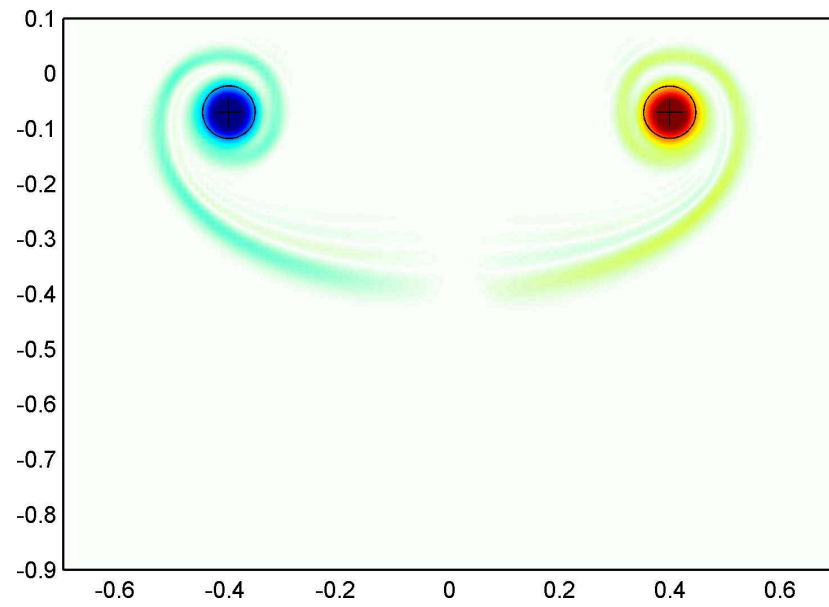


Figure 5.19: (a) Slice of the axial component of the vorticity field $\omega_x t_0$. (b) Slice of the axial component of the velocity field u_x/V_0 .

(a) $x/b = 4.0$ ($\tau = 0.13$)



(b)

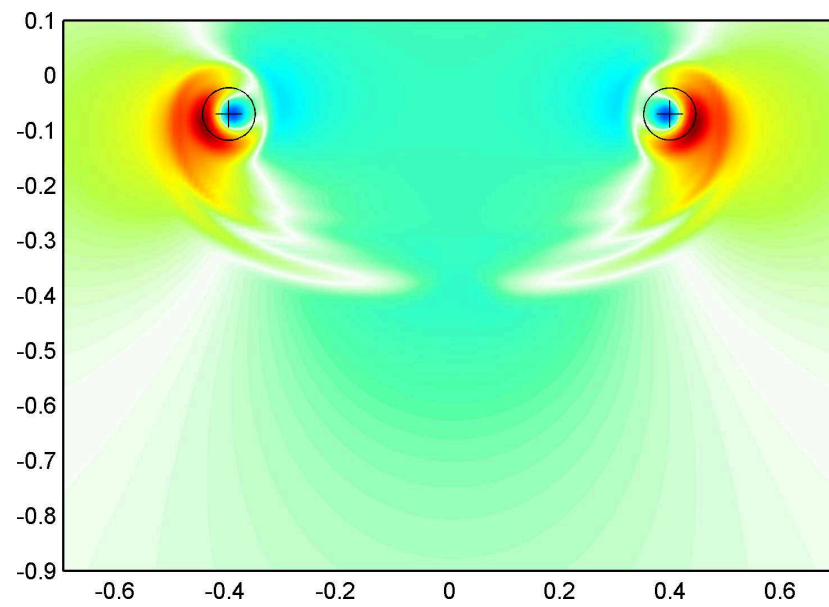
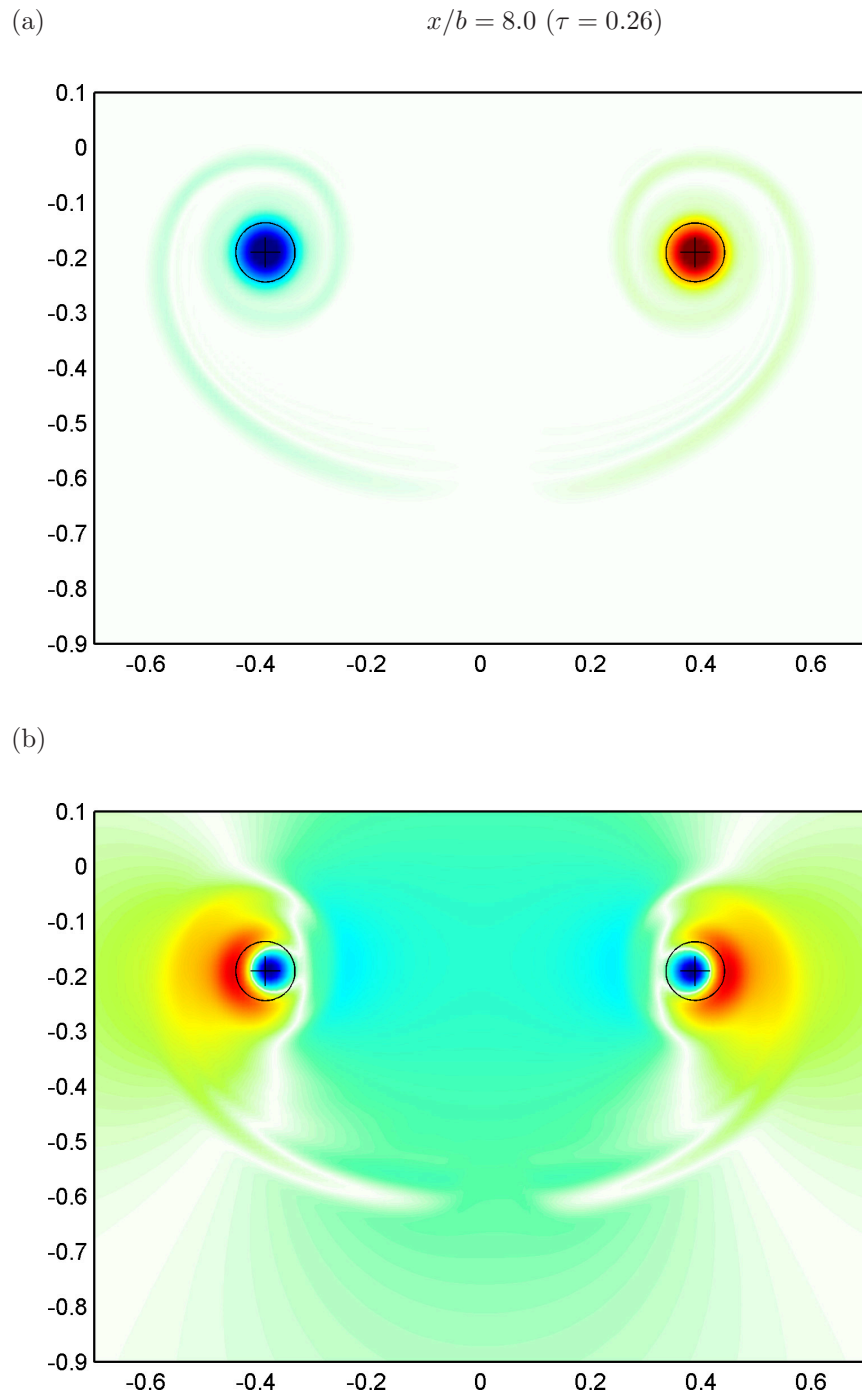
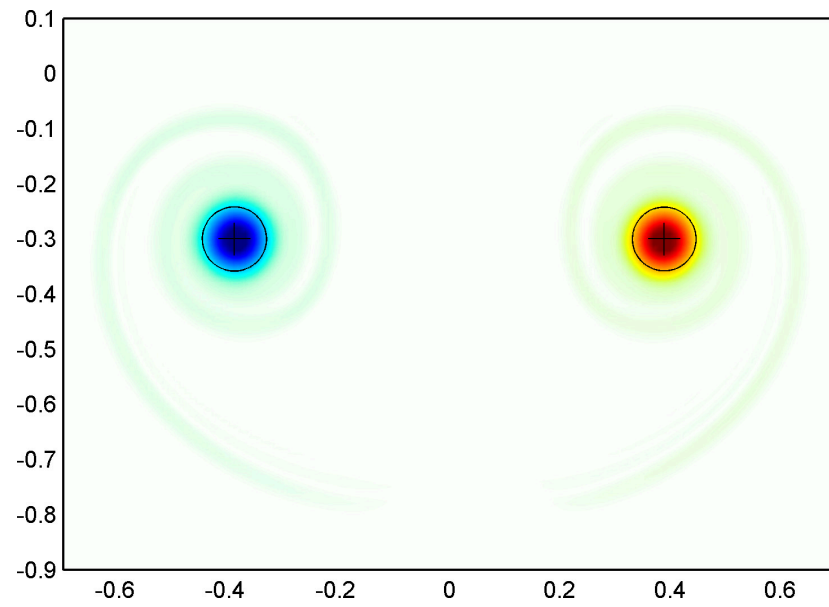


Figure 5.19: [Cont'd]

*Figure 5.19: [Cont'd]*

(a) $x/b = 12.0$ ($\tau = 0.39$)



(b)

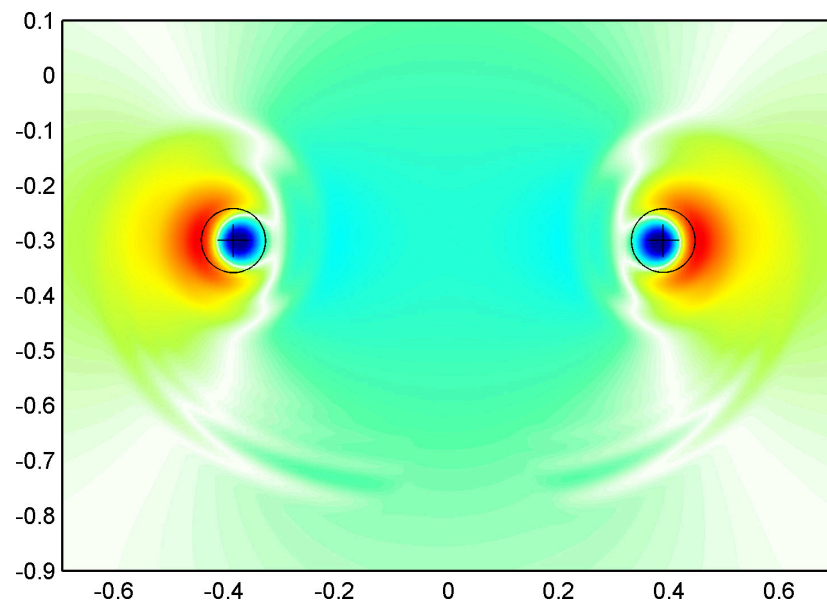


Figure 5.19: [Cont'd]

Global dynamics Fig. 5.20 presents the 3-D view of the S-D evolution of the accelerated wing and the resulting end-effect. While the wing accelerates, the vorticity generated by the acceleration rolls up around the two wake vortices. This creates an external deformation of the vortices. Afterwards, a helical instability develops that moves towards the stronger vortex and irremediably deforms the wake vortices. Fig. 5.21 shows the 3-D view of the end-effect for the T-D simulation. The same physics occur.

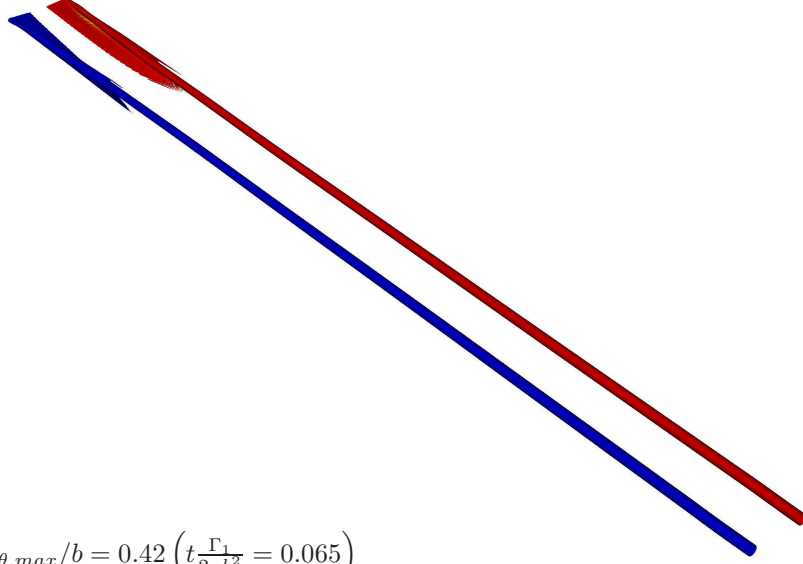
Visualization of the end-effect Fig. 5.22 shows a zoom, in the propagation region, of the vorticity isocontour at a low level of vorticity norm. In the S-D case, the end-effect appears to be more confined. The reason could be the fact that, in this case, an axial velocity (in the $\hat{\mathbf{e}}_x$ direction) surrounds the vortex and then convects, in the wake, the external deformation which propagates in the opposite direction. However, the pressure wave is not affected by the slowing down of the external helical instability. Fig. 5.23 shows two isocontours of vorticity norm for higher levels in order to visualize the inner vortices. A pressure wave propagates along each vortex which creates a strong axial velocity, destabilizes each vortex and allows the development of a helical instability.

Propagation of the instability Fig. 5.24 compares the evolution of the axial velocity front (u_w). This velocity is slightly higher in the S-D case : $u_w/u_{\theta,max} = 1.30$ instead of 1.20 for the T-D case. The difference might partially come from the additional axial velocity in the vortex core, valued at $\simeq 0.5V_0$. However, the global structure of the axial velocity is quite similar between the two cases, as shown in Fig. 5.25. The simulations were carried out at relatively high Reynolds number; the vortex core diffusion doesn't play much of a role and the front location evolves quasi linearly in time.

As done in the first section, we can analyze the axial component of the vorticity perturbation field ω'_x . Fig. 5.26 compares the isocontours of this field between the T-D simulation and the S-D simulation. The vorticity base fields $\omega_x^{(0)}$ are here taken as the vorticity field at the center slice of the high speed region for the T-D case, and as the vorticity field at the same time-evolving slice but for a rollup simulation at high speed regime for the S-D case. It confirms

the conclusion of Moet[59] : an axisymmetric instability is at first created which magnify the vortex core. This instability is followed by a helical one (the positive and the negative levels of vorticity perturbations are interlaced) that leads to the loss of the vortex core coherence.

$$tu_{\theta,max}/b = 0.0 \left(t \frac{\Gamma_1}{2\pi b_0^2} = 0.0 \right)$$



$$tu_{\theta,max}/b = 0.42 \left(t \frac{\Gamma_1}{2\pi b_0^2} = 0.065 \right)$$

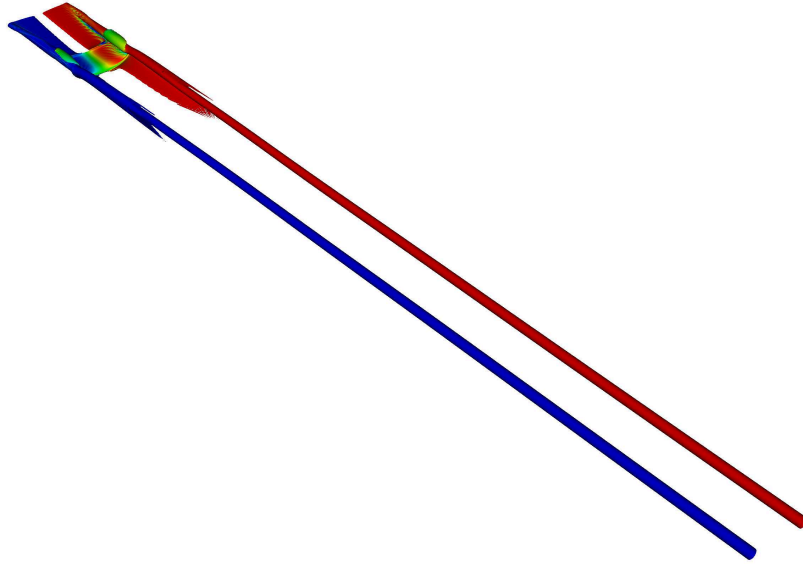
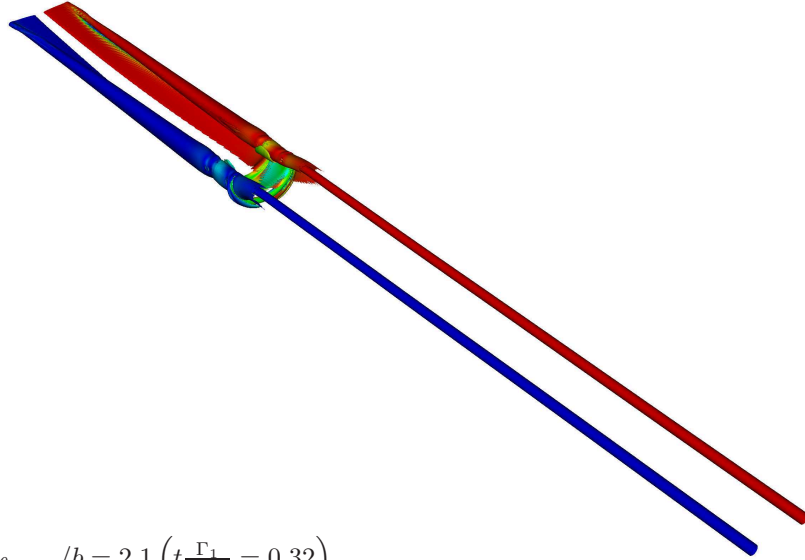


Figure 5.20: 3-D view of the wake for the space-developing simulation of an accelerated wing. Isocontour of the vorticity norm $\|\boldsymbol{\omega}\| b^2/\Gamma_1 = 8$ colored by the axial vorticity component.

$$tu_{\theta,max}/b = 1.2 \left(t \frac{\Gamma_1}{2\pi b_0^2} = 0.19 \right)$$



$$tu_{\theta,max}/b = 2.1 \left(t \frac{\Gamma_1}{2\pi b_0^2} = 0.32 \right)$$

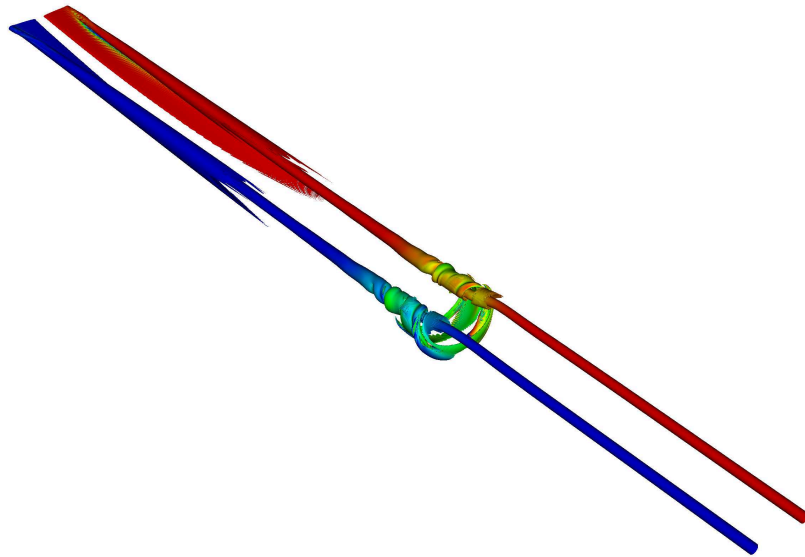
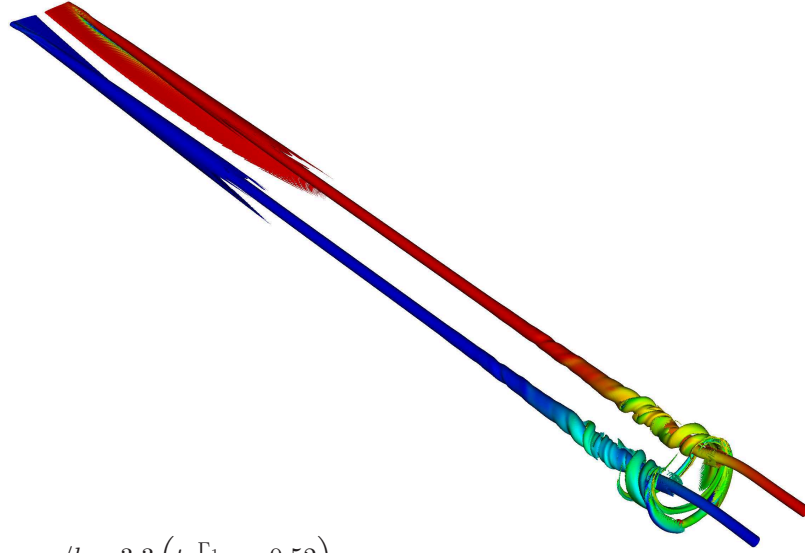


Figure 5.20: [Cont'd]

$$tu_{\theta, \max}/b = 2.9 \left(t \frac{\Gamma_1}{2\pi b_0^2} = 0.45 \right)$$



$$tu_{\theta, \max}/b = 3.3 \left(t \frac{\Gamma_1}{2\pi b_0^2} = 0.52 \right)$$

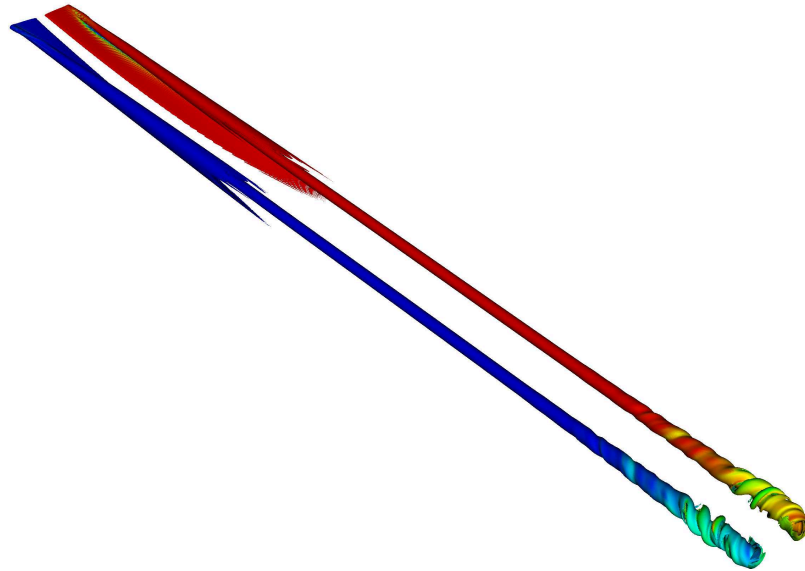
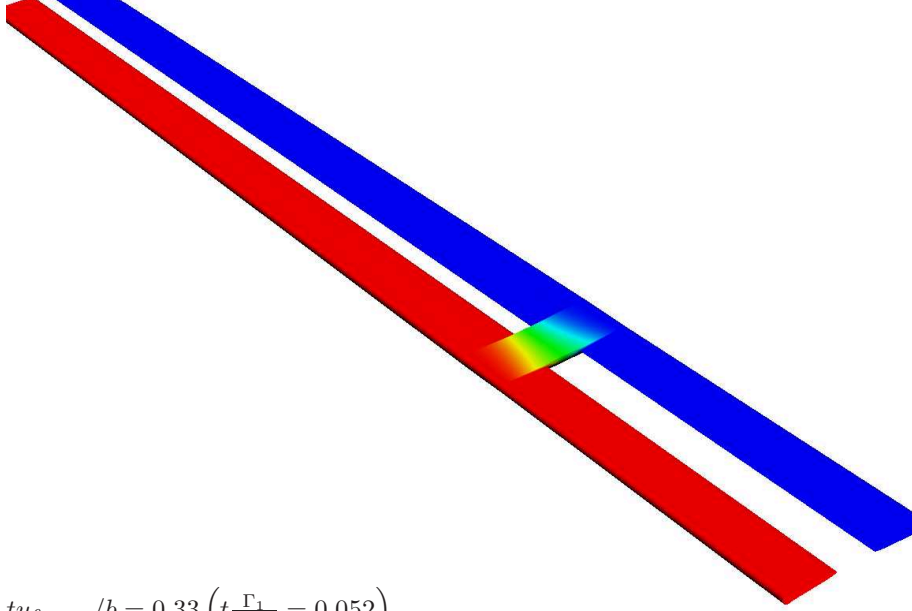


Figure 5.20: [Cont'd]

$$tu_{\theta,max}/b = 0.0 \left(t \frac{\Gamma_1}{2\pi b_0^2} = 0.0 \right)$$



$$tu_{\theta,max}/b = 0.33 \left(t \frac{\Gamma_1}{2\pi b_0^2} = 0.052 \right)$$

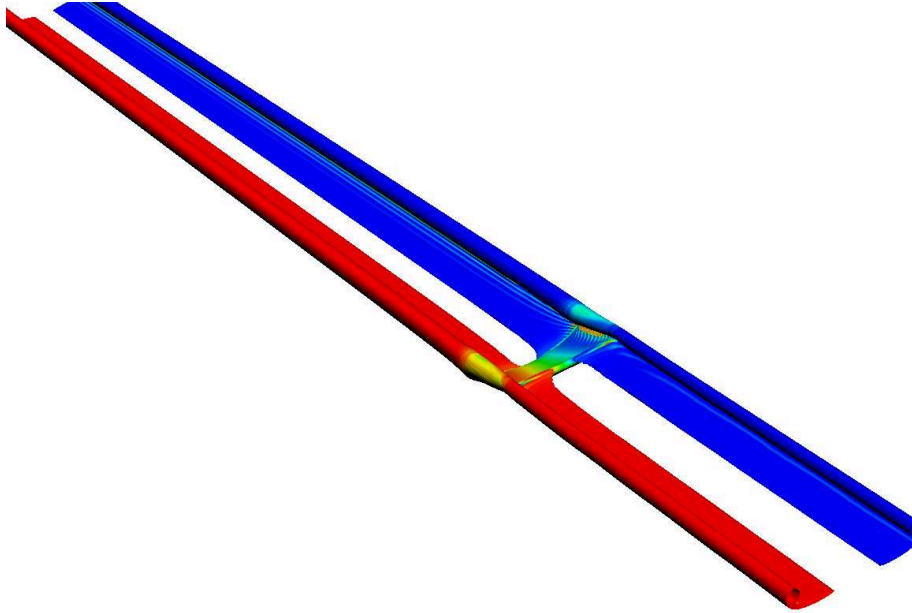
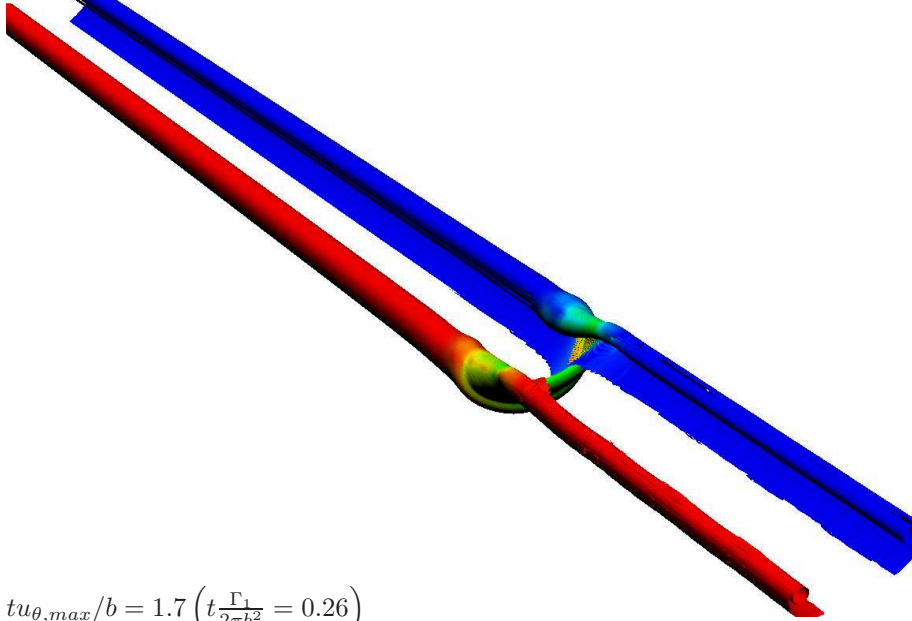


Figure 5.21: 3-D view of the wake for the time-developing simulation of an accelerated wing. Zoom of the vorticity norm isocontour $||\boldsymbol{\omega}|| b^2/\Gamma_1 = 8$ colored by the axial vorticity component.

$$tu_{\theta, \max}/b = 1.0 \left(t \frac{\Gamma_1}{2\pi b_0^2} = 0.15 \right)$$



$$tu_{\theta, \max}/b = 1.7 \left(t \frac{\Gamma_1}{2\pi b_0^2} = 0.26 \right)$$

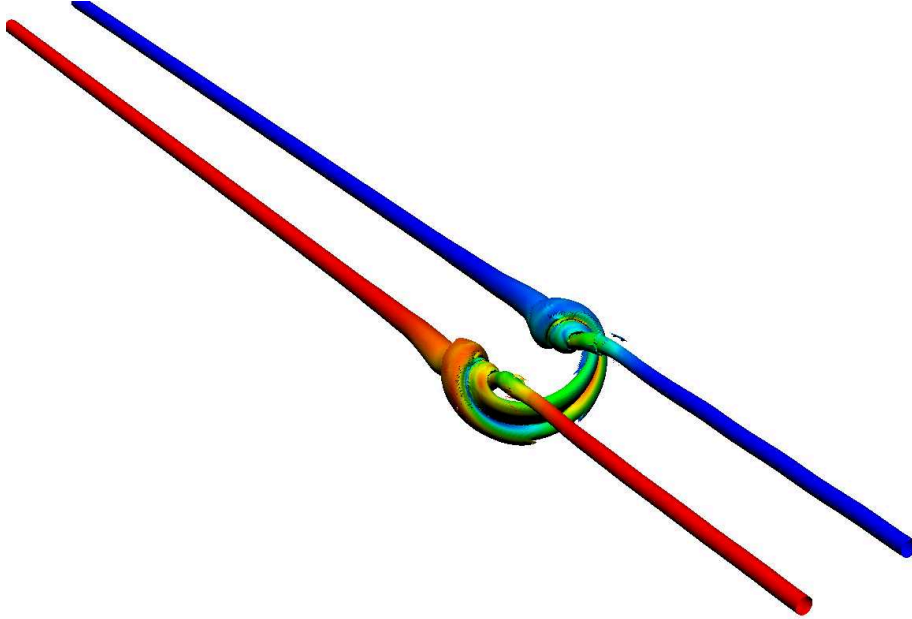
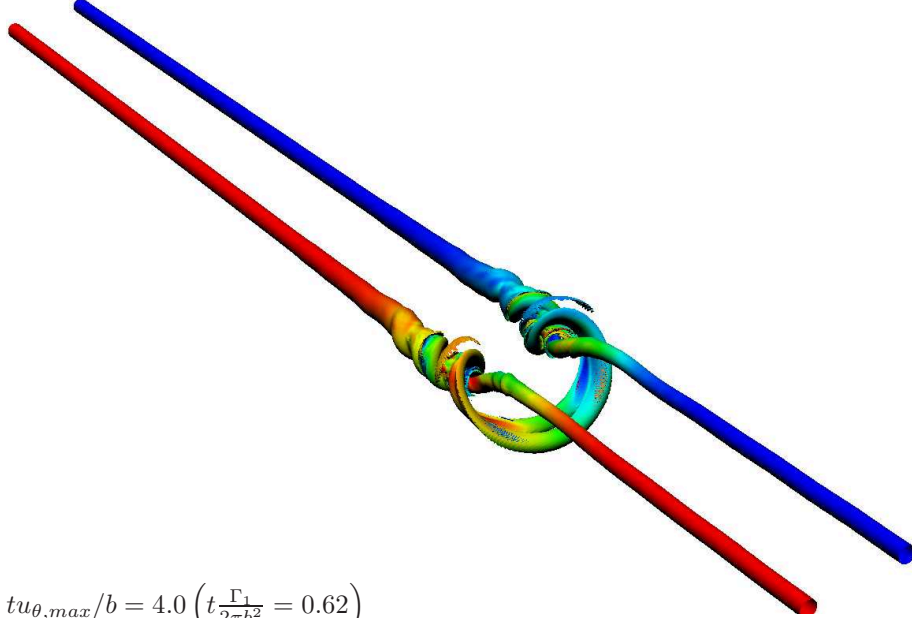


Figure 5.21: [Cont'd]

$$tu_{\theta, \max}/b = 2.7 \left(t \frac{\Gamma_1}{2\pi b_0^2} = 0.41 \right)$$



$$tu_{\theta, \max}/b = 4.0 \left(t \frac{\Gamma_1}{2\pi b_0^2} = 0.62 \right)$$

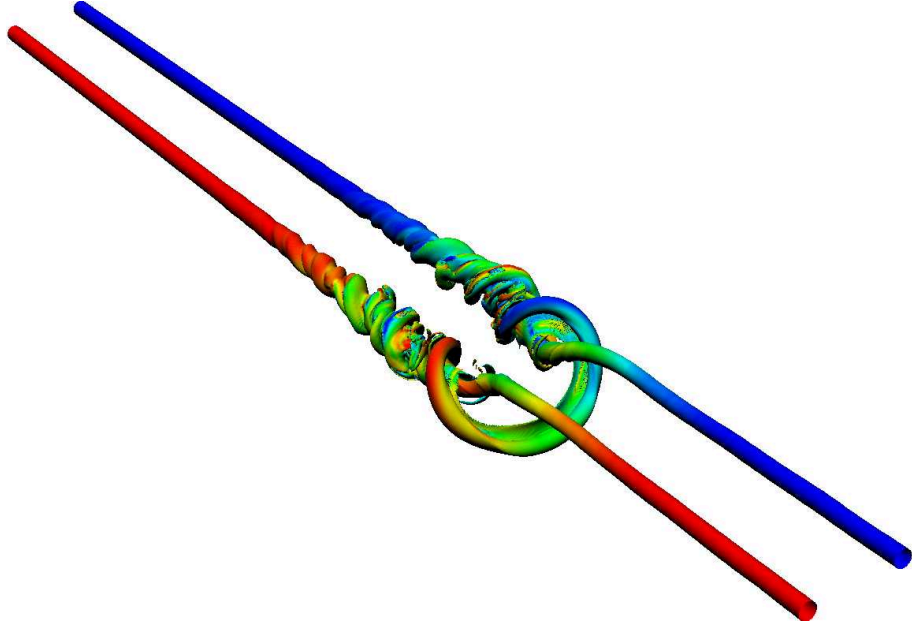


Figure 5.21: [Cont'd]

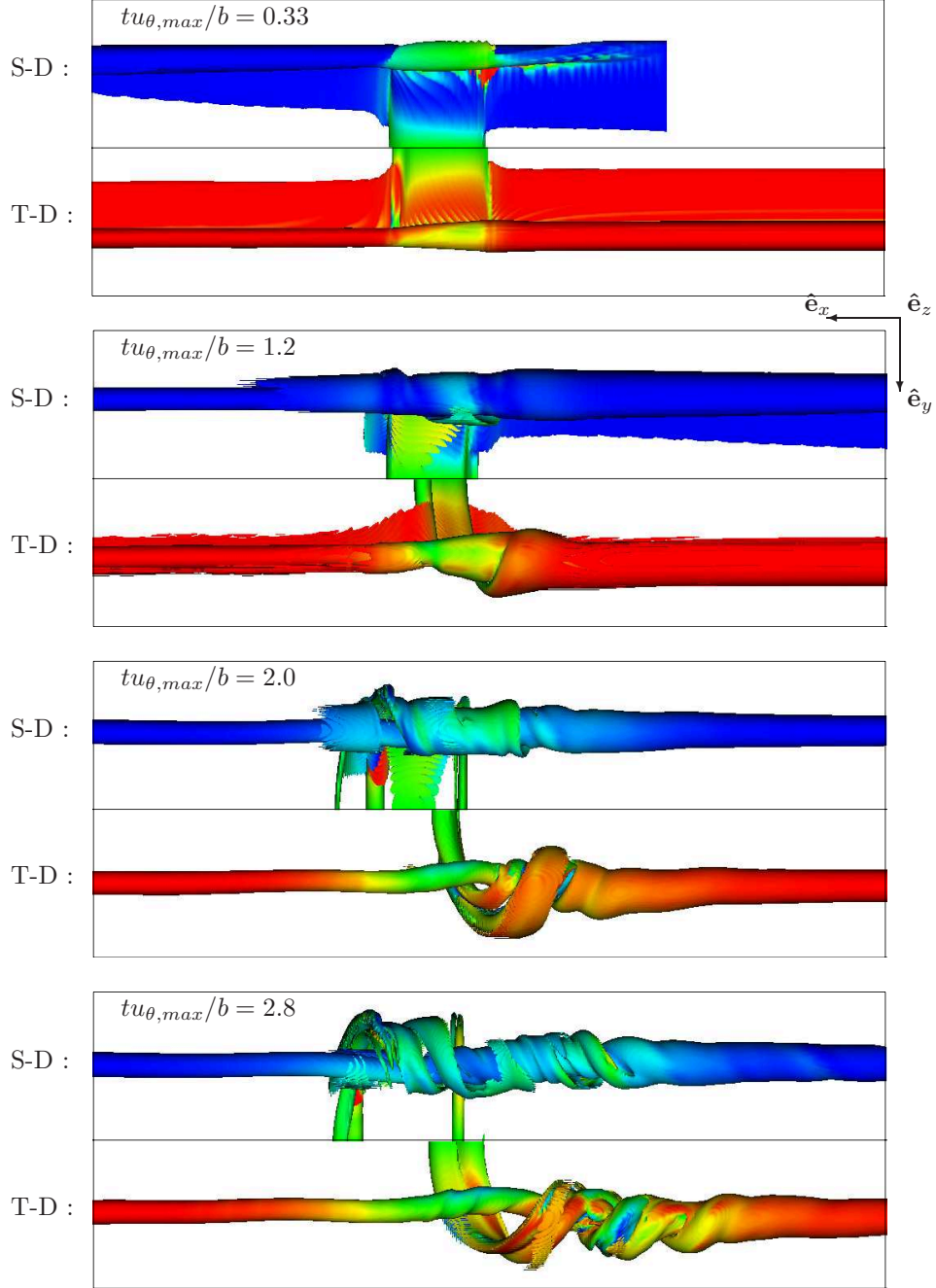


Figure 5.22: Top view of the vorticity field isocontour $||\omega||b^2/\Gamma_1 = 8$ colored by the axial vorticity component. Comparison of the instability evolution between the space-developing (S-D) and the time-developing (T-D) simulations for various times. The length of the visualization box is $4b$ centered relatively to x_{ref} .

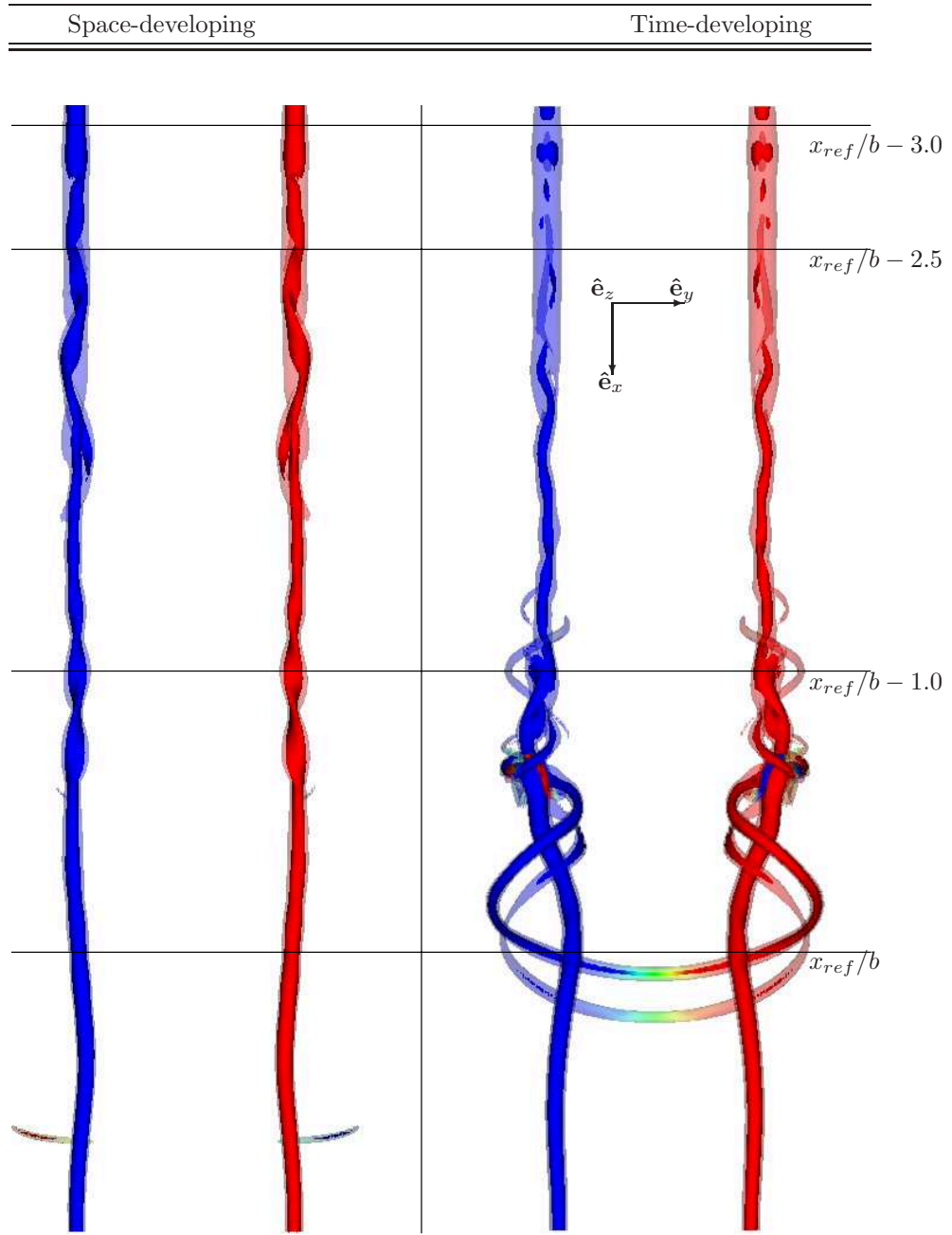


Figure 5.23: Top view of the vorticity isocontours colored by the axial vorticity component $\|\boldsymbol{\omega}\|b^2/\Gamma_1 = 40$ (low opacity) and $\|\boldsymbol{\omega}\|b^2/\Gamma_1 = 60$ (high opacity). Comparison between the space-developing simulation (left) and the time-developing simulation (right) at the time $tu_{\theta,max}/b = 2.8$. Also shown are the slices used Fig. 5.26.

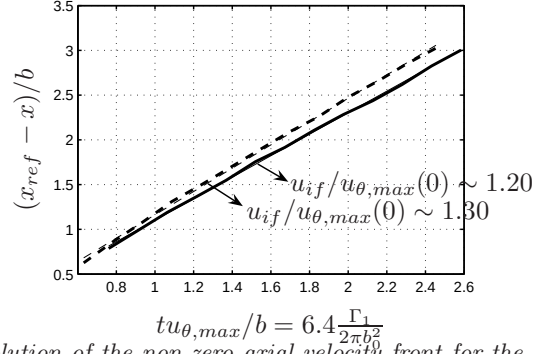


Figure 5.24: Evolution of the non-zero axial velocity front for the space-developing simulation (dash) and the time-developing simulation (solid). The thin lines corresponds to the least-squared fitted lines.

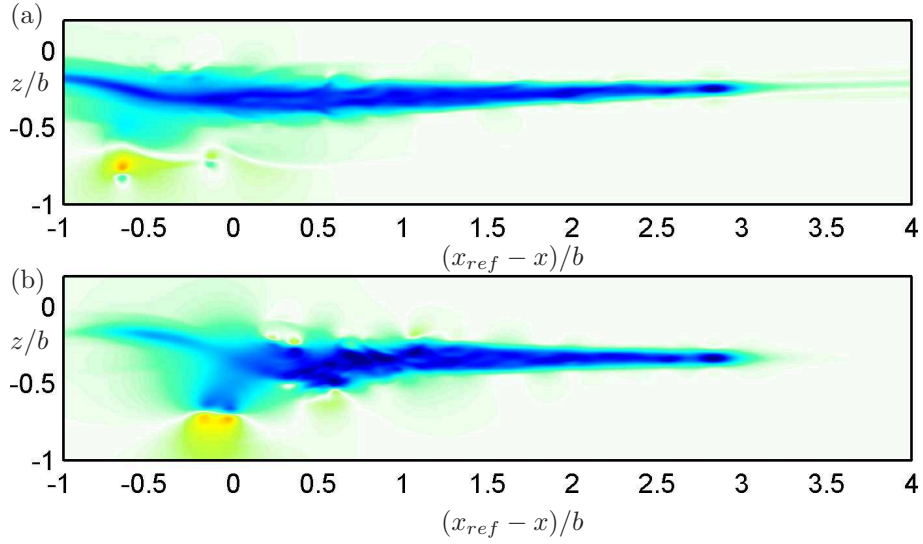


Figure 5.25: Cross plane view of the axial velocity field at $y/b = 0.5\pi/4$ for (a) the space-developing simulation and (b) the time-developing simulation at the time $tu_{\theta,max}/b = 2.8$.

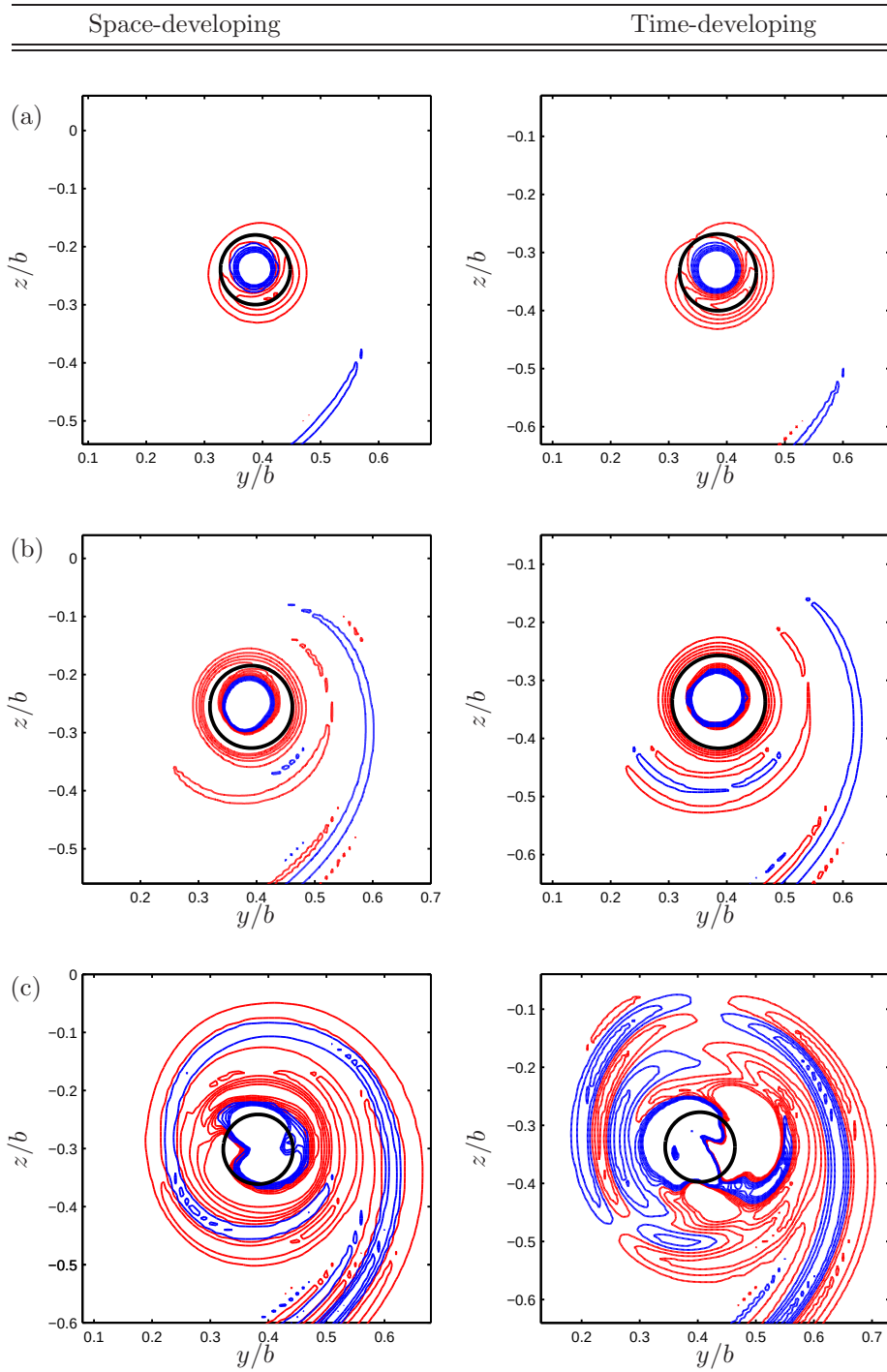


Figure 5.26: Ten equally spaced levels of the axial vorticity isocontour $\omega'_x b^2 / \Gamma_1$ from -5.5 to 5.5 , at $tu_{\theta, \max}/b = 2.8$, for various cross planes : (a) $x/b = x_{\text{ref}}/b - 3$, (b) $x/b = x_{\text{ref}}/b - 2.5$ and (c) $x/b = x_{\text{ref}}/b - 1$. Positive levels are plotted in red and the negative levels are plotted in blue. The circle with $r = r_c$ is also plotted in black.

5.5 Conclusion

The VIC-PFM method was shown to be suitable to simulate the dynamics of end-effects phenomena, both for the time-developing (T-D) and the space-developing (S-D) simulations. We observed comparable results to those of CNRS/IRPHE. The propagation speeds are found to be in good agreement with the theoretically predicted values. The same dynamics were observed for each simulated case, confirming the observations of Moet et al.[59] : at first an axisymmetric perturbation, corresponding to a pressure wave, propagates along the vortex core; if the resulting axial velocity is strong enough, as in the configurations investigated here, this wave can destabilize the vortex and a helical instability then develops.

The S-D simulation with wing acceleration points out a difference compared to the time-developing simulation in terms of propagation speed. The difference however remains minor and does not modify the dynamics.

Note however that the present S-D simulation, and the resulting 3-D rollup, underestimates the intensity of the axial velocity within the vortex core. It would likely be much higher considering a 3-D rollup with momentum deficit just behind the wing corresponding to the boundary layers and profile drag. This was not done in the present work; yet it constitutes an interesting follow up. It also can explain the different behaviour of the “end-effect” associated with an acceleration or a deceleration of the wing, as observed experimentally.

Chapter 6

Conclusions & perspectives

6.1 Achievements

This thesis has been devoted to the simulation of vortical flows using the vortex particle-mesh (VPM) method, also sometimes called “vortex-in-cell” (VIC) method. The contributions of this work are twofold: they consist of methodological improvements and of investigations of complex vortical flows.

Our work in the vortex particle-mesh method itself has brought two outstanding advances:

- In the spirit of the immersed boundary (IB) method, an approach has been successfully developed for the simulation of external flows past arbitrary-shape bodies (i.e. bluff- and profiled-bodies). The approach is based on the methodology proposed by Ploumhans et al. [68] that was then developed for the “classical” vortex particle method (VP, without grid). It was adapted and extended to the VPM framework where the Poisson equation is solved on a grid; here using the `FISHPACK` library. The high fidelity simulation of external flows benefits from the reduction of the computational cost associated to this combined approach. The developments were done in the framework developed by Coale et al. [11, 13] where the VPM method was combined with the parallel fast multipole method (PFM) for efficient domain decomposition simulations. The present VPM-PFM-IB was successfully validated against the

well-referenced test case of a flow past the sphere at $Re = 300$. Furthermore, the method was used to capture the flow past a low aspect-ratio wing at $Re = 3000$.

- A mesh refinement handling within this VPM method has also been developed; including the IB feature. The algorithm implements a wavelet-based conservative interpolation scheme of vortex particles patches with various resolutions. It also permits a consistent reinitialization of the vortex particle strengths. Good conservation properties were observed. The method has also been validated on the flow past a sphere at $Re = 300$. Furthermore, this multi-resolution capability has also been used, together with multiscale subgrid modeling, to perform a large eddy simulation of a wake-vortex flow at high Reynolds number ($Re = 10^6$). It is worth noting that this simulation would not be affordable using a uniform resolution. This contribution thus definitely increases the possibilities of turbulent vortical flow simulations, for which we typically observe multiple scales within the flow.

In the context of air traffic management, a preponderant factor is the aircraft induced wake-vortex hazard. It is at the origin of the separation standard for landing and take-off phases. Our work on the investigation of such vortical flows has also brought outstanding advances:

- We have investigated the influence of a velocity deficit (due to the friction drag) on the wake rollup. For that purpose, two space-developing (S-D) simulations, with and without velocity deficit effect, have been carried out. They have been compared. The former becomes quickly turbulent while the latter remains stationary. In addition to those simulations, a complementary time-developing (T-D) simulation with velocity deficit has been performed. We found some differences in the instability development. In opposition to the S-D simulation, a helical instability develops around the primary vortex-cores. Finally, the net-result of the rolled-up vortex-wake was characterized and the most common vortex-models were fitted: it was confirmed that two-length scale models are required to properly fit both the vortex-core size and the maximal tangential velocity.

- We have studied the end-effect phenomenon, associated to an axial variation of the vortex/wake circulation. Fundamental differences were found between the results obtained by the simulations and those obtained by simplified theory. The application of the VPM method to these flows has shown that the VPM method does constitute the method of choice for such problems.

6.2 Perspectives

Finally, the research work reported in this thesis raises a few questions and hints at potential future developments. We close this thesis with a discussion of these open issues.

6.2.1 A VPM method using a staggered grid

VPM methods are based on the resolution of the Poisson equation using an efficient grid solver such as **FISHPACK**: i) to obtain the streamfunction from the vorticity field and ii) to reproject the vorticity field in a solenoidal space. This provides non-fully consistent streamfunction: i.e., we do not exactly recover the vorticity field by taking twice the grid-evaluated curl of the obtained streamfunction. $\nabla_h \times (\nabla_h \times \Psi)$ is not exactly equal to ω . However, it turns out that for usual applications, the results are accurate enough if the vorticity divergence is maintained at an acceptable, and low, level. The same conclusion is drawn concerning the correction procedure of the vorticity field. Nevertheless, the initial level of vorticity divergence can be quite high for wall-bounded flows using immersed-boundary techniques. Moreover, this high level of divergence is convected in the wake. This can lead to local and/or global inaccuracies. An elegant way to address this issue would be to solve the Poisson equation using a staggered grid, in the same spirit as what is done in classical velocity-pressure based Navier-Stokes solvers, for which a Poisson equation is solved to make the velocity field divergence-free.

This approach was recently also proposed and tested by Uchiyama and Yoshii [84]. They definitely obtain good results in terms of self-consistency and accuracy.

6.2.2 Further improvements of the Poisson solver for hierarchically refined grids

The presented approach to solve the Poisson equations is not necessarily optimal in terms of computational cost and parallel efficiency. We here combined a fast direct solver (e.g. **FISHPACK**) with the parallel fast multipole (PFM) solver; in order to obtain the required boundary conditions. This approach is very versatile and robust but it requires that each processor be highly loaded. It results in a poor strong-scaling and an acceptable weak-scaling (Cocle et al. [13]). This behavior is explained by the following reason: the computational cost associated to the PFM evaluations is relatively high; given a fixed problem size, when increasing the number of processors, the PFM code is used more and more and that finally ends up with a computational time comparable to that of the classical vortex method using PFM only (i.e. no grid). This is even more true when using multi-resolution because many grid blocks that have few grid points are created leading to a less efficient load-balancing.

A good strategy for improvement would be the use of a multigrid solver for hierarchically refined grids; as first suggested by Brandt [5] and then improved by Brown and Lowe [7] and Meglicki et al. [56]. The Poisson equation would be solved on a bigger grid, requiring then fewer PFM evaluations.

An interesting project is the open development platform called “**CHOMBO**” which handles multi-physics and AMR abilities as well as a multigrid solver for Poisson equation (see Martin et al. [55]). An equivalent platform is the **Overture** framework (see Brown et al. [6]). We suggest to develop a VPM method using these platforms. This could be fairly easily implemented, leading to an even more efficient numerical tool for vortical flows. It is worth noting that another interesting method exists for solving a Poisson equation on hierarchically refined grids, and which is still a direct solver (see Huang and Greengard [38] and Ricker [71]).

The overall algorithm can also be further improved in terms of accuracy. So far, on a common surface between two blocks with different resolution, the Laplacian operator is computed independently from each side. Meaning that the diffusion flux is likely not identical across the surface. This potentially leads to some vorticity source in the flow. The Laplacian scheme can be modified to

avoid this issue (as done by Martin et al. [55]). We note that this feature is already implemented in the AMR frameworks cited above.

6.2.3 Improvement of the handling of no-slip

The approach to capture the flow past a no-slip boundary is summarized as follows: i) the panel sources that cancel the residual slip velocity are computed using the Boundary Element Method (BEM), ii) those panel sources are redistributed on the grid (and thus smoothed), in order to obtain the velocity by solving the Poisson equation $\nabla^2 \Psi = -\omega$ and iii) adding the panel sources into the flow through an equivalent diffusion flux. The major drawback of this approach is a certain inconsistency between, on the one hand, the velocity field used to compute the panel sources and, on the other hand, the velocity field obtained after solving the Poisson equation. This leads to a time oscillation of the vorticity flux at the wall and thus of the force acting on the body.

The consistency of the method can be improved by means of the evaluation of the boundary sources that are already redistributed on the grid (as done by Cottet and Poncet [17] and Poncet [70]). This however leads to a limited accuracy since the discontinuous vorticity solution is still not captured. A better approach is likely the use of a sharp-interface method (see Udaykumar et al. [85] and Marella et al. [54] and Yang and Udaykumar [92]) for which the near-wall operators are modified in order to handle boundary-discontinuities, preserving the discretization accuracy up to the wall. This approach can be enhanced using the immersed interface method (see Leveque and Li [50] and Li and Ito [51] for more details on the method and Morgenthal and Walther [61] for a 2D VPM application).

6.2.4 A coupling approach using an Eulerian “near-wall” solver

The combined VPM-PFM-IB method is costly for wall-bounded external flows at high Reynolds number. As the fundamental element of discretization is still isotropic ($h \times h \times h$), the method is intrinsically unadapted to capture thin boundary layers which are a main characteristic of high Reynolds number flows. A significant gain in terms of computational cost can be achieved by

means of a Eulerian-Lagrangian coupling (see Daeninck and Winckelmans [23] and Waerenburgh [86]). The coupling approach, proposed by Daeninck [23], has been validated in 2-D using a “classical” vortex particle method (i.e., no grid), together with a finite difference code for incompressible flows. We here suggest to couple, on the one hand, the efficient combined VPM-PFM-IB method and, on the other hand, a hybrid finite volume/finite element code for compressible flows (i.e. the **ARGO** code developed by CENAERO). This approach aims to make the two codes agree, at least for a highly subsonic regime (i.e. Mach number $< 0.1 \dots 0.2$).

The coupling approach is summarized as follows. The Lagrangian code has a global view of the flow; it contains the entire vorticity field. It is responsible for capturing the flow from the wall to the wake. The boundary layers are, however, much under-resolved, yet their particle strengths are correct (see below). The Eulerian code, allowing a body-fitted mesh, only captures the “near-wall” region. The Eulerian vorticity field is used to correct the near-wall Lagrangian vorticity field hence, the particle strengths in the near wall region are correct (as this corresponds to a fine to coarse procedure). The required boundary conditions on the outer part (i.e. velocity, pressure and temperature) are provided by the Lagrangian code. So far, only Dirichlet type boundary conditions were investigated. The pressure field is computed using a supplementary scalar Poisson equation (see Ploumhans [66]), whereas the temperature field is modeled.

• Pressure-field evaluation

A Poisson equation (and thus also an integral equation) is solved in order to compute the pressure field from the vorticity and the velocity fields :

$$\nabla^2 B = -\nabla \cdot (\boldsymbol{\omega} \times \mathbf{u}),$$

where $B = (p/\rho_\infty + \frac{1}{2}\mathbf{u} \cdot \mathbf{u}) - (p_\infty/\rho_\infty + \frac{1}{2}\mathbf{U}_\infty \cdot \mathbf{U}_\infty)$. The boundary condition to impose at the body surface is $\frac{\partial B}{\partial n} = -\nu(\nabla \times \boldsymbol{\omega}) \cdot \mathbf{n}$. This equation

is solved using the boundary element method :

$$\xi B(\mathbf{x}) - \int_S \frac{\partial G}{\partial n'}(|\mathbf{x} - \mathbf{x}'|) B(\mathbf{u}') d\mathbf{x}' = \nu \int_S G(|\mathbf{x} - \mathbf{x}'|) (\nabla'_x \times \boldsymbol{\omega}(\mathbf{x}')) \cdot \mathbf{n}(\mathbf{x}') d\mathbf{x}' \\ + \int_V G(|\mathbf{x} - \mathbf{x}'|) \nabla'_x \cdot (\boldsymbol{\omega}(\mathbf{x}') \times \mathbf{u}(\mathbf{x}')) d\mathbf{x}'.$$

where $\xi = \frac{1}{2}$ for a point $\mathbf{x}$ located on the surface body and $\xi = 1$ otherwise. The integral equation and the Poisson equation are solved in the same way as those involved in the immersed boundary technique.

- **Temperature-field estimation**

The VPM-PFM-IB solves the Navier-Stokes equation for incompressible flows and thus it does not provide the temperature field. Various hypotheses can be proposed to recover an image of the temperature field knowing the upstream conditions $(p_\infty, \rho_\infty, T_\infty, U_\infty)$:

1. Assuming the temperature is such that the density is conserved:
 $T(\mathbf{x}) = p(\mathbf{x})/(\rho_\infty R)$, where R is the gas constant.
2. Assuming an isothermal flow $T(\mathbf{x}) = T_\infty$.
3. Assuming a homentropic transformation: $T(\mathbf{x}) = T_\infty (p_\infty/p(\mathbf{x}))^{\frac{\gamma}{\gamma-1}}$
where $\gamma = \frac{c_p}{c_v}$ is the ratio of the specific heats.
4. Assuming a polytropic transformation $T(\mathbf{x}) = T_\infty (p/p_\infty)^{\frac{m}{m-1}}$, where
 $1 \leq m \leq \gamma$.

The first three hypotheses have been compared for the Euler flow past a sphere and at a Mach number $M = \frac{U_\infty}{\sqrt{\gamma R T_\infty}} = 0.1$. The ARGON code was used. The mesh is such that the external boundary is close to the body. It essentially consists in a sphere inside a cube, where the side $c = \frac{3}{2}D$ (D being the sphere diameter) using $\sim 220\,000$ tetrahedrons, see Fig. 6.1.(a). “Incompressible” boundary conditions (described before) are imposed. It turns out that the hypotheses 2 and 3 provide satisfactory results (and thus also likely hypothesis 4, not tested); the pressure field corresponding to the case 3 is presented in Fig. 6.1.(b).

The complete coupling approach (VPM-PFM-IB-ARGON), that is with viscosity effects, has been implemented but not yet validated. The impulsively

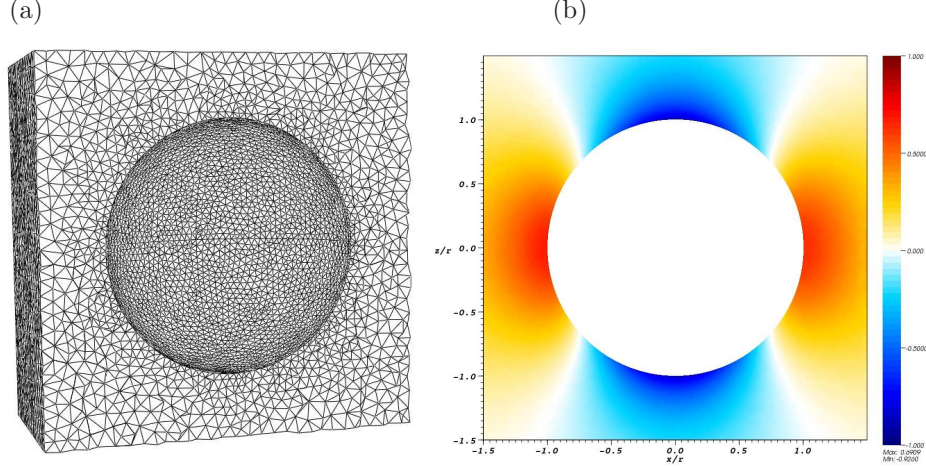
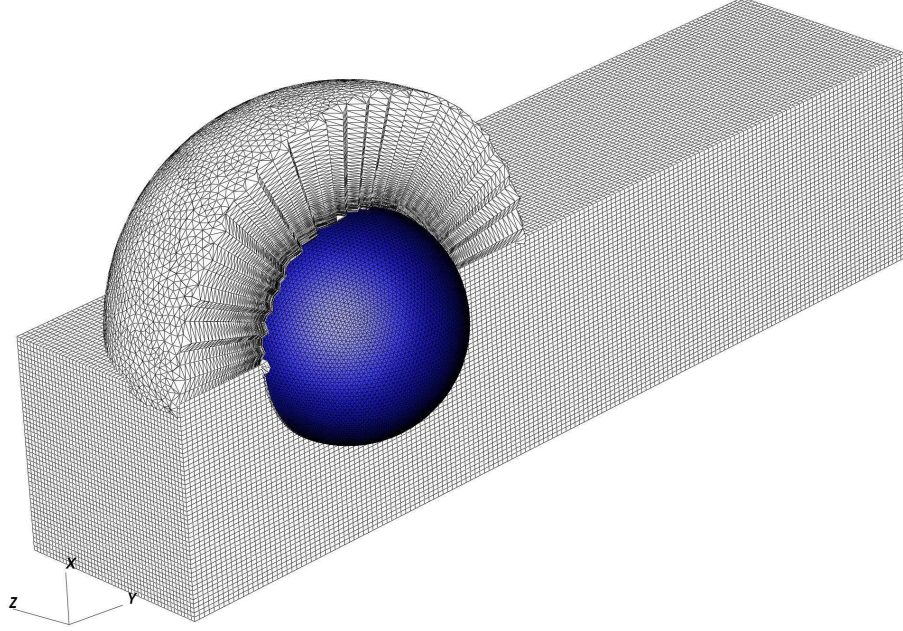


Figure 6.1: (a) View of the mesh of a sphere included in a box. (b) Slice of the pressure field $(p - p_\infty)/p_\infty * 100$ just after impulsive start assuming an isothermal flow.

started flow past a sphere at $Re = 300$ and $M = 0.1$ has been computed. The mesh used for each code is presented in Fig. 6.2.(a). Fig. 6.2.(b) shows a slice of the velocity norm for both domains at the early age $tU_\infty/D = 1.5$. We clearly see that velocity field is continuous through each domain.

These are preliminary results which seem to indicate that the methodology is coherent and could lead to a very efficient tool. Much work yet remains in order to “get there”. Wind farms computations constitute a good example of problems for which it is crucial to capture aerodynamics coupled to wake dynamics and for which this methodology should be investigated.

(a)



(b)

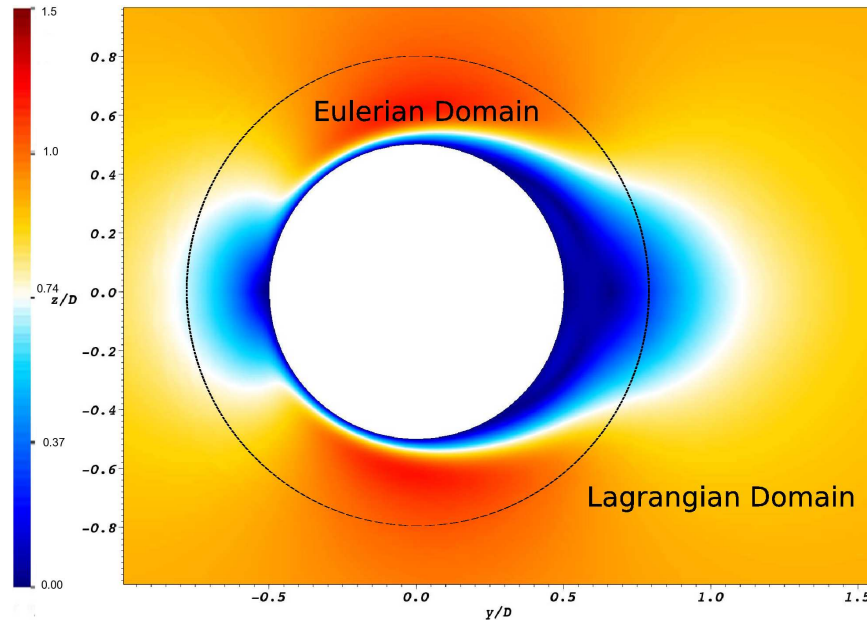


Figure 6.2: (a) Quarter of the mesh used by the compressible ARGO code (top) and by the incompressible VIC-PFM-IB code (bottom). (b) 2-D cut of the velocity norm $|\mathbf{u}|/U_\infty$ at $t \frac{U_\infty}{D} = 1.5$. Eulerian field inside the circle and Lagrangian field outside the circle.

Bibliography

- [1] J. Barnes and P. Hut. A hierarchical $\mathcal{O}(N \log N)$ force-calculation algorithm. *Nature*, 324:446–449, 1986.
- [2] M. Bergdorf, G.-H. Cottet, and P. Koumoutsakos. Multilevel adaptive particle methods for convection-diffusion equations. *Multiscale Modeling and Simulation: A SIAM Interdisciplinary Journal*, 4(1):28–357, 2005.
- [3] M. Bergdorf and P. Koumoutsakos. A lagrangian particle-wavelet method. *Multiscale Modeling and Simulation: A SIAM Interdisciplinary Journal*, 5(3):980–995, 2006.
- [4] M. J. Berger and J. Oliger. Adaptive mesh refinement for hyperbolic partial differential equations. *J. Comput. Phys.*, 53(3):484–512, 1984.
- [5] A. Brandt. Multi-level adaptive solutions to boundary-value problems. *Math. Comp.*, 31:333–390, 1977.
- [6] D. Brown, W. Henshaw, and D. Quinlan. Overture: An object-oriented framework for solving partial differential equations. In Yutaka Ishikawa, Rodney Oldehoeft, John Reynders, and Marydell Tholburn, editors, *Scientific Computing in Object-Oriented Parallel Environments*, volume 1343 of *Lecture Notes in Computer Science*, pages 177–184. Springer Berlin / Heidelberg, 1997.
- [7] D. Brown and L. Lowe. Multigrid elliptic equation solver with adaptive mesh refinement. *J. Comput. Phys.*, 209(2):582–598, 2005.
- [8] P. Chatelain, A. Curioni, M. Bergdorf, D. Rossinelli, W. Andreoni, and P. Koumoutsakos. Billion vortex particle direct numerical simulations of

- aircraft wakes. *Computer Methods in Applied Mechanics and Engineering*, 197(13-16):1296–1304, 2008.
- [9] P. Chatelain and P. Koumoutsakos. A fourier-based elliptic solver for vortical flows with periodic and unbounded directions. *J. Comput. Phys.*, 229(7):2425–2431, 4 2010.
- [10] J. P. Christiansen. Numerical solution of hydrodynamics by the method of point vortices. *J. Comput. Phys.*, 13:363–379, 1973.
- [11] R. Cocle. *Combining the vortex-in-cell and parallel fast multipole methods for efficient domain decomposition simulations : DNS and LES approaches*. PhD thesis, Université catholique de Louvain, Louvain-la-Neuve, 2007.
- [12] R. Cocle, L. Dufresne, and G. Winckelmans. Investigation of multiscale subgrid models for les of instabilities and turbulence in wake vortex systems. In *Lecture Notes in Computational Science and Engineering series*, Symposium on Complex Effects in Large-Eddy Simulation (CY-LES 2005), Limassol (Cyprus), January 2007. Springer, January 2007.
- [13] R. Cocle, G. Winckelmans, and G. Daeninck. Combining the vortex-in-cell and parallel fast multipole methods for efficient domain decomposition simulations. *J. Comput. Phys.*, 227:2263–2292, 2008.
- [14] G. Constantinescu and K. Squires. Numerical investigations of flow over a sphere in the subcritical and supercritical regimes. *Phys. Fluids*, 16(5):1449–1466, 2004.
- [15] M. Coquerelle and G.-H. Cottet. A vortex level set method for the two-way coupling of an incompressible fluid with colliding rigid bodies. *Journal of Computational Physics*, 227(21):9121 – 9137, 2008. Special Issue Celebrating Tony Leonard’s 70th Birthday.
- [16] G.-H. Cottet and P. Koumoutsakos. *Vortex Methods: Theory and Practice*. Cambridge University Press, first edition, 2000.
- [17] G.-H. Cottet and P. Poncet. Advances in direct numerical simulations of 3d wall-bounded flows by vortex-in-cell methods. *J. Comput. Phys.*, 193(1):136–158, 2004.

- [18] C. Cottin. Experimental investigation of the dynamics of counter-rotating vortex pairs in ground effect. Master's thesis, Chalmers University of Technology, 2006.
- [19] C. Cottin, G. Winckelmans, T. Lonfils, and R. Cocle. Roll-up of a temporally-evolving wing wake with velocity deficit. Far-wake research project technical report 2.2.2-3, Université catholique de Louvain (UCL), Louvain-la-Neuve, 2007.
- [20] S. C. Crow. Stability theory for a pair of trailing vortices. *AIAA Journal*, 8(12):2172–2178, 1970.
- [21] G. Daeninck. *Developments in hybrid approaches. Vortex method with known separation location. Vortex method with near-wall Eulerian solver. RANS-LES coupling*. PhD thesis, Université catholique de Louvain, 2006.
- [22] G. Daeninck, R. Cocle, and G. Winckelmans. FAR-Wake, TR-1.1.2-1-les of spatially evolving wakes in ground effect. FAR-Wake technical report TR3.1.2-4 (42 pages), Université catholique de Louvain, 2006.
- [23] G. Daeninck and G. Winckelmans. A new eulerian-lagrangian vortex method for flows with massive separation. In *6th National congress on Theoretical and Applied Mechanics, Ghent*, 1993.
- [24] A.C. de Bruin. Estimation of exhaust velocities and temperatures for various operation phases of a modern high by-pass turbofan. FAR-Wake research project deliverable d 2-1.1-1b, 2005.
- [25] A.C. de Bruin and G. Winckelmans. Cross-flow kinetic energy and core size growth of analytically defined wake vortex pairs. Technical report, Nationaal Lucht- en Ruimtevaartlaboratorium (NLR), Aug. 2005.
- [26] P. de Vos and M. Janssens. Investigation expérimentale de tourbillons de sillage d'aile en phase d'atterrissage. Master's thesis, Université catholique de Louvain (UCL), 2007.
- [27] O. Desenfans and S. Gigantelli. Instabilités des tourbillons de sillage d'aile avec décélération brusque : investigation dans le bassin de halage. Master's thesis, Université catholique de Louvain (UCL), 2004.

- [28] J. Eldredge, C. Colonius, and A. Leonard. A vortex particle method for two-dimensional compressible flow. *J. Comput. Phys.*, 179(2):371–399, 2002.
- [29] D. Fabre, Sipp D., and L. Jacquin. The kelvin waves and the singular modes of the lamb-oseen vortex. *J. Fluid Mech.*, 551:235–274, 2006.
- [30] D. Fabre and L. Jacquin. Viscous instabilities in trailing vortices at large swirl numbers. *J. Fluid Mech.*, 500:239–262, 2003.
- [31] L. Georges. *Development and validation of a LES methodology for complex wall-bounded flows : application to high-order structured and industrial unstructured solvers*. PhD thesis, Université catholique de Louvain, Louvain-la-Neuve, 2007.
- [32] C. Geuzaine and J.-F. Remacle. Gmsh: a three-dimensional finite element mesh generator with build-in pre- and post-processing facilities. *International Journal for Numerical Methods in Engineering*, 79(11):1309–1331, 2009.
- [33] L. Greengard. *The rapid evaluation of potential fields in particle systems*. PhD thesis, 1987.
- [34] L. Greengard and V. Rokhlin. A fast algorithm for particle simulations. *J. Comput. Phys.*, 73(2):325–348, 1987.
- [35] B. Hejazialhosseini, D. Rossinelli, M. Bergdorf, and P. Koumoutsakos. High order finite volume methods on wavelet-adapted grids with local time-stepping on multicore architectures for the simulation of shock-bubble interactions. *Journal of Computational Physics*, 229(22):8364 – 8383, 2010.
- [36] S. E. Hieber and P. Koumoutsakos. An immersed boundary method for smoothed particle hydrodynamics of self-propelled swimmers. *J. Comput. Phys.*, 227:8636–8654, October 2008.
- [37] R. W. Hockney. A fast direct solution of poisson’s equation using fourier analysis. *J. ACM*, 12(1):95–113, 1965.

- [38] J. Huang and L. Greengard. A fast direct solver for elliptic partial differential equations on adaptively refined meshes. *SIAM J. Sci. Comput.*, 21(4):1551–1566, 1999.
- [39] L. Jacquin, D. Fabre, D. Sipp, and E. Coustols. Unsteadiness, instability and turbulence in trailing vortices. *Comptes Rendus Physique*, 6:399, 2005.
- [40] L. Jacquin, D. Fabre, D. Sipp, V. Theofilis, and H/ Vollmers. Instability and unsteadiness of aircraft wake vortices. *Aerosp. Sci. Tech.*, 7:577–593, 2003.
- [41] J. Jeong and F. Hussain. On the identification of a vortex. *J. Fluid Mech.*, 285(-1):69–94, 1995.
- [42] T. A. Johnson and V. C. Patel. Flow past a sphere up to a Reynolds number of 300. *J. Fluid Mech.*, 378:19–70, 1999.
- [43] G. Karypis and V. Kumar. *METIS, a Software Package for Partitioning Unstructured Graphs and Computing Fill-Reduced Orderings of Sparse Matrices*.
- [44] G. Karypis and V. Kumar. A fast and high quality multilevel scheme for partitioning irregular graphs. *SIAM J. Sci. Comput.*, 20(1):359–392, 1998.
- [45] P. Koumoutsakos. *Direct numerical simulations of unsteady separated flows using vortex methods*. PhD thesis, California Institute of Technology, 1993.
- [46] P. Koumoutsakos. Multiscale flow simulations using particles. *Annual Review of Fluid Mechanics*, 37(1):457–487, 2005.
- [47] P. Koumoutsakos and A. Leonard. High-resolution simulations of the flow around an impulsively started cylinder using vortex methods. *J. Fluid Mech.*, 296(1):1–38, 1995.
- [48] S. Leibovich. Wave propagaton, instability, and breakdown of vortices. In H. G. Hornung and Muller E.-A., editors, *Vortex Motion*, pages 50–67, 1982.

- [49] S. Leibovich. Vortex breakdown: A coherent transition trigger in concentrated vortices. In M. Lesieur and O. Metais, editors, *Turbulence and coherent structures*, 1990.
- [50] R. J. Leveque and Z. Li. The immersed interface method for elliptic equations with discontinuous coefficients and singular sources. *SIAM Journal on Numerical Analysis*, 31(4):1019–1044, 1994.
- [51] Z. Li and K. Ito. *The Immersed Interface Method: Numerical Solutions of PDEs Involving Interfaces and Irregular Domains (Frontiers in Applied Mathematics)*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2006.
- [52] H. Liu, S. Krishnan, S. Marella, and H.S. Udaykumar. Sharp interface Cartesian grid method II: A technique for simulating droplet interactions with surfaces of arbitrary shape. *J. Comput. Phys.*, 210(1):32–54, 2005.
- [53] O. Lucca-Negro and T. O’Doherty. Vortex breakdown: a review. *Progress in Energy and Combustion Science*, 27(4):431 – 481, 2001.
- [54] S. Marella, S. Krishnan, H. Liu, and H.S. Udaykumar. Sharp interface Cartesian grid method I: An easily implemented technique for 3D moving boundary computations. *J. Comput. Phys.*, 210(1):1 – 31, 2005.
- [55] D. Martin, P. Colella, and D. Graves. A cell-centered adaptive projection method for the incompressible navier-stokes equations in three dimensions. *J. Comput. Phys.*, 227(3):1863–1886, 2008.
- [56] Z. Meglicki, S. Gray, and B. Norris. Multigrid FDTD with chombo. *Computer Physics Communications*, 176(2):109–120, 2007.
- [57] P. Meunier. FAR-Wake, TR-1.1.2-1 axial propagation of vortex perturbations. Technical report, CNRS-IRPHE, 2006.
- [58] R. Mittal and G. Iaccarino. Immersed boundary methods. *Ann. Rev. Fluid Mech.*, 37:239–261, 2005.
- [59] H. Moet, F. Laporte, G. Chevalier, and T. Poinso. Wave propagation in vortices and vortex bursting. *Phys. Fluids*, 17(5):054109 (15 pages), 2005.

- [60] J.J. Monaghan. Extrapolating b splines for interpolation. *J. Comput. Phys.*, 60(2):253–262, 1985.
- [61] G. Morgenthal and J.H. Walther. An immersed interface method for the vortex-in-cell algorithm. *Comput. Struct.*, 85(11-14):712–726, 2007.
- [62] F. Noca, D. Shiels, and D. Jeon. Measuring instantaneous fluid dynamic forces on bodies, using only velocity fields and their derivatives. *Journal of Fluids and Structures*, 11(3):345–350, 1997.
- [63] F. Noca, D. Shiels, and D. Jeon. A comparison of methods for evaluating time-dependent fluid dynamic forces on bodies, using only velocity fields and their derivatives. *Journal of Fluids and Structures*, 13(5):551–578, 1999.
- [64] M. Ould Salihi. *Couplage de méthodes numériques en simulation directe d'écoulements incompressibles*. PhD thesis, Université Joseph Fourier, Grenoble, 1998.
- [65] C.S. Peskin. The immersed boundary method. *Acta Numerica*, pages 479–517, 2002.
- [66] P. Ploumhans. *Simulation of High Reynolds Number Flows Past Bluff Bodies Using Vortex and Boundary Element Methods*. PhD thesis, Université catholique de Louvain, Louvain-la-Neuve, 2001.
- [67] P. Ploumhans and G. Winckelmans. Vortex methods for high-resolution simulations of viscous flow past bluff bodies of general geometry. *J. Comput. Phys.*, 165:354–406, 2000.
- [68] P. Ploumhans, G. Winckelmans, J. Salmon, A. Leonard, and M. Warren. Vortex methods for direct numerical simulation of three-dimensional bluff body flows: application to the sphere at $Re = 300, 500$, and 1000 . *J. Comput. Phys.*, 178(2):427–463, 2002.
- [69] P. Poncet. Vanishing of mode b in the wake behind a rotationally oscillating circular cylinder. *Phys. Fluids*, 14(6):2021–2023, 2002.

- [70] P. Poncet. Analysis of an immersed boundary method for three-dimensional flows in vorticity formulation. *J. Comput. Phys.*, 228:7268–7288, 2009.
- [71] P. M. Ricker. A direct multigrid poisson solver for oct-tree adaptive meshes. *The Astrophysical Journal Supplement Series*, 176(1):293–300, 2008.
- [72] D. Rossinelli, M. Bergdorf, G.-H. Cottet, and P. Koumoutsakos. Gpu accelerated simulations of bluff body flows using vortex particle methods. *Journal of Computational Physics*, 229(9):3316 – 3333, 2010.
- [73] D. Rossinelli, M. Bergdorf, B. Hejazialhosseini, and P. Koumoutsakos. Wavelet-based adaptive solvers on multi-core architectures for the simulation of complex systems. 5704:721–734, 2009.
- [74] J. K. Salmon and M. S. Warren. Skeletons from treecode closet. *SIAM J. Sci. Comput.*, 111(1):136–155, 1994.
- [75] K. Shariff and A. Leonard. Vortex rings. *Ann. Rev. Fluid Mech.*, 24:235, 1992.
- [76] K. Shariff, R. Verzicco, and P. Orlandi. A numerical study of three dimensional vortex ring instabilities : viscous corrections and early nonlinear stage. *J. Fluid Mech.*, 279:351, 1994.
- [77] P. R. Spalart. Airplane trailing vortices. *Ann. Rev. Fluid Mech.*, 30:107–138, 1998.
- [78] P.N. Swarztrauber. A direct method for the discrete solution of separable elliptic equations. *SIAM Journal of Numerical Analysis*, 11:1136–1150, 1974.
- [79] P.N. Swarztrauber, R. A. Sweet, and J. C. Adams. FISHPACK: Efficient FORTRAN Subprograms for the Solution of Elliptic Partial Differential Equations. Technical report, National Center for Atmospheric Research, <http://www.cisl.ucar.edu/css/software/fishpack/technote.ps>, July 1999.
- [80] P.N. Swarztrauber and R.A. Sweet. The fourier and cyclic reduction methods for solving Poisson’s equation. In J.A. Schetz and A.E. Fuhs, editors,

Handbook of fluid dynamics and fluid machinery. John Wiley & Sons, New York, 1996.

- [81] W. Sweldens and P. Schröder. Building your own wavelets at home. In Roland Klees and Roger Haagmans, editors, *Wavelets in the Geosciences*, volume 90 of *Lecture Notes in Earth Sciences*, pages 72–107. Springer Berlin / Heidelberg, 2000. 10.1007/BFb0011093.
- [82] F. Thirifay and G. Winckelmans. Development of a lagrangian method for combustion and application to the planar methane air jet diffusion flame. *Journal of Turbulence*, 3(59):1468–5248, 2002.
- [83] A. G. Tomboulides and S. A. Orszag. Numerical investigation of transitional and weak turbulent flow past a sphere. *J. Fluid Mech.*, 416:45–73, 2000.
- [84] T. Uchiyama and Y. Yoshii. Improvements of vortex in cell method for incompressible flow simulation, submitted paper. *Computers and Fluids*, 2011.
- [85] H.S. Udaykumar, R. Mittal, P. Rampunggoon, and A. Khanna. A Sharp Interface Cartesian Grid Method for Simulating Flows with Complex Moving Boundaries. *J. Comput. Phys.*, 174(1):345–380, 2001.
- [86] X. Waerenburgh. Simulation numérique d’écoulements 2d incompressibles autour de corps : Couplage d’une méthode eulérienne et d’une méthode lagrangienne. Travail de fin d’étude, Université catholique de Louvain (UCL), Louvain-la-Neuve, 2001.
- [87] G. Winckelmans. *Vortex Methods*, volume 3 (Fluids) of *Encyclopedia of Computational Mechanics*, chapter 5. John Wiley & Sons, 2004.
- [88] G. Winckelmans, R. Cocle, L. Dufresne, and R. Capart. Vortex methods and their application to trailing wake vortex simulations. *Comptes Rendus Physique*, 6(4-5):467–486, 2005. Aircraft trailing vortices.
- [89] G. Winckelmans, L. Dufresne, and L. Bricteux. LES investigation of aircraft wake two-vortex system in low atmospheric turbulence. *Bull. Amer. Phys. Soc.*, 50(10), 2005.

- [90] G. Winckelmans and A. Leonard. Contributions to vortex particle methods for the computation of three-dimensional incompressible unsteady flows. *J. Comput. Phys.*, 2(109):247–273, 1993.
- [91] G. Winckemans. *Topics in Vortex Methods for the Computation of Three- and Two-Dimensional Incompressible Unsteady Flows*. PhD thesis, California Institute of Technology, 1989.
- [92] Y. Yang and H.S. Udaykumar. Sharp interface Cartesian grid method III: Solidification of pure materials and binary solutions. *J. Comput. Phys.*, 210(1):55–74, 2005.
- [93] L. Zhang and J. Eldredge. A viscous vortex particle method for deforming bodies with application to biolocomotion. *International Journal for Numerical Methods in Fluids*, 59(12):1299–1320, 2009.

Appendix A

Space-developing simulation of a near-wake rollup using a modeled inflow vorticity field

This appendix presents the results of the space-developing simulation of the vortex sheet rollup with velocity deficit using an inflow. It is shown that this simulation provided non-physical results as the circulation of each vortex of the counter-rotating vortex-pair increase.

The straightforward space-developing extension of the time-developing velocity deficit model is here used as an inflow. It basically consists in imposing the vorticity field (such as described in Chapter 4; in the paragraph entitled “*Description of the vorticity-field inflow-model*”)

A global 3-D perspective view is presented in Fig. A.1. Fig. A.2 provides a zoom in the near-field region. Fig. A.4 presents various slices of the averaged streamwise vorticity. At the first sight, one could be tempted to conclude that the topology of the flow is similar to the one presented in Fig. 4.6 and Fig. 4.7. Yet, this is not the case. It turns out that the circulation of the two rolled-up vortices increases, see Fig. A.3. This also leads to a significant increase

of the linear impulse. This behavior is definitely not physically acceptable. That could be explained by analyzing the spanwise vorticity (i.e. the vorticity component due the modeling of the friction drag) flux at the inflow. Because of the presence of the “lifting-line” vorticity, the streamwise velocity component is not up-down anti-symmetric (which was the case in the time-developing simulation). Hence, the total spanwise vorticity flux (i.e. the integral over a cross-plane) is none zero. The resulting difference is gathered in both vortices.

To circumvent this issue, another approach is proposed for which the lift-force is directly modeled (exhaustively presented in Chapter 4).

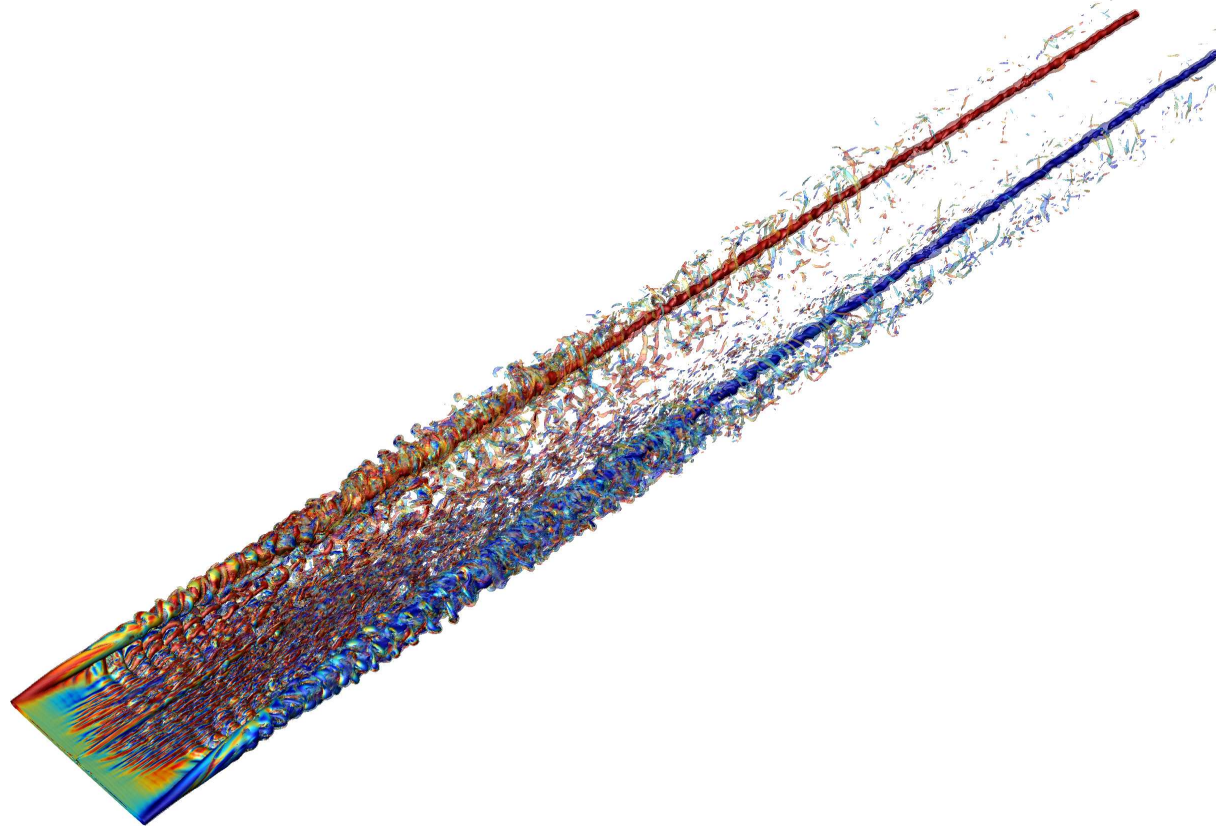


Figure A.1: Space-developing simulation of the vortex-sheet rollup with velocity deficit using an inflow: 3D perspective view of two isocontours of the vorticity norm $\|\boldsymbol{\omega}\|_{t_0} = [200\ 400]$ colored with the axial vorticity component.

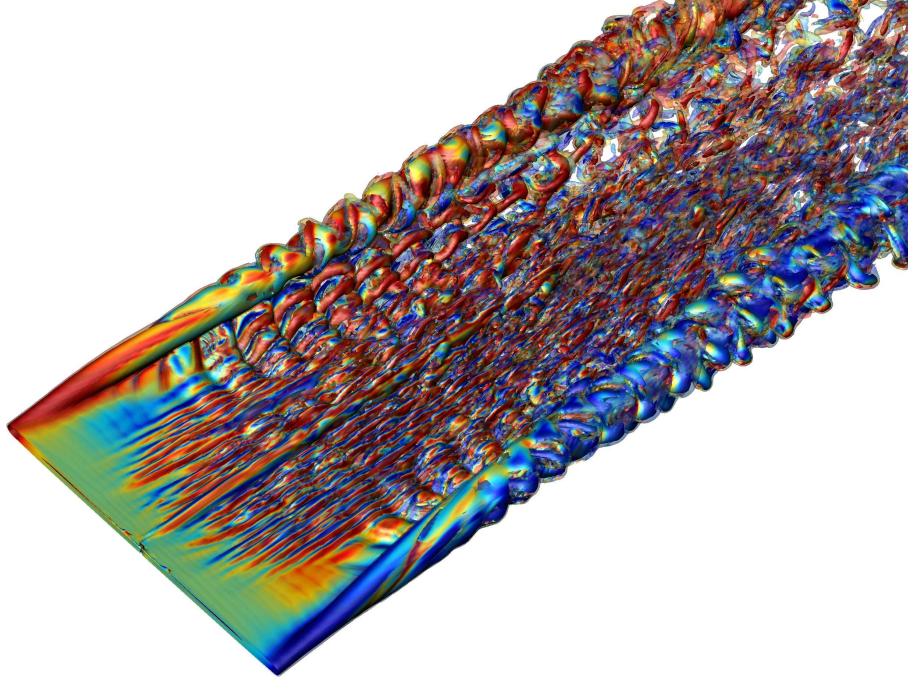


Figure A.2: Space-developing simulation of the vortex-sheet rollup with velocity deficit using an inflow: zoom in the near-wake region of the 3D perspective view of two isocontours of the vorticity norm $\|\boldsymbol{\omega}\|_{t_0} = [200\ 400]$ colored with the axial vorticity component.

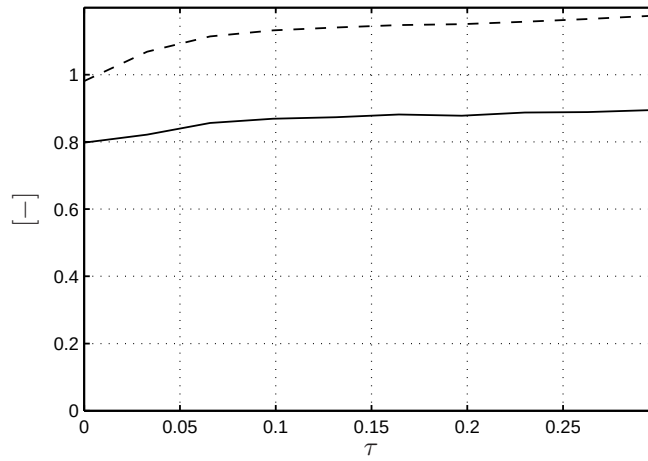


Figure A.3: Space-developing simulation of the vortex-sheet rollup with velocity deficit using an inflow: evolution of the linear-impulse $y_c \Gamma^+ / (\Gamma_0 b)$ (solid) and of the total circulation of the half-plane Γ^+ / Γ_0 (dash).

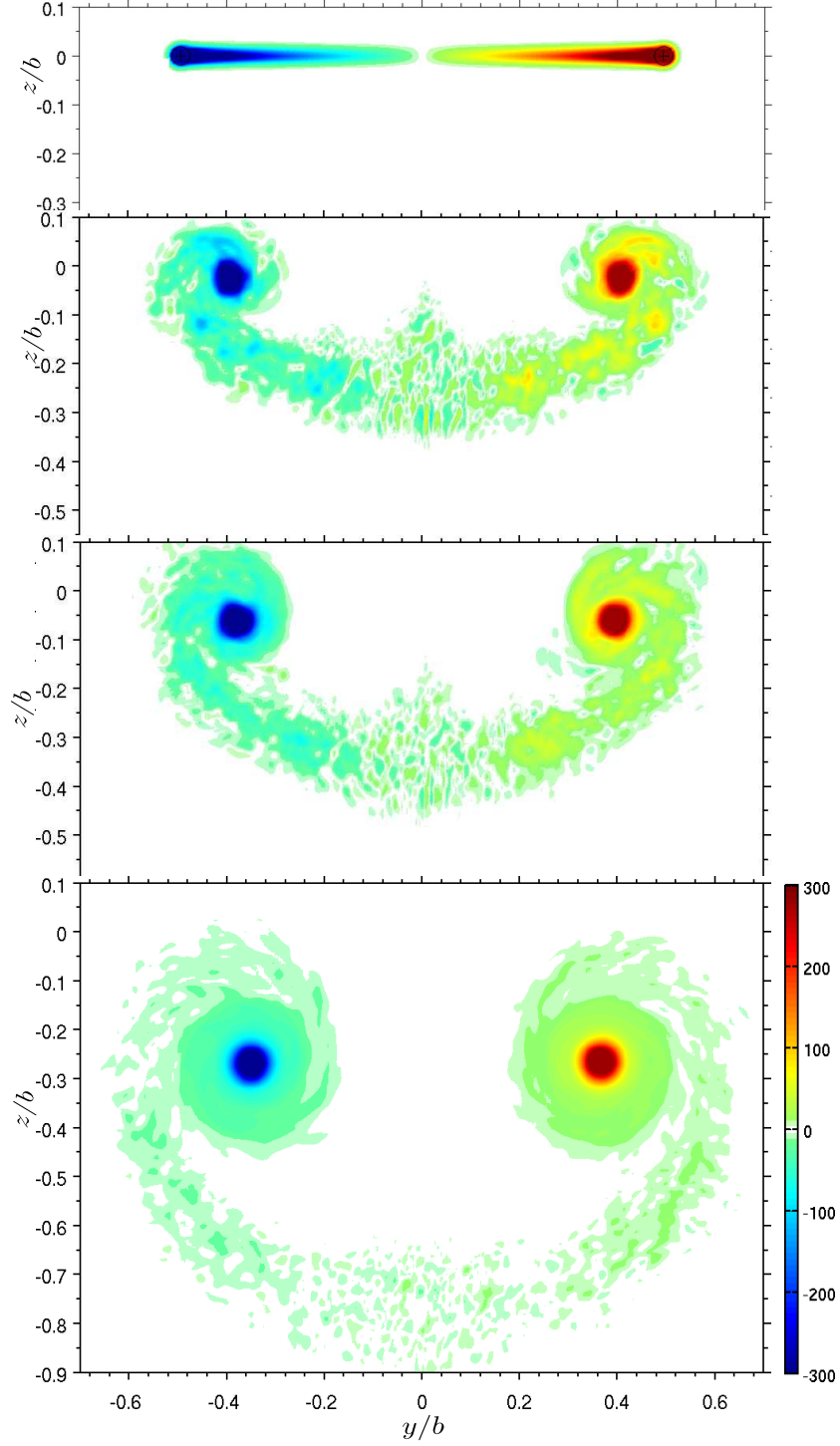


Figure A.4: Space-developing simulation of the vortex-sheet rollup with velocity deficit using an inflow: 2D averaged vorticity field $\overline{\omega_x t_0}$; from top to bottom, time $t = [0.0, 0.07, 0.1, 0.3]$.

Appendix B

3D filament simulations of Crow-like instabilities in ground effect

*This chapter was written within the FAR-Wake project as a technical report.
European project AST4-CT-2005-012238, Report TR3.1.1-1, February, 2006.
The authors are T. Lonfils, G. Daeninck and G. Winckelmans.*

B.1 Introduction

The purpose of this work is to investigate two vortex systems in ground effect (IGE) using a vortex filament method. The method being inviscid, the scope of this study is however reduced: 2-D and 3-D (DNS and LES) results clearly show that a very important effect for wake vortices (WV) IGE is the viscous interaction with the ground through the creation of secondary vortices. We therefore restrict the study to the analysis of long wave (Crow-type) instabilities for WV IGE.

After briefly describing the numerical method, we investigate two cases: (i) a configuration as close as possible to the one studied experimentally by CNRS-IRPHE (i.e., a thick vortex pair IGE with a forced long wave instability) and (ii) the case of aircraft like vortices IGE with a forced Crow instability (at

different initial perturbation levels).

B.2 Vortex filament method

In 3-D inviscid flows, vortex lines move as material lines: this constitutes the basis for the *method of vortex filaments*. For more details on the vortex filament method, and vortex methods in general, refer to [88, 87]. Each filament p corresponds to a vortex tube of circulation Γ_p : the strength associated to that filament. The method must be regularized, as singular filaments have a logarithmically infinite self-induced velocity everywhere their curvature is nonzero. For filaments using radially symmetric regularization functions, one has

$$\mathbf{u}_\sigma(\mathbf{x}) = -\frac{1}{4\pi} \sum_p \Gamma_p \int_{C_p} \frac{g_\sigma(|\mathbf{x} - \mathbf{x}_p|)}{|\mathbf{x} - \mathbf{x}_p|^3} (\mathbf{x} - \mathbf{x}_p) \times d\mathbf{x}_p$$

where the regularization function $g_\sigma(r) \triangleq g\left(\frac{r}{\sigma}\right)$ is such that $g(\rho) \rightarrow 1$ for ρ large and $g(\rho) \propto \rho^3$ for ρ small. The corresponding streamfunction is obtained as

$$\Psi_\sigma(\mathbf{x}) = \frac{1}{4\pi} \sum_p \Gamma_p \int_{C_p} G_\sigma(|\mathbf{x} - \mathbf{x}_p|) d\mathbf{x}_p \quad (\text{B.1})$$

with $G_\sigma(r) \triangleq \frac{1}{\sigma} G\left(\frac{r}{\sigma}\right)$ and $\frac{g(\rho)}{\rho^3} = -\frac{1}{\rho} \frac{dG}{d\rho}(\rho)$. This streamfunction also corresponds to the solution of $\nabla^2 \psi_\sigma = -\omega_\sigma$ with, as vorticity field,

$$\omega_\sigma(\mathbf{x}) = \sum_p \Gamma_p \int_{C_p} \zeta_\sigma(|\mathbf{x} - \mathbf{x}_p|) d\mathbf{x}_p \quad (\text{B.2})$$

where $\zeta_\sigma(|\mathbf{x}|) \triangleq \frac{1}{4\pi\sigma^3} \zeta\left(\frac{r}{\sigma}\right)$. One has that $g(\rho) = \int_0^\rho \zeta(s) s^2 ds$.

Typically, one uses parametric splines to represent numerically the filaments. The regularized method converges for regular vorticity fields when the number of filaments is increased, provided that the overlapping condition is satisfied. The regularization function has normalization $\int \zeta_\sigma(\mathbf{x}) d\mathbf{x} = 1$. It is of order r when it satisfies the moment properties $\int x_1^{p_1} x_2^{p_2} x_3^{p_3} \zeta_\sigma(\mathbf{x}) d\mathbf{x} = 0$ for $1 \leq p_1 + p_2 + p_3 < r$, and $\int |\mathbf{x}|^r |\zeta_\sigma(\mathbf{x})| d\mathbf{x} < \infty$. For radially symmetric functions, these become $\int_0^\infty \zeta(\rho) \rho^{2+s} d\rho = 0$ for s even and $2 \leq s < r$, and $\int_0^\infty |\zeta(\rho)| \rho^{2+r} d\rho < \infty$. Order higher than $r = 2$ calls for functions that are

not strictly positive. See, Table B.1 for usual functions. The *low order algebraic* (LOA) function has $r = 0$: it can only be used for global vortex tube modelling (as done in the present work), as it does not converge for detailed field discretizations (using many filaments per vortex tube).

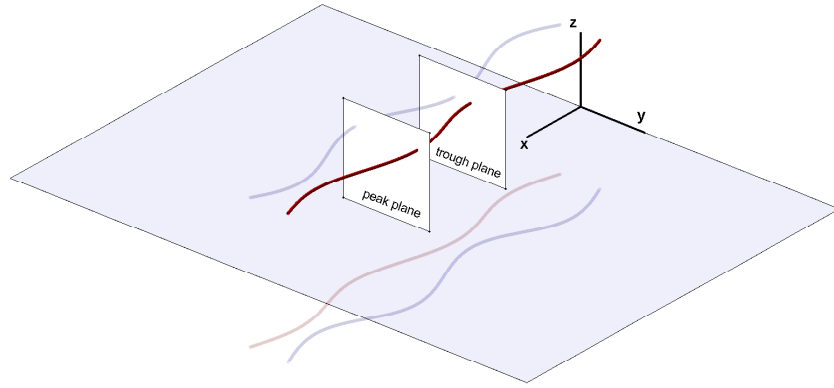


Figure B.1: Schematic of an infinite periodic filament and its image with respect to the vertical plane and to the horizontal plane (ground plane). The peak plane and the trough plane are also shown.

The filament method is here also “filtered”: indeed, if nothing is done, a high wave number spurious mode will pollute the simulation (and eventually make it blow up). Here, we use a discrete filter acting on the splines representation of the filaments (as in [88]).

Figure B.1 describes the geometry of the problem. The problem being periodic in the x -direction, we use a periodic version of the filament method. Planes of symmetry are forced following $\hat{O}xy$, to take into account the inviscid wall, and following $\hat{O}xz$ to ensure the symmetrical evolution of the vortices. We thus discretize and compute the evolution of one vortex filament only.

Table B.1: Examples of 3-D ($g(\rho)$ and $\zeta(\rho)$) regularization functions. The circulation $\Gamma(\rho)$ and vorticity $\omega(\rho)$ distributions for an equivalent 2-D vortex (i.e. straight 3-D filament) are also provided.

	$g(\rho)$	$\zeta(\rho)$	$\Gamma(\rho)/\Gamma_0$	$\pi\sigma^2\Gamma_0\omega(\rho)$	r
LOA	$\frac{\rho^3}{(\rho^2+1)^{3/2}}$	$\frac{3}{(\rho^2+1)^{5/2}}$	$\frac{\rho^2}{(\rho^2+1)}$	$\frac{1}{(\rho^2+1)^2}$	0
HOA	$\frac{\rho^3(\rho^2+5/2)}{(\rho^2+1)^{5/2}}$	$\frac{15/2}{(\rho^2+1)^{7/2}}$	$\frac{\rho^2(\rho^2+2)}{(\rho^2+1)^2}$	$\frac{2}{(\rho^2+1)^3}$	2
Gaussian	$\text{erf}(\rho) - \rho \frac{2}{\pi^{1/2}} e^{-\rho^2}$	$\frac{4}{\pi^{1/2}} e^{-\rho^2}$	$(1 - e^{-\rho^2})$	$e^{-\rho^2}$	2

B.3 Definition

One defines the Fourier transforms of the filament position in the plane $\hat{O}xy$ and $\hat{O}xz$ respectively:

$$\hat{y}_m = \frac{1}{N} \sum_{i=1}^N y(x_i) \exp^{-2I\pi mi/N}$$

$$\hat{z}_m = \frac{1}{N} \sum_{i=1}^N z(x_i) \exp^{-2I\pi mi/N}$$

where N the number of discrete points, $m = 1, 2, \dots, N/2$ and $I^2 = -1$. The amplitude of each mode is:

$$||\hat{a}_m||^2 = ||\hat{y}_m||^2 + ||\hat{z}_m||^2$$

The growth rate is then defined by:

$$\sigma^* \triangleq \sigma t_0 = \frac{d}{dt^*} (\log(||\hat{a}_m||))$$

where $t^* = t/t_0$ and t_0 is the characteristic time scale. This corresponds to the time for a pair of vortices characterized by an initial circulation Γ_0 to descend OGE a distance b_0 with the self-induced velocity $V_0 = \frac{\Gamma_0}{2\pi b_0}$.

B.4 Comparison: numerical simulation versus experiment

In the framework of FAR-wake Task 3.1.1, experiments have been performed by CNRS-IRPHE [18]. One of the aims of this work is to compare numerical results obtained using the filament method with experimental data. We first compare visualizations of vortex locations at different times. We then analyze the trajectories and temporal evolution of the vortex positions.

Experimental setup

The vortices are generated by two plates which are fixed to a common base and are able to rotate. The motion of the plates is properly chosen so as to have a spacing distance between the two vortices equals on average $b_0 = 2.4$ [cm]. In order to force a longwave instability, the edges of the plates describe a sinusoid. Here, the wavelength and the amplitude of this sinusoid is respectively $\lambda_f = 12$ [cm] $= 5.0 b_0$ and $a_0 = 1$ [mm] $= 0.0417 b_0$. The initial moment is here defined when the vortex spacing no longer varies. This corresponds to $h_0 = 6.2 b_0$. The experimental results show that the vortex circulation profile is relatively well fitted by a Gaussian vortex characterized by $\sigma_g = 0.17 b_0$.

Numerical setup

In order to compare the numerical and experimental results, the numerical parameters must be carefully chosen in order to match the experimental configuration. The two vortices are separated by a distance b_0 and are initialized at an altitude of $h_0 = 6.2 b_0$. The perturbation plane is put at 45° . In this plane, the perturbation is taken as:

$$a(x) = a_0 \sin \left(2\pi \frac{x}{\lambda_f} \right)$$

where $a_0 = 0.0417 b_0$ (it turns out that this initial amplitude produces results that best match those of the experiment, see later) and $\lambda_f = 5.0 b_0$. Notice that this wavelength is significantly shorter than the natural Crow instability (with $\lambda_C \simeq 8 - 9 b_0$); it is nevertheless the one that was forced in the CNRS-IRPHE

experiment and we here only aim at comparing with the experiment.

As mentioned above, the experimental vortex circulation profile can be fitted by a Gaussian vortex. However, the 3-D Gaussian regularization function is not implemented in the current filament code; this is because the Biot-Savart evaluation is expensive. To be able to compare two equivalent vortex evolutions, one uses vortices which have the same long wavelength dynamics.

In other words, in vortex filament methods, one can use vortices with another distribution function and require to obtain: the same translational velocity for thin vortex rings (Eq.B.3) and the same rotational velocity for long wavelength perturbations of a vortex tube (Eq.(B.4)). As shown below, those requirements are equivalent: the proper scaling fulfills both requirements.

- The self-induced translational velocity of a thin vortex ring ($\sigma/R \ll 1$) is given, for a regularized vortex by

$$U = \frac{\Gamma}{4\pi R} \left[\log \left(\frac{8R}{\sigma} \right) - C \right] \quad (\text{B.3})$$

with $C = 1$ for the low order algebraic function, $C = 1/2$ for the high order algebraic function and $C = (1 - \gamma/2) = 0.7114$ for the Gaussian function (where $\gamma = 0.5772$ is the Euler constant).

- The dispersion relation (self-induced rotational velocity) of long wavelength perturbations ($k\sigma \ll 1$) is given, for a regularized vortex by

$$\Omega = \pm \frac{\Gamma}{4\pi} k^2 \left[\left(\log \left(\frac{k\sigma}{2} \right) + \left(\gamma - \frac{1}{2} \right) \right) + C \right] \quad (\text{B.4})$$

with the same C values as above.

We here use the high order algebraic distribution and we scale σ to obtain the same dynamics as those of a Gaussian vortex.

Another error comes from the fact that we here discretize a vortex tube with only one vortex filament: because the velocity is computed at the center of the filament, it is not equal to the true translational velocity of the vortex tube. It can be corrected by using, again, an equivalent σ . Winckelmans [91] showed that using $C = 0.5580$ instead of 0.7114 in Eq.(B.3) and Eq.(B.4) leads to the correct dynamics of a Gaussian vortex characterized by σ_g .

Thus, in order to obtain the same dynamics as those in the experiment, one simply uses the filament method with a high order algebraic regularization and with σ_{hoa} such that:

$$\log \sigma_{hoa} + C_{hoa} = \log \sigma_g + C_g$$

with $C_g = 0.5580$, $C_{hoa} = 0.5$ and here with $\sigma_g = 0.17 b_0$. It leads to

$$\sigma_{hoa} = 1.060 \sigma_g = 0.180 b_0$$

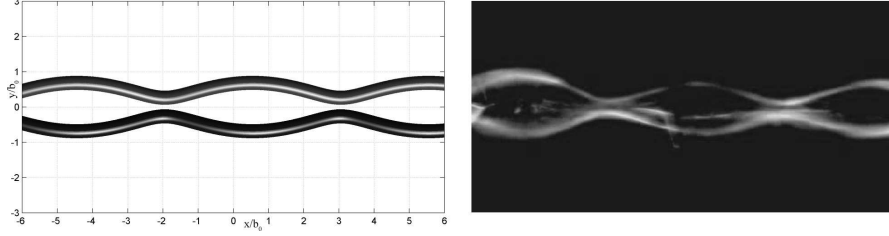
The vortex tube is represented by one vortex filament and is discretized by 320 points over λ_f , and cubic splines. The time step is $\Delta t^* = 10^{-2}/(2\pi)$. We use the same filter parameters as those detailed in [88].

Results

Comparison with the visualizations The qualitative vortex evolution is very similar to the experimental one. Figures B.2 and B.3 show the top and side views at $t^* = 2.7$ (during the linear growth phase) and $t^* = 3.8$ (at reconnection) respectively. Even if the Reynolds number of the experiment is not high ($Re \simeq 3500$ and thus there is some diffusion of the vortex core), the inviscid vortex filament method is in good agreement with the experimental results up to the reconnection phase: the position of the vortices and the amplitude of the instability are very similar.

Temporal evolution up to reconnection Experimental results are given for characteristic sections: the peak and the trough sections: i.e., the sections where the altitude are maximal and minimal respectively (Figure B.1). Up to the reconnection, trajectory, altitude and vortex spacing evolution are qualitatively similar as shown in Figure B.4.(a) and Figure B.5. The growth rate of the long wavelength instability is obtained as $\sigma^* = 0.66$, as shown in Figure B.4.(b).

(a)



(b)

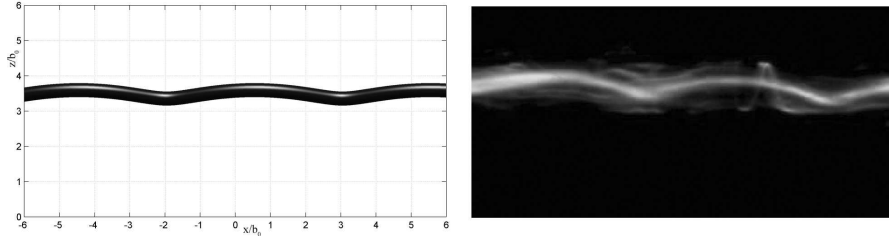


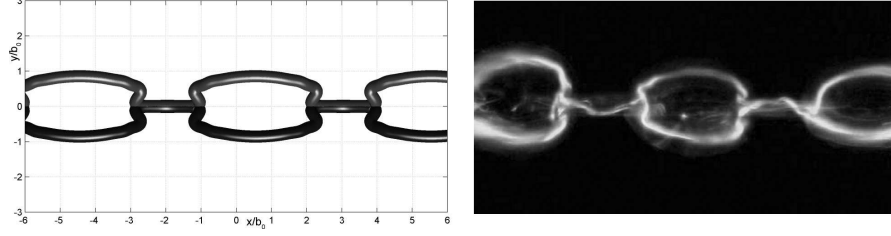
Figure B.2: Comparison of the vortex wake evolution: numerical case (left) at $t^* = 2.7$ and experimental case (right) at $t^* \simeq 2.7$. (a) top view and (b) side view, at same scale. The tube radius shown in the simulation is that for which the vortex induced velocity is maximal: $r = r_c = 1.121 \sigma_g$.

B.5 Analysis of the Crow instability in ground effect

Numerical simulations have been performed to investigate the behavior of the natural Crow instability in ground effect (before reconnection). Crow [20] has shown that the most unstable long wavelength mode of a pair of wake vortices ($\sigma/b_0 \ll 1$) has a wavelength: $\lambda = 8.6 b_0$. This mode grows in a plane inclined at about 48° to the horizontal. The growth rate is $\sigma^* = 0.83$. The purpose of this section is to investigate how the inviscid ground affects the Crow instability in terms of the deformation plane's angle and of the growth rate.

Numerical setup The following simulations are performed using the low order algebraic regularization function (see Table B.1) as it is a good fit usual for aircraft wake vortices. The value of σ is fixed to match a typical wake vortex core $\sigma = r_c = 0.05 b_0$. The vortices are initialized at an altitude out

(a)



(b)

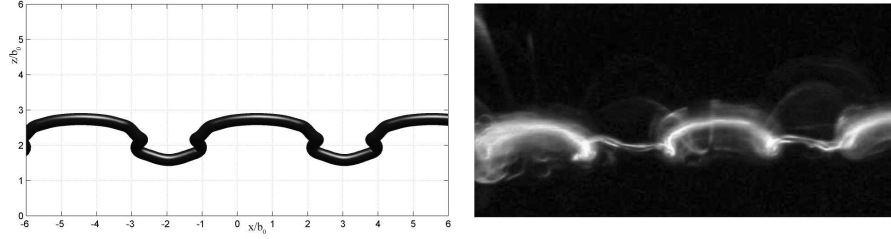
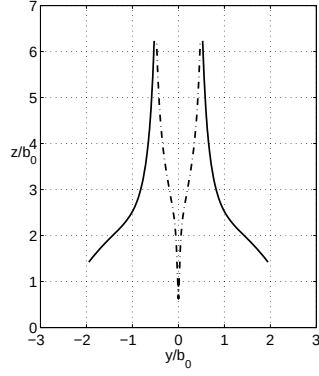


Figure B.3: Comparison of the vortex evolution: numerical case (left) at $t^* = 3.8$ and experimental case (right) at $t^* \simeq 3.9$. (a) top view and (b) side view, at same scale. In the simulation, the tube shown is that for which $r = r_c$.

(a)



(b)

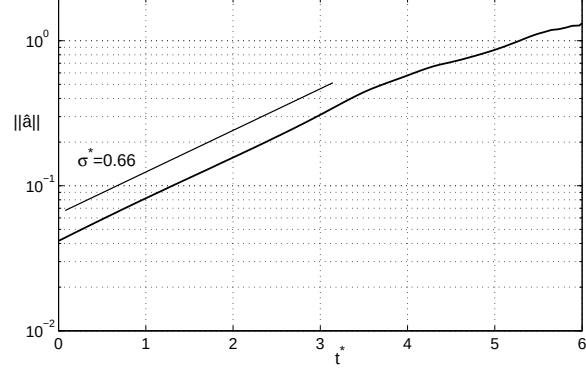


Figure B.4: Trajectory of the vortices (a) and evolution of the perturbation (b). The peak (solid line) and the trough (dash-dotted line).

of ground effect: $h_0 = 6b_0$. The perturbation is put in a plane inclined at 48.3° (accurate value determined by simulation carried OGE) to the horizontal and is initialized with a wavelength of $\lambda_C = 8.6b_0$. Different perturbation

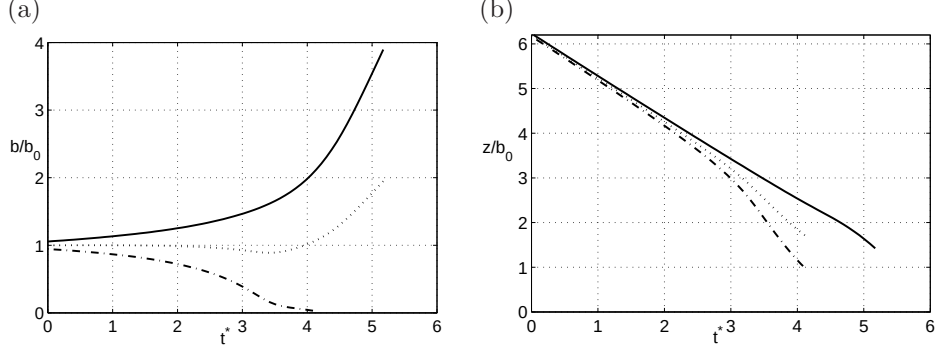


Figure B.5: Evolution of the spacing between the two vortices (a) and of the vortex altitudes (b). The peak (solid line), the trough (dash-dotted line) and the average (dotted line).

levels were investigated: $a_0/b_0 = [10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}]$. The filament is discretized using 512 points over λ_C , and cubic splines. The time step is $\Delta t^* = 5 \cdot 10^{-3}/(2\pi)$ and the filter parameters is again the same as those detailed in [88].

Instability analysis Figure B.6 and Figure B.7 show the visualization of the filament evolutions for two initial perturbation levels: $a_0/b_0 = 10^{-2}$ and $a_0/b_0 = 10^{-3}$ respectively. One can see that there are essentially two cases: (i) when the initial perturbation is high enough, the vortex filaments reconnect before entering IGE, as in Figure B.6 or (ii) for low initial perturbation level, the vortices enter IGE before reconnection and interact (and eventually reconnect) with their respective image.

Figures B.8 and B.9 describe the evolution of the amplitude and angle of the Crow mode for the different initial perturbation levels. Out of ground effect ($t^* < 4$), the growth rate of the Crow mode is $\sigma^* = 0.82$ and the perturbation angle is 48.3° . When the vortices are influenced by their respective images only ($t^* > 9$), the growth rate of the Crow mode is again $\sigma^* \simeq 0.82$; the perturbation angle is then -41.7° (thus 48.3° with respect to the vertical). Between these two phases, the angle of the perturbation plane changes (Figure B.9). Figure B.10 further shows this phenomenon: $\|\hat{z}_m\|$ changes sign while $\|\hat{y}_m\|$ slows down, stops, and then grows again.

As expected, if the initial perturbation level is low enough, the results show that the evolution of the Crow instability IGE is universal and independent of the initial perturbation amplitude. The rotation of the perturbation plane also happens at the same altitude $z/b_0 \simeq 0.53$ above the ground (here at the same time $t^* \simeq 7.13$ as all simulations were initially at $h_0/b_0 = 6$). Figure B.11 shows the trajectories of the vortices and the orientation of the perturbation planes. As expected, out of ground effect, and completely in ground effect, the instability characteristics (growth rate and angle) are those of the usual Crow instability.

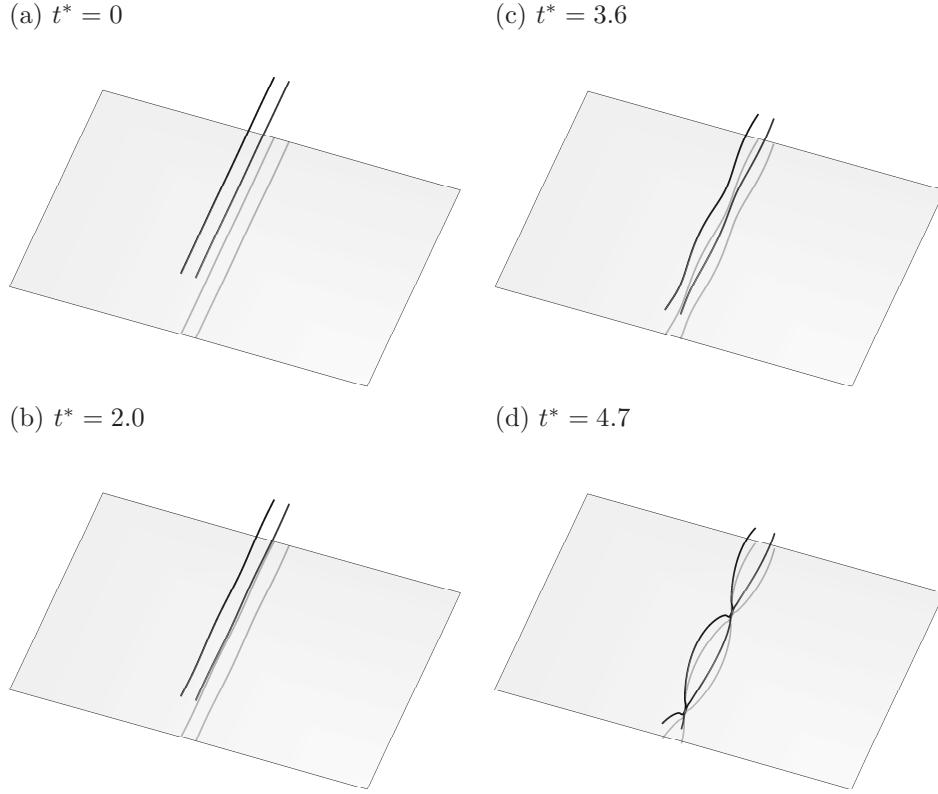


Figure B.6: Visualization of the vortex wake evolution. The shadow is also shown. The initial perturbation level is: $a_0/b_0 = 10^{-2}$.

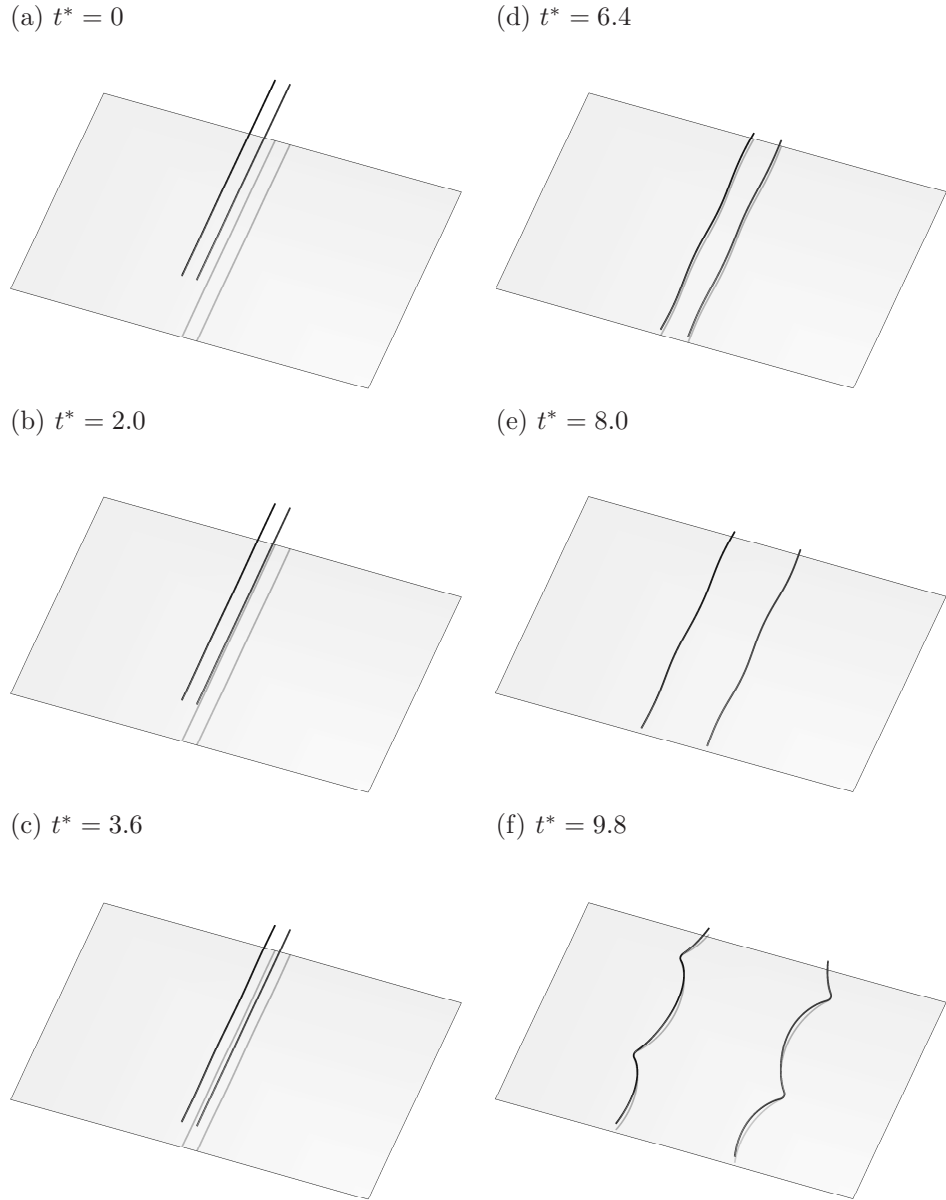


Figure B.7: Visualization of the vortex wake evolution. The shadow is also shown. The initial perturbation level is: $a_0/b_0 = 10^{-3}$.

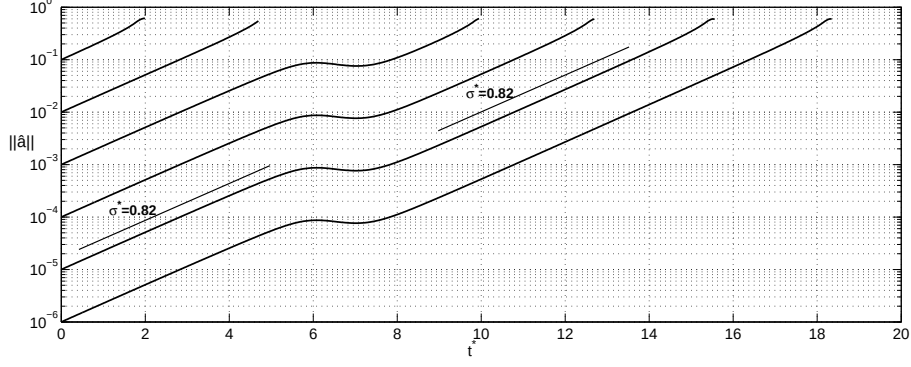


Figure B.8: Temporal evolution of the amplitude for the Crow mode $\lambda_C/b_0 = 8.6$ for different initial perturbation levels.

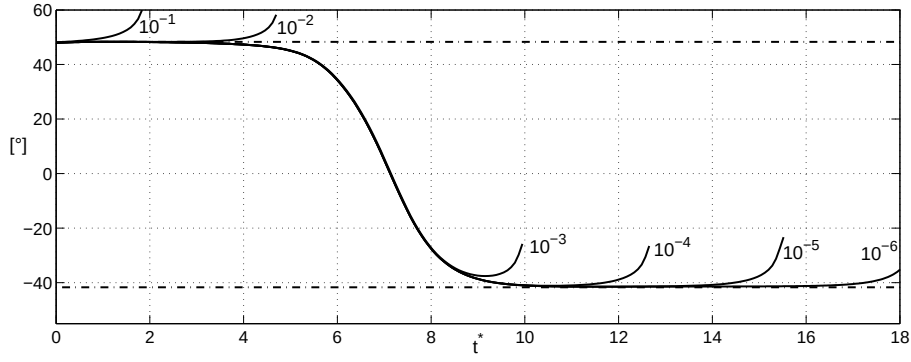


Figure B.9: Temporal evolution of the perturbation plane angle for each initial perturbation level a_0/b_0 . The time at which the orientation changes sign is $t^* \simeq 7.13$, which corresponds to $z/b_0 = 0.53$.

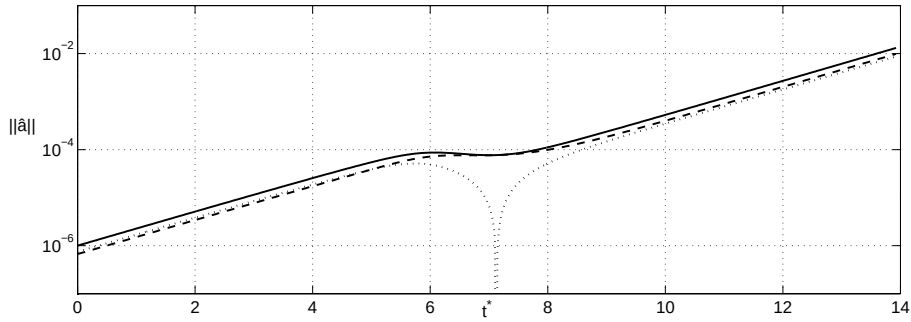


Figure B.10: Temporal evolution of the amplitude for the Crow mode $\lambda_C/b_0 = 8.6$. The initial perturbation level is: $a_0/b_0 = 10^{-6}$. $||\hat{a}||$ (solid line), $||\hat{y}||$ (dashed line) and $||\hat{z}||$ (dotted line).

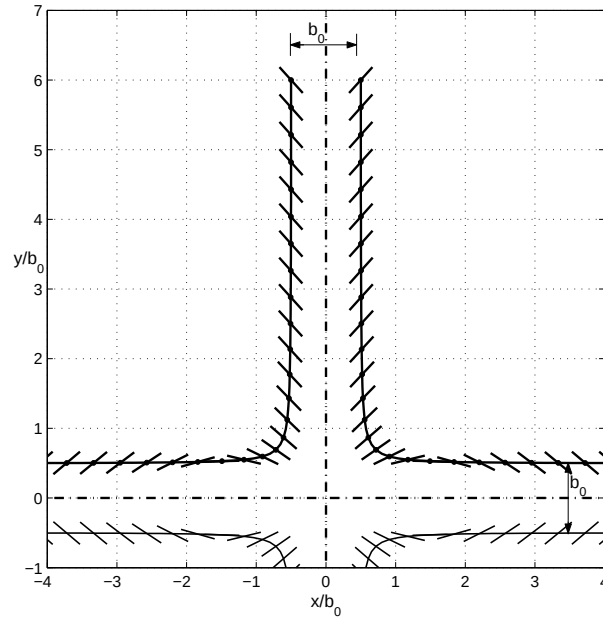


Figure B.11: Trajectories of the vortices and orientation of the perturbation planes (solid lines). Planes of symmetry (dash-dotted line). The time between each orientation shown is $\Delta t^* = 0.4$.