Università
di Genova

Department of Civil, Environmental and Architectural Engineering
University of Genoa

NUMERICAL SIMULATION OF THE UNSTEADY AERODYNAMICS OF FLAPPING FLIGHT

Joel GUERRERO

A thesis submitted for the degree of
Doctor of Philosophy

April 2009

This page intentionally left blank

Contents

List of Figures

List of Tables

Acknowledgements

Financial support of the Marie Curie actions EST project FLUBIO, through grant MEST-CT-2005-020228 is acknowledged.

I would like to specially thank my thesis advisor Prof. Alessandro Bottaro, for giving me the opportunity to conduct my doctoral studies at the University of Genova and for supervising my research work and for constantly keeping me busy with new challenging and interesting subjects.

I wish to express my sincere gratitude to all the people I met during these last three years, who in one or another way influenced on my decisions and helped me during this period. Just to name a few: Joao, Ana, Bart, Giovanni, Andreas, Damian, Julien, Houssam, Hakan, Laura, Thomas, Bill, Dominic, Anna, Antoine, Marco, Simona, Valeria, Stephano, Gaby, Ernesto, Max, Sam, Laly (and I am deeply sorry for those I do not remember at the moment).

My successes are largely due to the quiet support of my family. I thank my parents for supporting the choices that I have made in my life, I also thank my brother and sister for always being there. Finally, I cannot pass this page without thanking the universal architect.

E Pluribus Unum

Abstract

The aim of this dissertation is to contribute to a better comprehension of the mechanism of flapping airfoils/wings propulsion and the associated unsteady aerodynamics, independently of their possible practical applications. We describe an accurate and stable numerical method to numerically solve the incompressible Navier-Stokes equations, which, used together with the overlapping grids method and to the numerical tools implemented, constitutes a very powerful tool to solve fluid dynamics problems with fixed and moving/deforming boundaries in two and three space dimensions. The two-dimensional results are presented for airfoils undergoing heaving and coupled heaving-and-pitching motion. The interest here is to determine the values of flapping frequency and flapping amplitude best suited for flapping flight, in terms of maximum propulsive efficiency and thrust production. We also study the influence of airfoil cambering and airfoil flexibility on the aerodynamic performance. Finally, three-dimensional rigid finite-span wings undergoing heaving, coupled heaving-and-pitching and root-flapping motion modes are investigated, with focus on the wake topology and aerodynamic performance, and their dependence on the flapping motion parameters. We also establish the best criteria for vortical structures identification and assess whether the assumption of two-dimensionality has some validity in three-dimensional cases.

Chapter 1

Introduction

1.1 Overview

Biologists, naturalists and bioengineers, all agree that nature relies on reciprocating motions for locomotion and propulsion on land, in the air and sea. Legs for walking, flapping wings for flight and oscillating fins and tails for swimming [4, 21]. Tens, even hundreds of millions of years of evolution have led to the refined forms and motions we see today in birds, insects, fishes, sharks and cetaceans. Biologists, aerodynamicists, engineers and the general public, look on with wonder at the ease with which these creatures stay aloft, propel themselves, navigate and manoeuvre.

Over a million different species of insects fly with flapping wings, and of the living 13000 warm-blooded vertebrate species (*i.e.*, birds and mammals), 10000 types of birds and bats flap their wings to generate propulsion in the skies [165, 167]. Flapping wings for flying and oscillating fins for swimming stand out as one of the most complex yet widespread propulsion methods found in nature. Although aeronautical technology has advanced rapidly over the past 100 years, natural flyers, which have evolved over millions of years, are still impressive and represent one of nature's finest locomotion experiments. Considering that humans move at top speeds of about 5 body lengths per second, a race horse runs approximately 7 body lengths per second, a cheetah accomplishes 18 body lengths per second, a wide body commercial aircraft such as the Boeing 747 flying at top speed (910 kph) achieves 3.6 body lengths per second, a supersonic military aircraft such as the SR-71, traveling near Mach 3 covers about 32 body lengths per second or as the supersonic commercial aircraft, *"Concorde"*, flying at maximum cruise speed (Mach 2.0) only reaches about 10 body lengths per second; it is amazing that a common pigeon frequently attains speeds of 80 kph, which converts to 75 body lengths per second, a starling can travel at 120 body lengths per second, a swan can reach 23 body lengths per second, the desert locust travels at 180 body lengths per second and a common house fly, flying at 3 meters per second can travel 430 body lengths per second [3, 21, 156]. The roll rate of highly aerobatic aircrafts (*e.g.*, the MXS single place aerobatic airplane) is approximately 420 degrees per second, and a barn swallow has a roll rate in excess of 5000 degrees per second [167]. The maximum positive G forces permitted in most general aviation aircrafts is about 4 to 5 G's, select military aircrafts can withstand 8 to 10 G's and high performance aerobatic aircrafts support up to 14 G's. However, many birds routinely experience positive G forces in excess of 10 G's and up to 14 G's [156, 165, 167]. The primary reasons for such superior maneuvering and flight characteristics include the scaling laws (such as low stall velocity, low inertia and low weight) with respect to man-made vehicle's size, as well as intuitive but highly developed sensing, navigation and control capabilities. Quoting McMasters

and Henderson [119]: *"Humans fly commercially or recreationally, but animals fly professionally"*.

To gain the benefit of that evolutionary refinement, nature may serve as the inspiration for novel techniques (in human terms) to enhance or supplant traditional sources of propulsion (propellers in ships and submersibles, jet engines and propellers in aircrafts and rotors in helicopters) and lift generation mechanisms (fixed wings and helicopter rotors) in man-made vehicles.

Recently, the engineering community (particularly the aerospace field) has seen renewed interest in the low Reynolds number aerodynamics of flapping wings and hydrodynamics of oscillating fins, and this is chiefly due to the growing interest of developing Micro-Air-Vehicles (MAVs), Autonomous-Underwater-Vehicles (AUVs) and more recently, Nano-Air-Vehicles (NAVs) [43]. These vehicles may use or take advantage of such unconventional propulsion and lift generation methods, in order to achieve better performances than traditional methods.

MAVs are flying vehicles with a maximal dimension of 15 cm or less (which is comparable to the size of small birds or bats), and capable of reaching flight speeds around 10 to 20 meters per second [127]. These vehicles can perform surveillance and reconnaissance missions, sensing at remote or hazardous locations, traffic monitoring, forestry and wildlife surveys, inspection of power lines and aerial photography, among other tasks [120]. MAVs experience the same low Reynolds number as their biological counterparts (typically in the order of 10^3 to 10^5); in this regime fixed wings drop dramatically in aerodynamic performance. At these low Reynolds number values, the fluid flow is prone to separation, resulting in increased drag and loss of efficiency. Even without flow separation, the low Reynolds number results in low lift-to-drag ratio due to the thickness of the boundary layer. It becomes clear that, in order to develop practical MAVs, new ways of generating lift and thrust must be investigated with the aim of overcoming the drawbacks of fixed wings at low Reynolds number.

Both fixed and flapping wings have been explored in the development of MAV vehicles [127]. Fixed wings, turbines and propellers become less efficient as the size and speed of the vehicle (and hence the Reynolds number) decrease. Viscous drag increases due to relatively thicker boundary layer, and flow separation causes loss of lift and increased pressure drag. With the intention of overcoming the problems of fixed wings at low Reynolds number, flapping wings are being actively studied with the hope that they might provide better performance at this flight regime [165]. The use of flapping wings as an alternative to fixed wings is motivated by the observation of the flight of birds and insects, which use flapping wings not only to overcome the difficulties of low Reynolds number, but to exploit the associated aerodynamic phenomena. Traditional aircraft design using fixed wings attempts to ensure that flow stays attached to the airfoil (unstalled) at all times. In contrast, flapping wings rely on vortex separation from the trailing and leading edges of the wings, forming in this way low pressure regions that may be used to create higher lift and thrust than is possible with fixed wings [44, 46].

Several researchers [7, 136, 152, 166, 182, 191, 193, 207], have found that flying and swimming animals cruise at Strouhal numbers (St) corresponding to a regime of vortex growth and shedding in which the propulsion peaks it maximum efficiency. The St is a dimensionless parameter that describes the wing (or tail) kinematics of flying (or swimming) animals and is defined as $St = fA/U$ (where f is the flapping frequency, A the peak to peak amplitude of the flapping stroke and U is the forward velocity). Because natural selection is likely to tune animals for high propulsive efficiency, we expect it to constraint the range of St that natural swimmers and flyers

use. Triantafyllou *et al.* [191, 193], Anderson *et al.* [7] and Read *et al.* [152], found that most of the fast swimming fishes oscillate their tails with $0.2 < St < 0.4$, regardless of the size. Wang [207], Taylor *et al.* [182] and Nudds *et al.* [136], also observed that flying animals, converge on the same narrow range of St for cruise conditions (see figure 1.1). In this range, the propulsive efficiency (defined as the ratio of aerodynamic power output to mechanical power input) can be as high as 70% [182]. Optimal St depends subtly on kinematic parameters including angle of attack, amplitude-to-chord ratio, airfoil section, beating frequency and phase of motion.

Hence, it becomes evident that gaining a better understanding of the relationships between the forces produced and the flapping parameters, including St, and the wing motions driving the leading and trailing edge vortex separation, the manner in which the vortices interact with the airfoil and themselves, how they contribute to lift and propulsion and how to optimize the process, would aid in better understanding the propulsion mechanism of birds, insects and fishes for the design of lighter, more efficient and more maneuverable new generation of MAVs or similar applications.

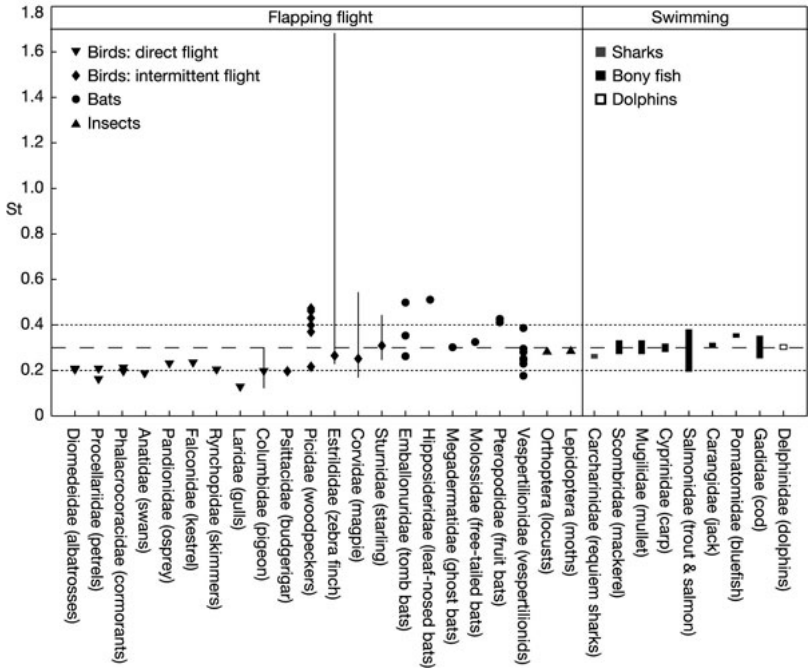

Figure 1.1: *Published ranges (taken from [182]) of St for cruising birds, bats, fishes, sharks and dolphins. Dotted lines mark the range* $0.2 < St < 0.4$*, in which propulsive efficiency usually peaks; dashed line marks the modal peak at* $St = 0.3$*. Unbroken lines indicate the range of variation in St across other non-zero flight speeds, where such data exist.*

Apart from the biomechanical design aspects, further motivation may be found in understanding the aerodynamics and hydrodynamics of natural flyers and swimmers. These areas of research include:

- The unsteady aerodynamics of helicopter rotor blades undergoing cyclical flapping motions.

- Flow control and flow reattachment through shape control.
- Blade-vortex interaction for noise prevention in rotorcrafts.
- Interaction and decay of vortex wakes to minimize hazards to aircrafts.
- Fluid-structure interactions and aerodynamic flutter.
- Wake vortex dynamics.

This dissertation investigates the aerodynamics of low Reynolds number flapping airfoils/wings, from the perspective of efficient thrust generation and propulsion efficiency (whether for better understanding the aerodynamics of natural flyers, the hydrodynamics of natural swimmers, or for MAV/AUV applications). It mainly deals with the aerodynamics of two-dimensional flapping rigid and flexible/deforming airfoils (as many of the phenomena of interest such as; thrust generation, inversion of the vortex street and leading edge vortex separation, can be captured with two-dimensional simulations), and to a lesser extent with the aerodynamics of three-dimensional flapping rigid wings. This limitation is mainly due to the unsteady nature of the flow and the requirement for high resolution in regions of flow separation, vortex shedding and vortex wake evolution, which added to the requirement of bigger and finer computational domains for three-dimensional simulations, would create an impractical computational load given the current resources available. As noted by Wang [207], *"a two-dimensional computation can serve both as a reliable tool in its own right and a useful reference point to be compared with three dimensional simulations"*.

The current work is fully based on numerical simulations. The governing equations of fluid dynamics and the highly unsteady aerodynamics of the flapping motion are solved using a Navier-Stokes flow solver on overlapping grids. The use of unsteady potential methods [86, 93, 94, 177, 183, 212] is not consider, this is chiefly due to the limitations imposed by these methods of assuming inviscid flow and flow separation only from the trailing edge of the airfoil (as imposed by the Kutta condition in potential methods), hence better resolution and representation of the flow physics is obtained by using Navier-Stokes solvers, but with the drawback of being more time consuming and computational expensive. Finally, the obtained results are interpreted and analyzed and where is possible, are compared against other experimental and computational results found in the literature in order to explain the observed phenomena and to characterize the parameters governing the mechanism of flapping wings propulsion, in terms of thrust production and propulsive efficiency.

1.2 Objectives

The objectives of this dissertation are, from the biomechanical point of view of flapping wings propulsion:

- the study of the mechanisms of thrust generation by flapping airfoils/wings at low Reynolds number, in the range typical of birds, large insects and potential MAV, NAV and UAV in order to:
  - explain the experimentally and numerically observed wake structures for flapping airfoils;

  - determine the relationships between thrust production and propulsive efficiency, wake structure, wing geometry and flapping parameters such as frequency, amplitude of motion (maximum heaving and pitching amplitude) and phase angle;

- compare the performance parameters and wake structures of the flapping motion of rigid and flexible/deforming airfoils;

- determine the relative importance of leading and trailing edge vortex shedding in the generation of the wake structures and aerodynamics forces;

- to examine the validity of a single parameter (Strouhal number), as the fundamental aerodynamic parameter insofar as high propulsive efficiency is concerned;

and from the computational point of view:

- to propose the best methodology to efficiently handle moving and deforming bodies;

- to compute efficient solutions in terms of problem definition (geometry, kinematics, Reynolds number, Strouhal number and so on) and computing time.

1.3 Outline of the dissertation

In order to meet the objectives stated above, this dissertation is divided into nine chapters. In chapter 1, a short introduction and an outline of the objectives of this dissertation are given.

In chapter 2, a literature review on the aerodynamics of low Reynolds number flapping airfoils is presented. This covers their use in nature, some experimental observations, various analytical and numerical techniques, a review of nonstationary airfoil aerodynamics including dynamic stall, vortex shedding and thrust generation, a presentation of flapping flight in terms of Reynolds number, Strouhal number and reduced frequency and a discussion of flapping wings performance parameters and flapping wings kinematics.

Chapter 3, presents the governing equations of fluid dynamics and their nondimensionalization; followed by a description of their transformation to generalized curvilinear coordinates and a presentation of the governing equations for the case of an incompressible viscous flow.

In chapter 4, the structured overlapping grids method is reviewed and discussed in the context of a method for the efficient solution of the governing equations around complex geometries and moving/deforming bodies.

Chapter 5, describes the numerical method used to solve the governing equations on overlapping grids. The numerical method presented is a split-step scheme, second-order accurate in space and time and solves the momentum equations for the velocity together to a Poisson equation for the pressure (the so called pressure-Poisson equation or PPE), this system of equations is known as the velocity-pressure formulation of the incompressible Navier-Stokes equations.

In Chapter 6, a qualitative and quantitative validation and verification of the proposed computational tool against experimental and numerical results is carried out in order to assess its

numerical accuracy. Also in this chapter, a grid dependency study is conducted in order to determine the best suited grid for the computations to be performed in the next chapter.

In chapter 7, we present several two dimensional results for heaving and coupled heaving-and-pitching motions. The interest here is to determine the values of flapping frequency and flapping amplitude best suited for flapping flight, in terms of maximum efficiency and thrust production. We also study the influence of airfoil cambering and airfoil flexibility on lift and thrust generation.

In chapter 8, we extend the two-dimensional results presented in chapter 7 to three-dimensional rigid finite-span flapping wings. In this chapter, we investigate the wake topology behind low aspect ratio flapping wings and their dependence on the Strouhal number and flapping parameters. We also present some results on the aerodynamic performance of flapping wings, as well as we establish the best criteria for vortical structures identification.

Finally, we conclude by presenting the major conclusion and future perspectives in Chapter 9.

Chapter 2

Aerodynamics of Flapping Flight

Flying animals flap their wings to generate lift and thrust as well as to perform remarkable maneuvers with rapid accelerations and decelerations. Insects, bats and birds provide illuminating examples of unsteady aerodynamics. In this chapter we present the various issues related to the aerodynamics of flapping flight. We first present flapping flight in nature. We next review both analytical and computational models and some experimental observations. Then, we present a review of nonstationary airfoil aerodynamics including dynamic stall, vortex shedding and thrust generation; followed by a presentation of flapping wing flight in terms of Reynolds number, Strouhal number and reduced frequency. Finally, we close this chapter with a discussion of flapping wings performance parameters and flapping wings kinematics.

2.1 Flight in Nature

The fundamentals of bird flight are similar to those of aircrafts. As the wings move through the air, they are held at a slight angle, which deflects the air gently downward. This causes air pressure to build up beneath the wings, while the pressure above the wings is reduced. The difference in pressure produces lift, a force that acts roughly perpendicular to the wing surface and keeps the bird or airplane from falling. Generally, bird flight can be divided into two modes of functioning, *i.e.*, unpowered flight (gliding and soaring flight) and powered flight (flapping and hovering flight).

When a bird is gliding, the wings are held out to the side of the body and do not flap. Lift force is produced by the action of air flow on the wings. The lift force occurs because the air has a lower pressure just above the wings and higher pressure below. But there is also air resistance or drag on the body and wings of the bird. This force would eventually cause the bird to slow down, up to the point where it would not have enough speed to fly. To make up for this, the bird can lean forward a little and go into a shallow dive. In that way, the lift force produced by the wings is angled forward slightly helping the bird to speed up. Really what the bird is doing here is giving up some height in exchange for increased speed (or putting it in another way, it is converting its gravitational potential energy into kinetic energy).

An alternative method to gliding used by many biological flyers to produce lift and thrust, is flapping wing flight. Flapping flight is a far more complicated process than gliding. During flapping flight, the bird's wings systematically change shape. Flapping involves up and down movement of the wings. During the downstroke (or power stroke), the wings move downward and forward. During the upstroke (or recovery stroke), the wings move upward and drawn in toward the body

to reduce drag. During flapping flight, the wings also change their angle of attack depending on the stroke. Flapping flight is basically rowing in the air with the added complication that lift needs to be generated as well.

2.1.1 Unpowered Flight: Gliding and Soaring

Flying animals usually flap their wings to generate both lift and thrust. But if they stop flapping and keep their wings stretched out, their wings actively produce only lift, nor thrust. Thrust can be produced by gravity force while the animal is descending. When this happens, we call them gliders. Many gliding birds (and soaring birds as well) appear to hang in the air effortlessly, gaining height with barely a twitch of a wing. These are birds like vultures, albatrosses, pelicans and storks with a high lift-to-drag ratio. Essentially, this means that their wings generate a lot of lift without producing much drag. Large birds have evolved to be gliders partly because gliding becomes easier the larger your wings are and obviously small birds can not have large wings. In addition to birds, gliders can also be found among bats, fish, amphibians, reptiles, and mammals [167].

Figure 2.1: *A bird while gliding. Notice the separation between the wingtip feathers; these natural slots, help to reduce the induced drag while gliding.*

To maintain level flight, a flying animal must produce both lift and thrust to balance the gravity force in the vertical direction and drag in the horizontal direction respectively. Because gliding occurs with no active thrust production, an animal always resorts to the gravity force to overcome the drag. In gliding, the animal tilts its direction of motion slightly downward relative to the air that it moves through. When the animal tilts downward, the resulting angle between the motion direction and the air becomes the gliding angle. The gliding angle directly controls the lift-to-drag ratio. The higher this ratio, the shallower the glide becomes. The lift-to-drag ratio increases with the Reynolds number, a parameter proportional to animal size and flight speed. Large flying animals fly at high Reynolds numbers and have a large lift-to-drag ratio. For example, a wandering albatross, with a wing span of over 3 meters, has a reported lift-to-drag ratio of 19, whereas the fruit fly, which has a span of 6 millimeters, has a lift-to-drag ratio of 1.8

[3, 200]. If the animal has a low lift-to-drag ratio, it must glide (if it can) with a considerably large glide angle. For example, the North American flying squirrel has a glide angle of about 18 to 26 degrees with a lift-to-drag ratio of 2 to 3 [3, 167].

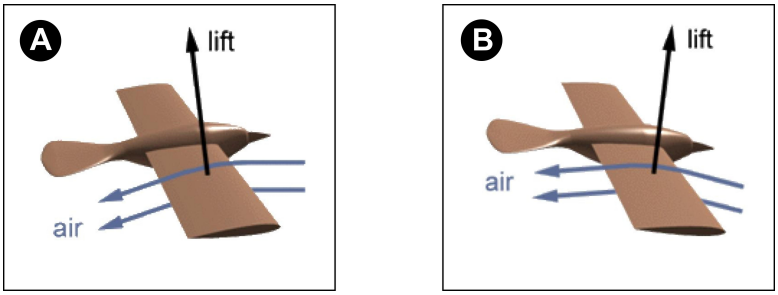

Figure 2.2: *In gliding flight, a bird's wing deflect air downward, causing a lift force that holds the bird up in the air (see figure A). By tilting forward and going into a slight dive (figure B), the bird can maintain forward speed.*

Gliding flight always results in a bird moving downward through the air. In order to maintain or gain height, birds resort to soaring (see figure 2.3). Soaring flight is a special kind of glide, in which the bird flies into a rising air current. Because the air is rising, the bird can maintain its height relative to the ground without the need of flapping its wings. Instead of using gravity, soaring uses energy in the atmosphere, such as rising air current.

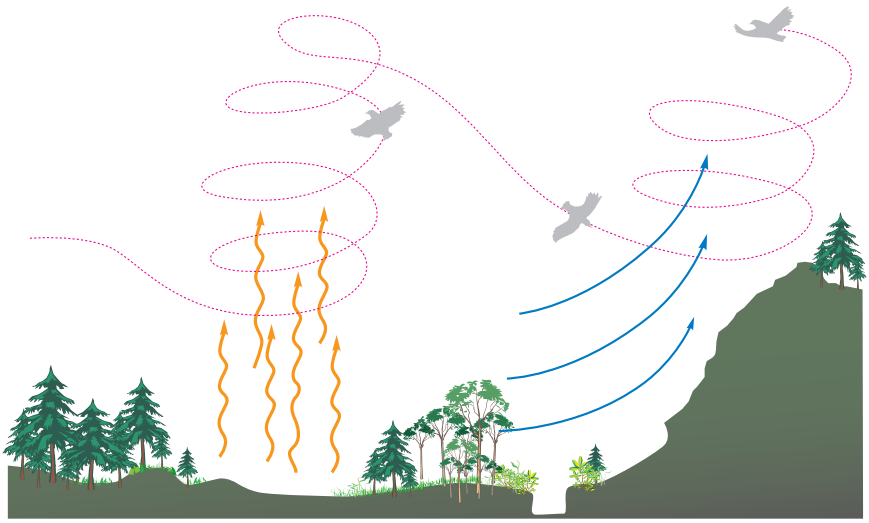

Figure 2.3: *In soaring flight, birds use both the updraft thermals and orographic lifting to maintain or gain altitude and save energy.*

2.1.2 Powered Flight: Flapping

Flapping flight is more complicated than flight with fixed wings because of the structural movement and the resulting unsteady fluid dynamics. Conventional airplanes with fixed wings are, in comparison, very simple. The forward motion relative to the air causes the wings to produce lift. However, in flapping flight the wings not only move forward relative to the air, they also flap up and down, bend, twist and sweep.

Figure 2.4: *A Mallard in powered flight (flapping flight).*

When a bird flaps, as opposed to gliding, its wings continue to develop lift as before, but they also create an additional forward and upward force, thrust, to counteract its weight and drag. Flapping involves two stages: the downstroke or power stroke, which provides the majority of the thrust, and the upstroke or recovery stroke, which can also (depending on the bird's wings) provide some upward force. At each upstroke the wing is slightly folded inwards to reduce upward resistance. Birds change the angle of attack between the upstroke and the downstroke of their wings. During the downstroke the angle of attack is increased, and is decreased during the upstroke.

When the wings move up and down, they are also moving forward through the air along with the rest of the bird. Close to the body, there is very little up and down movement. Farther out toward the wingtips, there is much more vertical motion. As the bird is flapping along, it needs to make sure it has the correct angle of attack all along its wingspan. Since the outer part of the wing moves up and down more steeply than the inner part, the wing has to twist (and bird's wings are very flexible), so that each part of the wing can maintain just the right angle of attack. As the wing twists, and as the outer part of the wing moves downward, the lift force in the outer part of the wing is angled forward. This is what would happen if the whole bird went into a steep dive. However, only the wing is moving downward, not the whole bird. Therefore the bird can generate a large amount of forward propulsive force without any loss of altitude. During this stroke, the air is not only deflected downward, but also to the rear. The air is forced backward just as it would be by a propeller.

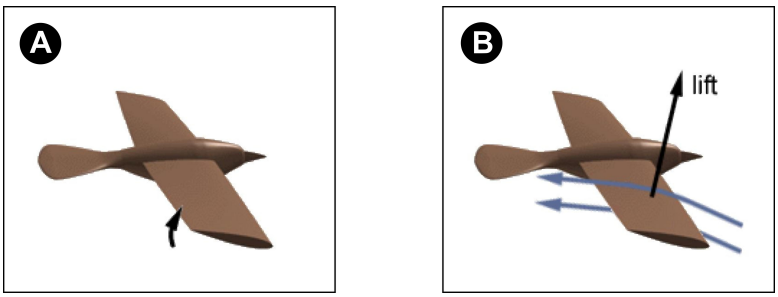

Figure 2.5: *In A, the wings twist as shown to maintain the correct angle of attack for the downstroke. In B, the bird's wings produce lift and thrust during the downstroke.*

During the upstroke, the outer part of the wing points straight along its line of travel so it can pass through the air with the least possible resistance. In other words, the angle of attack is reduced to zero. The bird partially folds its wings, which reduces the wingspan and eliminates the draggy outer part of the wing (this is not strictly necessary though, and most insects lack this capability). Also, the primaries (wingtip feathers) separate, these natural slots, allow passage of air through them, reducing in this way the skin friction.

The inner part of the wing is different. There is little up-and-down movement there, so that part of the wing continues to provide lift and functions more or less as it would when gliding. Because only the inner part of the wing produces lift in the upstroke, the upstroke as a whole offers less lift than the downstroke. As a result, the bird's body will bob up and down slightly during flight.

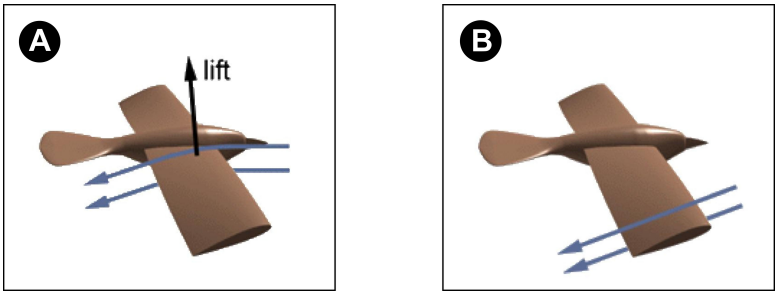

Figure 2.6: *In A, the inner part of the wing produces lift, even during the upstroke. In B, the outer part of the wing is angled to pass through the air with little resistance.*

What we have outlined so far is a basic description of how birds fly, when they are already in the air and cruising along. Birds also have other flying techniques, which they use when taking off or landing, or for other special maneuvers like hovering, as we will see later.

Birds, bats and insects apply a variety of different flapping patterns in hovering and forward flight to generate lift and thrust. Larger birds have relatively simple wingtip paths. For example, an oval tip path is often associated with albatrosses. Smaller flyers exhibit more complicated flapping patterns. Figure 2.7 illustrates some of these patterns for two natural flyers. In figure 2.9, hummingbird's wing lying eight pattern is shown.

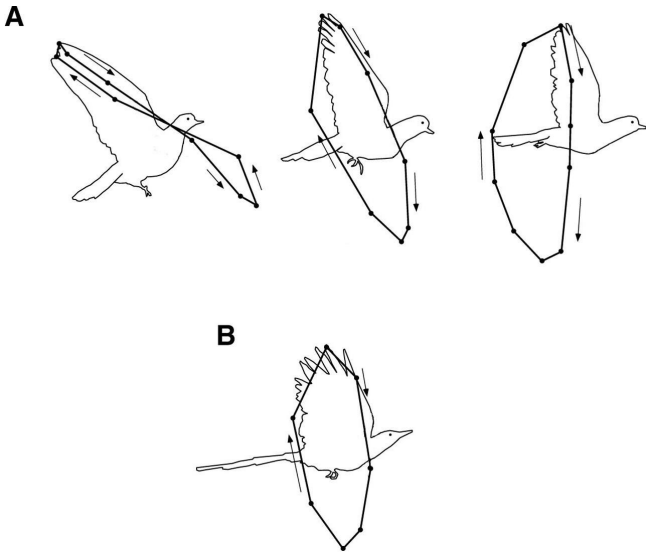

Figure 2.7: *Wingtip paths relative to the body for two natural flyers. (A) Pigeon (Columba Livia), here we see the path transition from tip-reversal upstrokes during slow flight to feathered upstrokes at intermediate speeds and a swept-wing upstroke during fast flight. (B) Black-billed magpie (Pica Hudsonica) wingtip path at all flight speeds [187].*

2.1.3 Hovering

Hovering is used by several species of birds. Hovering, which is generating only lift through flapping alone rather than as a product of thrust, demands a lot of energy. Whether a flying animal can hover or not depends on its size, moment of inertia of the wings, degrees of freedom in the movement of the wings and the wings shape [167]. As a result of these limitations, hovering is mainly performed by small birds and insects. The largest bird able to truly hover is the pied kingfisher, although larger birds can hover but for short periods of time [3, 167, 200].

Large birds can also hover and they do so in an artificial way by flying into a headwind, allowing them to utilize thrust to fly slowly but remain stationary to the ground (or water), this is known as wind-hovering. Kestrels, terns and even hawks use this wind-hovering.

Most birds that hover have high aspect ratio wings that are suited to low speed flying. One major exception to this are the hummingbirds, which are among the most accomplished hoverers among all birds. Hummingbird flight is different from other birds flight in that the wings are fully extended throughout the whole stroke, the stroke being a lying figure eight. Some hummingbirds can beat their wings 52 times a second, though others do so less frequently.

There are two kinds of hovering, symmetric hovering and asymmetric hovering (figure 2.8), as described by Norberg [135] and Shyy [167]. For large birds, which cannot rotate their wings between the forward and backward stroke, the wings are extended to provide more lift during downstroke, whereas during the upstroke the wings are flexed backward to reduce drag. In general the flex is more pronounced in the slow forward flight than in fast forward flight. This type of asymmetric hovering is usually called *"avian stroke"* [9, 167].

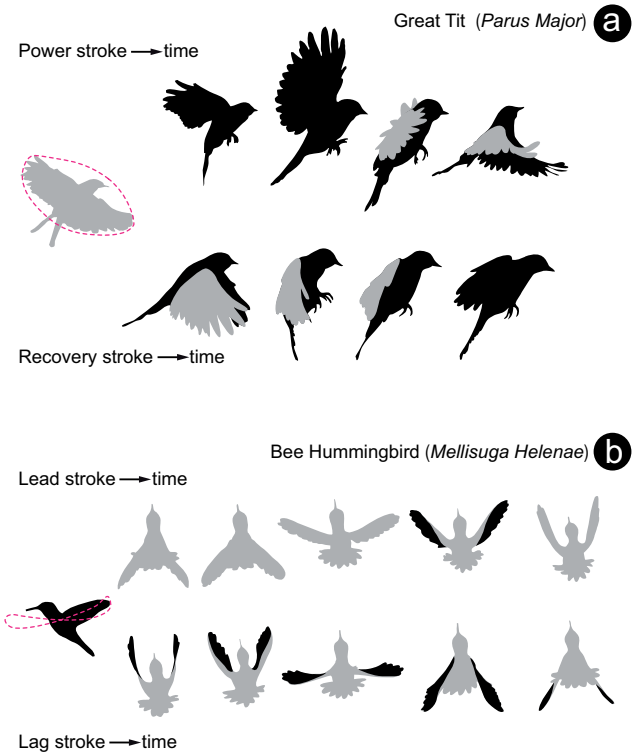

Figure 2.8: *Hovering flight: a) asymmetric hovering or "avian stroke" and b) symmetric hovering or "insect stroke".*

Symmetric hovering, also called normal or true hovering, or *"insect stroke"* [9, 167], is performed by hummingbirds or insects that hover with fully extended wings during the entire wing-beat cycle. Lift is produced during the entire wing stroke, except at the reversal points. The wings are rotated and twisted during the backstroke so that the leading edge of the wing remains the same throughout the cycle, but the upper surface of the wing during the forward stroke becomes the lower surface during the backward stroke. The wing movements during downstroke and upstroke can be seen in figure 2.9. Note that, during hovering, the body axis is inclined at a desirable angle and the wings describe a figure of a lying eight in the vertical plane.

2.1.4 Take-off and landing

Take-off can be one of the most energetically demanding aspects of flight, as the bird needs to generate enough airflow under the wing to create lift. In small birds a jump up will suffice, while for larger birds this is not possible. In this situation, birds need to take a run up in order to generate the airflow to take off. Large birds often simplify take off by facing into the wind, and if they can, perching on a branch or cliff so that all they need to do is drop off into the air.

Landing is also a problem for many large birds with high airspeeds. This problem is dealt with in some species by aiming for a point below the intended landing area (such as a nest on a cliff) then pulling up beforehand. If timed correctly, the airspeed once the target is reached is virtually zero. Landing on water is simpler, and the larger waterfowl species prefer to do so whenever

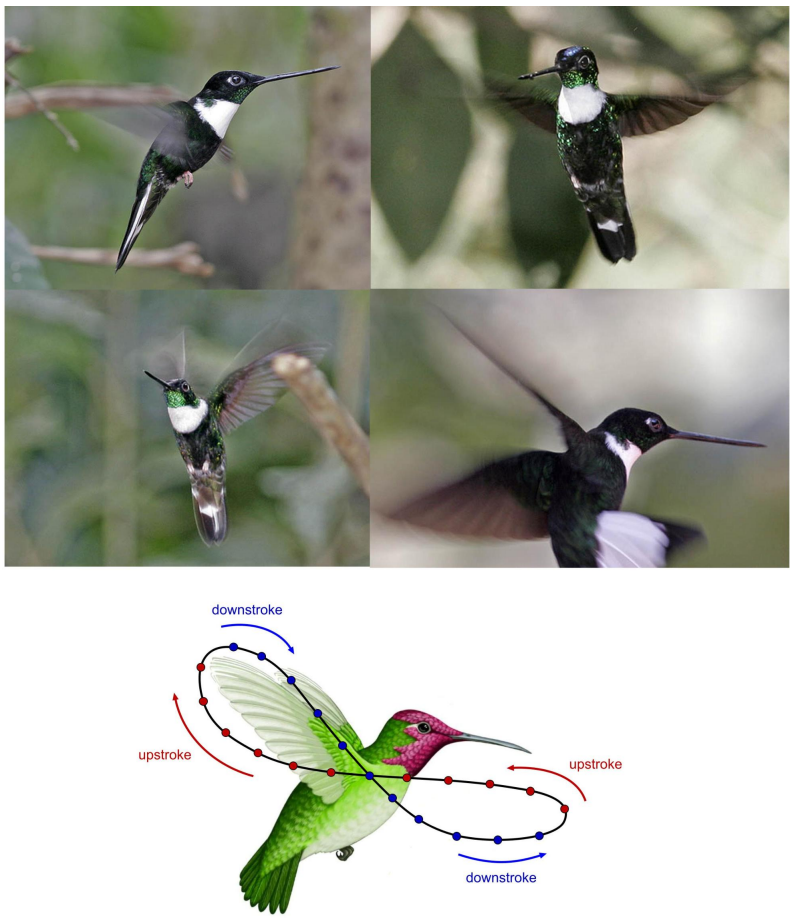

Figure 2.9: *Illustration of a hummingbird in hovering flight. In the bottom figure, hummingbird 's wing figure-eight pattern is shown.*

Figure 2.10: *A bufflehead running atop the water while taking off.*

possible and some species, such as swans, are only able to land on water. In order to lose height and velocity rapidly prior to landing, some large birds such as geese indulge in a rapid alternating series of sideslips in a maneuver termed as whiffling.

Figure 2.11: *Precision touchdown of an eastern imperial eagle on a tree branch.*

2.1.5 Summary

In this section, we just presented a brief overview of flapping flight in nature. In the next sections, we present a review of some of the computational models and experimental observations that constitute our current knowledge on flapping wing propulsion, this followed by a review of nonstationary airfoil aerodynamics (including dynamic stall, leading edge vortex shedding and thrust generation), flapping wing flight in terms of Reynolds number, Strouhal number and reduced frequency and finally we close this chapter with a presentation of flapping wings propulsion performance parameters and kinematics.

2.2 Brief History of Flapping Wing Research: Experimentation, Observations, Analytical and Computational Approaches

For thousands of years, the graceful flight of insects and birds has captivated those who have witnessed it. From those who dream of tasting the freedom of flight in man-made vehicles, to those who hope to further our understanding of some of nature's most fascinating creatures. There is a great variety of documented research in the study of flapping flight, since those studying the topic come from fields such as zoology, engineering and aerodynamics. These studies have been undertaken from analytical, experimental and computational standpoints. Hereafter we highlight some of the works which form the basis of our current understanding of flapping flight.

The desire to fly like birds is almost as old as humanity itself. Indeed, Leonardo da Vinci was so fascinated by the birds flight that he made many sketches of bird's wings and artificial wings and

summarized his flapping wing studies in a book manuscript named *"Sul volo degli Uccelli"* [200]. However, little progress was made during the following centuries until Otto Lilienthal, the great pioneer in human flight, began his flying experiments with his brother Gustav in the mid-1800s. While Lilienthal may be most famous for his glider experiments in the 1890s which proved that heavier than air flight is possible without the use of flapping wings, much of his knowledge in aeronautics was derived from experimentation with flapping wings and observation of birds. In 1889, Lilienthal published a book describing his experiments and detailing his predictions for the energy required for flapping wing flight [110]. Perhaps the most significant of his findings was the benefit of using cambered airfoils instead of flat plates, which he discovered by testing models on a rotating apparatus. Shortly after his death in a gliding accident in 1896, Lilienthal served as an inspiration to the Wright brothers in their successful effort to develop powered human flight. Following the sustained flight of the Wright Flyer in 1903 and the subsequent rapid development of fixed-wing airplanes in conjunction with the obvious mechanical complications introduced by flapping wings, the further development of man-made flapping wing vehicles and the research of flapping flight was discouraged.

The earliest scientific theories concerning flapping wing flight pertain to purely heaving airfoils. In independent studies in 1909 and 1912, Knoller [101] and Betz [20], perceived that flapping a wing in a free stream flow resulted in an effective angle of attack (α_{eff}) with a normal force vector containing both lift and thrust components. This phenomenon is now referred to as the Knoller-Betz effect and is illustrated in figure 2.12.

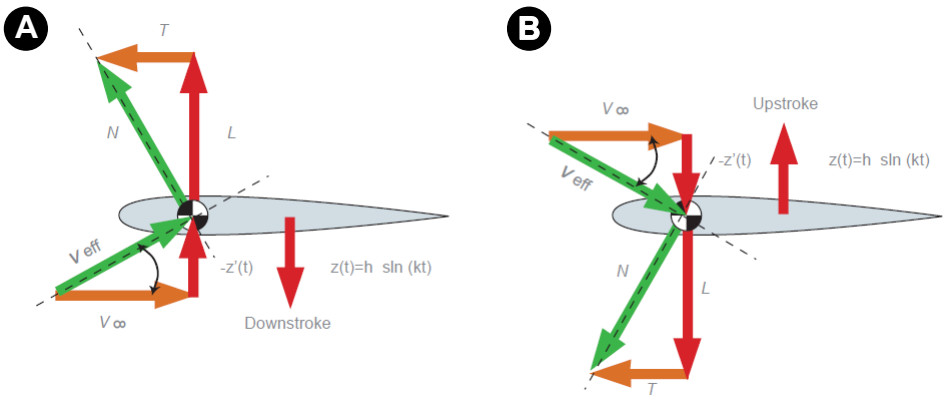

Figure 2.12: *Thrust (T) and lift (L) components of the normal force vector (N) during heaving motion.*

In 1922, Katzmayr [99] conducted wind tunnel tests to validate the Knoller-Betz effect. Rather than flapping the airfoil, Katzmayr sinusoidally oscillated the freestream velocity. Katzmayr's measurements conclusively proved that an airfoil mounted in an oscillating wind stream experienced a thrust force. Also adding to this increasing field of research was Prandtl's student Birnbaum [22], who in the same decade developed a solution for an incompressible flow past flapping airfoils and observed the conditions that lead to flutter or thrust generation. He also suggested the use of a sinusoidally flapping (heaving) wing as an alternative to the conventional propeller.

By the 1930s, fixed wing aircrafts had improved greatly in performance. At this point, the devel-

opment of an unsteady aerodynamic theory became important for the investigation of flutter. In 1935, Theodorsen published an analytical approach for estimating the unsteady lift and moment on harmonically oscillating airfoils [184]. In deriving this method, Theodorsen used the assumptions of an inviscid and incompressible flow. He also assumed that all oscillations were of very small amplitude; thus the flow was assumed to remain fully attached to the airfoil during the flapping cycle (the Kutta condition was applied at the trailing edge of the airfoil). Theodorsen assumed that the wake of the airfoil would take the form of a continuous vortex sheet of sinusoidally varying strength, stretching from the trailing edge to infinity in the downstream direction. The wake was not allowed to change shape in response to the velocity induced by the wake. This Theodorsen's theory will become in the years to come the standard tool to analyze airfoil flutter, rotorcraft aerodynamics and flapping flight problems.

In the following decade, the aerodynamics of heaving-and-pitching airfoils received much attention because of its importance for reliable flutter and gust response analyses. However, such analyses required only the determination of the lifting forces generated by heaving or pitching airfoils, and consequently, little effort was devoted over the years to the determination of the thrust forces.

Nevertheless, in the mid 1930s von Karman and Burgers [205] offered the first theoretical explanation of drag or thrust production based on the observed location and orientation of the wake vortices, as illustrated in figure 2.13. In their work, von Karman and Burgers [205] experimentally observed that a wake consisting of two rows of counter-rotating vortices could produce a thrust force on an airfoil in an incompressible flow. At about the same time, Garrick [57] applied Theodorsen's inviscid, incompressible, oscillatory, flat plate theory [184] to the determination of the thrust force (which remains the classical reference work to this day) and showed that heaving airfoils generate thrust over the whole frequency range considered, whereas pitching airfoils do so only with frequencies above a certain critical value and as function of the pivot location. The first experimental verification of Garrick predictions was provided by Silverstein and Joyner [168] in 1939. Later on, in 1950, Bratt [23] performed flow visualization experiments that corroborated von Karman and Burger's observations. Of particular interest, Bratt's experimental data include several cases where a non-symmetrical, deflected wake pattern was observed, but no comment was made on these deflected wakes.

Recognizing the fact that some of the flapping energy is lost in the form of vorticity shed in the wake, Schmidt [161] proposed that a stationary wing be placed in the oscillatory wake of a flapping wing to take back some of the vortical energy lost by the flapping airfoil, the aft wing thus is exposed to an oscillatory flow which generates thrust by virtue of the Katzmayr effect. Schmidt, as a result of his studies about flapping foils in the 1940s and 1950s, developed the wave propeller (which he claimed achieved efficiencies as good as those of conventional propellers) and demonstrated it on a catamaran boat.

In the early 1970s, Lighthill [109] performed a very similar analysis to that of Garrick [57], with an additional assumption of pitching motion leading plunging motion by 90 degrees, in the context of lunate (crescent-shaped) tail propulsion by fishes and cetaceans. Using an energy method, he obtained expressions for thrust and propulsive efficiency that are identical to those of Garrick when converted into similar terms. In principle, any other relative phase difference between pitching and plunging motion may be accounted for, by a change in the pitch axis of the airfoil, however this is strictly only true for very small amplitude motion. As the Garrick work makes no assumption about relative phases of plunging and pitching, Lighthill work it is slightly more

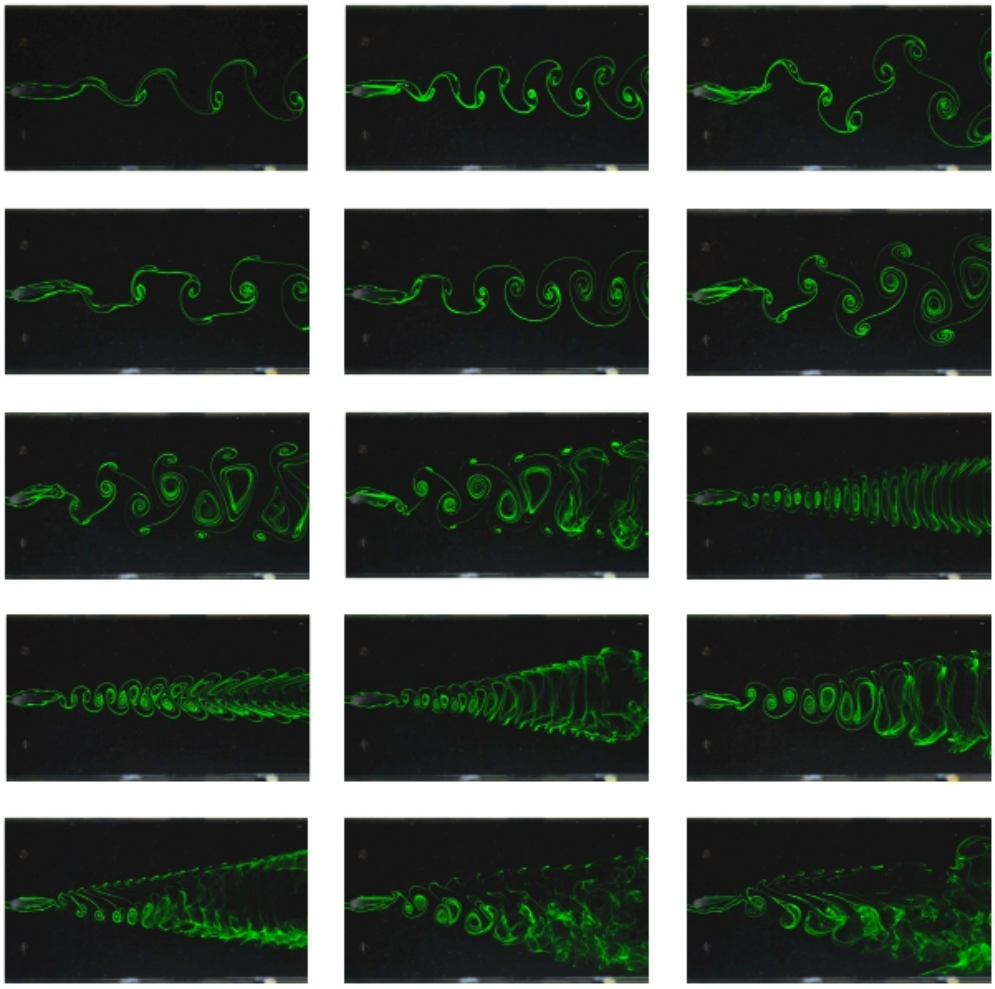

Figure 2.13: *Dye visualizations of different wakes behind an oscillating airfoil (from von Karman vortex street, to neutral wake, to reverse von Karman vortex street, to deflected wake).*

general. Similar analyses of the small amplitude motion of a $2\mathbb{D}$ flat plate in potential flow may be found in a variety of sources (*e.g.*, Ashley and Landahl [8] and Katz and Plotkin [98]).

The linearized potential flow analyses discussed above, particularly the Garrick analysis, contain a number of assumptions about the flow, the airfoil geometry and the airfoil motion. Firstly, the airfoil is assumed to be thin and is treated as a flat plate. Secondly, the flow is assumed to be inviscid and incompressible, and the Kutta condition is applied to the flow at the trailing edge of the airfoil. Finally, the motion of the airfoil is assumed to be of small amplitude and the wake does not evolve in response to its own induced velocity field. In order to avoid the limitations imposed by the previous assumptions and to improve the predictive capabilities, new computational models that account for the unsteady nature of the motion, arbitrary airfoil sections and three-dimensional effects were developed.

Of special note is the replacement of Theodorsen oscillatory thin-airfoil theory by an approach which enables the computation of incompressible potential flow past oscillating airfoils of arbi-

trary shape. This is being accomplished by the placement of sources and vortices on the airfoil surface rather than along the chord line. This so-called panel method was pioneered by Giesing [58], who generalized the method of Hess and Smith [82], for steady airfoil flow. The method was further developed by Teng [183], Platzer *et al.* [148], and Jones *et al.* [147]. More recently, two-dimensional unsteady panel methods have been used for the prediction of the aerodynamic performance of flapping wings MAVs [94, 96, 97]. Three-dimensional methods also have been used to predict the forces on insect wings and cetacean tails [171, 173].

Inviscid analyses require the separation point of the flow on the airfoil or wing to be known ahead of time. This is usually fixed at the trailing edge (the Kutta condition), and vorticity is shed into the wake from this point. Navier-Stokes flow solvers avoid this limitation by using the full viscous flow equations rather than potential flow, allowing flow separation and vortex shedding to be predicted rather than assumed *a priori*, allowing simulation of flows that involve leading and trailing edge separation. Also, thanks to the rapid increase in computer power over the past few years, such viscous flow solvers are becoming more popular and have been successfully used to model flapping wing aerodynamics, thus making possible to compute the strong viscous effects and three-dimensional flows characteristics of flapping flight.

Some recent works using Navier-Stokes solvers that are worth to mention, include the work done by Young and Lai [216], where it is shown that the vortical wake structures, and the lift and thrust characteristics of a heaving airfoil are strongly dependent on the oscillation frequency and amplitude. Isogai *et al.* [89] carried out Navier-Stokes computations for a single flapping airfoil. They calculated the thrust and the propulsive efficiency for various combinations of frequency and phase shift, and gave a detailed analysis of the effects of the dynamic stall phenomena on the behavior of the thrust and the propulsive efficiency. In a following work about the aerodynamic performance of a dragonfly, Isogai *et al.* [88], clarified the fundamental mechanism of the hovering flight of a dragonfly. Another flow simulation of an insect was presented by Togashi *et al.* [189] in which, by solving the Navier-Stokes equations, they gave the approximate numerical solution of the flow past a hornet in forward flight. Liu *et al.* [112], using a $3\mathbb{D}$ Navier-Stokes solver, successfully modeled the powered hovering mode of a Hawkmoth. Lewin and Haj-Hariri [108] and Wang [207], both examined the propulsive characteristics of an elliptical airfoil heaving sinusoidally over a range of frequencies and heave amplitudes in order to correlate viscous flow structures to thrust generation. Pedro *et al.* [140], numerically studied the propulsive efficiency of a flapping hydrofoil at a Reynolds number of 1100. In their work, Pedro *et al.* [140] studied airfoils undergoing pure pitching motion and combined pitching-and-heaving motion and they showed the sensitivity of thrust and efficiency to the Strouhal number, maximum pitch angle and phase angle. Hover *et al.* [87] used sinusoidal and non-sinusoidal effective angle of attack variations in time to investigate the propulsive performance of an airfoil undergoing combined heave and pitch oscillations. Lee *et al.* [106] identified the key physical flow phenomenon dictating the thrust generation of a heaving and/or pitching airfoil in terms of flow and geometry parameters. Tuncer *et al.* [196, 197, 198] performed several Navier-Stokes computations to explore the effect of flow separation on the thrust and the propulsive efficiency of a single flapping airfoil in combined pitch and heave oscillations.

Experimentally, Jones *et al.* [94] and Lai and Platzer [105], conducted water tunnel flow visualization experiments on flapping airfoils which have provided a considerable amount of information on the wake characteristics of thrust producing flapping airfoils. Koochesfahani [103] experimentally studied the wake structure behind an oscillating airfoil and found different topologies of the wake

with associated numbers of shed vortices per cycle of oscillations as function of the amplitude and the frequency of oscillation. Triantafyllou *et al.* [191], based on the experimental results of Koochesfahani [103] and on a linear stability analysis of an average velocity profile, assumed that optimal efficiency is obtained when an airfoil flaps at the frequency of maximum spatial amplification of the wake. Anderson *et al.* [7], in their experiments, also observed that the phase angle between pitch and heave oscillations plays a significant role in maximizing the propulsive efficiency. The experimental studies by Jones *et al.* [95] and Platzer and Jones [146] demonstrated that two airfoils arranged in a biplane configuration and oscillating in counter-phase show significant benefits of thrust and propulsive efficiency compared to a single flapping airfoil.

More recently, Heathcote and Gursul [67] carried out water tunnel experiments on a heaving-and-pitching flexible airfoil for low Reynolds numbers. They observed a peak in thrust coefficient at a particular value of the phase shift between heaving and pitching for fixed heave and pitch amplitudes. Schouveiler *et al.* [163] experimentally studied the performance of an aquatic propulsion system inspired from the thunniform swimming mode to investigate the effects of flapping parameters on the thrust force and the hydro-mechanical efficiency. In the computational area of flexible airfoils/fins, Miao and Ho [124], studied the effect of chordwise flexure amplitude on the unsteady aerodynamic characteristics for flapping airfoils with various combinations of Reynolds number and reduced frequency. In this study, they observed an enhancement in the propulsive efficiency for a flapping airfoil with flexure amplitude of 0.3 of the chord length, they also found that the flow conditions which yield the highest propulsive efficiency correspond to a Strouhal number St of 0.255. Liu and Kawachi [113], performed a numerical study of the undulatory locomotion of a swimming body. They successfully modeled the unsteady hydrodynamics of a realistic three-dimensional tadpole-shaped model, establishing the importance of accurately predicting a staggered array of reverse von Karman vortices, the jet stream and their correlation with thrust generation. They also pointed to an optimal propulsive mechanism appropriate to undulatory swimming, which is achieved by a best coupling of the geometry and the motion matched to the body.

Despite the potential of flapping wings for either pure propulsion or as an integrated lift/propulsion system, it was regarded as unattractive until very recently. Flapping wing studies therefore largely remained restricted to scientists interested in bird flight or fish propulsion problems. An unexpected revival of interest in the study of flapping wing flight phenomena occurred in the late 1990s with the announcement of a major initiative by the United States of America Defense Advanced Research Projects Agency (DARPA) to encourage the development of micro-air-vehicles (MAVs). The goal of the DARPA MAVs program was to determine whether evolving technologies could be favorably integrated into a mission capable flight system for military surveillance and reconnaissance applications. The only requirement was that the dimension of the vehicle should not exceed 15 cm. There were no other restrictions on the design. The use of flapping wings for vehicles with dimensions not exceeding 15 cm in length or span is an obvious option because of the low efficiency of conventional propellers and fixed wings at low Reynolds number. Therefore, this DARPA initiative sparked a large number of investigations in the field of flapping wing propulsion; some of these investigations are compiled in [127].

2.3 The Physics of Drag and Thrust Generation Due to Wing Flapping

As already mentioned, Knoller [101] and Betz [20] were the first ones to offer an explanation for the birds' ability to generate a propulsive force by means of flapping their wings. Consider the airfoil undergoing sinusoidal flapping while also flying forward. As the airfoil moves through its mean position during the downward stroke, it is effectively exposed to a flow with positive angle of incidence (see figure 2.12). Similarly, it sees a negative incidence angle during the upstroke. If, for simplicity, the resulting aerodynamic force is assumed to be essentially perpendicular to the instantaneous approach flow angle, then decomposition into a force component parallel to the flight velocity vector will produce a small sinusoidally varying thrust force. It is understood that this explanation is greatly simplified. The actual flow which is produced is considerably more complicated.

The flow over a stationary airfoil or a bluff body at low Reynolds number produces a von Karman vortex street as shown in figure 2.14. In this configuration, where the upper row of vortices rotates clockwise and the lower row counterclockwise (for flow from left to right), the measured time-averaged velocity distribution in the wake shows a distinct velocity or momentum deficit, indicative of drag. This vortex configuration is hereafter referred as drag producing wake.

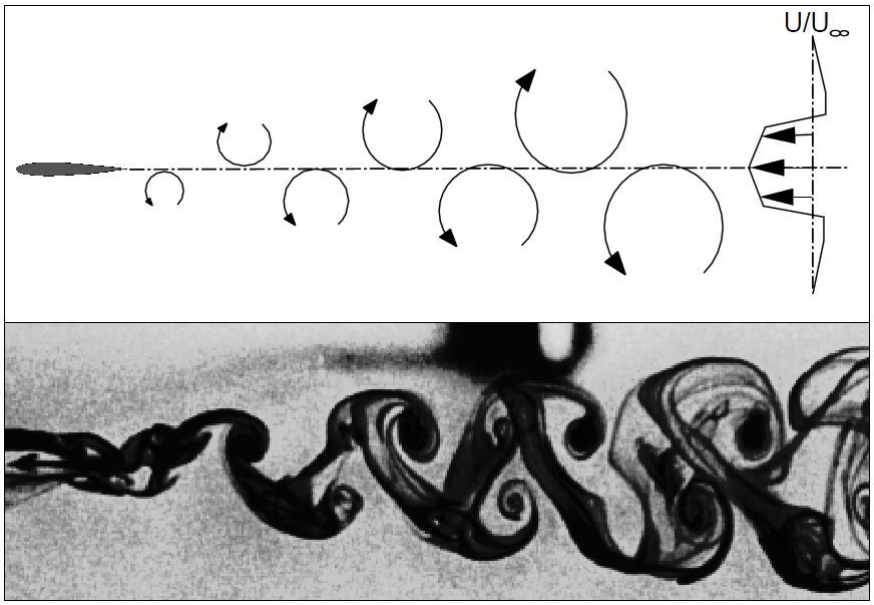

Figure 2.14: *Vortex street indicative of drag production (drag producing wake) [93].*

As is well known, every change in the incidence of the airfoil will produce a starting vortex which is shed from the trailing edge. Therefore, a sinusoidally oscillating airfoil will generate a vortex street behind the airfoil. This phenomenon can be reproduced easily experimentally or simulated with the computational approaches highlighted in the previous section. Depending of the heave velocity, the resulting vortex street can be drag producing wake or can consist of an upper row of counterclockwise vortices and a row of lower clockwise vortices (see figure 2.15). This vortex street therefore is just the opposite of the well known von Karman vortex street and is known

as reverse von Karman street. If time-averaging is applied at some location cutting the wake in the normal direction to the free-stream a jet profile or momentum surfeit wake is obtained. The vortex street produced by the flapping foil in effect produces a jet flow. This vortex configuration is hereafter referred to as thrust producing wake. This is to be expected since the thrust experienced by the airfoil must be found as momentum increase in the fluid. As shown by Jones *et al.* [94] and Lewin and Haj-Hariri [108], this jet flow can indeed be measured and is in good agreement with panel code and Navier-Stokes solver predictions.

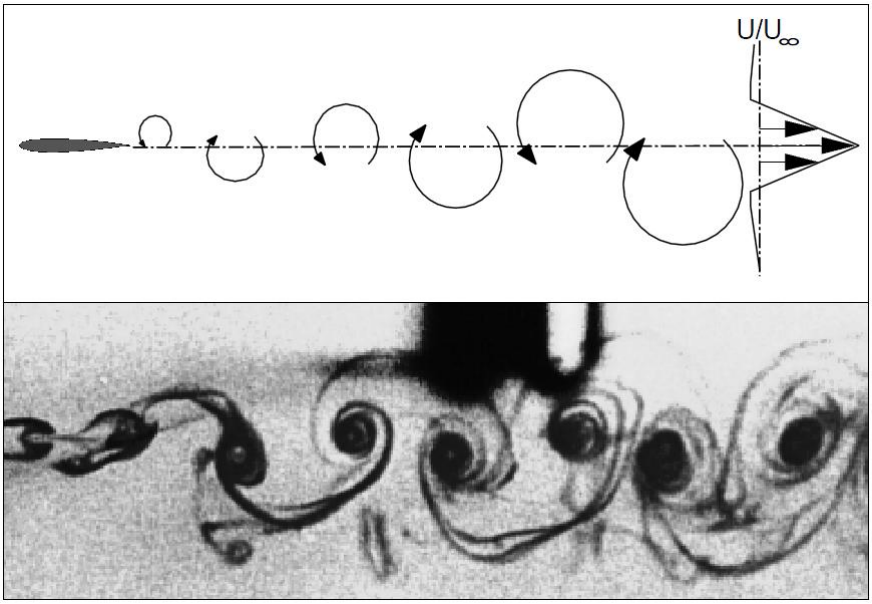

Figure 2.15: *Vortex street indicative of thrust production (thrust producing wake) [93].*

From this analysis, clearly we see that increasing the vertical spacing between the rows increases the region of lower time-averaged velocity (for von Karman street) or higher velocity (for reverse von Karman street) between the rows. This then increases the drag (or thrust) of the configuration in the direct proportion to the vertical spacing. It follows that reducing the vertical spacing to zero, so that the two vortex rows are interspersed as in figure 2.16, will result in zero net drag or thrust production. This vortex configuration is hereafter referred to as neutral wake.

Jones *et al.* [93], found that for large heave velocities, the symmetric vortex street changes into a dual-mode or nonsymmetric vortex street, as show in figure 2.17. In this case, in addition to a net thrust, a net lift is also observed according to the deflection of the vortex street. This vortex configuration is hereafter referred to as deflected wake or lift-thrust producing wake. This phenomena was previously observed by Bratt in 1950 [23], but he did not make any comments on these deflected wakes. Once again, the flow visualizations are in good agreement with panel code [94, 96, 171] and Navier-Stoke solver predictions [108, 115].

This encouraging agreement between the measurements and the inviscid flow predictions of Jones *et al.* [93, 147] and Platzer *et al.* [148], might give the impression that the physics of flapping airfoils is understood reasonably well and that the prediction of the achievable thrust can be made with considerable confidence by using inviscid methods. Unfortunately, even disregarding

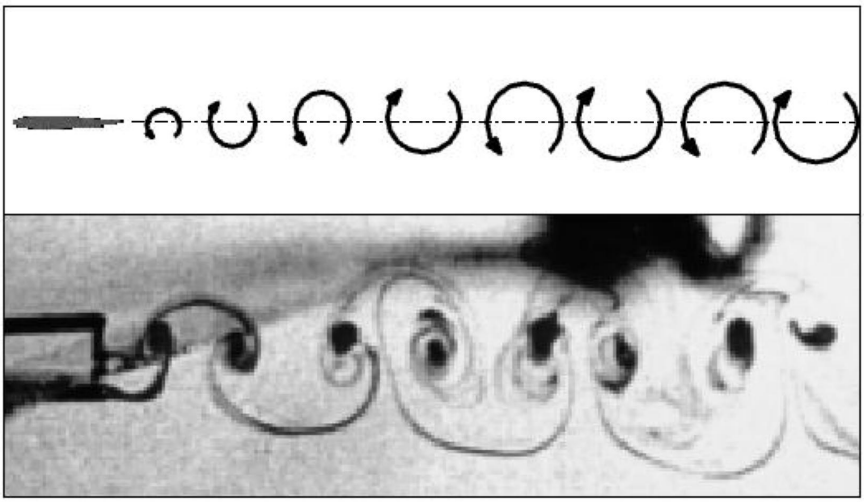

Figure 2.16: *Vortex street indicative of zero drag (neutral wake) [93].*

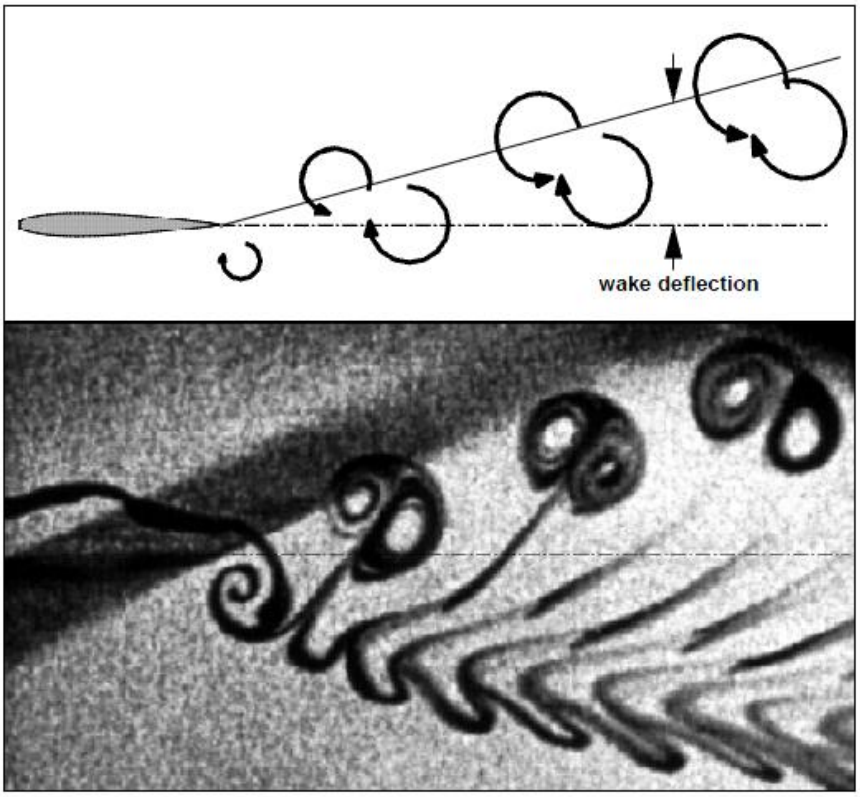

Figure 2.17: *Dual-mode or nonsymmetric vortex street indicative of thrust and lift production (deflected wake) [93].*

the three-dimensional flow effects introduced by finite-span wings, the range of validity of inviscid flow predictions is severely limited by the onset of dynamic stall. This seems to be particularly true at the low Reynolds numbers typically required for micro-air-vehicles. Hence panel methods

must be dropped in favor of more sophisticated Navier-Stokes solvers, which take into account viscous effects.

2.4 The Phenomenon of Dynamic Stall and Leading Edge Vortex (LEV) Shedding

Dynamic stall is a non-linear unsteady aerodynamic effect that occurs when airfoils rapidly change the angle of attack. The rapid change can cause a strong vortex to be shed from the leading edge of the airfoil, and travel backwards above the wing, strongly interacting with the vortex which forms at the trailing edge. The vortex, containing high velocity airflows, briefly increases the lift produced by the wing. As soon as it passes behind the trailing edge, however, rapid lift loss and changes in pitching moment occur, leading to a severe hysteresis loop in lift, drag and pitching moment.

Dynamic stall is an effect mostly associated with helicopters, turbines, windmill blades and recently to flapping wing propulsion. Helicopter aerodynamicists are quite familiar with this phenomenon. A helicopter blade in forward flight can experience dynamic stall while it is in the so-called retreating blade position, exposing the blade to high incidence angles. For this reason, the aerodynamics of pitching airfoils experiencing dynamic stall has been studied to a considerable extent both experimentally and computationally. The current state-of-the-art can be found in the papers of Carr and Chandrasekhara [31], Ekaterinaris and Platzer [47], McCroskey *et al.* [117] and Rozhdestvensky and Ryzhov [156].

Most dynamic stall studies have been limited to pitching airfoils because of the importance of this motion for helicopter blades. However, for thrust generation it is well known (and has been shown in detail by Jones and Platzer [97]) that pure heave or a combined pitch/heave motion is required in order to produce significant thrust forces. Very few experiments involving dynamic stall due to pure heave have been carried out, but several Navier-Stokes computations have been reported by Isogai *et al.* [89], Tuncer and Platzer [198] and Ramamurti and Sandberg [151], among others. These computations clearly show the possibility of occurrence of dynamic stall for values of flapping frequency, amplitude, and Reynolds number typical for MAV flight.

The basic physics of dynamic stall is illustrated in figure 2.18, where a sequence of images during the downstroke of a heaving airfoil is shown from a Navier-Stokes simulation of a NACA 0012 airfoil undergoing sinusoidal heaving motion. In the figure, a strong vortex forms near the leading edge (LEV), propagates over the upper surface and is then swept downstream past the trailing edge. While the vortex is over the airfoil upper surface, lift is enhanced. This increased lift is significantly greater than the static lift which would be generated at the corresponding static incidence angle. As soon as the vortex approaches the trailing edge this lift is reduced quite suddenly and dramatically. While the presence of dynamic stall is generally adverse on aircrafts and rotorcrafts, there is evidence that birds and insects may benefit from this effect as a high-lift mechanism, by simple relying on leading edge vortices (LEV) created by dynamic stall during the flapping motion [48, 107, 125]. Basically, LEVs are originated due to roll up of the shear layer separating from the leading edge region of the wing during flapping or pitching motion.

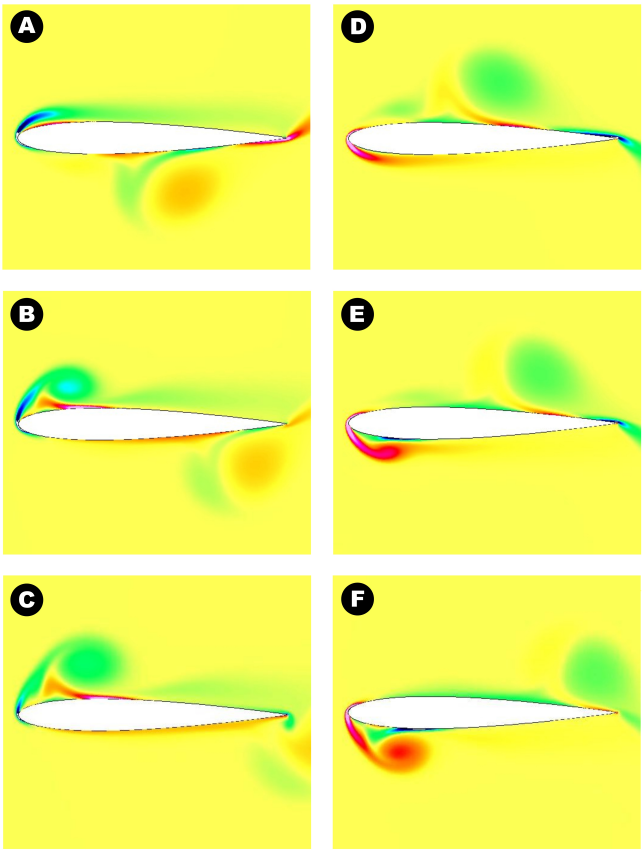

Figure 2.18: *Dynamic stall on a heaving airfoil during downstroke (sequence is top-to-bottom left column, then top-to-bottom right column).*

2.5 Reynolds Number in Terms of Flapping Flight

Given a reference length L_{ref} and a reference velocity U_{ref}, one normally defines the Reynolds number Re as

$$Re = \frac{\rho U_{ref} L_{ref}}{\mu} = \frac{U_{ref} L_{ref}}{\nu} \tag{2.1}$$

where ρ is the fluid density, μ is the dynamic viscosity of the fluid and ν is the fluid kinematic viscosity. Re represents the fluid ratio of inertial forces to viscous forces and is commonly used in fluid dynamics. In flapping flight, with consideration of the fact that flapping wings produce lift and thrust, the mean wing's chord length c_m is used as the reference length L_{ref}, whereas the body length or averaged length of the caudal fin is typically used in swimming animals. The reference velocity U_{ref} is also defined differently in hovering and forward flight.

In hovering flight, as there is no forward velocity, the mean wingtip velocity may be used as the reference velocity, which can be written as $U_{ref} = \omega R$, where R is the wing length (half wingtip-to-wingtip span) and ω is the mean angular velocity of the wing ($\omega = 2\Phi f$, where Φ is the wing-beat amplitude, measured in radians, and f is the flapping frequency). Therefore the

Reynolds number for a 3𝔻 flapping wing in hovering flight (Re_{fh3}) is given by

$$Re_{fh3} = \frac{U_{ref}L_{ref}}{\nu} = \frac{2\Phi f R c_m}{\nu} = \frac{\Phi f R^2}{\nu}\left(\frac{4}{\mathrm{AR}}\right) \tag{2.2}$$

where AR is the wing aspect ratio ($\mathrm{AR} = S^2/A$) which gives a relation between the wingspan S and the wing area A. In eq. 2.2, the aspect ratio is introduced in the form $\mathrm{AR} = (2R)^2/A$ with the wing area being the product of the wing span ($2R$) and the mean chord (c_m). Note that the Reynolds number here is proportional to the wing-beat amplitude Φ, the flapping frequency f, the square of the wing length R^2, but inversely proportional to the AR of the wing.

For a 2𝔻 flapping airfoil undergoing hovering motion, the Reynolds number (Re_{fh2}) is defined by the maximum heaving velocity, such as

$$Re_{fh2} = \frac{\rho U_{ref}L_{ref}}{\mu} = \frac{U_{ref}L_{ref}}{\nu} = \frac{2\pi f h_a c}{\nu} \tag{2.3}$$

where f is the flapping frequency, h_a is the heaving amplitude, and c is the airfoil chord length.

In forward flight, for both 2𝔻 and 3𝔻 applications, the forward velocity U is often used as the reference velocity U_{ref} and the mean chord length c_m as the reference length L_{ref}, hence eq. 2.1 is solely used to obtain the forward flight Reynolds number Re. Compared with the hovering flight Reynolds number, which is proportional to R^2, the forward flight Reynolds number is proportional to R.

2.6 Strouhal Number and Reduced Frequency

A fundamental dimensionless parameter in flows showing an unsteady aerodynamic nature is the Strouhal number (St), this number is well known for characterizing the vortex dynamics and shedding behavior of unsteady flows. In some St ranges, a flapping airfoil produces thrust, and the vortices in the wake are termed reverse von Karman vortices. In general, for flapping flight, the dimensionless parameter St is normally defined as

$$St = \frac{fL_{ref}}{U_{ref}} = \frac{2fh_a}{U_{ref}} \tag{2.4}$$

where f is the stroke (flapping) frequency in flapping flight, h_a is the stroke (flapping) amplitude, and U is the forward velocity. This definition describes a ratio between the oscillating (flapping) speed (fh_a) and the forward speed (U), which offers a measure of propulsive efficiency in flying and swimming animals. In the study of natural flyers and swimmers in cruising condition it is found that the Strouhal number, as defined by eq. 2.4, is often within a region of $0.2 < St < 0.4$; in this range of St, the propulsive efficiency (see eq. 2.20) is high, with an optimal St value of 0.3 [136, 182, 192].

Another dimensionless parameter that characterizes the unsteady aerodynamics of pitching and heaving airfoils is the reduced frequency, which is a measure of the residence time of a particle convecting over the airfoil chord compared to the period of motion. The forward flight reduced

frequency k is defined as

$$k = \frac{2\pi f L_{ref}}{2U_{ref}} = \frac{\pi f c_m}{U_{ref}} = \frac{\omega c_m}{2U_{ref}} \tag{2.5}$$

where ω is equal to $2\pi f$. In hovering $3\mathbb{D}$ flight, for which there is no forward speed, the reference speed U_{ref} is defined as the mean wingtip velocity $2\Phi f R$; then the reduced frequency k_{fh3} can be formulated as

$$k_{fh3} = \frac{\pi f c_m}{U_{ref}} = \frac{\pi c_m}{2\Phi R} = \frac{\pi}{\Phi \mathrm{AR}} \tag{2.6}$$

where the AR is introduced here again as in eq. 2.2. For the special case of $2\mathbb{D}$ hovering airfoils, the reference velocity U_{ref} is the maximum flapping velocity (see eq. 2.3), and the reduced frequency k_{fh2} is defined as

$$k_{fh2} = \frac{\pi f c}{U_{ref}} = \frac{c}{2h_a} \tag{2.7}$$

which is simply related to the normalized stroke amplitude.

The Strouhal number St (eq. 2.4), may be related to the reduced frequency k (eq. 2.5) as follows

$$St = \frac{f L_{ref}}{U_{ref}} = \frac{2 f h_a}{U_{ref}} = \frac{1}{\pi}\frac{\omega c_m}{2U_{ref}}\frac{2h_a}{c_m} = \frac{k}{\pi}\frac{2h_a}{c_m} = \frac{2}{\pi}\frac{k h_a}{c_m} = \frac{2}{\pi} kh \tag{2.8}$$

where h is the nondimensional heaving amplitude equal to h_a/c_m and where the product kh is defined as the maximum nondimensional heaving velocity.

In the case of forward flight, another dimensionless parameter is the advance ratio J. In a general $2\mathbb{D}$ or $3\mathbb{D}$ framework, J is defined as

$$J = \frac{U_{ref}}{2\pi f h_a} \tag{2.9}$$

which is related to St, specifically, $J = 1/(\pi St)$. In eq. 2.9, the reference velocity U_{ref} is the forward flying velocity U.

2.6.1 Strouhal Number as the Fundamental Aerodynamic Parameter in Flapping Flight

In studying natural flyers and swimmers and in designing a lifting and/or propulsive man-made vehicle based on flapping wings, an immediate question is the range of flapping parameters that may be chosen to optimize the design thrust, propulsive efficiency, or other criteria. Evolution is not guaranteed to find a solution that is globally optimized among the range of available parameters, and nature may be more limited in the range of parameters to change. However, a thorough examination of nature's techniques is a logical starting point in defining guiding principles.

Taylor *et al.* [182] performed a study of published wing beat frequencies and amplitudes and cruise speeds, across a range of birds, bats and insects, to determine Strouhal numbers in cruising flight.

They found 75% of the 42 species considered to fall within a narrow range of $0.19 < St < 0.41$, with a mean value of $St = 0.29$. They made the point that whilst similar species might be expected to exhibit similar Strouhal number due to similar morphological and physiological characteristics, the disparate variety of species used in their study (spanning five orders of magnitude of body mass) strongly implies that the narrow Strouhal number range is due to aerodynamic principles alone.

Triantafyllou *et al.* [193], provided a graph of measured Strouhal numbers for a range of fishes, sharks and cetaceans, with all falling largely within the Strouhal number range of $0.25 < St < 0.35$. In a later study, Triantafyllou *et al.* [191] expanded on these results, showing that this narrow range of selected Strouhal number holds over a Reynolds number range of $10^4 < Re < 10^6$, with the Re based on the averaged length of the caudal fin (tail) in the specimens examined. The velocities used in the calculation were stated to be at or near maximum velocity range for each species.

Nudds *et al.* [136], proposed a simple and accurate empirical model for predicting wing-beat frequency in birds, based on the Strouhal number. The proposed aerodynamic model predicted wing-beat frequency better than any other relationship proposed, explaining 90% of the observed variance in a sample of 60 birds species. In the results presented by Nudds *et al.*, they found that their calculations were consistent with the hypothesis that birds have converged upon a narrow optimum range of St in cruising flight. The best estimates of St for the empirical data given by them fell within the range of $0.2 < St < 0.4$ associated with high propulsive efficiency in other theoretical and experimental studies [152, 193]. These results lead Nudds *et al.* to conclude, *"Avian wing kinematics therefore appear to have been tuned by natural selection for high aerodynamic efficiency"*.

These results all support the proposition that the Strouhal number is the single, or at least the dominant parameter that controls the aerodynamics of flapping flight, insofar as high efficiency propulsion is concerned [108, 136, 182, 191, 192, 193, 207]. As stated by Taylor [182], *"The exact mechanism by which St controls the efficiency of force production has yet to be fully elucidated, but is generally thought to reflect the role of St in governing the time-scales of vortex growth and shedding"*.

Hence, throughout this dissertation the Strouhal number St (or equivalently the product kh according to eq. 2.8) will be used as the fundamental aerodynamic parameter when characterizing the flapping motion, unless otherwise specified.

2.7 Flapping Airfoils Performance Parameters

In the study of flapping airfoils, several parameters may be used to quantify the flow characteristics. Hereafter, we present the most representative ones. Two parameters that provide important information in the study of flapping wing propulsion are drag and lift coefficients, which are defined as follow

$$c_d = \frac{D}{\frac{1}{2}\rho U^2 c} \tag{2.10}$$

$$c_l = \frac{L}{\frac{1}{2}\rho U^2 c} \tag{2.11}$$

Since we are studying flapping wing propulsion, it is more convenient to think in terms of thrust instead of drag. Hence, thrust can be seen as the opposite in direction to the drag force (but equal in magnitude), therefore we obtain

$$c_t = -c_d = \frac{T}{\frac{1}{2}\rho U^2 c} \tag{2.12}$$

In equations 2.10, 2.11 and 2.12; D, L and T are the drag forces, lift forces and thrust forces (same as drag forces but opposite in sign), respectively; ρ is the fluid density, U is the forward velocity and c is the airfoil chord.

The instantaneous power input P, can be defined as the amount of energy imparted to the airfoil for it to overcome the fluid forces and is equal to

$$P(t) = -L(t) \times \dot{y}(t) - M(t) \times \dot{\alpha}(t) \tag{2.13}$$

where $\dot{y}(t)$ is the vertical velocity of the airfoil pivot point ($\dot{y}(t) = \frac{d}{dt}y(t)$), $y(t)$ is the plunging motion of the airfoil pivot point, $\dot{\alpha}(t)$ is the angular velocity of the airfoil about the pivot point ($\dot{\alpha}(t) = \frac{d}{dt}\alpha(t)$), $\alpha(t)$ is the pitching motion of the airfoil about the pivot point, and $M(t)$ is the moment created by the lift and drag forces at the pitching axis and is nondimensionalized by

$$c_m = \frac{M}{\frac{1}{2}\rho U^2 c^2} \tag{2.14}$$

In eq 2.13, the sign of both terms is negative as the lift force and the moment are reaction forces created by the fluid as the airfoil moves through it, where the products $L(t) \times \dot{y}(t)$ and $M(t) \times \dot{\alpha}(t)$ are the instantaneous supplied energy for vertical translation motion (heaving) and angular motion (pitching), respectively. The input power can be also nondimensionalized as follows

$$c_P = \frac{P}{\frac{1}{2}\rho U^3 c} \tag{2.15}$$

Power, thrust, lift and moment coefficients can be also averaged over time as

$$c_{P_{mean}} = \overline{c_P} = \frac{1}{\mathbb{T}} \int_t^{t+\mathbb{T}} C_P(t)dt \tag{2.16}$$

$$c_{t_{mean}} = \overline{c_t} = \frac{1}{\mathbb{T}} \int_t^{t+\mathbb{T}} C_T(t)dt \tag{2.17}$$

$$c_{l_{mean}} = \overline{c_l} = \frac{1}{\mathbb{T}} \int_t^{t+\mathbb{T}} C_L(t)dt \tag{2.18}$$

$$c_{m_{mean}} = \overline{c_m} = \frac{1}{\mathbb{T}} \int_t^{t+\mathbb{T}} C_M(t)dt \tag{2.19}$$

where $\mathbb{T}$ is the period of flapping motion and is equal to $\mathbb{T} = 2\pi/\omega$. Finally, the propulsive efficiency can be seen as a measure of the energy lost in the wake versus the energy used in creating the necessary thrust, and is given by

$$\eta = \frac{T_{mean}U}{P_{mean}} = \frac{c_{t_{mean}}}{c_{P_{mean}}} = \frac{\overline{c_t}}{\overline{c_P}} \tag{2.20}$$

2.8 Airfoil Geometry and Flapping Kinematics

For a generic 2$\mathbb{D}$ case, the airfoil geometry and airfoil motions are shown in figure 2.19. Vertical (heaving) and rotational (pitching) motions are shown. Horizontal motion (lagging or surging) will be not considered for 2$\mathbb{D}$ cases. Note that throughout this dissertation single-mode motions will be referred to as pure pitching or pure heaving, whereas the term flapping will imply a combination of heaving-and-pitching in 2$\mathbb{D}$.

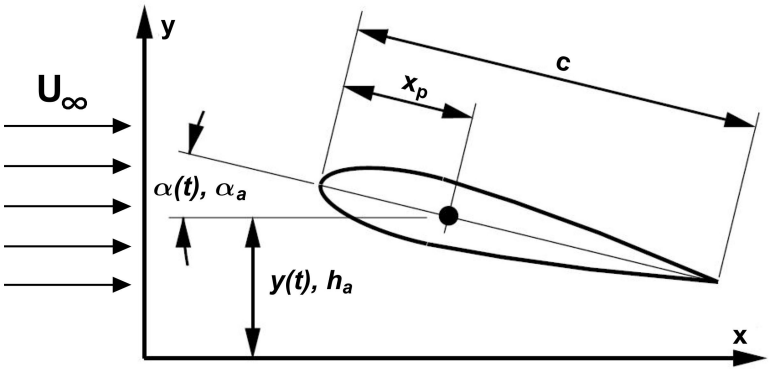

Figure 2.19: *Airfoil geometry and airfoil motion. In the figure, heaving motion $y(t)$ of the pivot point, pitching motion $\alpha(t)$ of the airfoil about the pivot point, maximum heaving amplitude h_a, maximum pitching amplitude α_a, airfoil chord c, pivot point x_p and free-stream velocity U_∞ are shown.*

In figure 2.19, c is the airfoil chord, x_p is the distance from the leading edge to the pivot point which simultaneously translates vertically (heave) and rotate (pitch), h_a is the maximum heaving amplitude, α_a is the maximum pitching amplitude, $y(t)$ is the heaving motion and $\alpha(t)$ is the pitching motion.

The 2$\mathbb{D}$ kinematics of the airfoil undergoing a combination of time-dependent heaving ($y(t)$) and pitching ($\alpha(t)$) motions, can be describe by symmetric, periodic, harmonic functions as follows

$$y(t) = h_a cos(\omega_h t + \varphi_h) = h_a cos(2\pi f_h t + \varphi_h) \tag{2.21}$$

$$\alpha(t) = \alpha_a cos(\omega_\alpha t + \varphi_\alpha) = \alpha_a cos(2\pi f_\alpha t + \varphi_\alpha) \tag{2.22}$$

where h_a is the heaving amplitude and is defined positive upwards, α_a is the pitch amplitude and is defined positive clockwise, ω_h is the heaving angular frequency ($\omega_h = 2\pi f_h$), f_h is the heaving oscillating frequency, ω_α is the pitching angular frequency ($\omega_h = 2\pi f_\alpha$), f_α is the pitching oscillating frequency, φ_h is the phase angle of the heaving motion and φ_α is the phase angle of the pitching motion.

Based on the definition of reduced frequency eq. 2.18, the airfoil kinematics eq. 2.21 and eq. 2.22 can be rewritten as

$$y(t) = h_a cos(2k\tilde{t} + \varphi_h) \tag{2.23}$$

$$\alpha(t) = \alpha_a cos(2k\tilde{t} + \varphi_\alpha) \tag{2.24}$$

where $\tilde{t}$ is the dimensionless time and is equal to $\tilde{t} = \frac{t_{ref} U_{ref}}{L_{ref}}$.

Natural flyers generally use a combination of pitching and heaving motion rather than a single degree of freedom pitch or heave motion. With combined pitching and heaving motions the parameter space becomes larger, in addition to the pitch and heave amplitudes and oscillating frequencies, one now has to consider the phase angle between the pitch and heave motions. In figure 2.20, this situation is illustrated. Cases (a) and (b) represent the pure heave and pitch modes. In case (c) the airfoil is both pitching and heaving with a phase angle of 90 degrees (pure feathering). In case (d) and (e), the motion of case (c) is duplicated, but with a phase angle different that 90 degrees between the pitch and heave motions.

Above, we just presented the general $2\mathbb{D}$ case. For $3\mathbb{D}$ cases [9, 167], the scenario is far more complex as shown in figure 2.21. The wing-beat kinematics can be described by three positional angles within the stroke plane: (i) flapping about the x axis (rolling or flapping motion) in the wing-fixed coordinate system described by the positional angle ϕ, (ii) rotation of the wing about the z axis (lagging motion) described by the elevation angle θ, and (iii) rotation of the wing about the y axis (feathering motion) described by the angle of attack α. The angle of attack α is used to describe the orientation of a chordwise strip of a beating wing relative to the stroke plane, which may change significantly in the spanwise direction because of the wing torsion often observed in birds and insect flapping flight.

For a general $3\mathbb{D}$ case, definitions of the positional angle, the elevation angle, and the angle of attack, all in radians, are

$$\phi(t) = \sum_{n=0}^{3} \left[\phi_{cn} cos(2n\pi ft) + \phi_{sn} cos(2n\pi ft)\right], \quad n = integer \tag{2.25}$$

$$\theta(t) = \sum_{n=0}^{3} \left[\theta_{cn} cos(2n\pi ft) + \theta_{sn} cos(2n\pi ft)\right], \quad n = integer \tag{2.26}$$

$$\alpha(t) = \sum_{n=0}^{3} \left[\alpha_{cn} cos(2n\pi ft) + \alpha_{sn} cos(2n\pi ft)\right], \quad n = integer \tag{2.27}$$

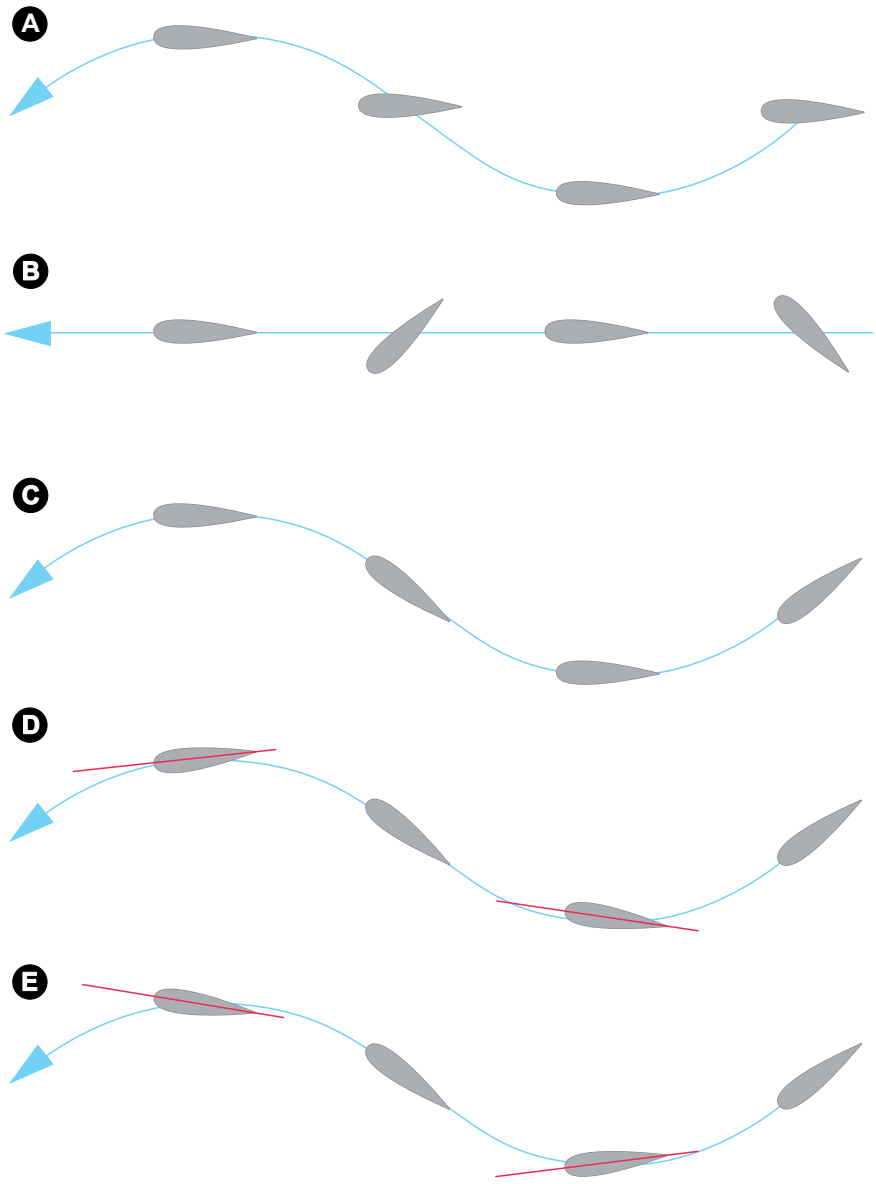

Figure 2.20: *Different possible combination of motions and effect of phase angle in* $2\mathbb{D}$.

Note that t is the time and f is the flapping frequency. The Fourier coefficients ϕ_{cn}, ϕ_{sn}, θ_{cn}, θ_{sn}, α_{cn} and α_{sn}, are determined from empirical kinematic data [114, 188, 213]. Based on the Fourier coefficients gathered by analysis of the kinematics of a hovering hawkmoth [114], the positional, elevation and feathering angle variation for one period are plotted in figure 2.22.

Equations 2.25, 2.26 and 2.27, represent the general $3\mathbb{D}$ case, where most if not all of the coefficients must be obtained from experimental data. However, simpler kinematics can be consider when dealing with $3\mathbb{D}$ cases, which are very similar or analogous to the kinematics in $2\mathbb{D}$ cases. Examples of such kinematics can be, a pure pitching wing, a pure heaving wing, a wing undergoing heaving-and-pitching motions, a wing undergoing lagging (or surging) and pitching motion

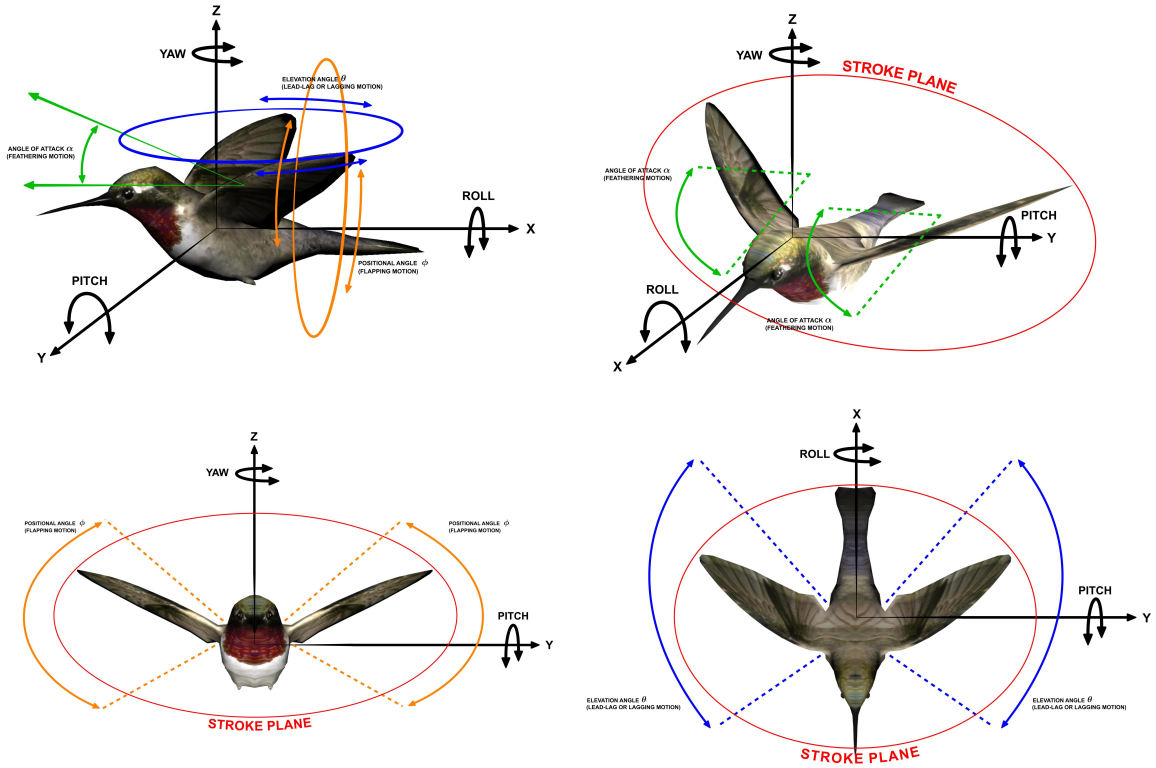

Figure 2.21: 3𝔻 *flapping wing kinematics.*

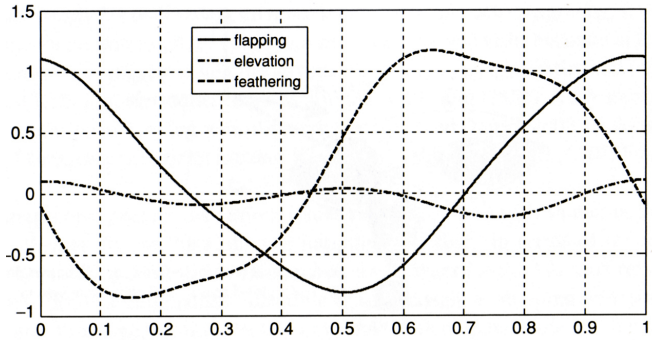

Figure 2.22: *Positional, elevation and feathering angle variations for one period for a hovering hawkmoth [167].*

or a wing rolling about the traveling axis, among others.

Even though 3𝔻 effects are important for predicting low Reynolds number flapping wing aerodynamics, 2𝔻 experiments and computations do provide important insight into the unsteady physics related to flapping wings, that is why a lot of research has been done and is being actively done in the field of 2𝔻 flapping airfoils aerodynamics [108, 140, 196, 198, 207, 216].

Chapter 3

Governing Equations of Fluid Dynamics

The starting point of any numerical simulation are the governing equations of the physics of the problem to be solved. In this chapter, we first present the governing equations of fluid dynamics and their nondimensionalization. Then, we describe their transformation to generalized curvilinear coordinates. And finally, we close this chapter by presenting the governing equations for the case of an incompressible viscous flow.

3.1 Navier-Stokes System of Equations

The equations governing the motion of a fluid can be derived from the statements of the conservation of mass, momentum, and energy [5]. In the most general form, the fluid motion is governed by the time-dependent three-dimensional compressible Navier-Stokes system of equations. For a viscous Newtonian, isotropic fluid in the absence of external forces, mass diffusion, finite-rate chemical reactions, and external heat addition, the strong conservation form of the Navier-Stokes system of equations in compact differential form can be written as

$$\begin{aligned}
\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) &= 0 \\
\frac{\partial (\rho \mathbf{u})}{\partial t} + \nabla \cdot (\rho \mathbf{u}\mathbf{u}) &= -\nabla p + \nabla \cdot \tau \\
\frac{\partial (\rho e_t)}{\partial t} + \nabla \cdot (\rho e_t \mathbf{u}) &= k \nabla \cdot \nabla T - \nabla p \cdot \mathbf{u} + (\nabla \cdot \tau) \cdot \mathbf{u}
\end{aligned}$$

This set of equations can be rewritten in vector form as follows

$$\frac{\partial \mathbf{Q}}{\partial t} + \frac{\partial \mathbf{E}_i}{\partial x} + \frac{\partial \mathbf{F}_i}{\partial y} + \frac{\partial \mathbf{G}_i}{\partial z} = \frac{\partial \mathbf{E}_v}{\partial x} + \frac{\partial \mathbf{F}_v}{\partial y} + \frac{\partial \mathbf{G}_v}{\partial z} \tag{3.1}$$

where $\mathbf{Q}$ is the vector of the conserved flow variables given by

$$\mathbf{Q} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho e_t \end{bmatrix} \tag{3.2}$$

and $\mathbf{E_i} = \mathbf{E_i(Q)}$, $\mathbf{F_i} = \mathbf{F_i(Q)}$ and $\mathbf{G_i} = \mathbf{G_i(Q)}$ are the vectors containing the inviscid fluxes in the x, y and z directions and are given by

$$\mathbf{E_i} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ (\rho e_t + p)\, u \end{bmatrix}, \qquad \mathbf{F_i} = \begin{bmatrix} \rho v \\ \rho vu \\ \rho v^2 + p \\ \rho vw \\ (\rho e_t + p)\, v \end{bmatrix}, \qquad \mathbf{G_i} = \begin{bmatrix} \rho w \\ \rho wu \\ \rho wv \\ \rho w^2 + p \\ (\rho e_t + p)\, w \end{bmatrix} \tag{3.3}$$

where $\mathbf{u}$ is the velocity vector containing the u, v and w velocity components in the x, y and z directions and p, ρ and e_t are the pressure, density and total energy per unit mass respectively.

The vectors $\mathbf{E_v} = \mathbf{E_v(Q)}$, $\mathbf{F_v} = \mathbf{F_v(Q)}$ and $\mathbf{G_v} = \mathbf{G_v(Q)}$ contain the viscous fluxes in the x, y and z directions and are defined as follows

$$\begin{aligned} \mathbf{E_v} &= \begin{bmatrix} 0 \\ \tau_{xx} \\ \tau_{xy} \\ \tau_{xz} \\ u\tau_{xx} + v\tau_{xy} + w\tau_{xz} - q_x \end{bmatrix} \\ \mathbf{F_v} &= \begin{bmatrix} 0 \\ \tau_{yx} \\ \tau_{yy} \\ \tau_{yz} \\ u\tau_{yx} + v\tau_{yy} + w\tau_{yz} - q_y \end{bmatrix} \\ \mathbf{G_v} &= \begin{bmatrix} 0 \\ \tau_{zx} \\ \tau_{zy} \\ \tau_{zz} \\ u\tau_{zx} + v\tau_{zy} + w\tau_{zz} - q_z \end{bmatrix} \end{aligned} \tag{3.4}$$

where the heat fluxes q_x, q_y and q_z are given by the Fourier's law of heat conduction as follows

$$\begin{aligned} q_x &= -k\, \frac{\partial T}{\partial x} \\ q_y &= -k\, \frac{\partial T}{\partial y} \\ q_z &= -k\, \frac{\partial T}{\partial z} \end{aligned} \tag{3.5}$$

and the viscous stresses τ_{xx}, τ_{yy}, τ_{zz}, τ_{xy}, τ_{yx}, τ_{xz}, τ_{zx}, τ_{yz} and τ_{zy}, are given by the following

relationships

$$\begin{aligned}
\tau_{xx} &= \frac{2}{3}\mu\left(2\frac{\partial u}{\partial x} - \frac{\partial v}{\partial y} - \frac{\partial w}{\partial z}\right) \\
\tau_{yy} &= \frac{2}{3}\mu\left(2\frac{\partial v}{\partial y} - \frac{\partial u}{\partial x} - \frac{\partial w}{\partial z}\right) \\
\tau_{zz} &= \frac{2}{3}\mu\left(2\frac{\partial w}{\partial z} - \frac{\partial u}{\partial x} - \frac{\partial v}{\partial y}\right) \\
\tau_{xy} &= \mu\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) \\
\tau_{xz} &= \mu\left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right) \\
\tau_{yz} &= \mu\left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y}\right) \\
\tau_{yx} &= \tau_{xy} \\
\tau_{zx} &= \tau_{xz} \\
\tau_{zy} &= \tau_{yz}
\end{aligned} \tag{3.6}$$

where μ is the laminar viscosity.

Examining closely equations eq. 3.1, eq. 3.2, eq. 3.3 and eq. 3.4 and counting the number of equations and unknowns, we clearly see that we have five equations in terms of seven unknown flow field variables u, v, w, ρ, p, T, and e_t. It is obvious that two additional equations are required to close the system. These two additional equations can be obtained by determining relationships that exist between the thermodynamic variables (p, ρ, T, e_i) through the assumption of thermodynamic equilibrium. Relations of this type are known as equations of state, and they provide a mathematical relationship between two or more state functions (thermodynamic variables). Choosing the specific internal energy e_i and the density ρ as the two independent thermodynamic variables, then equations of state of the form

$$p = p\left(e_i, \rho\right), \qquad T = T\left(e_i, \rho\right) \tag{3.7}$$

are required.

For most problems in aerodynamics and gasdynamics, it is generally reasonable to assume that the gas behaves as a perfect gas (a perfect gas is defined as a gas whose intermolecular forces are negligible), *i.e.*,

$$p = \rho R_g T \tag{3.8}$$

where R_g is the specific gas constant and is equal to $287\frac{m^2}{s^2 K}$ for air. Assuming also that the working gas behaves as a calorically perfect gas (a calorically perfect gas is defined as a perfect gas with constant specific heats), then the following relations hold

$$e_i = c_v T, \qquad h = c_p T, \qquad \gamma = \frac{c_p}{c_v}, \qquad c_v = \frac{R_g}{\gamma - 1}, \qquad c_p = \frac{\gamma R_g}{\gamma - 1} \tag{3.9}$$

where γ is the ratio of specific heats and is equal to 1.4 for air, c_v the specific heat at constant volume, c_p the specific heat at constant pressure and h is the enthalpy. By using eq. 3.8 and eq. 3.9, we obtain the following relations for pressure p and temperature T in the form of eq. 3.7

$$p = (\gamma - 1)\,\rho e_i, \qquad T = \frac{p}{\rho R_g} = \frac{(\gamma - 1)\,e_i}{R_g} \tag{3.10}$$

where the specific internal energy per unit mass $e_i = p/(\gamma - 1)\rho$ is related to the total energy per unit mass e_t by the following relationship,

$$e_t = e_i + \frac{1}{2}\left(u^2 + v^2 + w^2\right) \tag{3.11}$$

In our discussion, it is also necessary to relate the transport properties (μ, k) to the thermodynamic variables. Then, the laminar viscosity μ is computed using Sutherland's formula

$$\mu = \frac{C_1 T^{\frac{3}{2}}}{(T + C_2)} \tag{3.12}$$

where for the case of the air, the constants are $C_1 = 1.458 \times 10^{-6} \frac{kg}{ms\sqrt{K}}$ and $C_2 = 110.4K$.

The thermal conductivity, k, of the fluid is determined from the Prandtl number ($Pr = 0.72$ for air) which in general is assumed to be constant and is equal to

$$k = \frac{c_p \mu}{Pr} \tag{3.13}$$

where c_p and μ are given by equations eq. 3.9 and eq. 3.12 respectively.

The first row in eq. 3.1 corresponds to the continuity equation. Likewise, the second, third and fourth rows are the momentum equations, while the fifth row is the energy equation in terms of total energy per unit mass.

The Navier-Stokes system of equations eq. 3.1, eq. 3.2, eq. 3.3 and eq. 3.4, is a coupled system of nonlinear partial differential equations (PDE), and hence is very difficult to solve analytically. There is no general closed-form solution to this system of equations; hence we look for an approximate solution of this system of equation in a given domain $\mathcal{D}$ with prescribed boundary conditions $\partial\mathcal{D}$ and given initial conditions $\mathcal{D}\mathring{\mathbf{U}}$.

If in eq. 3.1 we set the viscous fluxes $\mathbf{E}_{\boldsymbol{v}} = 0$, $\mathbf{F}_{\boldsymbol{v}} = 0$ and $\mathbf{G}_{\boldsymbol{v}} = 0$, we get the Euler system of equations, which governs inviscid fluid flow. The Euler system of equations is a set of hyperbolic equations while the Navier-Stokes system of equations is a mixed set of hyperbolic (in the inviscid region) and parabolic (in the viscous region) equations. Therefore, time marching algorithms are used to advance the solution in time using discrete time steps.

3.2 Nondimensionalization of the Governing Equations

The governing fluid dynamic equations shown previously may be nondimensionalized to achieve certain objectives. The advantage in doing this is that, firstly, it will provide conditions upon

which dynamic and energetic similarity may be obtained for geometrically similar situations. Secondly, by nondimensionalizing the equations appropriately, the flow variables are normalized so that their values fall between certain prescribed limits such as zero and one. Thirdly, the procedure of nondimensionalization, also allows the solution to be independent of any system of units and helps to reduce the sensitivity of the numerical algorithm to round-off-errors. And finally, by nondimensionalizing the governing equations, characteristic parameters such as Mach number, Reynolds number and Prandtl number can be varied independently. Among many choices, in external flow aerodynamics it is reasonable to normalize with respect to the freestream parameters so that

$$\begin{aligned} \tilde{x} &= \frac{x}{L}, \quad \tilde{y} = \frac{y}{L}, \quad \tilde{z} = \frac{z}{L} \\ \tilde{u} &= \frac{u}{U_\infty}, \quad \tilde{v} = \frac{v}{U_\infty}, \quad \tilde{w} = \frac{w}{U_\infty} \\ \tilde{\rho} &= \frac{\rho}{\rho_\infty}, \quad \tilde{T} = \frac{T}{T_\infty}, \quad \tilde{p} = \frac{p}{\rho_\infty U_\infty^2} \\ \tilde{t} &= \frac{tU_\infty}{L}, \quad \tilde{e_t} = \frac{e_t}{U_\infty^2}, \quad \tilde{\mu} = \frac{\mu}{\mu_\infty} \end{aligned} \tag{3.14}$$

where $\tilde{}$ denotes nondimensional quantities, the subscript $_\infty$ denotes freestream conditions, L is some dimensional reference length (such as the chord of an airfoil or the length of a vehicle), and U_∞ is the magnitude of the freestream velocity. The reference length L is used in defining the nondimensional Reynold's number, this parameter represents the ratio of inertia forces to viscous forces, and is given by

$$Re_L = \frac{\rho_\infty U_\infty L}{\mu_\infty} \tag{3.15}$$

where the freestream laminar viscosity μ_∞ is computed using the freestream temperature T_∞ according to eq. 3.12.

When dealing with high speed compressible flow, it is also useful to introduce the Mach number. The Mach number is a nondimensional parameter that measures the speed of the gas motion in relation to the speed of sound a,

$$a = \left[\left(\frac{\partial p}{\partial \rho}\right)_s\right]^{\frac{1}{2}} = \sqrt{\gamma \frac{p}{\rho}} = \sqrt{\gamma R_g T} \tag{3.16}$$

Then the Mach number M_∞ is given by,

$$M_\infty = \frac{U_\infty}{a} = \frac{U_\infty}{\sqrt{\gamma\,(p/\rho)}} = \frac{U_\infty}{\sqrt{\gamma R_g T}} \tag{3.17}$$

Finally, the remaining nondimensional quantities are defined as follows

$$\begin{aligned}
\tilde{R}_g &= \frac{R_g}{U_\infty^2/T_\infty} = \frac{1}{\gamma M_\infty^2} \\
\tilde{c}_p &= \frac{1}{(\gamma-1)\,M_\infty^2} \\
\tilde{C}_1 &= C_1 \frac{T_\infty^{1/2}}{\mu_\infty} \\
\tilde{C}_2 &= \frac{C_2}{T_\infty}
\end{aligned} \tag{3.18}$$

Now, by simple replacing into the governing equations eq. 3.1 the dimensional quantities by their corresponding nondimensional equivalent, the following nondimensional equations are obtained

$$\frac{\partial \tilde{\mathbf{Q}}}{\partial \tilde{t}} + \frac{\partial \tilde{\mathbf{E}}_{\boldsymbol{i}}}{\partial \tilde{x}} + \frac{\partial \tilde{\mathbf{F}}_{\boldsymbol{i}}}{\partial \tilde{y}} + \frac{\partial \tilde{\mathbf{G}}_{\boldsymbol{i}}}{\partial \tilde{z}} = \frac{\partial \tilde{\mathbf{E}}_{\boldsymbol{v}}}{\partial \tilde{x}} + \frac{\partial \tilde{\mathbf{F}}_{\boldsymbol{v}}}{\partial \tilde{y}} + \frac{\partial \tilde{\mathbf{G}}_{\boldsymbol{v}}}{\partial \tilde{z}} \tag{3.19}$$

where $\tilde{\mathbf{Q}}$ is the vector of the nondimensional conserved flow variables given by

$$\tilde{\mathbf{Q}} = \begin{bmatrix} \tilde{\rho} \\ \tilde{\rho}\tilde{u} \\ \tilde{\rho}\tilde{v} \\ \tilde{\rho}\tilde{w} \\ \tilde{\rho}\tilde{e}_t \end{bmatrix} \tag{3.20}$$

and $\tilde{\mathbf{E}}_{\boldsymbol{i}} = \tilde{\mathbf{E}}_{\boldsymbol{i}}(\tilde{\mathbf{Q}})$, $\tilde{\mathbf{F}}_{\boldsymbol{i}} = \tilde{\mathbf{F}}_{\boldsymbol{i}}(\tilde{\mathbf{Q}})$ and $\tilde{\mathbf{G}}_{\boldsymbol{i}} = \tilde{\mathbf{G}}_{\boldsymbol{i}}(\tilde{\mathbf{Q}})$ are the vectors containing the nondimensional inviscid fluxes in the $\tilde{x}$, $\tilde{y}$ and $\tilde{z}$ directions and are given by

$$\tilde{\mathbf{E}}_{\boldsymbol{i}} = \begin{bmatrix} \tilde{\rho}\tilde{u} \\ \tilde{\rho}\tilde{u}^2 + \tilde{p} \\ \tilde{\rho}\tilde{u}\tilde{v} \\ \tilde{\rho}\tilde{u}\tilde{w} \\ (\tilde{\rho}\tilde{e}_t + \tilde{p})\,\tilde{u}, \end{bmatrix}, \qquad \tilde{\mathbf{F}}_{\boldsymbol{i}} = \begin{bmatrix} \tilde{\rho}\tilde{v} \\ \tilde{\rho}\tilde{v}\tilde{u} \\ \tilde{\rho}\tilde{v}^2 + \tilde{p} \\ \tilde{\rho}\tilde{v}\tilde{w} \\ (\tilde{\rho}\tilde{e}_t + \tilde{p})\,\tilde{v}, \end{bmatrix}, \qquad \tilde{\mathbf{G}}_{\boldsymbol{i}} = \begin{bmatrix} \tilde{\rho}\tilde{w} \\ \tilde{\rho}\tilde{w}\tilde{u} \\ \tilde{\rho}\tilde{w}\tilde{v} \\ \tilde{\rho}\tilde{w}^2 + \tilde{p} \\ (\tilde{\rho}\tilde{e}_t + \tilde{p})\,\tilde{w} \end{bmatrix} \tag{3.21}$$

and $\tilde{\mathbf{E}}_{\boldsymbol{v}} = \tilde{\mathbf{E}}_{\boldsymbol{v}}(\tilde{\mathbf{Q}})$, $\tilde{\mathbf{F}}_{\boldsymbol{v}} = \tilde{\mathbf{F}}_{\boldsymbol{v}}(\tilde{\mathbf{Q}})$ and $\tilde{\mathbf{G}}_{\boldsymbol{v}} = \tilde{\mathbf{G}}_{\boldsymbol{v}}(\tilde{\mathbf{Q}})$ are the vectors containing the nondimensional viscous fluxes in the $\tilde{x}$, $\tilde{y}$ and $\tilde{z}$ directions and are given by

$$
\begin{aligned}
\tilde{\mathbf{E}}_{\boldsymbol{v}} &= \begin{bmatrix} 0 \\ \tilde{\tau}_{xx} \\ \tilde{\tau}_{xy} \\ \tilde{\tau}_{xz} \\ \tilde{u}\tilde{\tau}_{xx} + \tilde{v}\tilde{\tau}_{xy} + \tilde{w}\tilde{\tau}_{xz} - \tilde{q}_x \end{bmatrix} \\
\tilde{\mathbf{F}}_{\boldsymbol{v}} &= \begin{bmatrix} 0 \\ \tilde{\tau}_{yx} \\ \tilde{\tau}_{yy} \\ \tilde{\tau}_{yz} \\ \tilde{u}\tilde{\tau}_{yx} + \tilde{v}\tilde{\tau}_{yy} + \tilde{w}\tilde{\tau}_{yz} - \tilde{q}_y \end{bmatrix} \\
\tilde{\mathbf{G}}_{\boldsymbol{v}} &= \begin{bmatrix} 0 \\ \tilde{\tau}_{zx} \\ \tilde{\tau}_{zy} \\ \tilde{\tau}_{zz} \\ \tilde{u}\tilde{\tau}_{zx} + \tilde{v}\tilde{\tau}_{zy} + \tilde{w}\tilde{\tau}_{zz} - \tilde{q}_z \end{bmatrix}
\end{aligned}
\tag{3.22}
$$

However, in the process of nondimensionalizing the equations, the terms M_∞ and Re_L arises from the nondimensional viscous flux vectors. Therefore, the definition of the heat flux components and the viscous stresses may be modified as follows

$$
\begin{aligned}
\tilde{q}_x &= -\frac{\tilde{\mu}}{(\gamma-1)\,M_\infty^2 Re_L Pr}\,\frac{\partial \tilde{T}}{\partial \tilde{x}} \\
\tilde{q}_y &= -\frac{\tilde{\mu}}{(\gamma-1)\,M_\infty^2 Re_L Pr}\,\frac{\partial \tilde{T}}{\partial \tilde{y}} \\
\tilde{q}_z &= -\frac{\tilde{\mu}}{(\gamma-1)\,M_\infty^2 Re_L Pr}\,\frac{\partial \tilde{T}}{\partial \tilde{z}}
\end{aligned}
\tag{3.23}
$$

and

$$
\begin{aligned}
\tilde{\tau}_{xx} &= \frac{2}{3}\frac{\tilde{\mu}}{Re_L}\left(2\frac{\partial\tilde{u}}{\partial\tilde{x}} - \frac{\partial\tilde{v}}{\partial\tilde{y}} - \frac{\partial\tilde{w}}{\partial\tilde{z}}\right) \\
\tilde{\tau}_{yy} &= \frac{2}{3}\frac{\tilde{\mu}}{Re_L}\left(2\frac{\partial\tilde{v}}{\partial\tilde{y}} - \frac{\partial\tilde{u}}{\partial\tilde{x}} - \frac{\partial\tilde{w}}{\partial\tilde{z}}\right) \\
\tilde{\tau}_{zz} &= \frac{2}{3}\frac{\tilde{\mu}}{Re_L}\left(2\frac{\partial\tilde{w}}{\partial\tilde{z}} - \frac{\partial\tilde{u}}{\partial\tilde{x}} - \frac{\partial\tilde{v}}{\partial\tilde{y}}\right) \\
\tilde{\tau}_{xy} &= \frac{\tilde{\mu}}{Re_L}\left(\frac{\partial\tilde{u}}{\partial\tilde{y}} + \frac{\partial\tilde{v}}{\partial\tilde{x}}\right) \\
\tilde{\tau}_{xz} &= \frac{\tilde{\mu}}{Re_L}\left(\frac{\partial\tilde{u}}{\partial\tilde{z}} + \frac{\partial\tilde{w}}{\partial\tilde{x}}\right) \\
\tilde{\tau}_{yz} &= \frac{\tilde{\mu}}{Re_L}\left(\frac{\partial\tilde{v}}{\partial\tilde{z}} + \frac{\partial\tilde{w}}{\partial\tilde{y}}\right) \\
\tilde{\tau}_{yx} &= \tilde{\tau}_{xy} \\
\tilde{\tau}_{zx} &= \tilde{\tau}_{xz} \\
\tilde{\tau}_{zy} &= \tilde{\tau}_{yz}
\end{aligned}
\tag{3.24}
$$

Finally, by nondimensionalizing the equations of state eq. 3.10, we obtain

$$
\tilde{p} = (\gamma - 1)\,\tilde{\rho}\tilde{e}_i, \qquad \tilde{T} = \frac{\tilde{p}}{\tilde{\rho}\tilde{R}_g} = \frac{(\gamma-1)\,\tilde{e}_i}{\tilde{R}_g} \tag{3.25}
$$

where the nondimensional specific internal energy per unit mass $\tilde{e}_i = \tilde{p}/(\gamma-1)\tilde{\rho}$ is related to the nondimensional total energy per unit mass $\tilde{e}_t$ by the following relationship,

$$
\tilde{e}_t = \tilde{e}_i + \frac{1}{2}\left(\tilde{u}^2 + \tilde{v}^2 + \tilde{w}^2\right) \tag{3.26}
$$

Note that the nondimensional form of the equations given by eq. 3.19, eq. 3.20, eq. 3.21 and eq. 3.22 are identical (except for the˜) to the dimensional form given by equations eq. 3.1, eq. 3.2, eq. 3.3 and eq. 3.4. For the sake of simplicity, the notation ˜ will be dropped for the remainder of this dissertation. Thus, all the equations will be given in nondimensional form unless otherwise specified.

3.3 Transformation of the Governing Equations to Generalized Curvilinear Coordinates

The Navier-Stokes system of equation (eq. 3.1, eq. 3.2, eq. 3.3 and eq. 3.4) are valid for any coordinate system. We have previously expressed these equations in terms of a Cartesian coordinate system. For many applications it is more convenient to use a generalized curvilinear coordinate system. The use of generalized curvilinear coordinates implies that a distorted region in physical space is mapped into a rectangular region in the generalized curvilinear coordinate space (figure 3.1). Often, the transformation is chosen so that the discretized equations are solved in a uniform logically rectangular domain for $2\mathbb{D}$ applications and an equivalent uniform logically hexahedral domain for $3\mathbb{D}$ applications. The transformation shall be such that there is a one-to-one correspondence of the grid points from the physical space (Cartesian coordinates)

to computational space (generalized curvilinear coordinates).

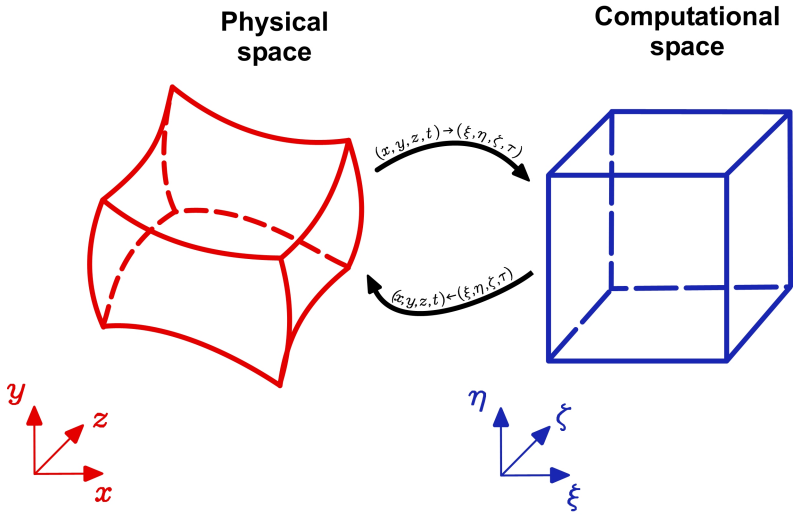

Figure 3.1: *Correspondence between the physical space (Cartesian coordinates) and the computational space (generalized curvilinear coordinates).*

Hereafter, we will describe the general transformation of the nondimensional Navier-Stokes system of equations (eq. 3.19, eq. 3.20, eq. 3.21 and eq. 3.22) given in the previous section between the physical space (Cartesian coordinates) and the computational space (generalized curvilinear coordinates). The governing equations are written in strong conservation form and expressed in terms of the generalized curvilinear coordinates as independent variables, thus the computations are performed in the generalized curvilinear coordinate space.

The governing equations of fluid dynamics are transformed from the physical space $\mathcal{P} = \mathcal{P}(x, y, z, t)$ to the computational space $\mathcal{C} = \mathcal{C}(\xi, \eta, \zeta, \tau)$ by using the following transformations

$$\begin{aligned} \tau &= \tau(t) = t \\ \xi &= \xi(x, y, z, t) \\ \eta &= \eta(x, y, z, t) \\ \zeta &= \zeta(x, y, z, t) \end{aligned} \tag{3.27}$$

where τ is considered to be equal to t and thus the transformation with respect to time is simple defined as $\tau = t$ as shown in eq. 3.27.

Applying the chain rule, the partial derivatives of any quantity $\phi = \phi(x, y, z, t)$ with respect to the Cartesian coordinates can be written as

$$\begin{aligned}
\frac{\partial \phi}{\partial t} &= \frac{\partial \phi}{\partial \tau} + \xi_t \frac{\partial \phi}{\partial \xi} + \eta_t \frac{\partial \phi}{\partial \eta} + \zeta_t \frac{\partial \phi}{\partial \zeta} \\
\frac{\partial \phi}{\partial x} &= \xi_x \frac{\partial \phi}{\partial \xi} + \eta_x \frac{\partial \phi}{\partial \eta} + \zeta_x \frac{\partial \phi}{\partial \zeta} \\
\frac{\partial \phi}{\partial y} &= \xi_y \frac{\partial \phi}{\partial \xi} + \eta_y \frac{\partial \phi}{\partial \eta} + \zeta_y \frac{\partial \phi}{\partial \zeta} \\
\frac{\partial \phi}{\partial z} &= \xi_z \frac{\partial \phi}{\partial \xi} + \eta_z \frac{\partial \phi}{\partial \eta} + \zeta_z \frac{\partial \phi}{\partial \zeta}
\end{aligned} \tag{3.28}$$

Then the governing equations may be transformed from physical space $\mathcal{P}$ to computational space $\mathcal{C}$ by replacing the Cartesian derivatives by the partial derivatives given in eq. 3.28, where the terms $\xi_x, \eta_x, \zeta_x, \xi_y, \eta_y, \zeta_y, \xi_z, \eta_z, \zeta_z, \xi_t, \eta_t$ and ζ_t are called metrics (they represents the ratio of arc lengths in the computational space $\mathcal{C}$ to that of the physical space $\mathcal{P}$) and where ξ_x represents the partial derivative of ξ with respect to x, *i.e.* $\partial \xi / \partial x$, and so forth.

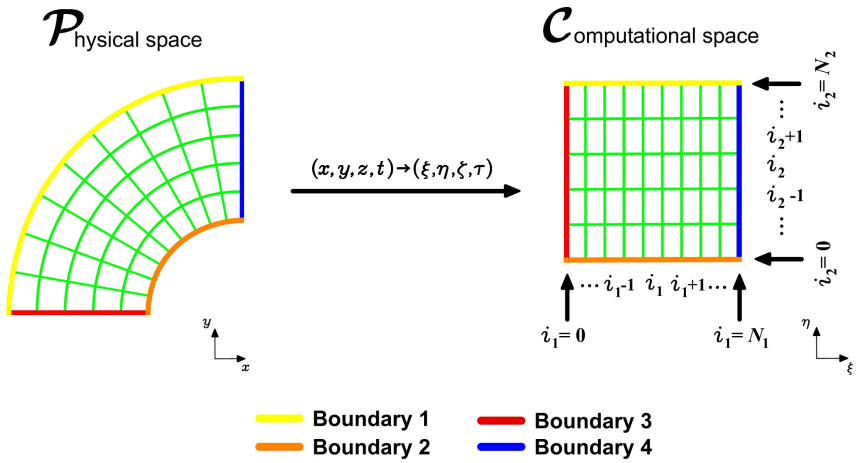

Figure 3.2: *Transformation from physical space to computational space. Left: structured grid in physical space. Right: logically uniform grid in computational space.*

In most cases, the transformation eq. 3.27 from physical space $\mathcal{P}$ to computational space $\mathcal{C}$ is not known analytically, rather it is generated numerically by a grid generation scheme. That is, we usually are provided with just the x, y and z coordinates of the grid points and we numerically generate the metrics using finite differences. The metrics $\xi_x, \eta_x, \zeta_x, \xi_y, \eta_y, \zeta_y, \xi_z, \eta_z, \zeta_z, \xi_t, \eta_t$ and ζ_t appearing in eq. 3.28 can be determined in the following manner. First, we write down the differential expressions of the inverse of the transformation eq. 3.27,

$$\begin{aligned}
dt &= t_\tau d\tau + t_\xi d\xi + t_\eta d\eta + t_\zeta d\zeta \\
dx &= x_\tau d\tau + x_\xi d\xi + x_\eta d\eta + x_\zeta d\zeta \\
dy &= y_\tau d\tau + y_\xi d\xi + y_\eta d\eta + y_\zeta d\zeta \\
dz &= z_\tau d\tau + z_\xi d\xi + z_\eta d\eta + z_\zeta d\zeta
\end{aligned} \tag{3.29}$$

where the inverse of the transformation eq. 3.27 is

$$\begin{aligned} t &= t(\tau) = \tau \\ x &= x(\xi, \eta, \zeta, \tau) \\ y &= y(\xi, \eta, \zeta, \tau) \\ z &= z(\xi, \eta, \zeta, \tau) \end{aligned} \tag{3.30}$$

and recalling that for a grid that is not changing (moving, adapting or deforming)

$$\begin{aligned} &\frac{\partial t}{\partial \tau} = 1 \quad \text{and} \\ &\frac{\partial t}{\partial \xi} = \frac{\partial t}{\partial \eta} = \frac{\partial t}{\partial \zeta} = 0 \quad \text{thus} \\ &dt = d\tau \end{aligned}$$

Expressing eq. 3.29 in matrix form, we obtain

$$\begin{bmatrix} dt \\ dx \\ dy \\ dz \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ x_\tau & x_\xi & x_\eta & x_\zeta \\ y_\tau & y_\xi & y_\eta & y_\zeta \\ z_\tau & z_\xi & z_\eta & z_\zeta \end{bmatrix} \begin{bmatrix} d\tau \\ d\xi \\ d\eta \\ d\zeta \end{bmatrix} \tag{3.31}$$

In a like manner, we proceed with the transformation eq. 3.27, and we obtain the following differential expressions

$$\begin{aligned} d\tau &= dt \\ d\xi &= \xi_t dt + \xi_x dx + \xi_y dy + \xi_z dz \\ d\eta &= \eta_t dt + \eta_x dx + \eta_y dy + \eta_z dz \\ d\zeta &= \zeta_t dt + \zeta_x dx + \zeta_y dy + \zeta_z dz \end{aligned} \tag{3.32}$$

which can be written in matrix form as

$$\begin{bmatrix} d\tau \\ d\xi \\ d\eta \\ d\zeta \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ \xi_t & \xi_x & \xi_y & \xi_z \\ \eta_t & \eta_x & \eta_y & \eta_z \\ \zeta_t & \zeta_x & \zeta_y & \zeta_z \end{bmatrix} \begin{bmatrix} dt \\ dx \\ dy \\ dz \end{bmatrix} \tag{3.33}$$

By relating the differential expressions eq. 3.33 of the transformation eq. 3.27 to the differential expressions eq. 3.31 of the transformation eq. 3.30, so that the metrics

$$\xi_x, \eta_x, \zeta_x, \xi_y, \eta_y, \zeta_y, \xi_z, \eta_z, \zeta_z, \xi_t, \eta_t, \zeta_t$$

can be found, we conclude that

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ \xi_t & \xi_x & \xi_y & \xi_z \\ \eta_t & \eta_x & \eta_y & \eta_z \\ \zeta_t & \zeta_x & \zeta_y & \zeta_z \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ x_\tau & x_\xi & x_\eta & x_\zeta \\ y_\tau & y_\xi & y_\eta & y_\zeta \\ z_\tau & z_\xi & z_\eta & z_\zeta \end{bmatrix}^{-1} \tag{3.34}$$

This yields the following metrics relationships

$$\begin{aligned}
\xi_x &= J_x\left(y_\eta z_\zeta - y_\zeta z_\eta\right)\\
\xi_y &= J_x\left(x_\zeta z_\eta - x_\eta z_\zeta\right)\\
\xi_z &= J_x\left(x_\eta y_\zeta - x_\zeta y_\eta\right)\\
\xi_t &= -\left(\tau_t x_\tau \xi_x + \tau_t y_\tau \xi_y + \tau_t z_\tau \xi_z\right)\\
\eta_x &= J_x\left(y_\zeta z_\xi - y_\xi z_\zeta\right)\\
\eta_y &= J_x\left(x_\xi z_\zeta - x_\zeta z_\xi\right)\\
\eta_z &= J_x\left(x_\zeta y_\xi - x_\xi y_\zeta\right)\\
\eta_t &= -\left(\tau_t x_\tau \eta_x + \tau_t y_\tau \eta_y + \tau_t z_\tau \eta_z\right)\\
\zeta_x &= J_x\left(y_\xi z_\eta - y_\eta z_\xi\right)\\
\zeta_y &= J_x\left(x_\eta z_\xi - x_\xi z_\eta\right)\\
\zeta_z &= J_x\left(x_\xi y_\eta - x_\eta y_\xi\right)\\
\zeta_t &= -\left(\tau_t x_\tau \zeta_x + \tau_t y_\tau \zeta_y + \tau_t z_\tau \zeta_z\right)
\end{aligned} \tag{3.35}$$

For ξ_t, η_t and ζ_t the following values are obtained after some manipulation

$$\begin{aligned}
\xi_t &= J_x\left[x_\tau\left(y_\zeta z_\eta - y_\eta z_\zeta\right) + y_\tau\left(x_\eta z_\zeta - x_\zeta z_\eta\right) + z_\tau\left(x_\zeta y_\eta - x_\eta y_\zeta\right)\right]\\
\eta_t &= J_x\left[x_\tau\left(y_\xi z_\zeta - y_\zeta z_\xi\right) + y_\tau\left(x_\zeta z_\xi - x_\xi z_\zeta\right) + z_\tau\left(x_\xi y_\zeta - x_\zeta y_\xi\right)\right]\\
\zeta_t &= J_x\left[x_\tau\left(y_\eta z_\xi - y_\xi z_\eta\right) + y_\tau\left(x_\xi z_\eta - x_\eta z_\xi\right) + z_\tau\left(x_\eta y_\xi - x_\xi y_\eta\right)\right]
\end{aligned} \tag{3.36}$$

In eq. 3.35 and eq. 3.36, J_x is the determinant of the Jacobian matrix of the transformation defined by

$$J_x = \left|\frac{\partial\left(\xi,\eta,\zeta\right)}{\partial\left(x,y,z\right)}\right|$$

or

$$J_x = \frac{1}{x_\xi\left(y_\eta z_\zeta - y_\zeta z_\eta\right) - x_\eta\left(y_\xi z_\zeta - y_\zeta z_\xi\right) + x_\zeta\left(y_\xi z_\eta - y_\eta z_\xi\right)} \tag{3.37}$$

which can be interpreted as the ratio of the areas (volumes in $3\mathbb{D}$) in the computational space $\mathcal{C}$ to that of the physical space $\mathcal{P}$.

Once relations for the metrics and for the Jacobian of the transformation are determined, the governing equations eq. 3.19 are then written in strong conservation form as

$$\frac{\partial\hat{\mathbf{Q}}}{\partial t} + \frac{\partial\hat{\mathbf{E}}_{\boldsymbol{i}}}{\partial\xi} + \frac{\partial\hat{\mathbf{F}}_{\boldsymbol{i}}}{\partial\eta} + \frac{\partial\hat{\mathbf{G}}_{\boldsymbol{i}}}{\partial\zeta} = \frac{\partial\hat{\mathbf{E}}_{\boldsymbol{v}}}{\partial\xi} + \frac{\partial\hat{\mathbf{F}}_{\boldsymbol{v}}}{\partial\eta} + \frac{\partial\hat{\mathbf{G}}_{\boldsymbol{v}}}{\partial\zeta} \tag{3.38}$$

where

$$
\begin{aligned}
\hat{\mathbf{Q}} &= \frac{\mathbf{Q}}{J_x} \\
\hat{\mathbf{E}}_{\boldsymbol{i}} &= \frac{1}{J_x}\left(\xi_t \mathbf{Q} + \xi_x \mathbf{E}_{\boldsymbol{i}} + \xi_y \mathbf{F}_{\boldsymbol{i}} + \xi_z \mathbf{G}_{\boldsymbol{i}}\right) \\
\hat{\mathbf{F}}_{\boldsymbol{i}} &= \frac{1}{J_x}\left(\eta_t \mathbf{Q} + \eta_x \mathbf{E}_{\boldsymbol{i}} + \eta_y \mathbf{F}_{\boldsymbol{i}} + \eta_z \mathbf{G}_{\boldsymbol{i}}\right) \\
\hat{\mathbf{G}}_{\boldsymbol{i}} &= \frac{1}{J_x}\left(\zeta_t \mathbf{Q} + \zeta_x \mathbf{E}_{\boldsymbol{i}} + \zeta_y \mathbf{F}_{\boldsymbol{i}} + \zeta_z \mathbf{G}_{\boldsymbol{i}}\right) \\
\hat{\mathbf{E}}_{\boldsymbol{v}} &= \frac{1}{J_x}\left(\xi_x \mathbf{E}_{\boldsymbol{v}} + \xi_y \mathbf{F}_{\boldsymbol{v}} + \xi_z \mathbf{G}_{\boldsymbol{v}}\right) \\
\hat{\mathbf{E}}_{\boldsymbol{v}} &= \frac{1}{J_x}\left(\eta_x \mathbf{E}_{\boldsymbol{v}} + \eta_y \mathbf{F}_{\boldsymbol{v}} + \eta_z \mathbf{G}_{\boldsymbol{v}}\right) \\
\hat{\mathbf{E}}_{\boldsymbol{v}} &= \frac{1}{J_x}\left(\zeta_x \mathbf{E}_{\boldsymbol{v}} + \zeta_y \mathbf{F}_{\boldsymbol{v}} + \zeta_z \mathbf{G}_{\boldsymbol{v}}\right)
\end{aligned}
\tag{3.39}
$$

The viscous stresses given by eq. 3.24 in the transformed computational space are

$$
\begin{aligned}
\hat{\tau}_{xx} &= \frac{2}{3}\frac{\mu}{Re_L}[2\left(\xi_x u_\xi + \eta_x u_\eta + \zeta_x u_\zeta\right) - \left(\xi_y v_\xi + \eta_y v_\eta + \zeta_y v_\zeta\right)\ldots \\
&\qquad \ldots - \left(\xi_z w_\xi + \eta_z w_\eta + \zeta_z w_\zeta\right)] \\
\hat{\tau}_{yy} &= \frac{2}{3}\frac{\mu}{Re_L}[2\left(\xi_y v_\xi + \eta_y v_\eta + \zeta_y v_\zeta\right) - \left(\xi_x u_\xi + \eta_x u_\eta + \zeta_x u_\zeta\right)\ldots \\
&\qquad \ldots - \left(\xi_z w_\xi + \eta_z w_\eta + \zeta_z w_\zeta\right)] \\
\hat{\tau}_{zz} &= \frac{2}{3}\frac{\mu}{Re_L}[2\left(\xi_z w_\xi + \eta_z w_\eta + \zeta_z w_\zeta\right) - \left(\xi_x u_\xi + \eta_x u_\eta + \zeta_x u_\zeta\right)\ldots \\
&\qquad \ldots - \left(\xi_y v_\xi + \eta_y v_\eta + \zeta_y v_\zeta\right)]
\end{aligned}
\tag{3.40}
$$

$$
\begin{aligned}
\hat{\tau}_{xy} = \hat{\tau}_{yx} &= \frac{\mu}{Re_L}(\xi_y u_\xi + \eta_y u_\eta + \zeta_y u_\zeta + \xi_x v_\xi + \eta_x v_\eta + \zeta_x v_\zeta) \\
\hat{\tau}_{xz} = \hat{\tau}_{zx} &= \frac{\mu}{Re_L}(\xi_z u_\xi + \eta_z u_\eta + \zeta_z u_\zeta + \xi_x w_\xi + \eta_x w_\eta + \zeta_x w_\zeta) \\
\hat{\tau}_{yz} = \hat{\tau}_{zy} &= \frac{\mu}{Re_L}(\xi_z v_\xi + \eta_z v_\eta + \zeta_z v_\zeta + \xi_y w_\xi + \eta_y w_\eta + \zeta_y w_\zeta)
\end{aligned}
$$

and the heat flux components given by eq. 3.23 in the computational space are

$$
\begin{aligned}
\hat{q}_x &= -\frac{\mu}{\left(\gamma-1\right)M_\infty^2 Re_L Pr}\ \left(\xi_x T_\xi + \eta_x T_\eta + \zeta_x T_\zeta\right) \\
\hat{q}_y &= -\frac{\mu}{\left(\gamma-1\right)M_\infty^2 Re_L Pr}\ \left(\xi_y T_\xi + \eta_y T_\eta + \zeta_y T_\zeta\right) \\
\hat{q}_z &= -\frac{\mu}{\left(\gamma-1\right)M_\infty^2 Re_L Pr}\ \left(\xi_z T_\xi + \eta_z T_\eta + \zeta_z T_\zeta\right)
\end{aligned}
\tag{3.41}
$$

Equations eq. 3.38 and eq. 3.39 are the generic form of the governing equations written in strong conservation form in the transformed computational space $\mathcal{C}$ (see [14], [85] and [181] for a detailed derivation). The coordinate transformation presented in this section, follows the same development proposed by Viviand [202] and Vinokur [201], where they show that the governing equations

of fluid dynamics can be put back into strong conservation form after a coordinate transformation has been applied.

Comparing the original governing equations eq. 3.19, eq. 3.20, eq. 3.21 and eq. 3.22 and the transformed equations eq. 3.38 and eq. 3.39, it is obvious that the transformed equations are more complicated than the original equations. Thus, a trade-off is introduced whereby advantages gained by using the generalized curvilinear coordinates are somehow counterbalanced by the resultant complexity of the equations. However, the advantages (such as the capability of using standard finite differences schemes and solving the equations in a uniform rectangular logically grid) by far outweigh the complexity of the transformed governing equations.

One final word of caution. The strong conservation form of the governing equations in the transformed computational space $\mathcal{C}$ is a convenient form for applying finite difference schemes. However, when using this form of the equations, extreme care must be exercised if the grid is changing (that is moving, adapting or deforming). In this case, a constraint on the way the metrics are differenced, called the geometric conservation law or GCL (see [50], [55] and [185]), must be satisfied in order to prevent additional errors from being introduced into the solution.

3.4 Simplification of the Navier-Stokes System of Equations: Incompressible Viscous Flow Case

Equations eq. 3.1, eq. 3.2, eq. 3.3 and eq. 3.4 with an appropriate equation of state and boundary and initial conditions, governs the unsteady three-dimensional motion of a viscous Newtonian, compressible fluid. In many applications the fluid density may be assumed to be constant. This is true not only for liquids, whose compressibility may be neglected, but also for gases if the Mach number is below 0.3 [6, 53]; such flows are said to be incompressible. If the flow is also isothermal, the viscosity is also constant. In this case, the dimensional governing equations in primitive variable formulation (u, v, w, p) and written in compact conservative differential form reduce to the following set

$$\nabla \cdot (\mathbf{u}) = 0$$
$$\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{u}\mathbf{u}) = \frac{-\nabla p}{\rho} + \nu \nabla^2 \mathbf{u}$$

where ν is the kinematic viscosity and is equal $\nu = \mu/\rho$. The same set of equations in nondimensional form is written as follows

$$\nabla \cdot (\mathbf{u}) = 0$$
$$\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{u}\mathbf{u}) = -\nabla p + \frac{1}{Re_L} \nabla^2 \mathbf{u}$$

which can be also written in nonconservative form (or advective/convective form [60])

$$\nabla \cdot \mathbf{u} = 0$$
$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = -\nabla p + \frac{1}{Re_L} \nabla^2 \mathbf{u}$$

or in expanded three-dimensional Cartesian coordinates

$$
\begin{aligned}
\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}+\frac{\partial w}{\partial z} &= 0 \\
\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}+w\frac{\partial u}{\partial z} &= -\frac{\partial p}{\partial x}+\frac{1}{Re_L}\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}+\frac{\partial^2 u}{\partial z^2}\right) \\
\frac{\partial v}{\partial t}+u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}+w\frac{\partial v}{\partial z} &= -\frac{\partial p}{\partial x}+\frac{1}{Re_L}\left(\frac{\partial^2 v}{\partial x^2}+\frac{\partial^2 v}{\partial y^2}+\frac{\partial^2 v}{\partial z^2}\right) \\
\frac{\partial w}{\partial t}+u\frac{\partial w}{\partial x}+v\frac{\partial w}{\partial y}+w\frac{\partial w}{\partial z} &= -\frac{\partial p}{\partial x}+\frac{1}{Re_L}\left(\frac{\partial^2 w}{\partial x^2}+\frac{\partial^2 w}{\partial y^2}+\frac{\partial^2 w}{\partial z^2}\right)
\end{aligned}
\tag{3.42}
$$

This form (the advective/convective form), provides the simplest form for discretization and is widely used when implementing numerical methods for solving the incompressible Navier-Stokes equations, as noted by Gresho [60].

Equation eq. 3.42 governs the unsteady three-dimensional motion of a viscous, incompressible and isothermal flow. This simplification is generally not of a great value, as the equations are hardly any simpler to solve. However, the computing effort may be much smaller than for the full equations (due to the reduction of the unknowns and the fact that the energy equation is decoupled from the system of equation), which is a justification for such a simplification. The set of equations eq. 3.42 can be rewritten in vector form as follow

$$
\frac{\partial \mathbf{Q}}{\partial t}+\frac{\partial \mathbf{E_i}}{\partial x}+\frac{\partial \mathbf{F_i}}{\partial y}+\frac{\partial \mathbf{G_i}}{\partial z}=\frac{\partial \mathbf{E_v}}{\partial x}+\frac{\partial \mathbf{F_v}}{\partial y}+\frac{\partial \mathbf{G_v}}{\partial z} \tag{3.43}
$$

where $\mathbf{Q}$ is the vector containing the primitive variables and is given by

$$
\mathbf{Q}=\begin{bmatrix} 0 \\ u \\ v \\ w \end{bmatrix} \tag{3.44}
$$

and $\mathbf{E_i}$, $\mathbf{F_i}$ and $\mathbf{G_i}$ are the vectors containing the inviscid fluxes in the x, y and z directions and are given by

$$
\mathbf{E_i}=\begin{bmatrix} u \\ u^2+p \\ uv \\ uw \end{bmatrix}, \qquad \mathbf{F_i}=\begin{bmatrix} v \\ vu \\ v^2+p \\ vw \end{bmatrix}, \qquad \mathbf{G_i}=\begin{bmatrix} w \\ wu \\ wv \\ w^2+p \end{bmatrix} \tag{3.45}
$$

The viscous fluxes in the x, y and z directions, $\mathbf{E_v}$, $\mathbf{F_v}$ and $\mathbf{G_v}$ respectively, are defined as follows

$$
\mathbf{E_v}=\begin{bmatrix} 0 \\ \tau_{xx} \\ \tau_{xy} \\ \tau_{xz} \end{bmatrix}, \qquad \mathbf{F_v}=\begin{bmatrix} 0 \\ \tau_{yx} \\ \tau_{yy} \\ \tau_{yz} \end{bmatrix}, \qquad \mathbf{G_v}=\begin{bmatrix} 0 \\ \tau_{zx} \\ \tau_{zy} \\ \tau_{zz} \end{bmatrix} \tag{3.46}
$$

Since we made the assumptions of an incompressible flow, appropriate nondimensional terms and

expressions for shear stresses must be used, these expressions are given as follows

$$\begin{aligned}
\tau_{xx} &= \frac{2}{Re_L}\frac{\partial u}{\partial x} \\
\tau_{yy} &= \frac{2}{Re_L}\frac{\partial v}{\partial y} \\
\tau_{zz} &= \frac{2}{Re_L}\frac{\partial w}{\partial z} \\
\tau_{xy} &= \frac{1}{Re_L}\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) \\
\tau_{xz} &= \frac{1}{Re_L}\left(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z}\right) \\
\tau_{yz} &= \frac{1}{Re_L}\left(\frac{\partial w}{\partial y} + \frac{\partial v}{\partial z}\right) \\
\tau_{yx} &= \tau_{xy} \\
\tau_{zx} &= \tau_{xz} \\
\tau_{zy} &= \tau_{yz}
\end{aligned} \tag{3.47}$$

Following the procedure presented in the previous section, the nondimensional incompressible Navier-Stokes system of equations eq. 3.43 in the computational space $\mathcal{C}$ is expressed as

$$\frac{\partial \hat{\mathbf{Q}}}{\partial t} + \frac{\partial \hat{\mathbf{E}}_{\boldsymbol{i}}}{\partial \xi} + \frac{\partial \hat{\mathbf{F}}_{\boldsymbol{i}}}{\partial \eta} + \frac{\partial \hat{\mathbf{G}}_{\boldsymbol{i}}}{\partial \zeta} = \frac{\partial \hat{\mathbf{E}}_{\boldsymbol{v}}}{\partial \xi} + \frac{\partial \hat{\mathbf{F}}_{\boldsymbol{v}}}{\partial \eta} + \frac{\partial \hat{\mathbf{G}}_{\boldsymbol{v}}}{\partial \zeta} \tag{3.48}$$

where

$$\begin{aligned}
\hat{\mathbf{Q}} &= \frac{\mathbf{Q}}{J_x} \\
\hat{\mathbf{E}}_{\boldsymbol{i}} &= \frac{1}{J_x}\left(\xi_x \mathbf{E}_{\boldsymbol{i}} + \xi_y \mathbf{F}_{\boldsymbol{i}} + \xi_z \mathbf{G}_{\boldsymbol{i}}\right) \\
\hat{\mathbf{F}}_{\boldsymbol{i}} &= \frac{1}{J_x}\left(\eta_x \mathbf{E}_{\boldsymbol{i}} + \eta_y \mathbf{F}_{\boldsymbol{i}} + \eta_z \mathbf{G}_{\boldsymbol{i}}\right) \\
\hat{\mathbf{G}}_{\boldsymbol{i}} &= \frac{1}{J_x}\left(\zeta_x \mathbf{E}_{\boldsymbol{i}} + \zeta_y \mathbf{F}_{\boldsymbol{i}} + \zeta_z \mathbf{G}_{\boldsymbol{i}}\right) \\
\hat{\mathbf{E}}_{\boldsymbol{v}} &= \frac{1}{J_x}\left(\xi_x \mathbf{E}_{\boldsymbol{v}} + \xi_y \mathbf{F}_{\boldsymbol{v}} + \xi_z \mathbf{G}_{\boldsymbol{v}}\right) \\
\hat{\mathbf{F}}_{\boldsymbol{v}} &= \frac{1}{J_x}\left(\eta_x \mathbf{E}_{\boldsymbol{v}} + \eta_y \mathbf{F}_{\boldsymbol{v}} + \eta_z \mathbf{G}_{\boldsymbol{v}}\right) \\
\hat{\mathbf{G}}_{\boldsymbol{v}} &= \frac{1}{J_x}\left(\zeta_x \mathbf{E}_{\boldsymbol{v}} + \zeta_y \mathbf{F}_{\boldsymbol{v}} + \zeta_z \mathbf{G}_{\boldsymbol{v}}\right)
\end{aligned} \tag{3.49}$$

In eq. 3.49, $\hat{\mathbf{Q}}$ is the vector containing the primitive variables and $\hat{\mathbf{E}}_{\boldsymbol{i}}$, $\hat{\mathbf{F}}_{\boldsymbol{i}}$ and $\hat{\mathbf{G}}_{\boldsymbol{i}}$ are the vectors containing the inviscid fluxes in the ξ, η and ζ directions respectively, and are given by

$$\begin{aligned}
\hat{\mathbf{Q}} &= \frac{1}{J_x}\begin{bmatrix}0\\u\\v\\w\end{bmatrix}, \qquad \hat{\mathbf{E}}_{\boldsymbol{i}} = \frac{1}{J_x}\begin{bmatrix}U\\uU+p\xi_x\\vU+p\xi_y\\wU+p\xi_z\end{bmatrix}, \\
\hat{\mathbf{F}}_{\boldsymbol{i}} &= \frac{1}{J_x}\begin{bmatrix}V\\uV+p\eta_x\\vV+p\eta_y\\wV+p\eta_z\end{bmatrix}, \qquad \hat{\mathbf{G}}_{\boldsymbol{i}} = \frac{1}{J_x}\begin{bmatrix}W\\uW+p\zeta_x\\vW+p\zeta_y\\wW+p\zeta_z\end{bmatrix}
\end{aligned} \tag{3.50}$$

where U, V and W are the contravariant velocities

$$U = u\xi_x + v\xi_y + w\xi_z, \quad V = u\eta_x + v\eta_y + w\eta_z, \quad W = u\zeta_x + v\zeta_y + w\zeta_z$$

The shear stresses given by eq. 3.47 expressed in the computational space $\mathcal{C}$ are as follow

$$\begin{aligned}
\tau_{xx} &= \frac{2}{Re_L}\left(\xi_x u_\xi + \eta_x u_\eta + \zeta_x u_\zeta\right) \\
\tau_{yy} &= \frac{2}{Re_L}\left(\xi_y v_\xi + \eta_y v_\eta + \zeta_y v_\zeta\right) \\
\tau_{zz} &= \frac{2}{Re_L}\left(\xi_z w_\xi + \eta_z w_\eta + \zeta_z w_\zeta\right) \\
\tau_{xy} &= \frac{1}{Re_L}\left(\xi_y u_\xi + \eta_y u_\eta + \zeta_y u_\zeta + \xi_x v_\xi + \eta_x v_\eta + \zeta_x v_\zeta\right) \\
\tau_{xz} &= \frac{1}{Re_L}\left(\xi_z u_\xi + \eta_z u_\eta + \zeta_z u_\zeta + \xi_x w_\xi + \eta_x w_\eta + \zeta_x w_\zeta\right) \\
\tau_{yz} &= \frac{1}{Re_L}\left(\xi_y w_\xi + \eta_y w_\eta + \zeta_y w_\zeta + \xi_z v_\xi + \eta_z v_\eta + \zeta_z v_\zeta\right) \\
\tau_{yx} &= \tau_{xy} \\
\tau_{zx} &= \tau_{xz} \\
\tau_{zy} &= \tau_{yz}
\end{aligned} \tag{3.51}$$

Substituting the expressions for the shear stresses given by eq. 3.51 into the viscous flux vectors $\hat{\mathbf{E}}_{\boldsymbol{v}}$, $\hat{\mathbf{F}}_{\boldsymbol{v}}$ and $\hat{\mathbf{G}}_{\boldsymbol{v}}$ (given by eq. 3.49) in the ξ, η and ζ directions respectively, we obtain the following equations

$$
\begin{aligned}
\hat{\mathbf{E}}_{\boldsymbol{v}} &= \frac{1}{J_x Re_L}\begin{bmatrix} 0 \\ a_1u_\xi + b_1u_\eta - c_1v_\eta + c_2w_\eta + b_2u_\zeta - d_1v_\zeta + d_2w_\zeta \\ a_1v_\xi + c_1u_\eta + b_1v_\eta - c_3w_\eta + d_1u_\zeta + b_2v_\zeta - d_3w_\zeta \\ a_1w_\xi - c_2u_\eta + c_3v_\eta + b_1w_\eta - d_2u_\zeta + d_3v_\zeta + b_2w_\zeta \end{bmatrix} \\
\hat{\mathbf{F}}_{\boldsymbol{v}} &= \frac{1}{J_x Re_L}\begin{bmatrix} 0 \\ a_2u_\eta + b_1u_\xi + c_1v_\xi - c_2w_\xi + b_2u_\zeta - e_1v_\zeta + e_2w_\zeta \\ a_2v_\eta - c_1u_\xi + b_1v_\xi + c_3w_\xi + e_1u_\zeta + b_3v_\zeta - e_3w_\zeta \\ a_2w_\eta + c_2u_\xi - c_3v_\xi + b_1w_\xi - e_2u_\zeta + e_3v_\zeta + b_3w_\zeta \end{bmatrix} \\
\hat{\mathbf{G}}_{\boldsymbol{v}} &= \frac{1}{J_x Re_L}\begin{bmatrix} 0 \\ a_3u_\zeta + b_2u_\xi + d_1v_\xi - d_2w_\xi + b_3u_\eta + e_1v_\eta - e_2w_\eta \\ a_3v_\zeta - c_4u_\xi + b_2v_\xi + d_3w_\xi - e_1u_\eta + b_3v_\eta + e_3w_\eta \\ a_3w_\zeta + d_2u_\xi - d_3v_\xi + b_2w_\xi + c_8u_\eta - e_3v_\eta + b_3w_\eta \end{bmatrix}
\end{aligned}
\tag{3.52}
$$

where

$$
\begin{aligned}
&a_1 = \xi_x^2 + \xi_y^2 + \xi_z^2, \qquad a_2 = \eta_x^2 + \eta_y^2 + \eta_z^2, \qquad a_3 = \zeta_x^2 + \zeta_y^2 + \zeta_z^2, \\
&b_1 = \xi_x\eta_x + \xi_y\eta_y + \xi_z\eta_z, \qquad b_2 = \xi_x\zeta_x + \xi_y\zeta_y + \xi_z\zeta_z, \\
&b_3 = \zeta_x\eta_x + \zeta_y\eta_y + \zeta_z\eta_z, \\
&c_1 = \xi_x\eta_y - \eta_x\xi_y, \qquad c_2 = \eta_x\xi_z - \xi_x\eta_z, \qquad c_3 = \xi_y\eta_z - \eta_y\xi_z, \\
&d_1 = \xi_x\zeta_y - \zeta_x\xi_y, \qquad d_2 = \zeta_x\xi_z - \xi_x\zeta_z, \qquad d_3 = \xi_y\zeta_z - \zeta_y\xi_z, \\
&e_1 = \eta_x\zeta_y - \zeta_x\eta_y, \qquad e_2 = \zeta_x\eta_z - \eta_x\zeta_z, \qquad e_3 = \eta_y\zeta_z - \zeta_y\eta_z
\end{aligned}
\tag{3.53}
$$

equations eq. 3.52 and eq. 3.53 written in a more compact way, can be expressed as

$$
\begin{aligned}
\hat{\mathbf{E}}_{\boldsymbol{v}} &= \frac{1}{J_x Re_L}\begin{bmatrix} 0 \\ (\nabla\xi\cdot\nabla\xi)\,u_\xi + (\nabla\xi\cdot\nabla\eta)\,u_\eta + (\nabla\xi\cdot\nabla\zeta)\,u_\zeta \\ (\nabla\xi\cdot\nabla\xi)\,v_\xi + (\nabla\xi\cdot\nabla\eta)\,v_\eta + (\nabla\xi\cdot\nabla\zeta)\,v_\zeta \\ (\nabla\xi\cdot\nabla\xi)\,w_\xi + (\nabla\xi\cdot\nabla\eta)\,w_\eta + (\nabla\xi\cdot\nabla\zeta)\,w_\zeta \end{bmatrix} \\
\hat{\mathbf{F}}_{\boldsymbol{v}} &= \frac{1}{J_x Re_L}\begin{bmatrix} 0 \\ (\nabla\eta\cdot\nabla\xi)\,u_\xi + (\nabla\eta\cdot\nabla\eta)\,u_\eta + (\nabla\eta\cdot\nabla\zeta)\,u_\zeta \\ (\nabla\eta\cdot\nabla\xi)\,v_\xi + (\nabla\eta\cdot\nabla\eta)\,v_\eta + (\nabla\eta\cdot\nabla\zeta)\,v_\zeta \\ (\nabla\eta\cdot\nabla\xi)\,w_\xi + (\nabla\eta\cdot\nabla\eta)\,w_\eta + (\nabla\eta\cdot\nabla\zeta)\,w_\zeta \end{bmatrix} \\
\hat{\mathbf{G}}_{\boldsymbol{v}} &= \frac{1}{J_x Re_L}\begin{bmatrix} 0 \\ (\nabla\zeta\cdot\nabla\xi)\,u_\xi + (\nabla\zeta\cdot\nabla\eta)\,u_\eta + (\nabla\zeta\cdot\nabla\zeta)\,u_\zeta \\ (\nabla\zeta\cdot\nabla\xi)\,v_\xi + (\nabla\zeta\cdot\nabla\eta)\,v_\eta + (\nabla\zeta\cdot\nabla\zeta)\,v_\zeta \\ (\nabla\zeta\cdot\nabla\xi)\,w_\xi + (\nabla\zeta\cdot\nabla\eta)\,w_\eta + (\nabla\zeta\cdot\nabla\zeta)\,w_\zeta \end{bmatrix}
\end{aligned}
\tag{3.54}
$$

Equation eq. 3.48, together with eq. 3.49, eq. 3.50 and eq. 3.54, are the governing equations of an incompressible viscous flow written in strong conservation form in the transformed computational space $\mathcal{C}$. Hence, we look for an approximate solution of this set of equations in a given domain $\mathcal{D}$ with prescribed boundary conditions $\partial\mathcal{D}$ and given initial conditions $\mathcal{D}\mathring{\mathbf{U}}$. So far, we have just presented the governing equations; in the following chapters the grid generation method as well as the numerical scheme for solving the governing equations will be explained.

Chapter 4

On Structured Overlapping Grids

A major task for calculating the approximate numerical solution of a set of partial differential equations (PDEs) on complex domains, is the problem of mesh or grid generation. Different methods of spatial discretization exist, including Cartesian meshes, unstructured meshes and block structured body-fitted conforming grids. In this chapter, the structured overlapping grids method (which falls within the block structured body-fitted conforming grids classification) is briefly reviewed and discussed in the context of a methodology for the solution and analysis of flows around complex geometries and moving/deforming bodies.

4.1 Approaches to Grid Generation

Grid generation can be defined as the process of breaking up a continuous physical domain into smaller discrete sub-domains, in order to compute the numerical approximate solution of a PDE. The grid generation methods can be classified as structured (body-fitted grids), unstructured (body-fitted grids) or Cartesian (non-body-fitted grids). Hereafter, the different grid generation methods will be briefly surveyed and presented. Since the purpose of this section is to present various techniques to generate grids, the detailed and theoretical derivations for those techniques will be avoided as much as possible.

There is a large body of literature [50, 185, 186] and software packages [137, 162] dealing with structured grid generation. Structured grid methods take their name from the fact that the grid is laid out in a regular repeating pattern called a block. Strictly speaking, in a structured grid the computational domain is selected to be rectangular in shape where the interior grid points are distributed along grid lines, therefore, the grid points can be identified easily with reference to the appropriate grid lines. These types of grids use quadrilateral elements in $2\mathbb{D}$ and hexahedral elements in $3\mathbb{D}$. Algebraic methods, elliptic methods and hyperbolic methods are often employed to generate these grids [84, 85, 185, 186], complex iterative smoothing techniques are also used to align elements with boundaries or physical domains in order to improve the orthogonality and uniformity. Where non-trivial boundaries are required, block structured techniques can be employed which allow the user to break the domain up into several aligned topological blocks, obtaining in this way a multiblock grid. While multiblock grids give the user more freedom in constructing the mesh, the block connection requirements can be restrictive and are often difficult to construct.

There is another block structured grid method which seeks to avoid the problems associated with

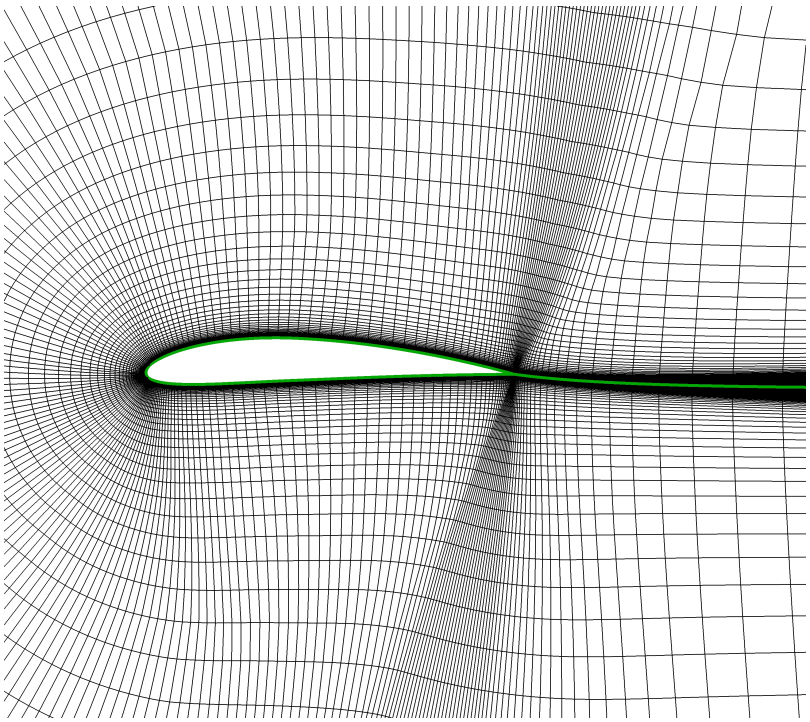

Figure 4.1: *Single-block C-type structured grid around a NACA 4412 airfoil.*

block connections in multiblock grids. Structured overlapping grid methods allow the individual blocks to conform to the physical boundaries, but, different from the multiblock grids, the blocks boundaries not necessary have to be aligned, they are allow to overlap. Sophisticated grid assembly tools are used to compute domain connectivity information and to remove unnecessary grid points. What these methods gain in user convenience, they usually give up in solution accuracy. However, these methods are very efficient when dealing with geometries which would be too daunting a task with conventional block structured methods or when dealing with moving bodies (*e.g.*, helicopters with moving rotor blades and aircraft store separation).

Structured grids enjoy a considerable advantage over other grid methods in that they allow the user a high degree of control. Because the user places control points and edges interactively, she/he has total freedom when positioning the mesh. In addition, hexahedral and quadrilateral elements, which are very efficient at filling space, support a high amount of skewness and stretching before the solution is significantly affected. This allows the user to naturally concentrate points in regions of high gradients in the flowfield and to coarsen the grid away from these areas. Also, because the user interactively lays out the elements, the grid is most often flow aligned, thereby yielding greater accuracy within the solver. Structured flow solvers typically require the lowest amount of memory for a given grid size and execute faster because they are optimized for the structured layout of the grid.

The major drawback of structured grids is the time and expertise required to lay out an optimal grid for a complex geometry. Often this comes down to past user experience and brute force placement of control points and edges is required. Grid generation times are usually measured in days if not weeks.

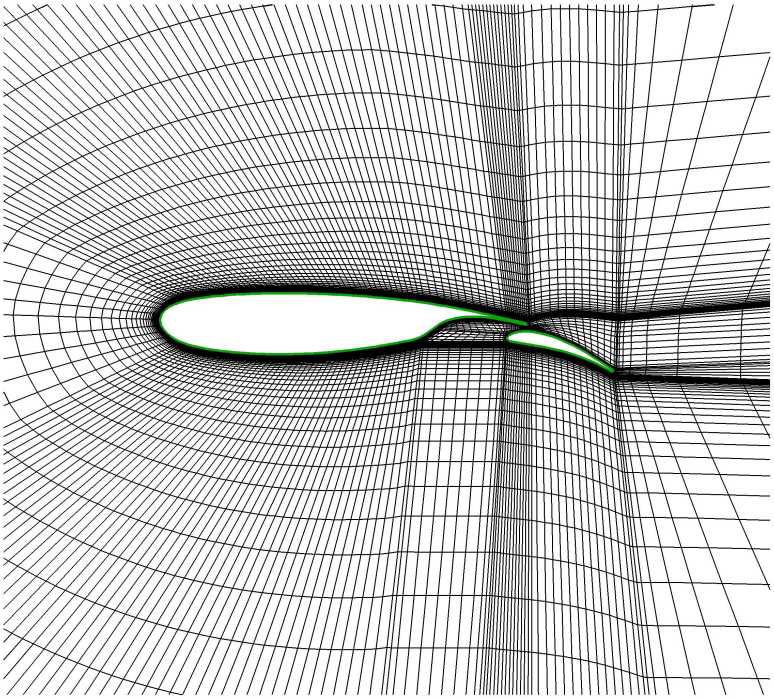

Figure 4.2: *Multi-block structured grid around a NLR 7301 airfoil with flap.*

Unstructured mesh methods [19, 137, 162, 164, 186], on the other hand, use an arbitrary collection of elements to fill the domain. Because the arrangement of elements have no recognizable pattern, the mesh is called unstructured. These types of meshes typically use triangles in $2\mathbb{D}$ and tetrahedrals in $3\mathbb{D}$, although quadrilateral and hexahedral meshes can also be used. As with structured grids, the elements can be stretched and twisted to fit the domain. These methods have the ability to be automated to a very large degree. Given a good CAD model, a good mesher can automatically place triangles on the surfaces and tetrahedrals in the volume with very little input from the user. The automatic meshing algorithm typically involves meshing the boundaries and then either adding elements touching the boundary (advancing front methods) or adding points in the interior and reconnecting the elements (Delaunay methods).

The advantage of unstructured mesh methods is that they are very automated and therefore, require little user time or effort. The user need not worry about laying out block structures or connections. Additionally, unstructured mesh methods are well suited to inexperienced users because they require little user input and will generate a valid mesh under most circumstances. Mesh generation times are usually measured in minutes or hours.

The major drawback of unstructured meshes is the lack of user control when laying out the mesh. Typically any user involvement is limited to the boundaries of the mesh with the mesher automatically filling the interior. Triangle and tetrahedral elements have the problem that they do not stretch or twist well, therefore, the grid is limited to being largely isotropic, *i.e.* all the elements have roughly the same size and shape. This is a major problem when trying to locally refine the mesh, often the entire mesh must be made much finer in order to get the locally desired point

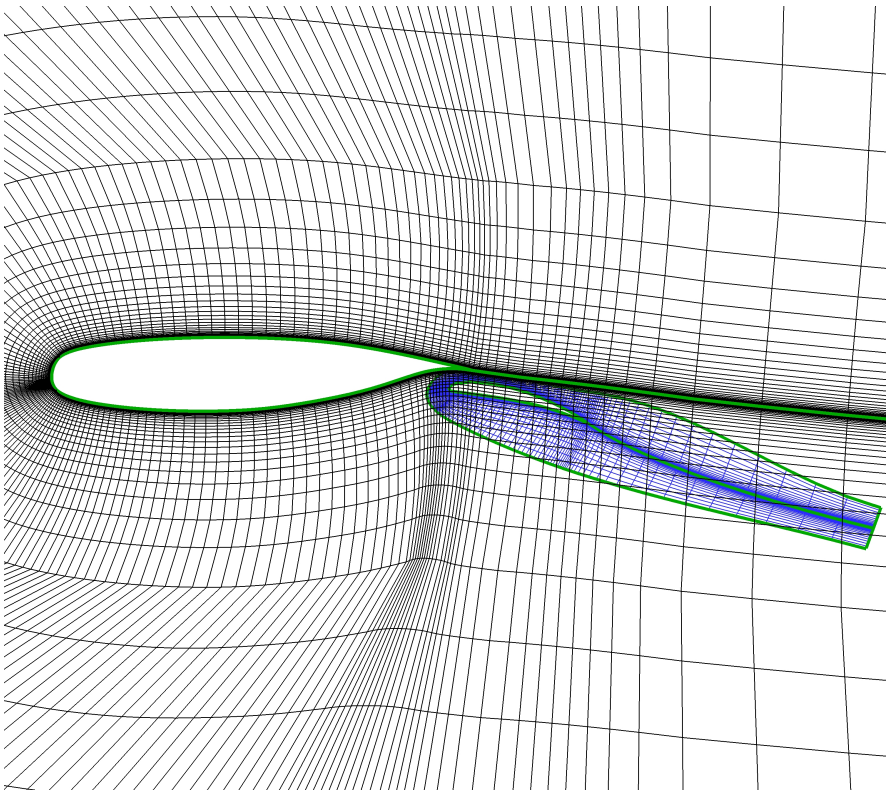

Figure 4.3: *Overlapping structured grid around a NLR 7301 airfoil with flap.*

densities. Unstructured flow solvers typically require more memory and have longer execution times than structured grid solvers on a similar mesh.

While both structured and unstructured approaches have enjoyed reasonable success in their application to real world problems, neither method has offered a truly fully automatic method for discretizing the domain around arbitrarily complex geometries. One reason for this stems from the fact that both techniques are body-fitted, *i.e.* cells neighboring the body must conform to the surface. This implies that the connectivity of the computational mesh is intimately linked to the body's geometry and topology. As a result, the surface mesh is subject to conflicting requirements of resolving both the local geometry and the expected flow variation.

Cartesian methods, as the name suggests, use a regular underlying Cartesian non-body-fitted grid. Solid objects are carved out from the interior of the mesh, leaving a set of irregularly shaped cells along the surface boundary. Early work with Cartesian grids used a stair-cased representation of the boundary. In contrast, modern Cartesian grids allow planar surface approximations at walls, and some even retain sub-cell descriptions of the boundary within the body-intersected cells. Obviously, this additional complexity places a greater burden on the flow solver, and recent research has focused on developing numerical methods to accurately integrate along the surface boundaries of a Cartesian grid. Since most of the volume mesh is completely regular, highly efficient and accurate flow solvers can be used. All the overhead for the geometric complexity is at the boundary, where the Cartesian cells are cut by the body.

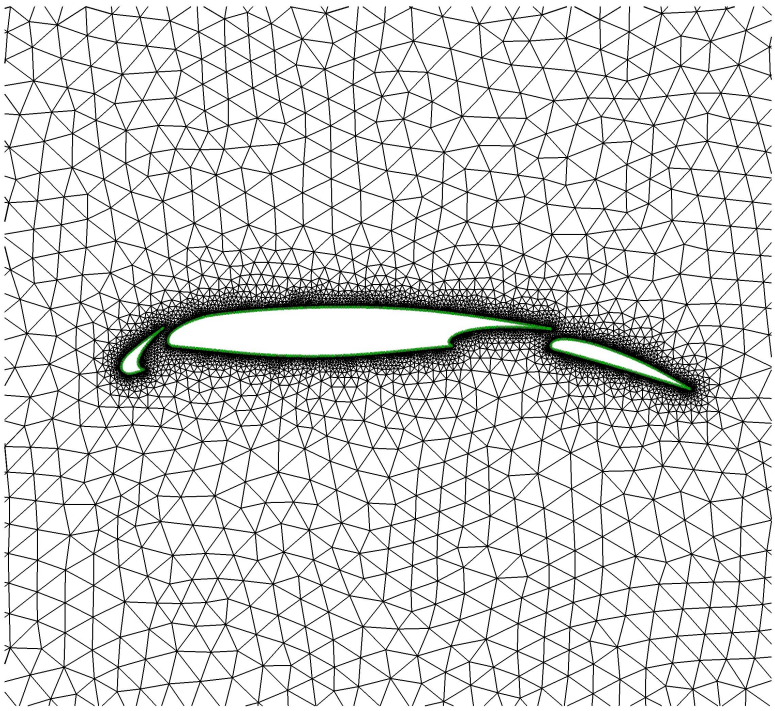

Figure 4.4: *Unstructured mesh around a NHLP-2D three element airfoil.*

Although Cartesian grid methods date back to the 1970s, it was only with the advent of adaptive mesh refinement (AMR) that their use became practical [1, 18]. Without some provision for grid refinement, Cartesian grids would lack the ability to efficiently resolve fluid and geometry features of various sizes and scales. This resolution is readily incorporated into structured meshes via grid point clustering. Many algorithms for automatic Cartesian grid refinement have, however, been developed in the last decade, largely alleviating this shortcoming. A fairly extensive literature on the flow solvers developed for Cartesian grids with embedded adaptation is available, for a more thorough discussion of Cartesian mesh topics refer to [1, 123, 137, 162].

The last decade has witnessed a resurgence of interest in Cartesian mesh methods. In contrast to body-fitted structured or unstructured methods, Cartesian grids are inherently non-body-fitted. This characteristic promotes extensive automation, dramatically eases the burden of surface preparation, and greatly simplifies the re-analysis processes when the topology of a configuration changes. By taking advantage of these important characteristics, well-designed Cartesian approaches virtually eliminate the difficulty of grid generation for complex configurations. Typically, meshes with millions of cells can be generated in minutes on a modest workstations [1, 2].

The most serious current drawback of Cartesian grids is that their use is restricted to inviscid or low Reynolds number flows. An area of active research is their coupling to prismatic grids or other methods for incorporating boundary layer zoning into the Cartesian grid framework [2].

The approach discussed in this chapter, using overlapping grids, may be viewed as a combination of Cartesian grids and structured grids methods. Body-fitted conforming structured grids are used in order to achieve high-quality representations of near-body boundaries. At the same

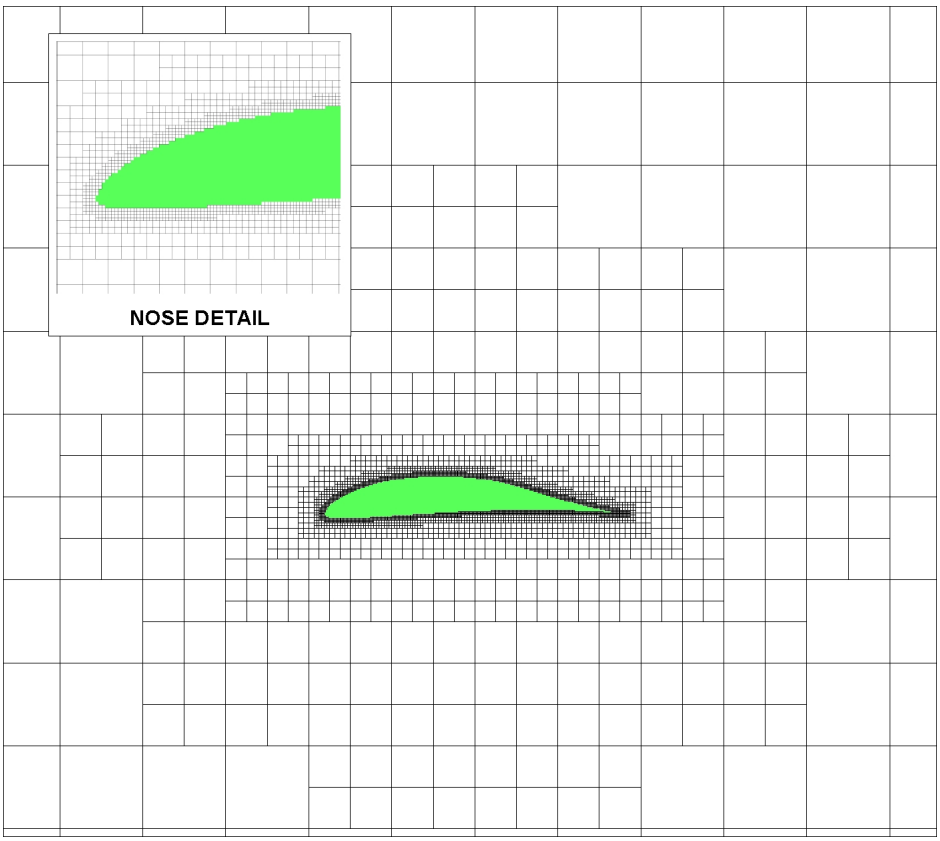

Figure 4.5: *Cartesian grid around a Drela DAE11 low Reynolds number airfoil.*

time, the majority of grid points in an overlapping grid system tend to belong to Cartesian grids (off-body) so that the numerical and computational efficiencies inherent with such grids can be exploited. In table 4.1, some of the advantages and disadvantages of some of the currently used grid generation methods are listed.

4.2 Overview and Historical Background of the Structured Overlapping Grids Method

The overlapping grids method, also known as overset composite grids or Chimera grids (named like this after the composite monster of Greek mythology), provides a flexible and efficient spatial discretization method for numerically solving a PDE on a general $1\mathbb{D}$, $2\mathbb{D}$ or $3\mathbb{D}$ domain.

The structured overlapping grids method consists in generating a set of body-fitted conforming structured components grids that completely cover the physical domain that is being modeled and overlap where they meet [141] (see figure 4.6). Reducing in this way a single, complex domain into a series of smaller, potentially simpler ones. The governing PDEs are solved separately on each component grid and domain connectivity is obtained through proper interpolation in the overlapping areas. The geometry of the components of the domain can be defined individually and hence the grids around them can be generated separately. Body-fitted conforming grids are

Gridding method	Advantages	Disadvantages
Cartesian	Small memory and CPU requirements which results in very fast flow solvers; easiness in the grid generation process; does not requires much user experience; easy to parallelize; user grid generation process is almost automatic, it requires little user time and effort	Difficulties in resolving boundary layers; difficulties in resolving complex boundaries
Single Structured	Small memory and CPU requirements which results in fast flow solvers; very well fitted for viscous flow computations; can handle very efficiently trivial geometries, accurate representation of boundaries	Restricted to simple geometries; requires user experience; not amenable to local refinement schemes, user grid generation process can be time consuming
Block Structured	Small memory and CPU requirements which results in fast flow solvers; very well fitted for viscous flow computations; can efficiently handle complex geometries; provides natural domain decomposition, accurate representation of boundaries	Requires a lot of user experience; user grid generation process is very time consuming, not amenable to local refinement schemes; requires block connectivity information
Unstructured	Easiness in grid generation process; they do not require a lot of user input or experience; user grid generation process is almost automatic, it requires little user time and effort; can efficiently handle very complex geometries; allows arbitrary use of element types; allows adaptivity and local refinement, accurate representation of boundaries	Large memory and CPU requirements which results in slower flow solver; not suited for viscous flow computations
Overlapping	Small memory and CPU requirements which results in fast flow solvers; very well fitted for viscous flow computations; can efficiently handle complex and moving geometries; eases grid generation burden for very complex geometries or when working in collaborative environments; provides natural domain decomposition, accurate representation of boundaries	Requires a lot of user experience; user grid generation process can be very time consuming; non-conservative interpolation issues in overlapping area; often needs separate tools to handle domain connectivity information

Table 4.1: *Some of the currently used grid generation methods.*

used near the components boundaries while one or more background Cartesian grids are used to handle the rest of the domain, all without any constraints on the grid boundaries as long as overlap exists between adjacent grids. While originally developed as a means to address complex geometries [128, 139], the overlapping grids method have also been employed to simulate multiple bodies in relative motion [121, 149] and to resolve fine-scale flow features through the use of adaptive mesh refinement (AMR) [73]. The use of structured grids, together with the use of optimized discretizations for large regions typically covered by Cartesian grids and the intrinsic domain decomposition nature of the overlapping grids methodology, leads to an efficient method in both computer time and computer memory, highly scalable to parallel computing platforms [149]. Finally, the overlapping grids method is advantageous for performing simulations in a production environment, where component grids for a complex system may be developed concurrently by different team members, libraries of grids of common grid components may be developed for reusability, and small changes may be quickly incorporated into a grid system by modifying only the impacted component grids and not the entire grid system [13].

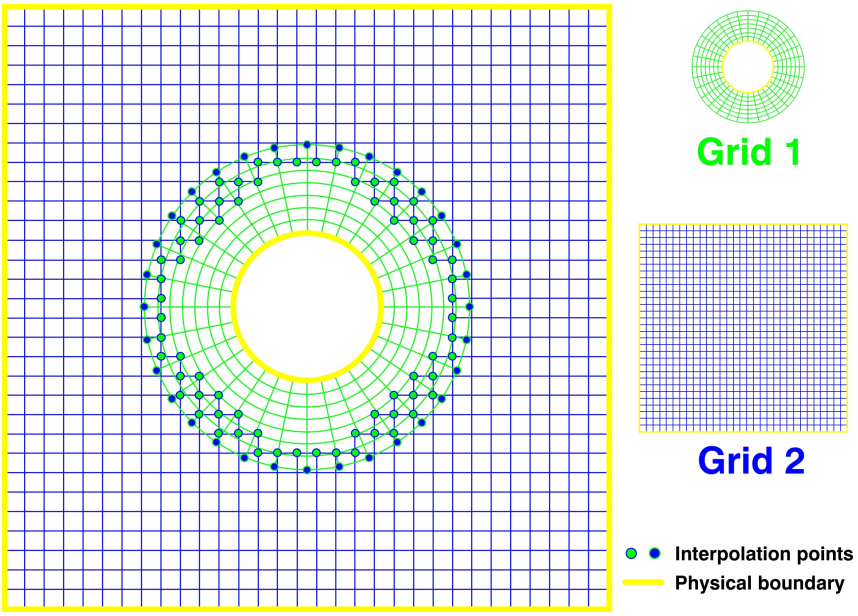

Figure 4.6: *Simple overlapping grid system in physical space $\mathcal{P}$.*

In many ways, the overlapping grids method is similar to the so-called patched or block structured approach. What differentiates overlapping grids from multiblock grids is that alignment constraints set in multiblock grids are relaxed. Overlapping grids, are only required to overlap so that no part of the computational domain is left uncovered. Clearly, the discretization becomes more complicated at overlap boundaries, but the flexibility of having smooth overlapping grids seems to be worthwhile.

The overlapping grids method has been in use for some time. Apparently, the first use of overlapping grids was described by Volvov in 1966 [203, 204], who considered approximations to Poisson's equation on regions with corners. The method was further developed and promoted by Starius and Kreiss. Starius, in 1977, looked at the convergence of elliptic problems on two overlapping meshes using the Schwarz alternating procedure [174]. In a later paper ([175]), he considered

the numerical solution of hyperbolic problems on overlapping grids. In [175], the stability of the Lax-Wendroff method was shown for a model problem on a one-dimensional overlapping grid. Moreover, he solved the shallow water equations in a two-dimensional basin, showing that despite the overlap the mass was conserved to within a few percent. Later on, a method for the construction of composite meshes and the solution of hyperbolic PDEs was described by Kreiss [104]. Also of interest is the work of Berger [17], where she indicates how to obtain conservative difference approximations at grid interfaces. However, despite the fact that the method had been around for a while, it was first introduced into the CFD community about two decades ago by Steger *et al.* [176] and Benek *et al.* [15]; and it has been further developed by Meakin and Suhs [122], Chesshire and Henshaw [37] and Noack *et al.* [133, 130, 131, 132]. It is now recognized as an attractive approach for treating problems with complex geometries and moving boundaries. The solution process discussed in this dissertation uses a grid system that discretizes the problem domain by using separately generated but overlapping body-fitted conforming structured grids; however, the use of unstructured meshes have also been considered by Togashi *et al.* [190] and Wang and Kannan [208]. In industry and academia, the overlapping grids method has been used to solve a wide variety of problems in fields such as: aerodynamics [139, 149], rotor dynamics [41, 172], combustion [26], fluid-structure interaction [56, 179], reactive flow with detonations [73, 77], incompressible flows [78], biological flows [100], non-Newtonian flows [52] and flows with deforming boundaries [51, 144], to name a few. For example, a Reynolds-averaged Navier-Stokes calculations for a prototype Martian rotorcraft vehicle was carried out successfully using the overlapping grids method [41], in this study, solutions for hovering rotor performance at Mars experimental flow conditions were produced for a series of collective pitch angles, in a moving grid system of about 10 millions grids points. Also, the unsteady viscous flow around a V-22 tiltrotor helicopter in high-speed forward flight was solved using moving overlapping grids [121].

The maturation process for overlapping grids generation tools is ongoing and is an area of active research. Historically, users of the overlapping grids method have used grid generation tools designed for block structured grids to generate required component grids to be used in the overlapping grid system. It is just recently, that grid generation tools that exploit the flexibility inherent to overlapping grids have been efficiently implemented and coupled with flow solvers. Table 4.2 lists some of the codes that are currently available for assembling overlapping grid systems. But in general, the major distinguishing features between these different approaches to overlapping grids generation lie in the grid construction algorithm, the manner of performing interpolation, the data structures, amount of user data input and the details of implementation.

In this dissertation, the Ogen[1] grid generator [70], is used to assemble the overlapping grid system. In Ogen, the user may first generate the component grids that describe the geometry or may import the component grids into it in a readable format. The overlapping grid then is constructed. This latter step consists of determining how the different component grids interpolate from each other, removing grid points from holes in the domain and eliminating unnecessary grid points in regions of excess overlap.

[1] https://computation.llnl.gov/casc/Overture/

Code	Comments
DiRTlib [131, 132]	Donor interpolation Receptor Transaction library. It is a solver neutral library designed to provide the required capability for using overlapping grids in any general flow solver. It is designed with the idea of minimizing the number of modifications required in the flow solver. It is not freely available.
SUGGAR [130, 131]	Structured, Unstructured, Generalized overset Grid AssembleR. It is used to build domain connectivity information for the wide range of grid topologies and solver formulations in codes that use DiRTlib. It can handle moving bodies simulations. It is not freely available.
Chimera Grid Tools [33, 34]	Chimera Grid Tools (CGT) is a software package containing a variety of tools for generating overlapping grids for solving complex configuration problems. It is part of the OVERFLOW-D general purpose Navier-Stokes solver. It can handle moving bodies simulations. It is not freely available.
BEGGAR [10, 116]	It is a flow solution environment, specially designed for store separation problems. It provides automated grid assembly. It is specially targeted to production work environments. It can handle moving bodies simulations and 6-DOF rigid body motion. It is not freely available.
Ogen [70, 74]	It is part of the Overture framework, which is a collection of C++ libraries for solving PDEs on overlapping grids. Provides tools for structured grid generation and overlapping grids assembly. It can handle moving bodies simulations and adaptive mesh refinement on moving bodies. It is freely available for research and academic purposes.

Table 4.2: *Some of the codes that are currently available for assembling overlapping grid systems.*

4.3 Problem Formulation

Let us suppose we want to solve some PDE on a domain $\mathcal{D}$ in $\mathbb{N}$ space dimensions ($\mathbb{N}$=1, 2 or 3), using overlapping grids. Then, an overlapping grid system $\mathbb{G}$ of the domain $\mathcal{D}$, consists of a set of $\mathcal{N}$ structured component grids $\mathcal{G}_g$,

$$\mathbb{G} = \{\mathcal{G}_g\}, \qquad g = 1, 2, ..., \mathcal{N}$$

that entirely cover the domain $\mathcal{D}$ and overlap where the component grids $\mathcal{G}_g$ meet. Each component grid is a logically rectangular structured grid in $\mathbb{N}$ space dimensions and is defined by a smooth mapping $\mathbf{M}_g$ from the computational space $\mathcal{C} = \mathcal{C}(\xi, \eta, \zeta, \tau)$ (the unit interval for $1\mathbb{D}$ applications, unit square for $2\mathbb{D}$ applications and an equivalent hexahedral domain for $3\mathbb{D}$ applications) to the physical space $\mathcal{P} = \mathcal{P}(x, y, z, t)$, such that

$$\mathcal{P} = \mathbf{M}_g(\mathcal{C}), \qquad \mathcal{C} \in [0, 1]^{\mathbb{N}}, \qquad \mathcal{P} \in \Re^{\mathbb{N}}$$

Here $\mathcal{P}$ is equal to $\mathbf{x} = (x, y, z)$ for $\mathbb{N} = 3$ and contains all the coordinates in physical space and $\mathcal{C}$ is equal to $\mathbf{r} = (\xi, \eta, \zeta)$ for $\mathbb{N} = 3$ and contains the logically uniform array in computational space. Variables defined on a component grid, such as the coordinates of the grid points, are stored in rectangular arrays. For example, grid vertices are represented as the array

$$\mathcal{P}^g_{\mathbf{i}} : \text{grid vertices}, \qquad \mathbf{i} = (i_1, \ldots, i_{\mathbb{N}}), \qquad i_\alpha = 0, \ldots, N^g_\alpha, \qquad \alpha = 1, \ldots, \mathbb{N}$$

where N^g_α is the number of grid points in the i_α-*coordinate* direction. In the case of Cartesian grids, the grid vertex information and other mapping information is not stored, which results in a considerable savings in memory use.

Figure 4.7 shows a simple overlapping grid system consisting of two component grids, an annular boundary fitted grid and a background Cartesian grid. The top view shows the overlapping grid system in physical space $\mathcal{P}$ while the bottom view shows each grid in computational space $\mathcal{C}$. In this example the annular component grid cuts a hole in the Cartesian background grid, so that the latter grid has a number of unused points (chimera hole). These unused points are tagged and no computation is performed there. They are either outside of the computational domain $\mathcal{C}$, or are eliminated to make the total number of grid points in the overlapping grid system $\mathbb{G}$ smaller. The other points on the component grids are marked as either discretization points or interpolation points. The discretization points are those where the discretization of the governing equations or boundary conditions is applied and the interpolation points provide domain connectivity by interpolating their solution values back-and-forth between the different overlapping grids.

The classification of points on a component grid into discretization, interpolation and unused points and as well the computation of all the metrics and Jacobians used when forming discrete approximations is done by Ogen [70], which is highly optimized to treat overlapping grids and moving overlapping grids. Ogen takes as input a set of overlapping component grids along with a classification of the boundaries of each grid as a physical boundary, an interpolation boundary or a periodic boundary. This boundary information is held in a generic array $\text{flag}_g(\beta, \alpha)$, where $\beta = 1$ or 2 denotes the boundary side and $\alpha = 1, \ldots, \mathbb{N}$ is the i_α-*coordinate* direction, *i.e.*,

$$\text{flag}_g(\beta, \alpha) = \begin{cases} > 0 & \text{physical boundary} \\ = 0 & \text{interpolation boundary} \\ < 0 & \text{periodic boundary} \end{cases}$$

and

$$\begin{aligned} &\text{flag}_g(1,1) = \text{left side} \\ &\text{flag}_g(2,1) = \text{right side} \\ &\text{flag}_g(1,2) = \text{bottom side} \\ &\text{flag}_g(2,2) = \text{top side} \\ &\text{flag}_g(1,3) = \text{front side (for } 3\mathbb{D} \text{ domains)} \\ &\text{flag}_g(2,3) = \text{back side (for } 3\mathbb{D} \text{ domains)} \end{aligned}$$

Physical boundaries are discretization points. Interpolation boundaries are non-physical boundaries where the grid generator will attempt to interpolate the points from other components grids.

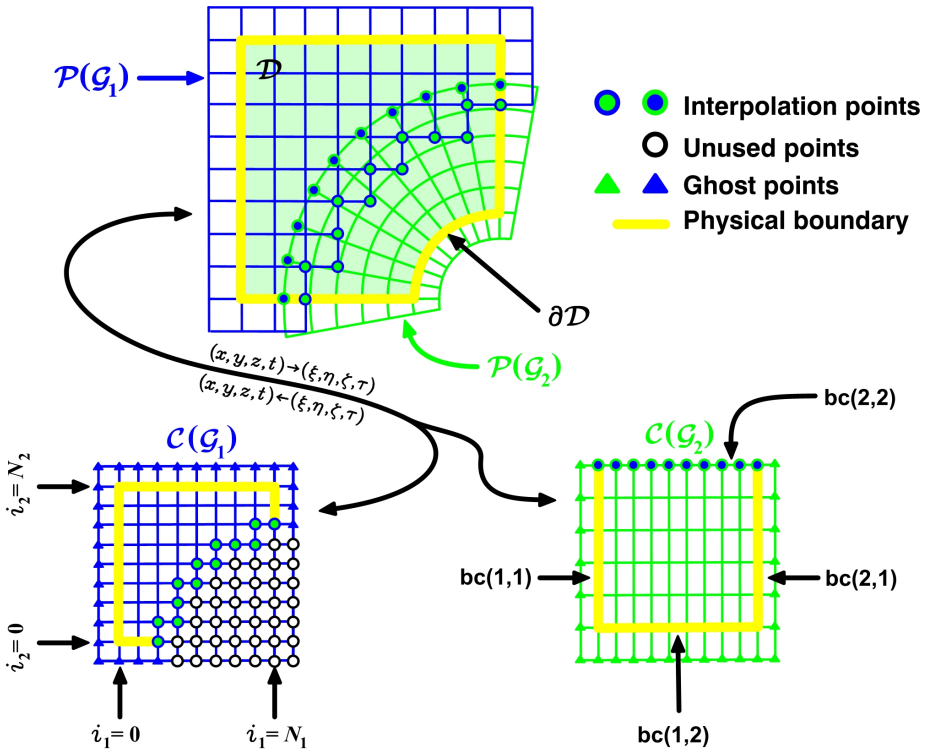

Figure 4.7: *Simple overlapping grid system consisting of two component grids $\mathcal{G}_g$. An annular boundary fitted grid ($\mathcal{G}_2$) and a background Cartesian grid ($\mathcal{G}_1$). The top view shows the overlapping grid in physical space $\mathcal{P}$ while the bottom view shows each grid in computational space $\mathcal{C}$.*

A periodic boundary can either be a branch cut (as on an annulus) or it can indicate a periodic domain. Unused points are determined by Ogen using physical boundaries to mark points exterior to the domain following a hole-cutting algorithm [37, 70, 141]. The remaining interior points are classified as either discretization points or interpolation points.

To determine which component grid to prefer when there are two or more grids that overlap each other, the component grids are ordered with respect to their priority such that grid $\mathcal{G}_g$ has priority g. When there is a choice which grid points to use in the overlap domain, the basic strategy of the overlapping grid algorithm is to prefer grid points from component grids with higher priority.

When adaptive mesh refinement (AMR) is used on an overlapping grid system, new refinement grids are added where an error in the numerical solution is estimated to be large. The approach used by Ogen [70], follows the pioneering work of Berger and Oliger [18], but with some modifications for moving grids [73]. The refinement grids are added to each component grid and are aligned with the computational space $\mathcal{C}$. The refinement grids are arranged in a hierarchical way, with the base grids belonging to a refinement level $r_{level} = 0$, the next finer grids belonging to $r_{level} = 1$, and so on. The grids on refinement level r_{level} are a factor r_{factor} finer than the grids on level $r_{level} - 1$. An AMR regridding procedure is performed every r_{regrid} time steps, where r_{regrid} is typically equal to $2 \times r_{factor}$. This procedure begins with the computation of an error estimate based on the current solution. Once the error estimate is obtained, grid points are flagged if the error is larger than a tolerance. A new set of refinement grids is generated to

cover all flagged points, and the solution is transferred from the old grid hierarchy to the new one. Since the regridding procedure takes place at a fixed time, it is effectively decoupled from moving grids cases.

In computational space $\mathcal{C}$, solution values at interpolation points are generally determined by a tensor-product polynomial interpolation scheme [76]. A simple interpolation method is to directly transfer the flow variables from the donor cell or point to the receptor cell or point. With little additional work, however, a more accurate higher-order interpolation scheme can be used, *e.g.*, a polynomial interpolation scheme. Hence, it is clear that higher-order interpolation schemes can be used with no major difficulties, but a larger overlapping region with more stencil points will be needed. In this dissertation, solution values at interpolation points are determined by using a non-conservative Lagrange interpolation scheme, whose interpolation stencil is fully compatible with the stencil of the numerical scheme discussed in the following chapter. More sophisticated interpolation algorithms that maintain conservation are presented by Chesshire and Henshaw [38], Wang [206] and Zheng and Liou [217, 218].

Consider the situation depicted in figure 4.8, in which a point $\mathcal{P}_{\mathbf{i}}^{\mathcal{G}_2}$ (where $\mathbf{i} = (i_1, i_2), i_\alpha = 0, \ldots, N_\alpha^g, \alpha = 1, 2$ and N_α^g is the number of grid points in the i_α-*coordinate* direction) is to be interpolated from component grid $\mathcal{G}_1$. Ogen supplies the $\mathcal{C}_{\mathbf{i}}^{\mathcal{G}_1}$ coordinates in computational space $\mathcal{C}$ ($\mathbf{r} = (r_1, r_2)$) of point $\mathcal{P}_{\mathbf{i}}^{\mathcal{G}_2}$, such that $\mathcal{C}_{\mathbf{i}}^{\mathcal{G}_1} = \mathbf{M}_{\mathcal{G}_1}^{-1}(\mathcal{P}_{\mathbf{i}}^{\mathcal{G}_2})$. Therefore it is only necessary to know how to interpolate a point from a rectangular grid. A list of the overlapping grid system $\mathbb{G}$ domain connectivity information such as interpolation points, the donor grid from which they interpolate, the location of the interpolation point in the computational space $\mathcal{C}$ of the donor grid and so on, is provided and kept by Ogen [70]. In particular, if grid $\mathcal{G}_g$ has n_{ip}^g interpolation points, then for each $n = 1, 2, \ldots, n_{\text{ip}}^g$

$$
\begin{aligned}
\mathbf{ip} &= \text{ip}_n^g && \text{(interpolation point } n \text{ on grid } g\text{)} \\
dg &= \text{dg}_n^g && \text{(donor grid for interpolation point ip}_n^g\text{)} \\
iw &= \text{iw}_n^g && \text{(interpolation width of the overlap area of grid } g\text{)} \\
\mathbf{r} &= \text{dgm}_n^g && \text{(donor grid location of interpolation point ip}_n^g\text{, } \mathbf{r} = \mathbf{M}_{dg}^{-1}(\mathcal{P}_{\mathbf{i}}^g)\text{)} \\
\mathbf{j} &= \text{dgs}_n^g && \text{(lower left corner of the donor grid stencil of ip}_n^g\text{)}
\end{aligned}
$$

denote the interpolation information associated with the interpolation point. The width of the interpolation stencil iw is chosen based on the order of accuracy, the type of PDE (elliptic, parabolic, hyperbolic, mixed), and by the behavior of the overlap when the grid size decreases; see [37] for details. In the following, we will assume that $iw \geqslant 2$.

Then, the two dimensional interpolation formula of any quantity ϕ on a component grid $\mathcal{G}_g$ is given by standard Lagrange interpolation,

$$
\phi_{\mathbf{ip}}^g = \sum_{m_1=0}^{iw-1} \sum_{m_2=0}^{iw-1} \beta_{\mathbf{m}} \phi_{\mathbf{j}+\mathbf{m}}^{dg} \tag{4.1}
$$

where

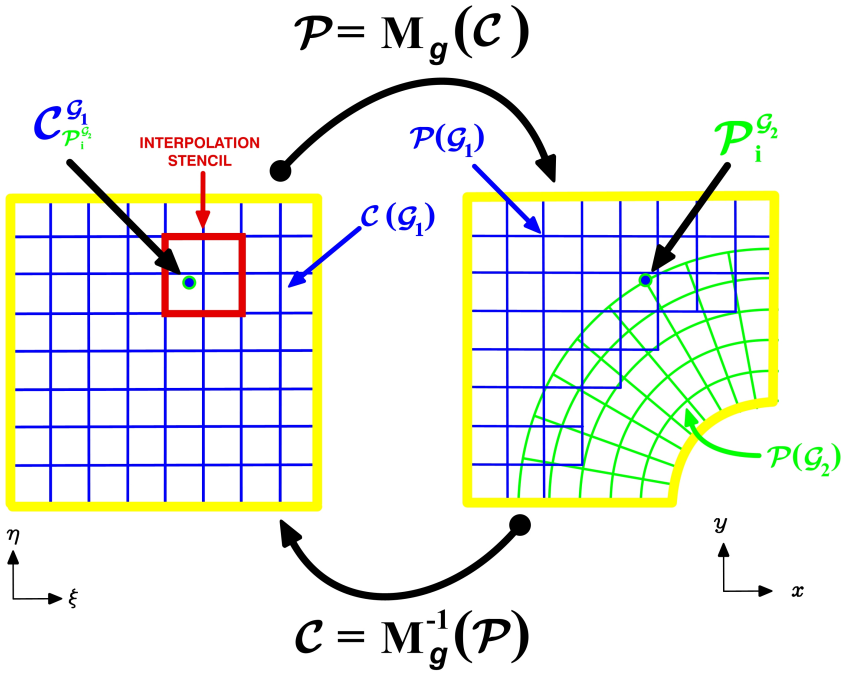

Figure 4.8: *Interpolation scheme for overlapping grids. The interpolation is performed in computational space $\mathcal{C}$.*

$$\beta_{\mathbf{m}} = \mathcal{L}^{iw}_{m_1}(\tilde{r}_1)\,\mathcal{L}^{iw}_{m_2}(\tilde{r}_2)\,, \qquad \tilde{r}_\alpha = (r_\alpha - j_\alpha)\,\Delta r_\alpha$$

Here $\mathbf{m} = (m_1, m_2)$, $\Delta r_\alpha = 1/N^g_\alpha$, and the Lagrange polynomials $\mathcal{L}^{iw}_\mu$ are defined in the usual way as

$$\mathcal{L}^{iw}_\mu(r) = \frac{\prod_{j=0, j\neq\mu}^{iw-1}(r-j)}{\prod_{j=0, j\neq\mu}^{iw-1}(\mu-j)}$$

Extension of eq. 4.1 to three dimensional cases is straightforward. The above approach to interpolation not only permits an easy way of implementing arbitrary order interpolation schemes (*e.g.*, bi-linear, bi-quadratic, tri-linear, Lagrange, spline cubic), but also makes the interpolation step less prone to error [37].

There are two different ways to interpolate in an overlapping grid [143] (figure 4.9). When the interpolation type is implicit, the solution values at the interpolation points are coupled, because they interpolate from both discretization and interpolation points in the donor grid *dg*. This makes the required overlap smaller compared to when explicit interpolation is used, since in that case only discretization points are allowed to be donor or interpolee points. When implicit interpolation is used, a small system of equations must be solved to obtain the solution at the interpolation points in terms of the values at other points, hence it becomes necessary to solve a linear system of equations in order to update the solution values at all interpolation points after each time step for time-dependent PDEs. The advantage of implicit interpolation is that the

amount of overlap is less and thus there are fewer grid points. Explicit interpolation is sometimes preferred when a time-dependent problem is solved on the overlapping grid, because it simplifies the solution procedure, but with the shortcoming of having more overlap between the grids.

Finally, when building an overlapping grid system $\mathbb{G}$, where we want the overlap to be as small and centred as possible, the following practical issues regarding to the type of interpolation must be considered. When the interpolation is implicit, the component grids must overlap one another by at least half a grid cell (for second or third order interpolation). Furthermore, the required amount of overlap is independent of the number of components grids that overlap. The situation is different for explicit interpolation. For instance, if the discretization and interpolation stencils are three points wide in each grid direction, the amount of overlap must exceed one and a half grid cells where the two component grids overlap each other. Also, the overlap must be up to three grid cells wide, close to where more than two component grids overlap each other. For further details and a complete demonstration, the interested reader should refer to [37, 141, 143].

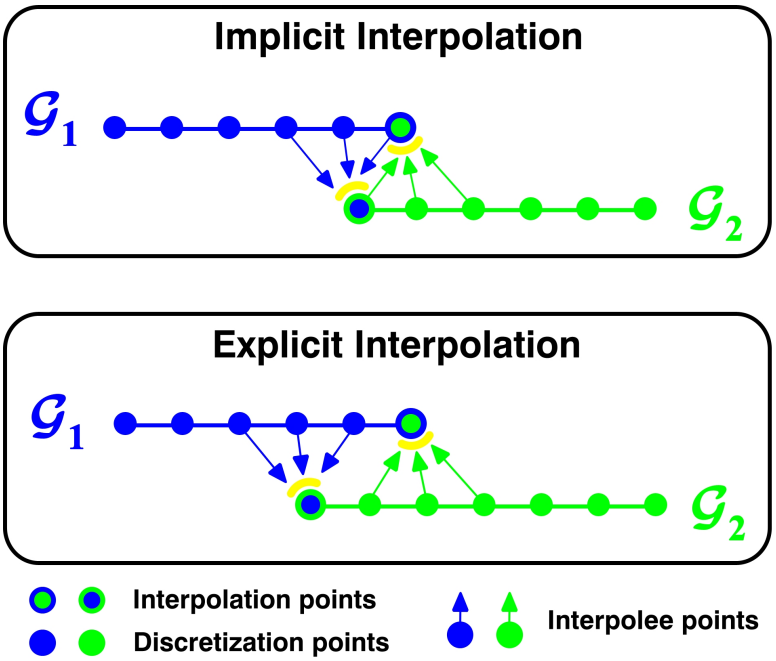

Figure 4.9: *Explicit and implicit interpolation for a one-dimensional overlapping grid.*

In addition, component grids are usually created with one or more rows of auxiliary ghost cells around the boundary of each component grid $\mathcal{G}_g$, these ghost cells are used to facilitate the discretization of boundary conditions. In the present dissertation, two rows of ghost cells are used in order to be compatible with the stencil of the numerical method and the order of the interpolation scheme. Clearly, the number of rows of ghost cells depends of the size of the numerical stencil, the interpolation width *iw* and the order of the interpolation scheme.

4.3.1 Extension of the Overlapping Grids Method to Moving Boundaries Problems

The presence of moving bodies changes the relative position of the overlapping grids continuously during the flow simulation. As the component grid (around a moving body) traverses through the computational domain, overlapping connectivity information, such as interpolation stencils and unused points regions (Chimera holes), is recomputed. The automation of hole cutting and interpolation stencils computation, makes the present methodology a powerful tool for the simulation of flows with one or multiple moving bodies, since the grids do not have to be regenerated as the solution evolves. Only Chimera holes and the interpolation stencils used to provide domain connectivity are recomputed at each time step, an operation which can be performed very efficiently. In general, the motion of the component grids and/or boundaries may be an user defined time dependent function, may obey the Newton-Euler equations for the case of rigid body motion or may be the boundary nodes displacement in response to the stresses exerted by the fluid pressure for the case of fluid structure interaction problems (FSI).

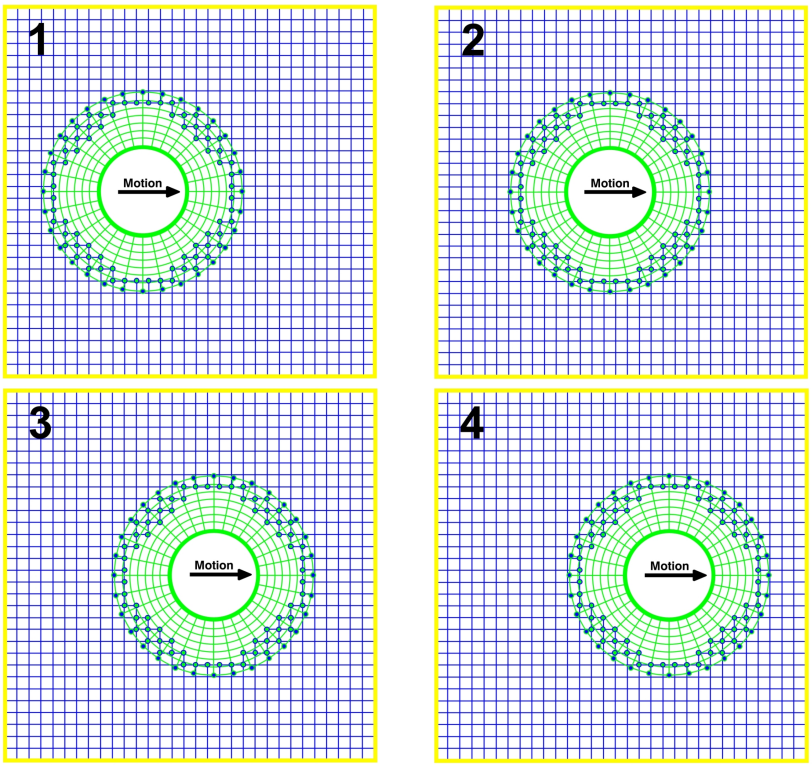

Figure 4.10: *Moving overlapping grid. The new overlapping grid system $\mathbb{G}$ interpolation stencils and chimera holes are determined by Ogen at each time step.*

When a component grid changes position during a moving grid computation, the overlapping grid generator Ogen [70] is called at each time step in order to update the interpolation stencils and Chimera holes. The component grids themselves do not have to be recomputed unless they deform in shape. An optimized algorithm is used to determine the new points classification for each grid. The algorithm only considers component grids affected by the moving boundary and a new classification of points begins by assuming that the actual structure is similar to that of

the grids at the previous time. After identifying interpolation points that are no longer valid, a local search is made for new candidates. In the event that the local search algorithm fails in completing the new classification of points, then the general overlapping grid algorithm is used (global search). Without any automation, such problems require the user to provide a priori all the input that would enable hole cutting and interpolation stencil identification for any given configuration of the overlapping grid, a cumbersome and time consuming operation.

Finally, as the problem of moving boundaries is a transient problem, the governing equations are intrinsically time dependent, thus they must be modified in order to handle the unsteady nature of the moving boundaries problem. The time derivative of any quantity $\phi = \phi(x, y, z, t)$ at a fixed point of the physical space $\mathcal{P}$ is related to its time-derivative at a fixed point of the computational space $\mathcal{C}$ by the equation

$$\left(\frac{\partial \phi}{\partial t}\right)_{\mathcal{P}} = \left(\frac{\partial \phi}{\partial t}\right)_{\mathcal{C}} - \dot{\mathbf{G}} \cdot \nabla \phi \tag{4.2}$$

with $\nabla\phi$ evaluated in the physical space $\mathcal{P}$. In eq. 4.2, $\dot{\mathbf{G}}$ is the rate of change of position of a given set of grid points $\mathcal{P}_{\mathbf{i}}^{g}$ in the physical space and can be called the grid points velocity or grid velocity and is equal to

$$\dot{\mathbf{G}} = \left(\frac{\partial \mathcal{P}_{\mathbf{i}}^{g}}{\partial t}\right)_{\mathcal{C}} \tag{4.3}$$

In eq. 4.2 and eq. 4.3, $t_{\mathcal{P}} = t$ and $t_{\mathcal{C}} = \tau$ and from eq. 3.27 we know that $t = \tau$. Thus,

$$\left.\frac{\partial \phi}{\partial t}\right|_{\mathcal{P}} = \left.\frac{\partial \phi}{\partial t}\right|_{\mathcal{C}} - \dot{\mathbf{G}} \cdot \nabla \phi \quad \text{where} \quad \dot{\mathbf{G}} = \left.\frac{\partial \mathcal{P}_{\mathbf{i}}^{g}}{\partial t}\right|_{\mathcal{C}} \tag{4.4}$$

By replacing eq. 4.4 into the respective governing equations, they are now expressed in a reference frame moving with the component grid. It is important to mention that the new governing equations expressed in the moving reference frame, must be accompanied by the proper boundary conditions. For a moving body with a corresponding moving no-slip wall, only one constraint may be applied and this corresponds to the velocity on the wall

$$\mathbf{u}\left(\mathcal{P}_{\mathbf{i}}^{g}|_{wall}, t\right) = \dot{\mathbf{G}}, \quad \text{where} \quad \mathcal{P}_{\mathbf{i}}^{g}|_{wall} \in \partial\mathcal{D}_{wall}\left(t\right) \tag{4.5}$$

4.3.2 Time Stepping Algorithm

In figure 4.11, the pseudo C++ code for the basic time-stepping algorithm for moving overlapping grids with AMR regridding used by Ogen is presented [74]. In the algorithm, $\mathbf{Q}_i^n$ denotes the numerical solution of a system of PDEs on a domain represented by an overlapping grid system $\mathbb{G}$. The input of the algorithm is an overlapping grid system $\mathbb{G} = \mathcal{G}_g^n$ generated by Ogen and the final time t_{final} over which the equations are to be integrated (for the purpose of the present discussion the precise governing equations involved or their numerical approximation are not of interest). The algorithm begins with the specification of the initial conditions according to the class **Set_initial_Conditions** and the possibly creation of an initial AMR hierarchy of grids according to an AMR algorithm contained in the **if-end** *clause*. The AMR steps involve estimating the error, regridding to better resolve the solution and interpolation of the solution from the old overlapping grid, including its hierarchy of refined grids, to a new one. These steps are repeated

until either the error tolerance is met or until the maximum number of refinement levels have been added.

Once the initial solution and initial AMR grid hierarchy have been determined, the discrete solution is advanced in time. At the top of the **while** *loop*, the solution, $\mathbf{Q}_i^n$, and grid, $\mathcal{G}_g^n$, are known at the current time t. Before advancing the grid and solution to the next time level, the algorithm checks to see whether the AMR grids need to be regenerated, then an AMR regridding procedure is performed every n_{regrid} steps.

```
1  PDE_To_Solve(𝔾, t_final)
2  {
3    t = 0;   n = 0;   𝔾 = 𝒢_g^n;
4    Q_i^n = Set_Initial_Conditions(𝒢_g^n);
5    while  t < t_final
6      if(n mod r_regrid = 0)                                   /* rebuild the AMR grids */
7        e_i = Apply_Error_Estimator(𝒢_g^n, Q_i^n);
8        𝒢_g^* = regrid(𝒢_g^n, e_i);
9        Q_i^* = Interpolate_To_New_Grid(Q_i^n, 𝒢_g^n, 𝒢_g^*);
10       𝒢_g^n = 𝒢_g^*;   Q_i^n = Q_i^*;
11     end
12     Δt = Compute_TimeStep(𝒢_g^n, Q_i^n);                     /* compute overlapping grids timestep */
13     𝒢_g^{n+1} = Move_Components_Grids(𝒢_g^n, Q_i^n);         /* predict the new grid state */
14     Update_Overlapping_Information(𝒢_g^{n+1});               /* update overlapping grid connectivity */
15     Q_i^{n+1} = Advanced_TimeStep(𝒢_g^n, 𝒢_g^{n+1}, Q_i^n, Δt);   /* advance the solution */
16     interpolate(𝒢_g^{n+1}, Q_i^{n+1});                       /* interpolate overlapping grid points */
17     Apply_Boundary_Conditions(𝒢_g^n, 𝒢_g^{n+1}, Q_i^n, Q_i^{n+1}, t + Δt);
18     t = t + Δt;   𝒢_g^n = 𝒢_g^{n+1};   n = n + 1;
19   end
20 }
```

Figure 4.11: *Pseudo C++ Code for the basic time stepping algorithm for overlapping grids.*

The first step in the main time-stepping loop moves the grids one time step according to the class **Move_Components_Grids**. The motion of the component grids $\mathcal{G}_g$ and/or boundaries may be an user defined time dependent function, may be determined by the Newton-Euler equations of motion (*i.e.*, rigid body motion) or may be the response to the stresses exerted on the boundaries by the fluid pressure (*i.e.* fluid structure interaction). Then, the new grid position $\mathcal{G}_g^{n+1}$ is determined at $t+\Delta t$. After the grids have moved, the overlapping grid generator is called to update the overlapping grid connectivity information for $\mathcal{G}_g^{n+1}$ (class **Update_Overlapping_Information**) which includes the new classification of points as discretization, unused, or interpolation points. The numerical solution may now be advanced to the next time level according to the current state $\mathbf{Q}_i^n$ and grids $\mathcal{G}_g^n$ and $\mathcal{G}_g^{n+1}$, as indicated by the class **Advanced_Time_Step**. After the solution is advanced at all discretization points, the **interpolate** class is called to update the solution on overlapping grid interpolation points and on the interpolation points on the refinement grids. The boundary conditions are then applied (class **Apply_Boundary_Conditions**) so that the solution $\mathbf{Q}_i^{n+1}$ is now specified at all interior, boundary and ghost points.

4.4 Overlapping Grids Assembling Algorithm

In this section, the algorithm for assembling overlapping grids is outlined and illustrated. In general, the method consist of three major steps. The first step is simply the geometry definition and component grids generation. The second step consist in detecting all hole points outside of the computational domain, and the third step consist in finding the grid points to interpolate

from (donor or interpolee points) for all interpolation points on the fringe of the hole (receptor or interpolation points). As we are only interested in illustrating the assembling algorithm, the detailed and theoretical definitions will be avoided as much as possible. For a detailed explanation of the algorithm see for example [37, 70, 143, 141, 149].

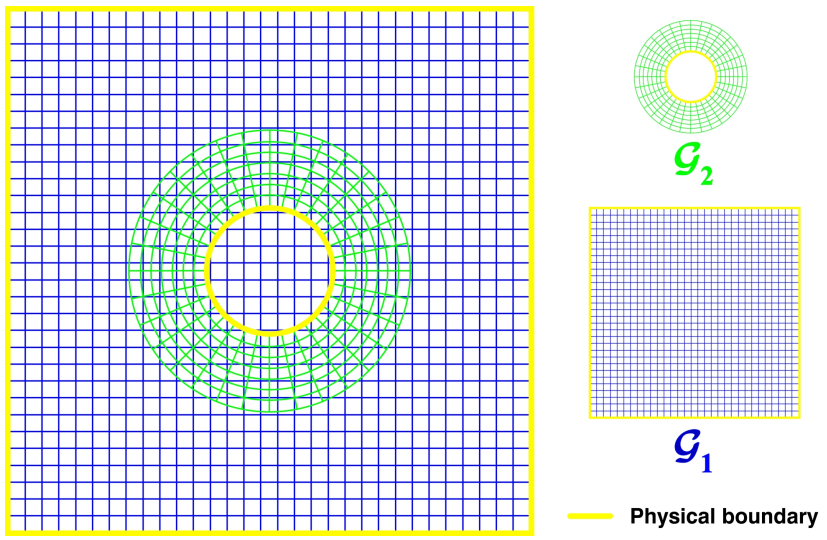

Figure 4.12: *Left. Initial overlapping grids system* $\mathbb{G}$. *Right. Individual component grids* $\mathcal{G}_g$.

We now proceed to describe the major steps and its corresponding substeps. First, the algorithm must start with a set of component grids $\mathcal{G}_g$ and a set of boundary conditions and constraints (figure 4.12), this step can be seen as mainly CAD and grid generation work.

The second step consist in marking hole boundaries and removing exterior or unused points (figure 4.13). This step can be seen as a two-substeps process. Firstly, for each physical boundary we find points on other component grids $\mathcal{G}_g$ that are near to and inside or outside of the physical boundary. After this substep, the holes in the grid will be bounded by a boundary of exterior or unused points next to a boundary of interpolation points. Next, all remaining points within the hole are marked as exterior or unused points. These points can be now easily swept out since the hole cutting algorithm ensures that all holes are bounded by interpolation points. At this stage, we have created a Chimera hole.

The third step consist in finding and classifying all valid interpolations points. This step starts with the highest grid priority and proceed in decreasing priority order such that fewer restrictions are enforced on the grids with higher priority. Here, the points on the physical boundaries and interpolation boundaries are collected into a list of interpolation points. Then, we proceed to classify these points by using improper interpolation [70]. A point is said to interpolate in an improper way from another grid if it simply lies within that grid. Since all the points in the list lie within the domain they must interpolate from some other grid or else there is something wrong (figure 4.14). Next, we proceed to find all the proper interpolation points. In this step, the points belonging to the list of improper interpolation points are classified as proper interpolation points or discretization points. A point of a grid is said to be a proper interpolation point if the appropriate stencil of points exist on the donor grid and consists of the correct types of points

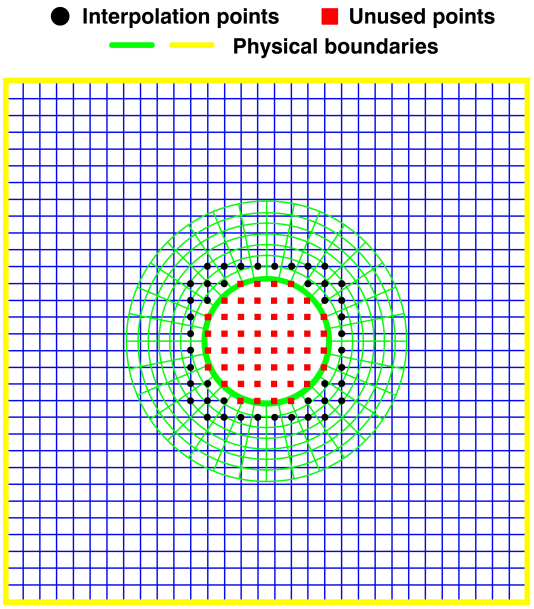

Figure 4.13: *Overlapping grid system $\mathbb{G}$ after cutting holes and removing all exterior or unused points. The hole cutting algorithm generates a barrier of unused points and interpolation points that bounds the entire hole region.*

for the implicit or explicit interpolation. We also attempt to interpolate the discretization points on each grid from grids of higher priority (figure 4.15). At this step, we should have a valid overlapping grid system $\mathbb{G}$.

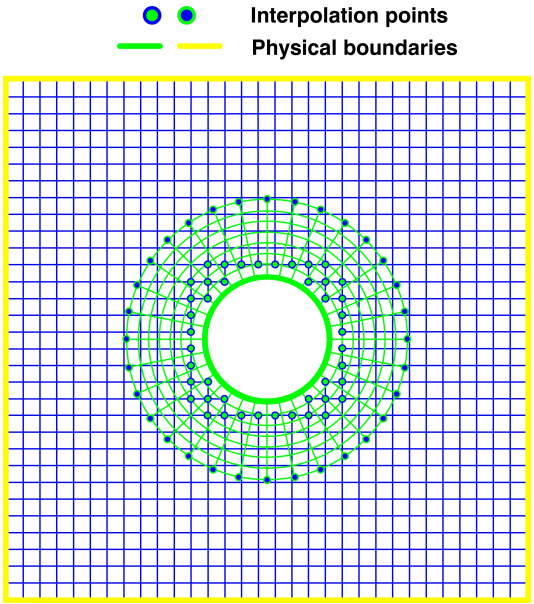

Figure 4.14: *Overlapping grid system $\mathbb{G}$ after marking points on the physical boundaries (stairstep boundary) and interpolation boundaries.*

At this stage, an additional step known as trimming (figure 4.16), is performed. Basically, this step consist in reducing the amount of overlap. Here, any interpolation point that is not needed is removed from the computation according to an user defined criteria or minimum overlap requirement. Interpolation points that are needed but can just as well be used as discretization points are turned into discretization points.

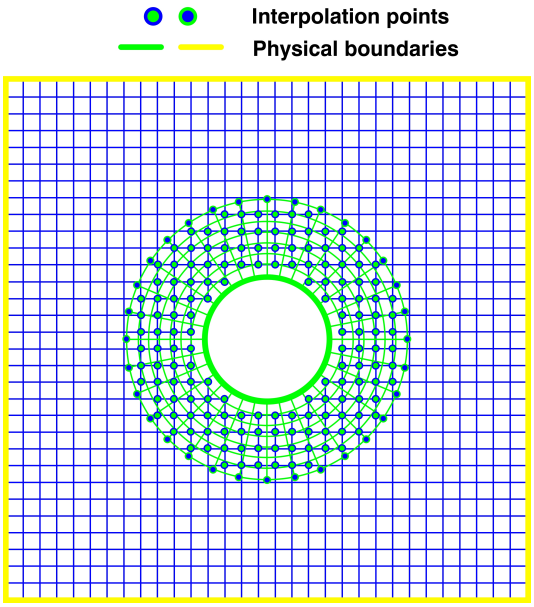

Figure 4.15: *Overlapping grid system* $\mathbb{G}$ *after marking all proper interpolation.*

By now, the overlapping grid system $\mathbb{G}$ has been assembled and optimized for the minimum or desired overlap. The last step simply consist in a consistency check, where we check if the classification of the grids points is consistent and if all discretization and interpolation points satisfy the necessary requirements. It is worth to mention that this consistency check is recursively done after each one of the previous steps. This check will mark all points that fail to satisfy the requirements and the output may be used as a reference for troubleshooting the overlapping grid system generated by Ogen [70].

In the above algorithm, one of the most important operations which is recursively performed is the task of inverting a component grid mapping $\mathbf{M}_g$ corresponding to a general component grid $\mathcal{G}_g$ in order to find a donor or interpolee point for an interpolation point $\mathcal{P}_{\mathbf{i}}^{g}$. Hence, it is essential to perform this operation as quickly as possible. Using figure 4.8 as a reference, the algorithm used by Ogen is as follows [37, 70, 141, 143],

1. Before attempting to invert the mapping, we first must check if $\mathcal{P}_{\mathbf{i}}^{\mathcal{G}_2}$ lies inside the rectangle $\Omega^{\mathcal{G}_1}$ that bounds the donor component grid $\mathcal{G}_1$. If the point does not lies inside $\Omega^{\mathcal{G}_1}$, then it cannot be interpolated and we are done. Else, if the point $\mathcal{P}_{\mathbf{i}}^{\mathcal{G}_2}$ is inside $\Omega^{\mathcal{G}_1}$, the second problem is to generate a sufficiently good initial guess for Newton's method, which is used to determine the $\mathcal{C}_{\mathbf{i}}^{\mathcal{G}_1}$ coordinates of the point $\mathcal{P}_{\mathbf{i}}^{\mathcal{G}_2}$.

2. If $\mathcal{P}_{\mathbf{i}}^{\mathcal{G}_2}$ lies within the rectangle $\Omega^{\mathcal{G}_1}$, we need a good initial guess for Newton's method. In order to get a good initial guess for this step, we do an optimized local search through all

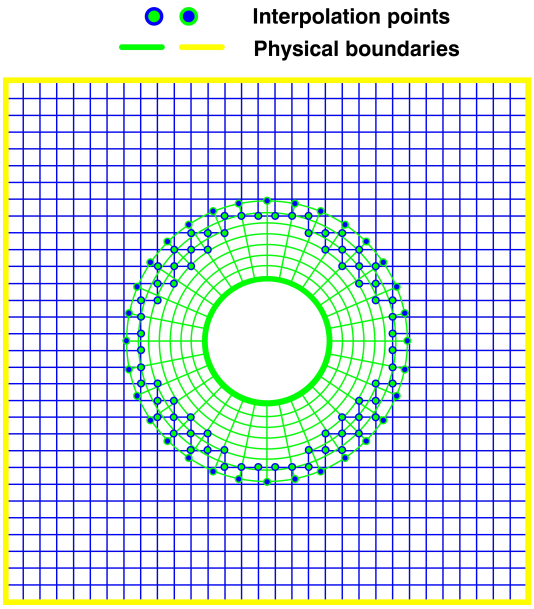

Figure 4.16: *Final overlapping grid system* $\mathbb{G}$ *after removing excess of interpolation points*

grid points in order to find the closest grid point to $\mathcal{P}_{\mathbf{i}}^{\mathcal{G}_2}$ on the donor component grid $\mathcal{G}_1$. As soon as the closest point to $\mathcal{P}_{\mathbf{i}}^{\mathcal{G}_2}$ is determined, we proceed to the next step.

3. Now we can determine the $\mathcal{C}_{\mathbf{i}}^{\mathcal{G}_1}$ coordinates of the point $\mathcal{P}_{\mathbf{i}}^{\mathcal{G}_2}$ by inverting the component grid $\mathcal{G}_1$ transformation $\mathbf{M}_g$ with a Newton iteration, using the closest point found in the previous step as an initial guess. As soon as $\mathcal{C}_{\mathbf{i}}^{\mathcal{G}_1}$ of $\mathcal{P}_{\mathbf{i}}^{\mathcal{G}_2}$ has been found ($\mathcal{C}_{\mathbf{i}}^{\mathcal{G}_1} = \mathbf{M}_{\mathcal{G}_1}^{-1}(\mathcal{P}_{\mathbf{i}}^{\mathcal{G}_2})$), it is trivial to determine the enclosing grid cell and the interpolation stencil according to the interpolation scheme.

We emphasize the above point, since the construction of a composite grid requires some computation and it is important to do certain tasks efficiently, so that the turnaround time for the grid assembly algorithm is small (specially for moving grids where the domain connectivity information have to be recomputed at each time step).

4.5 Discretization on Overlapping Grids

On an overlapping grid system $\mathbb{G}$, the solution of a PDE can be seen as the solution of the transformed PDE on a set of unit domains in computational space $\mathcal{C}$. The governing PDE is transformed from physical space $\mathcal{P}$ to computational space $\mathcal{C}$ by replacing the Cartesian derivatives by their equivalent in the transformed computational space $\mathcal{C}$ (see Chapter 3, Section 3). Hereafter, we present a simple example of how to discretize a model one-dimensional PDE on a structured overlapping grid system $\mathbb{G}$. For the sake of simplicity, the following example considers Cartesian coordinates, nevertheless, the discretization on the unit interval in the transformed computational space $\mathcal{C}$ is straightforward. Let us consider the boundary value problem (BVP) for the one-dimensional Poisson equation,

$$\begin{aligned} u_{xx} &= f, \qquad x \in \mathcal{P}\,[0,1]^{\mathbb{N}}, \qquad \mathcal{P} \in \Re^{\mathbb{N}}, \qquad \mathbb{N} = 1 \\ u\,(0) &= g_0, \qquad \text{(Dirichlet boundary condition)} \\ u_x\,(1) &= g_1, \qquad \text{(Neumann boundary condition)} \end{aligned} \tag{4.6}$$

eq. 4.6 is to be discretized on the overlapping grid system $\mathbb{G}$ shown in figure 4.17, by using standard finite differences. A second-order discretization to this problem is

$$\begin{aligned} \mathbf{U}_0 &= g_0 \,(\text{Dirichlet } bc \text{ on } \mathcal{G}_1) \\ \frac{\mathbf{U}_{i-1} - 2\mathbf{U}_i + \mathbf{U}_{i+1}}{h^2_{\mathcal{G}_1}} &= f(\mathcal{P}_i^{\mathcal{G}_1}), \quad i = 1, 2, \ldots, N_1 - 1, \quad (dp \text{ on } \mathcal{G}_1) \\ \mathbf{U}_{N_1} - (\alpha_0\mathbf{V}_0 + \alpha_1\mathbf{V}_1 + \alpha_2\mathbf{V}_2) &= 0, \quad (ip \text{ on } \mathcal{G}_1) \\ \mathbf{V}_0 - (\beta_0\mathbf{U}_{N_1-2} + \beta_1\mathbf{U}_{N_1-1} + \beta_2\mathbf{U}_{N_1}) &= 0, \quad (ip \text{ on } \mathcal{G}_2) \\ \frac{\mathbf{V}_{i-1} - 2\mathbf{V}_i + \mathbf{V}_{i+1}}{h^2_{\mathcal{G}_2}} &= f(\mathcal{P}_i^{\mathcal{G}_2}), \quad i = 1, 2, \ldots, N_2, \quad (dp \text{ on } \mathcal{G}_2) \\ \frac{\mathbf{V}_{N_2+1} - \mathbf{V}_{N_2-1}}{2h_{\mathcal{G}_2}} &= g_1 \quad (\text{Neumann } bc \text{ on } \mathcal{G}_2) \end{aligned} \tag{4.7}$$

where *bc* stands for boundary conditions, *dp* stands for discretization points, *ip* stands for interpolations points and $i = i_\alpha$-*coordinate* direction with $\alpha = 1$. As the problem is solved in one-dimension, the notation $variable_\alpha$-*coordinate* direction will be dropped for all index variables.

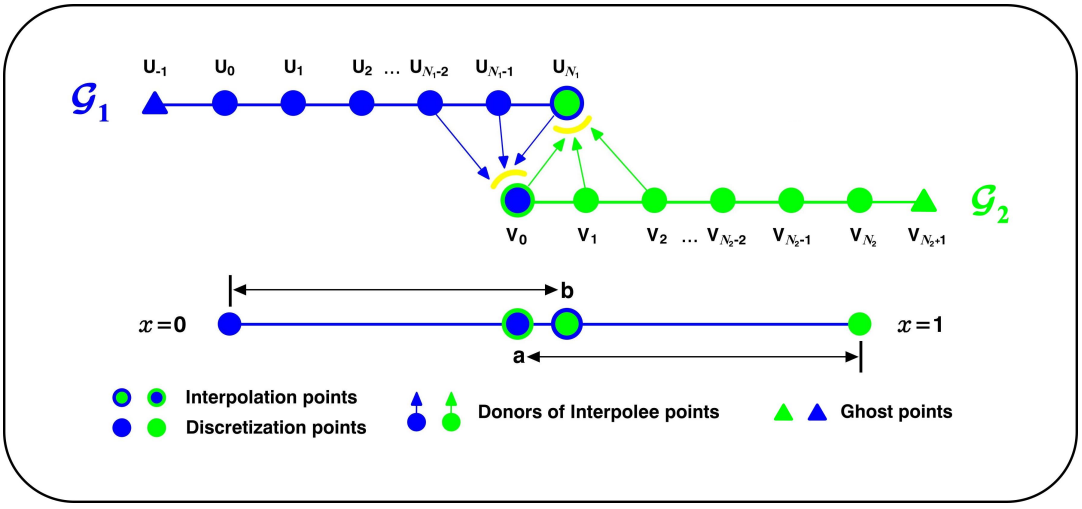

Figure 4.17: *Overlapping grid discretization in one dimension.*

In eq. 4.7, $\mathbf{U}_i$ is the approximate solution on grid $\mathcal{G}_1$ on the subinterval $[0, b]$, with $\mathcal{P}_i^{\mathcal{G}_1} = ih_{\mathcal{G}_1}$, $h_{\mathcal{G}_1} = b/N_1$ and N_1 the number of nodes on grid $\mathcal{G}_1$. Respectively, $\mathbf{V}_i$ is the approximate solution on grid $\mathcal{G}_2$ on the subinterval $[a, 1]$, with $\mathcal{P}_i^{\mathcal{G}_2} = ih_{\mathcal{G}_2}$, $h_{\mathcal{G}_2} = (1 - a)/N_2$ and N_2 the number of nodes on grid $\mathcal{G}_2$. The interpolation weights α_m and β_n are chosen appropriately according to

the interpolation scheme (see section 4.3). For this example, we assumed implicit interpolation and an interpolation width *iw* equal to 3, hence the interpolation stencil is equal to $j+m$, where j is the lower left corner of the donor grid *dg* interpolation stencil and $0 < m \leqslant iw - 1$. The Neumann boundary condition at $i = N_2$ on grid $\mathcal{G}_2$ is implemented by adding a ghost point $\mathbf{V}_{N_2+1}$; then, by using centred finite difference approximations, we obtain the value of the ghost point which is considered to be defined by the Neumann boundary condition and is equal to $\mathbf{V}_{N_2+1} = 2h_{\mathcal{G}_2} g_1 + \mathbf{V}_{N_2-1}$. At this point, the system of equations eq. 4.7 can be solved by using any direct or iterative solution method, obtaining in this way the approximate solution of eq. 4.6 on the discrete overlapping domain $\mathcal{P}_{\mathbf{i}}^{\mathbb{G}} \in (\mathcal{P}_i^{\mathcal{G}_1} \cup \mathcal{P}_i^{\mathcal{G}_2})$.

4.6 Comments on Overlapping Grids

The main advantage of the overlapping grids scheme is that individual body-fitted conforming structured grids can be created separately for each component defining the overall geometry and then superimposed to form one complete grid system that covers the entire physical domain, all without any constraints on the grid boundaries as long as overlap exists between adjacent grids. This allows complex geometries to be treated more easily, theoretically reducing the time and effort to generate a grid; also, as several overlapping grids are used, only the Chimera holes and the interpolation stencils are recomputed as the solution evolves in time when dealing with moving boundaries.

Of course, the main disadvantage of the overlapping grid scheme is that the algorithm is far more complex than a single-block or multi-block structured grid algorithm due to the use of multiple structured overlapping grids. Drawbacks to using overlapping grids include having to interpolate in a non-conservative way data points along overlapping zones, which in practice rarely seems to be an issue if good standard practices are followed [32]. In addition, the data structure bookkeeping can be especially complex if more than two grids overlap one another. Additional to the flow solver, class libraries are needed to interconnect the overlapping grids, create proper hole regions (Chimera holes), define hole boundaries, and determine the interpolation stencils for properly transmitting information among overlapping grids. Though this is perhaps a bit complex and time consuming, the computational efficiency and convenience that is gained by using such a scheme when dealing with complex or moving geometries makes the method worthwhile.

There are two additional practical complications related to generating an overlapping grid. First, it can be hard to judge *a priori* if the component grids overlap each another sufficiently. Second, the user can make a mistake when labeling the boundaries of the component grids, which can lead to an inconsistent definition of the overlapping grid system. Creating an overlapping grid is therefore sometimes an iterative process which requires much user experience and care, and where the component grids are changed by the user until a valid overlapping grid can be formed.

Finally, the use of overlapping grids does not eliminate the planning stage in grid generation process. Care must be taken to ensure that the grid distributions in the overlapping grid system are not drastically different. Radical differences between overlapping grids can lead to poor interpolation results. Poor interpolation can in turn lead to increased computational time, due to poor solution convergence.

Chapter 5

Numerical Method

In this chapter, an accurate and stable method is described for the solution of the time-dependent incompressible Navier-Stokes equations with finite differences on structured body-fitted overlapping grids in one, two or three space dimensions. For the incompressible Navier-Stokes equations, there is no direct link for the pressure between the continuity and momentum equations; the governing equations are said to be decoupled. To establish a connection between the two equations, mathematical manipulations are introduced. The numerical method presented is a split-step scheme, second-order accurate in space and time and solves the momentum equations for the velocity together with a Poisson equation for the pressure (the so called pressure-Poisson equation or PPE), this system of equations is known as the velocity-pressure formulation of the incompressible Navier-Stokes equations.

5.1 Primitive Variable Formulation of the Incompressible Navier-Stokes Equations

In primitive variables (u, v, w, p), the initial-boundary-value problem (IBVP) for the incompressible Navier-Stokes equations is

$$\begin{aligned} \frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} &= \frac{-\nabla p}{\rho} + \nu \nabla^2 \mathbf{u} \qquad \text{for} \quad \mathbf{x} \in \mathcal{D}, \qquad t > 0, \qquad (5.1) \\ \nabla \cdot \mathbf{u} &= 0 \qquad \text{for} \quad \mathbf{x} \in \mathcal{D}, \qquad t > 0, \qquad (5.2) \end{aligned}$$

with the following boundary conditions and initial conditions

$$\begin{aligned} B(\mathbf{u}, p) &= \mathbf{g} \qquad \text{for} \quad \mathbf{x} \in \partial\mathcal{D}, \qquad t > 0, \\ \mathcal{D}\mathring{\mathbf{Q}}(\mathbf{x}, 0) &= \mathbf{q}_0(\mathbf{x}) \qquad \text{for} \quad \mathbf{x} \in \mathcal{D}, \qquad t = 0. \end{aligned} \qquad (5.3)$$

in this IBVP, $\mathbf{x} = (x, y, z)$ (for $\mathbb{N} = 3$ where $\mathbb{N}$ is the number of space dimensions) is the vector containing the Cartesian coordinates in physical space $\mathcal{P}$, $\mathcal{D}$ is a bounded domain in $\mathcal{P} \in \Re^{\mathbb{N}}$ ($\mathbb{N} = 1, 2, 3$), $\partial\mathcal{D}$ is the boundary of the domain $\mathcal{D}$, t is the physical time, $\mathbf{u} = (u, v, w)$ is the vector containing the velocity field in $\mathcal{P}$, p is the pressure, ν is the kinematic viscosity which is equal to $\nu = \mu/\rho$, B is a boundary operator, $\mathbf{g}$ is the boundary data and $\mathbf{q}_0$ is the initial data. The system of equations eq. 5.1, eq. 5.2 and eq. 5.3 will be called the velocity-divergence formulation of the governing equations in primitive variables [78, 84, 118].

Typical boundary conditions for the system of equations eq. 5.1 and eq. 5.2 might be those for a non-penetrating no-slip wall

$$\begin{aligned} \mathbf{u}\cdot\hat{\mathbf{n}} &= 0 \quad \text{for} \quad \partial\mathcal{D}_{wall} \quad \text{“no-through-flow (Dirichlet boundary condition)”} \\ \mathbf{u}\cdot\hat{\mathbf{t}} &= 0 \quad \text{for} \quad \partial\mathcal{D}_{wall} \quad \text{“no-slip (Dirichlet boundary condition)”} \end{aligned} \tag{5.4}$$

or those for an inflow such as

$$\mathbf{u} = \mathbf{g} \quad \text{for} \quad \partial\mathcal{D}_{in} \quad \text{“inflow (Dirichlet boundary condition)”} \tag{5.5}$$

Specifying the pressure, its normal derivative, or a combination of the two at the outflow is also often used

$$\begin{aligned} \hat{n}\cdot\nabla p &= g \quad \text{for} \quad \partial\mathcal{D}_{out} \quad \text{“outflow (Neumann boundary condition)”} \\ \alpha p + \beta\hat{n}\cdot\nabla p &= g \quad \text{for} \quad \partial\mathcal{D}_{out} \quad \text{“outflow (Mixed boundary condition)”} \end{aligned} \tag{5.6}$$

with α and β suitable coefficients.

Also, the specification of zero velocity gradient at the outflow may be appropriate for most applications

$$\nabla\mathbf{u} = 0 \quad \text{for} \quad \partial\mathcal{D}_{out} \quad \text{“outflow (Neumann boundary condition)”} \tag{5.7}$$

Before continuing with our discussion of the numerical method, it is important to make a few comments with regard to the velocity-divergence formulation of the incompressible Navier-Stokes equations.

- The governing equations are a mixed elliptic-parabolic system of equations which are solved simultaneously. The unknowns in the equations are velocity field $\mathbf{u} = (u, v, w)$ and pressure p.
- There is no direct link for the pressure between the continuity and momentum equations. To establish a connection between the two equations, mathematical manipulations are introduced. Generally speaking there are three procedures for this purpose. The first is that of generating a Poisson equation for the pressure (the so-called PPE equation), which is developed in this chapter; the second is the introduction of artificial compressibility into the continuity equation, and the third is the use of projection methods, which encompasses similarities with the PPE approach. Projection methods also produce a Poisson equation that is solved for the pressure in the incompressible flow, this new equation is obtained by using Hodge decomposition theorem, which basically decompose the velocity field into a sum of a divergence-free part (solenoidal) and curl-free part (irrotational) [45]. Note that this difficulty does not exist for the compressible Navier-Stokes equations. That is because in the compressible case there is a link between the continuity and momentum equations through the density which appears in both equations.
- Straight-forward discretizations of eq. 5.1 and eq. 5.2 can lead to checker-board instabilities [81, 118]. Centred finite-difference approximations on unstaggered grids permit discrete

satisfaction of the divergence-free constraint by non-physical velocity fields. Similarly, centred finite-differences approximations to the pressure gradient terms (on unstaggered grids) in the momentum equations allow non-physical pressure fields to go undetected and thus, uncorrected (spurious oscillations).

- Many approaches require extra boundary conditions, either for the pressure or for an intermediate velocity field, which can be non-trivial to choose and difficult to implement.
- For efficiency, it is useful to decouple the solution of the velocity from the solution of the pressure (split-step scheme).

5.2 Pressure-Poisson Equation (PPE) or Velocity-Pressure Formulation in Primitive Variables

In this formulation, the PPE equation for the pressure is used in place of the continuity equation eq. 5.2. The new IBVP is expressed as follows

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = \frac{-\nabla p}{\rho} + \nu \nabla^2 \mathbf{u} \qquad \text{for} \quad \mathbf{x} \in \mathcal{D}, \qquad t > 0 \tag{5.8}$$

$$\frac{\nabla^2 p}{\rho} + \nabla u \cdot \mathbf{u}_x + \nabla v \cdot \mathbf{u}_y + \nabla w \cdot \mathbf{u}_z = 0 \qquad \text{for} \quad \mathbf{x} \in \mathcal{D}, \qquad t > 0 \tag{5.9}$$

with the following boundary and initial conditions

$$\begin{aligned} B(\mathbf{u}, p) &= \mathbf{g} && \text{for} \quad \mathbf{x} \in \partial\mathcal{D}, \quad t > 0 \\ \nabla \cdot \mathbf{u} &= 0 && \text{for} \quad \mathbf{x} \in \partial\mathcal{D}, \quad t > 0 \\ \mathcal{D}\mathring{\mathbf{Q}}(\mathbf{x}, 0) &= \mathbf{q}_0(\mathbf{x}) && \text{for} \quad \mathbf{x} \in \mathcal{D}, \quad t = 0 \end{aligned} \tag{5.10}$$

The system of equations eq. 5.8, eq. 5.9 and eq. 5.10 will be called the velocity-pressure formulation of the governing equations in primitive variables. Equation eq. 5.9 implies that the pressure can be calculated provided the velocity field is known. This is the form of the equations that will be discretized in the method described in this chapter (where the equations are solved on unstaggered grids), and which is based in the method developed by Brown *et al.* [25], Chesshire and Henshaw [37], Henshaw [78], Hewshaw, Kreiss and Reyna [80] and Henshaw and Petterson [81], for solving the velocity-pressure formulation of the incompressible Navier-Stokes equations on overlapping grids. The pressure equation (eq. 5.9) is derived by taking the divergence of the momentum equation eq. 5.1 and using the divergence-free constraint $\nabla \cdot \mathbf{u} = 0$, then, eq. 5.2 is replaced by the elliptic equation for the pressure. For the system of equations eq. 5.8 and eq. 5.9 an extra boundary condition is required in order to make the problem well-posed. The condition $\nabla \cdot \mathbf{u} = 0$ for $\mathbf{x} \in \partial\mathcal{D}$ is added as the extra boundary condition. This latter condition is an essential boundary condition for this formulation and ensures that the system of equations (eqs. 5.8 - 5.10) is equivalent to the original formulation (eqs. 5.1 - 5.3).

5.3 Remarks on the Pressure Boundary Condition

Perhaps, it is appropriate to make some remarks regarding the choice of $\nabla \cdot \mathbf{u} = 0$ as the extra boundary condition for the velocity-pressure formulation. The extra boundary condition required

by the velocity-pressure formulation should satisfy three general conditions. First, it should be chosen so that eq. 5.8 and eq. 5.9 are well posed. Second, it should be consistent with the original formulation eq. 5.1 and eq. 5.2. And lastly, it should be chosen so that the velocity-pressure formulation is equivalent to the velocity-divergence formulation. These three conditions are satisfied by the boundary condition $\nabla \cdot \mathbf{u} = 0$, which, despite not looking like a pressure boundary condition, is in some sense the natural extra condition to be added in order to fulfill the three requirement previously mentioned.

There has been a great deal of confusion as to the proper boundary condition for the PPE equation (*i.e.* eq. 5.9). Several articles (*e.g.*, [61, 78, 80, 92, 126, 142, 159]), discuss the issue whether it is appropriate to use the tangential or normal component of the momentum equation on the boundary as a boundary condition for the pressure equation or another type of boundary condition. However, it appears that these methods also impose (implicitly or explicitly) the boundary condition $\nabla \cdot \mathbf{u} = 0$ on $\partial\mathcal{D}$. Often the fact that this condition is applied is not emphasized [81].

In [61], Gresho and Sani proposed an important hypothesis regarding the pressure Poisson equation (PPE) for the incompressible Navier-Stokes equations. They stated there (but did not prove it) a so-called equivalence theorem that claimed that if the Navier-Stokes momentum equation is solved simultaneously with the PPE equation whose boundary condition is the Neumann boundary condition obtained by applying the normal component of the momentum equation on the boundary on which the normal component of velocity is specified as a Dirichlet boundary condition, the solution $(\mathbf{u}, p)$ would be exactly the same as if the primitive equations (in which the PPE equation plus Neumann boundary condition is replaced by the usual divergence-free constraint $(\nabla \cdot \mathbf{u} = 0)$) were solved instead. This issue is explored in sufficient detail by Sani *et al.* in [159], so as to actually prove the theorem for at least some situations. Additionally, like the primitive equations that require no boundary condition for the pressure, the new results establish the same requirement when the PPE equation approach is employed.

5.4 Spatial Discretization of the Velocity-Pressure Formulation of the Incompressible Navier-Stokes Equations

We now describe in more detail how we discretize equations (5.8 - 5.10). But before continuing, let us recall some basic features of overlapping grids (as illustrated in figure 4.7). An overlapping grid $\mathbb{G}$ of the domain $\mathcal{D}$ in $\mathbb{N}$ space dimensions, consists of a set of $\mathcal{N}$ structured component grids $\mathcal{G}_g$,

$$\mathbb{G} = \{\mathcal{G}_g\}, \qquad g = 1, 2, ..., \mathcal{N}$$

that entirely cover the domain $\mathcal{D}$ and overlap where the component grids $\mathcal{G}_g$ meet. Each component grid is a logically rectangular structured grid in $\mathbb{N}$ space dimensions and is defined by a smooth mapping $\mathbf{M}_g$ from the computational space $\mathcal{C} = \mathcal{C}(\xi, \eta, \zeta, \tau)$ to the physical space $\mathcal{P} = \mathcal{P}(x, y, z, t)$, such that

$$\mathcal{P} = \mathbf{M}_g(\mathcal{C}), \qquad \mathcal{C} \in [0, 1]^{\mathbb{N}}, \qquad \mathcal{P} \in \Re^{\mathbb{N}}$$

Here $\mathcal{P}$ is equal to $\mathbf{x} = (x, y, z)$ (for $\mathbb{N} = 3$) and contains all the coordinates in physical space and

$\mathcal{C}$ is equal to $\mathbf{r} = (\xi, \eta, \zeta)$ (for $\mathbb{N} = 3$) and contains the logically uniform array in computational space. Variables defined on a component grid, are stored in rectangular arrays. The grid vertices are represented as the array

$$\mathbf{x}^g_{\mathbf{i}} : \text{grid vertices}, \qquad \mathbf{i} = (i_1, \ldots, i_{\mathbb{N}}), \qquad i_\alpha = 0, \ldots, N^g_\alpha, \qquad \alpha = 1, \ldots, \mathbb{N}$$

where N^g_α is the number of grid points or nodes in the i_α-*coordinate* direction. Each component grid is usually created with one or more lines of ghost points, which are useful for applying boundary conditions. Domain connectivity is obtained through proper interpolation of the overlapping areas of the component grids $\mathcal{G}_g$.

For ease of presentation we describe here the solution of the velocity-pressure formulation of the incompressible Navier-Stokes equations in two space dimensions on a square grid $\mathcal{G}_g = \mathbf{G}$ in physical space $\mathcal{P}$ with grid spacing $h_{i_\alpha} > 0 (h_{i_\alpha} = 1/N_{i_\alpha})$ and with $h_{i_1} = h_{i_2}$, such that

$$\mathbf{G} = \{\mathbf{x_i} = (x_{1\mathbf{i}}, x_{2\mathbf{i}}) = (x_\mathbf{i}, y_\mathbf{i}) = (ih, jh) \quad \text{for} \quad i, j = -1, 0, 1, \ldots, N+1\}$$

here $\mathbf{i} = (i_1, i_2) = (i, j)$ is a multi-index. We include one row of ghost points at the boundaries to aid in the discretization. The discretization on the unit square in the transformed computational space $\mathcal{C}$ is straightforward and is done by replacing the Cartesian derivatives in the velocity-pressure formulation by their equivalent in the transformed computational space $\mathcal{C}$ (*i.e.*, eq. 3.28), as explained in Chapter 3, Section 3.

The equations defining the velocity-pressure formulation are discretized using second-order centred finite-difference approximations on overlapping grids. Let $\mathbf{U_i}(t)$ and $P_\mathbf{i}(t)$ denote the numerical approximations to $\mathbf{u}$ and p so that

$$\mathbf{U_i}(t) \approx \mathbf{u}(\mathbf{x_i}, t) \qquad \text{and} \qquad P_\mathbf{i}(t) \approx p(\mathbf{x_i}, t)$$

Here $\mathbf{U_i}(t) = (u_{1\mathbf{i}}(t), u_{2\mathbf{i}}(t)) = (u_\mathbf{i}(t), v_\mathbf{i}(t))$ is the vector containing the Cartesian components of the numerical approximation of the velocity. The spatial approximations of equations (5.8 - 5.10) are

$$\frac{d\mathbf{U_i}}{dt} = -(\mathbf{U_i} \cdot \nabla_h)\mathbf{U_i} - \frac{\nabla_h P_\mathbf{i}}{\rho} + \nu \nabla^2_h \mathbf{U_i}, \qquad i, j = 0, 1, 2, \ldots, N, \tag{5.11}$$

$$\frac{\nabla^2_h P_\mathbf{i}}{\rho} = -\sum_{m=1}^{\mathbb{N}} \nabla_h u_{m,\mathbf{i}} \cdot \frac{\partial \mathbf{U_i}}{\partial x_m}, \qquad i, j = 0, 1, 2 \ldots, N, \tag{5.12}$$

$$B_h(\mathbf{U_i}, \mathbf{P_i}) = \mathbf{g}(\mathbf{x_i}, t) \equiv (g_u(\mathbf{x_i}, t), g_v(\mathbf{x_i}, t)), \qquad i = 0, j = 0, 1, 2, \ldots, N, \tag{5.13}$$

$$\nabla_h \cdot \mathbf{U_i} = 0, \qquad i = 0, j = 0, 1, 2, \ldots, N, \tag{5.14}$$

$$\mathcal{D}\mathring{\mathbf{Q}}(\mathbf{x_i}, 0) = \mathbf{q}_0(\mathbf{x_i}, 0), \qquad i, j = 0, 1, 2 \ldots, N, \tag{5.15}$$

For the purposes of this discussion, the boundary conditions have only been specified at $i = 0, 0 < j < N$; at this boundary, we have considered a Dirichlet boundary condition for the velocity such as

$$\mathbf{u}(\mathbf{x}, t) = \mathbf{g}(\mathbf{x}, t) \qquad \text{for} \quad \mathbf{x} \in \partial\mathcal{D} \tag{5.16}$$

similar or more general expressions for the boundary conditions will hold at other boundaries (if there are other boundaries), although some of the details of implementation may vary [61, 92, 159]. In equations (5.11 - 5.15), subscript h denotes the order of accuracy of the numerical approximation, which for our case is equal to $h = 2$ (second-order centred finite-difference approximations). For the sake of simplicity, the subscript h will be dropped for the remainder of this dissertation.

Higher-order accurate methods based on the velocity-pressure formulation have been successfully used for solving the incompressible Navier-Stokes equations. Henshaw, Kreiss and Reyna [80] developed a fourth-order finite difference scheme based on this approach and also gave a stability analysis. They also presented a general principle for deriving numerical boundary conditions for higher-order accurate difference schemes. In addition, Henshaw [78], adapted the scheme to compute three-dimensional flows on complex domains using overlapping grids, where he introduced extra boundary conditions to make the scheme accurate and stable. Moreover, Browning [27], used sixth-order finite-difference methods on overlapping grids to solve the shallow water equations on a sphere. In [214], Wright and Shyy present a fourth-order accurate pressure-based composite grid method for solving the incompressible Navier-Stokes equations on domains composed by an arbitrary number of overlain grid blocks, where a conservative internal boundary scheme is devised to ensure that global conservation is maintained.

The discrete operators appearing in equations (5.11 - 5.15), are defined as follows,

$$
\begin{aligned}
\nabla \cdot \mathbf{U_i} &= D_{0x} u_{\mathbf{i}} + D_{0y} v_{\mathbf{i}} \,, \\
\nabla^2 \mathbf{U_i} &= \left(D_{+x} D_{-x} + D_{+y} D_{-y} \right) \mathbf{U_i} \\
\nabla P_{\mathbf{i}} &= \left(D_{0x} P_{\mathbf{i}}, D_{0y} P_{\mathbf{i}} \right)^T \\
\nabla \mathbf{U_i} &= \left(D_{0x} u_{\mathbf{i}}, D_{0y} v_{\mathbf{i}} \right)^T
\end{aligned}
$$

where

$$
\begin{aligned}
D_x \mathbf{U_i} = D_{0x} \mathbf{U_i} &= \frac{\mathbf{U}_{i+1,j} - \mathbf{U}_{i-1,j}}{2h} \approx \frac{\partial}{\partial x} \,, \\
D_y \mathbf{U_i} = D_{0y} \mathbf{U_i} &= \frac{\mathbf{U}_{i,j+1} - \mathbf{U}_{i,j-1}}{2h} \approx \frac{\partial}{\partial y} \,, \\
D_{+x} \mathbf{U_i} &= \frac{\mathbf{U}_{i+1,j} - \mathbf{U}_{i,j}}{h} \approx \frac{\partial}{\partial x} \,, \\
D_{-x} \mathbf{U_i} &= \frac{\mathbf{U}_{i,j} - \mathbf{U}_{i-1,j}}{h} \approx \frac{\partial}{\partial x} \,, \\
D_{+y} \mathbf{U_i} &= \frac{\mathbf{U}_{i,j+1} - \mathbf{U}_{i,j}}{h} \approx \frac{\partial}{\partial y} \,, \\
D_{-y} \mathbf{U_i} &= \frac{\mathbf{U}_{i,j} - \mathbf{U}_{i,j-1}}{h} \approx \frac{\partial}{\partial y} \,, \\
D_x^2 \mathbf{U_i} = D_{0x}^2 \mathbf{U_i} = D_{+x} D_{-x} \mathbf{U_i} = D_{+x} \mathbf{U_i} - D_{-x} \mathbf{U_i} &= \frac{\mathbf{U}_{i+1,j} - 2\mathbf{U}_{i,j} + \mathbf{U}_{i-1,j}}{h^2} \approx \frac{\partial^2}{\partial x} \,, \\
D_y^2 \mathbf{U_i} = D_{0y}^2 \mathbf{U_i} = D_{+y} D_{-y} \mathbf{U_i} = D_{+x} \mathbf{U_i} - D_{-x} \mathbf{U_i} &= \frac{\mathbf{U}_{i,j+1} - 2\mathbf{U}_{i,j} + \mathbf{U}_{i,j-1}}{h^2} \approx \frac{\partial^2}{\partial y} \,,
\end{aligned}
$$

where D_{+x} is the forward divided difference operator, D_{-x} is the backward divided difference operator and $D_x = D_{0x}$ is the centred divided difference operator, with analogous definitions for

the y direction.

Notice that by using compact difference approximations discretization, the checker-board instability is avoided [81]. The lack of a proper explicit boundary condition for the PPE equation has traditionally been a troubling point when designing or implementing numerical schemes based on the velocity-pressure formulation of the incompressible Navier-Stokes equations. Here, as a boundary condition for the PPE equation (eq. 5.9), we use the normal component $\hat{\mathbf{n}}$ of the momentum equation eq. 5.8 at the boundary $i = 0, 0 < j < N$, as discussed by Gresho and Sani [61], Henshaw [78], Johnston and Liu [92], Petersson [142] and Sani *et al.* [159], where

$$\left.\frac{\partial p}{\partial \hat{\mathbf{n}}}\right|_{\partial\mathcal{D}} = \left.\hat{\mathbf{n}}\right|_{\partial\mathcal{D}} \cdot \left(-\mathbf{g}_t - (\mathbf{g}\cdot\nabla)\,\mathbf{u} + \nu\nabla^2\mathbf{u}\right)\rho\,, \tag{5.17}$$

and we extrapolate the tangential component $\hat{\mathbf{t}}$ of the velocity, such as

$$(D_+)^{pe}\, v_{\mathbf{i}} = 0\,, \qquad \text{for} \qquad i = -1, j = 0, 1, 2, \ldots, N \tag{5.18}$$

where pe is the order of the polynomial extrapolation.

We call eq. 5.17 the div-grad pressure boundary condition and is obtained by taking the dot product between the momentum equation (eq. 5.8) and the unit normal $\hat{\mathbf{n}}$ to the boundary $\partial\mathcal{D}$. Note that by itself it adds no new information to the continuous PDE (since the momentum equation already is satisfied on the boundary) and cannot replace $\nabla\cdot\mathbf{u} = 0$ as the extra essential boundary condition required by the velocity-pressure formulation [81, 142]. Let us now obtain the discrete form of eq. 5.17,

$$D_{0x}P_{0j} = \nu D_{+x}D_{-x}u_{0j}\,, \qquad i = 0, j = 0, 1, 2, \ldots, N, \tag{5.19}$$

where for simplicity we assume $\mathbf{g}|_{\partial\mathcal{D}}$ to be equal to $\mathbf{g}|_{\partial\mathcal{D}} = 0$ (no-slip wall), which can be done without loss of generality [159]. Note that eq. 5.19 requires the value of a flow variable outside the physical domain $\mathcal{P}$ (ghost points), namely $u_{-1,j}$. If we discretize the divergence-free boundary condition $\nabla\cdot\mathbf{u} = 0$ (eq. 5.14) of the velocity-pressure formulation, we obtain

$$0 = \nabla\cdot\mathbf{U}_{\mathbf{i}} = D_{0x}u_{\mathbf{i}} + D_{0y}v_{\mathbf{i}} = D_{0x}u_{\mathbf{i}} + 0 = D_{0x}u_{\mathbf{i}}\,, \qquad i = 0, j = 0, 1, 2, \ldots, N, \tag{5.20}$$

from eq. 5.20 it is obvious that $u_{1,j} = u_{-1,j}$. This implies that in eq. 5.19 one should take $u_{1,j} = u_{-1,j}$, resulting in the following approximation for the Neumann boundary condition eq. 5.19 for the PPE equation

$$D_{0x}P_{0j} = \nu\frac{2}{h}D_{+x}u_{0j}\,, \qquad i = 0, j = 0, 1, 2, \ldots, N, \tag{5.21}$$

We can now see how $\nabla\cdot\mathbf{u} = 0$ provides a boundary condition for the pressure; the discrete divergence-free boundary condition eq. 5.14 determines the ghost line value of the normal component of the velocity $u_{-1,j}$, which is used in the right hand side of eq. 5.19. It is important to emphasizes that in order to achieve a stable scheme using eq. 5.17, it is extremely important to also enforce the essential boundary condition $\nabla\cdot\mathbf{u} = 0$ [81, 142].

5.5 Time-Stepping Algorithm for the Velocity-Pressure Formulation of the Incompressible Navier-Stokes Equations

The method of lines approach is used to solve the discretized equations in time. The method of lines (MOL) [65, 160, 210], is a technique for solving partial differential equations (PDEs) where all but one dimension is discretized. The resulting semi-discrete problem is a set of ordinary differential equations (ODEs) or differential algebraic equations (DAEs) that is then integrated in the undiscretized dimension. The basic idea of the MOL is to replace the spatial (boundary value) derivatives in the PDE with algebraic approximations. Once this is done, the spatial derivatives are no longer stated explicitly in terms of the spatial independent variables. Thus, in effect only the initial value variable (typically time t) remains. In other words, with only one remaining independent variable, we have a system of ODEs that approximate the original PDE. Once this is done, we can apply any integration algorithm for initial value ODEs to compute the approximate numerical solution of the PDE. One significant advantage of the MOL is that it allows the use of existing and generally well established numerical methods for ODEs. For PDEs where it is suitable, MOL is an efficient solution method [65].

After discretizing the equations in space on an overlapping grid system $\mathbb{G}$, one can regard the resulting system as a system of ordinary differential equations ODEs of the form

$$\frac{d\mathbf{U}}{dt} = \mathcal{F}\left(t, \mathbf{U}, P\right)$$

where the pressure P is considered to be a function of the velocity, $P = p(\mathbf{U})$. Now we can use any time integrator on a MOL fashion to solve equations eq. 5.11, eq. 5.12, eq. 5.13, eq. 5.14, eq. 5.18 and eq. 5.19.

In order to keep the solution of the pressure equation decoupled from the solution of the velocity components, we choose a time stepping scheme for the velocity components that only involves the pressure from the previous time steps (split-step scheme). Let us introduce the operators $\mathbb{L} = \mathbb{L}_{\mathbf{E}} + \mathbb{L}_{\mathbf{I}}$ representing the various terms in the momentum equations, as follows

$$\begin{aligned}
\mathbb{L} = \mathbb{L}\mathbf{U_i} &\equiv -\left(\mathbf{U_i}\cdot\nabla\right)\mathbf{U_i} - \frac{\nabla P_{\mathbf{i}}}{\rho} + \nu\nabla^2\mathbf{U_i}\,, \\
\mathbb{L}_{\mathbf{E}} = \mathbb{L}_{\mathbf{E}}\mathbf{U_i} &\equiv -\left(\mathbf{U_i}\cdot\nabla\right)\mathbf{U_i} - \frac{\nabla P_{\mathbf{i}}}{\rho}\,, \\
\mathbb{L}_{\mathbf{I}} = \mathbb{L}_{\mathbf{I}}\mathbf{U_i} &\equiv \nu\nabla^2\mathbf{U_i}\,,
\end{aligned}$$

where $\mathbb{L}_{\mathbf{E}}$ and $\mathbb{L}_{\mathbf{I}}$ are the operators that we treat explicitly and implicitly respectively. Then, the equations (5.11 - 5.14) and (5.18 - 5.19) are integrated using a semi-implicit multistep method, that uses a Crank-Nicolson scheme for the viscous terms and a second-order Adams-Bashforth predictor-corrector approach for the advection terms and pressure. We choose to implicitly treat the viscous terms because if they were treated explicitly we could have a severe time step restriction, proportional to the spatial discretization squared.

By using this time-stepping scheme, the velocity is advanced in time using a second-order Adams-Bashforth predictor step as follows

$$\frac{\mathbf{U}_{\mathbf{i}}^{p}-\mathbf{U}_{\mathbf{i}}^{n}}{\Delta t} = \frac{3}{2}\mathbb{L}_{\mathbf{E}}^{n}-\frac{1}{2}\mathbb{L}_{\mathbf{E}}^{n-1}+\alpha\mathbb{L}_{\mathbf{I}}^{p}+(1-\alpha)\,\mathbb{L}_{\mathbf{I}}^{n}\,, \qquad \text{for} \qquad i,j=1,2,\ldots,N-1, \quad (5.22)$$

followed by a second-order Adams-Moulton corrector step or the form

$$\frac{\mathbf{U}_{\mathbf{i}}^{c}-\mathbf{U}_{\mathbf{i}}^{n}}{\Delta t} = \frac{1}{2}\mathbb{L}_{\mathbf{E}}^{p}+\frac{1}{2}\mathbb{L}_{\mathbf{E}}^{n}+\alpha\mathbb{L}_{\mathbf{I}}^{c}+(1-\alpha)\,\mathbb{L}_{\mathbf{I}}^{n}\,, \qquad \text{for} \qquad i,j=1,2,\ldots,N-1, \quad (5.23)$$

where only one corrector step has been used (one may optionally correct more than one time as that should be inexpensive and allows a bigger time step for moving grids [71]). In equations eq. 5.22 and eq. 5.23 the super-script p stands for predicted value, the super-script c stands for corrected value, α is the implicit parameter and $\mathbf{U}_{\mathbf{i}}^{n} \approx \mathbf{u}(\mathbf{x}_{\mathbf{i}}, n\Delta t)$. For $\alpha = 1/2$ we obtain a second-order Crank-Nicolson method, whereas for $\alpha = 1$ we obtain a first-order backward Euler method.

Equations eq. 5.22 and eq. 5.23 are advanced to time $n+1$ together with the following equations

$$\begin{aligned}
\mathbf{U}_{\mathbf{i}}^{n+1} &= \mathbf{g}\left(\mathbf{x}_{\mathbf{i}}, t^{n+1}\right), && \text{for} \quad i=0, j=0,1,2,\ldots,N, && (5.24)\\
D_{0x}u_{\mathbf{i}}^{n+1} &= -D_{0y}g_v\left(\mathbf{x}_{\mathbf{i}}, t^{n+1}\right), && \text{for} \quad i=0, j=0,1,2,\ldots,N, && (5.25)\\
\left(D_{+}\right)^{pe} v_{-1j}^{n+1} &= 0\,, && \text{for} \quad i=-1, j=0,1,2,\ldots,N, && (5.26)
\end{aligned}$$

Equations (5.22 - 5.26), determine $\mathbf{U}_{\mathbf{i}}^{n+1}$ at all points including the ghost points. We then solve for the pressure at time $n+1$ using

$$\begin{aligned}
\frac{\nabla^2 P_{\mathbf{i}}}{\rho}^{n+1} &= -\sum_{m=1}^{\mathbb{N}} \nabla u_{m,\mathbf{i}}^{n+1} \cdot \frac{\partial \mathbf{U}_{\mathbf{i}}^{n+1}}{\partial x_m}\,, && i,j=0,1,2.\ldots,N, && (5.27)\\
D_{0x}P_{\mathbf{i}}^{n+1} &= \nu D_{+x}D_{-x}u_{\mathbf{i}}^{n+1} + B_p\left(\mathbf{U}_{\mathbf{i}}^{n+1}, \mathbf{g}_{\mathbf{i}}^{n+1}\right), && i=0, j=0,1,2,\ldots,N, && (5.28)
\end{aligned}$$

where the boundary forcing $B_p(\mathbf{U},\, \mathbf{g})$ satisfies

$$B_p(\mathbf{U},\, \mathbf{g}) = -\frac{\partial g_u}{\partial t} - g_u D_{0x}u - g_v D_{0y}g_v + \nu D_{+y}D_{-y}g_u \quad (5.29)$$

5.6 Velocity-Pressure Formulation for Moving Overlapping Grids

On a non-moving overlapping grid system $\mathbb{G}$, each component grid $\mathcal{G}_g$ is defined by a smooth mapping $\mathbf{M}_g$ from the computational space $\mathcal{C} = \mathcal{C}(\xi, \eta, \zeta, \tau)$ to the physical space $\mathcal{P} = \mathcal{P}(x, y, z, t)$, such that

$$\mathcal{P} = \mathbf{M}_g\left(\mathcal{C}\right) \qquad \text{or equivalently} \qquad \mathbf{x} = \mathbf{M}_g\left(\mathbf{r}\right)$$

where $\mathbf{x}$ denotes the coordinates in physical space $\mathcal{P}$ and $\mathbf{r}$ denotes the coordinates in computational space $\mathcal{C}$. On a moving grid, the moving mapping depends on time, such as

$$\mathbf{x} = \mathbf{M}_g\left(\mathbf{r}, t\right)$$

On moving grids we solve the governing PDE in a frame that moves with the grid. Thus, if

we are solving the velocity-pressure formulation of the incompressible Navier-Stokes equations in physical space $\mathcal{P} = \mathbf{x}$

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = \frac{-\nabla p}{\rho} + \nu \nabla^2 \mathbf{u}$$
$$\frac{\nabla^2 p}{\rho} + \nabla u \cdot \mathbf{u}_x + \nabla v \cdot \mathbf{u}_y + \nabla w \cdot \mathbf{u}_z = 0$$

then on each moving component grid $\mathcal{G}_g$ of the overlapping grid system $\mathbb{G}$, we make the change of variables from $\mathcal{P} = \mathbf{x}(x, y, z, t)$ to $\mathcal{C} = \mathbf{r}(\xi, \eta, \zeta, \tau)$ defined by

$$\begin{aligned} \mathbf{x} &= \mathbf{M}_g(\mathbf{r}, \tau) \\ t &= \tau \\ \mathbf{u}(\mathbf{x}, t) &= \mathbf{u}(\mathbf{M}_g(\mathbf{r}, \tau)) \equiv \mathbf{U}(\mathbf{r}, \tau) \end{aligned}$$

and as already outlined in Chapter 4, Section 3.1, the time derivative of $\mathbf{u}(\mathbf{x}, t)$ at a fixed point of the physical space $\mathbf{x}$ is related to its time-derivative of a fixed point of the computational space $\mathbf{r}$ by the following equation

$$\frac{\partial \mathbf{u}}{\partial t} = \frac{\partial \mathbf{U}}{\partial \tau} - \dot{\mathbf{G}} \cdot \nabla \mathbf{u} = \frac{\partial \mathbf{U}}{\partial t} - \dot{\mathbf{G}} \left(\nabla_{\mathbf{x}} \mathbf{r} \cdot \nabla_{\mathbf{r}} \right) \mathbf{U} \tag{5.30}$$

where

$$\dot{\mathbf{G}} = \frac{\partial \mathbf{M}_g(\mathbf{r}, t)}{\partial t} \tag{5.31}$$

is the grid velocity.

By replacing equations eq. 5.30 and eq. 5.31 into the incompressible Navier-Stokes equations eq. 5.8 and eq. 5.9 we obtain,

$$\frac{\partial \mathbf{U}}{\partial t} + \left[\left(\mathbf{U} - \dot{\mathbf{G}} \right) \cdot \nabla_{\mathbf{r}} \right] \mathbf{U} = \frac{-\nabla_{\mathbf{r}} p}{\rho} + \nu \nabla_{\mathbf{r}}^2 \mathbf{U} \tag{5.32}$$

$$\frac{\nabla_{\mathbf{r}}^2 p}{\rho} + \sum_{m=1}^{\mathbb{N}} \nabla_{\mathbf{r}} \mathbf{U}_m \cdot \partial_{\mathbf{x}_m} \mathbf{U} = 0 \tag{5.33}$$

which is the velocity-pressure formulation of the incompressible Navier-Stokes equations expressed in a moving frame in computational space $\mathbf{r}$.

5.6.1 Boundary Conditions for Moving Walls

The new governing equations expressed in the moving reference frame, must be accompanied by the proper boundary conditions. For a moving body with a corresponding moving wall, only one constraint may be applied and this corresponds to the velocity on the wall

$$\mathbf{U}(\mathbf{r},t) = \dot{\mathbf{G}}(\mathbf{r},t) \quad \text{for a no-slip wall} \tag{5.34}$$

$$\hat{\mathbf{n}} \cdot \mathbf{U}(\mathbf{r},t) = \hat{\mathbf{n}} \cdot \dot{\mathbf{G}}(\mathbf{r},t) \quad \text{for a slip wall} \tag{5.35}$$

On a moving no-slip wall the boundary condition for the pressure equation is obtained by dotting the normal $\hat{\mathbf{n}}$ into the momentum equation

$$\frac{1}{\rho}\frac{\partial p}{\partial \hat{\mathbf{n}}}\bigg|_{\partial \mathcal{D}_{wall}} = \hat{\mathbf{n}}\bigg|_{\partial \mathcal{D}_{wall}} \cdot \left(-\ddot{\mathbf{G}} + \nu \nabla_{\mathbf{r}}^2 \mathbf{U}\right) \tag{5.36}$$

Note that the acceleration of the wall appears on the right hand side of eq. 5.36.

5.7 Boundary Conditions

Imposing appropriate boundary conditions to the incompressible Navier-Stokes equations is of paramount importance for the success of every numerical algorithm. The type of boundary conditions to be imposed are dependent on the physics of the flow once the geometry and topology of the selected problem have been determined. In this dissertation, the applications and the flow geometry solved in general belong to external flow problems, where the normal unit vector out of the solid surface points away from the surface towards the computational domain (see figure 5.1).

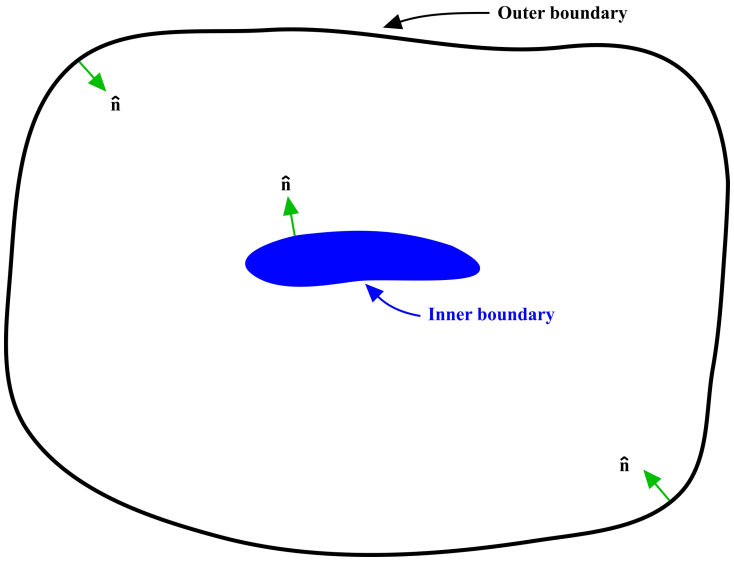

Figure 5.1: *General boundary configuration for external flows.*

The incompressible Navier-Stokes equations in their velocity-pressure formulation are numerically solved using the Overture[1] framework together with the PETSc[2] library. Using Overture, elementary boundary conditions such as Dirichlet boundary conditions, Neumann boundary conditions and mixed boundary conditions, extrapolation boundary conditions, symmetry boundary

[1] https://computation.llnl.gov/casc/Overture/
[2] http://www-unix.mcs.anl.gov/petsc/petsc-as/

conditions and so on, can be easily implemented. Besides the boundary conditions enforced in the velocity-pressure formulation, the following boundary conditions may be used,

$$\text{No-slip wall} = \begin{cases} \mathbf{u} = \mathbf{g} & \text{velocity specified} \\ \nabla \cdot \mathbf{u} = 0 & \text{zero divergence} \end{cases}$$

$$\text{Slip wall} = \begin{cases} \hat{\mathbf{n}} \cdot \mathbf{u} = g & \text{normal velocity specified} \\ \partial_{\hat{\mathbf{n}}} \left(\hat{\mathbf{t}} \cdot \mathbf{u} \right) = 0 & \text{normal derivative of} \\ & \text{tangential velocity is zero} \\ \nabla \cdot \mathbf{u} = 0 & \text{zero divergence} \end{cases}$$

$$\text{Inflow with velocity given} = \begin{cases} \mathbf{u} = \mathbf{g} & \text{velocity specified} \\ \partial_{\hat{\mathbf{n}}} p = 0 & \text{normal derivative of the pressure zero} \end{cases}$$

$$\text{Outflow} = \begin{cases} \text{extrapolate } \mathbf{u} & \text{velocity specified} \\ \alpha p + \beta \partial_{\hat{\mathbf{n}}} p = g & \text{mixed derivative of } p \text{ given} \end{cases}$$

$$\text{Dirichlet boundary condition} = \begin{cases} \mathbf{u} = \mathbf{g} & \text{velocity specified} \\ p = P & \text{pressure given} \end{cases}$$

$$\text{Symmetry} = \begin{cases} \hat{\mathbf{n}} \cdot \mathbf{u} : \text{odd}, \ \hat{\mathbf{t}} \cdot \mathbf{u} : \text{even} & \text{vector symmetry} \\ \partial_{\hat{\mathbf{n}}} p = 0 & \text{normal derivative of the pressure zero} \end{cases}$$

On moving walls, the boundary conditions are those specified in the previous section (equations eq. 5.34, eq. 5.35 and eq. 5.36).

5.8 Discrete Divergence Damping

Due to truncation errors and because of the interpolation between the component grids $\mathcal{G}_g$ of the overlapping grid system $\mathbb{G}$, the divergence ($\delta = \partial u/\partial x + \partial v/\partial y$) will not be identically zero in the numerical computation. Hence, an extra-term, namely discrete divergence damping $\alpha_{\mathbf{i}} \nabla \cdot \mathbf{U}_{\mathbf{i}}$, is often added in the pressure equation eq. 5.12 in order to suppress the spurious divergence. Equation 5.12 becomes,

$$\frac{\nabla_h^2 P_{\mathbf{i}}}{\rho} = \alpha_{\mathbf{i}} \nabla \cdot \mathbf{U}_{\mathbf{i}} - \sum_{m=1}^{\mathbb{N}} \nabla_h u_{m,\mathbf{i}} \cdot \frac{\partial \mathbf{U}_{\mathbf{i}}}{\partial x_m}, \qquad i, j = 0, 1, 2 \ldots, N, \tag{5.37}$$

This technique of adding a damping term is well known and has been used previously by a number of researchers in the field of incompressible flows (*e.g.*, the MAC method of Harlow and Welch [66] or the fourth-order velocity-pressure method of Henshaw, Kreiss and Reyna [80]). This term can be seen as a divergence sink, since it appears as a sink in the PPE equation, helping to keep the discrete divergence small. A detailed description of the coefficient $\alpha_{\mathbf{i}}$ is given by Henshaw in [72, 78] and Henshaw and Kreiss in [79].

One might wonder whether this divergence damping term, which is a potentially order one ad-

dition to the pressure equation, will destroy the accuracy of the method. In [79], Henshaw and Kreiss presented an analysis of this damping term using normal-mode stability analysis and showed why this term does not degrade the accuracy of the numerical method. They also found that increasing $\alpha_{\mathbf{i}}$ will result in a decrease of the maximum divergence (up to a point), but it can also increase the error in the pressure.

Chapter 6

Validation and Verification of the Navier-Stokes Flow Solver

Before proceeding to extensively use the Navier-Stokes flow solver for our calculations, it must be first validated. In this chapter, a qualitative and quantitative validation and verification of the proposed flow solver against experimental and numerical results is carried out in order to assess its numerical accuracy.

6.1 Flow Solver Validation and Verification. General Issues

Software Verification and Validation (V&V) is the process of ensuring that the code being developed or changed (in our case a flow solver) is: a) able to model with accuracy a real world problem, that is, *"it solves the right equations"* (validation) and b) it yields the right results or in other words, *"it solves the equation right"* (verification). For our purposes, the differences between validation and verification are unimportant and are just of interest to the theorist. Hereafter, the term V&V or just validation will be used to refer to all the qualitative and quantitative comparisons done to assess the accuracy of the proposed tools.

The proposed flow solver is Overture[1] together with PETSc[2]. Overture, is an object-oriented code framework for solving partial differential equations (PDEs) in serial and parallel environments. It is implemented as a collection of C++ libraries that enable the use of finite difference approximations to solve the governing PDEs in structured and overlapping structured grids. PETSc [11] is a suite of data structures and routines that includes a large series of linear and nonlinear equation solvers, preconditioners and Krylov subspace methods for the scalable (parallel) solution of large-scale scientific applications modeled by PDEs. Both tools used together provide a portable, scalable and flexible software development environment for applications that involve the simulation of physical processes in complex fixed or moving $2\mathbb{D}$ and $3\mathbb{D}$ geometries.

[1] https://computation.llnl.gov/casc/Overture/
[2] http://www-unix.mcs.anl.gov/petsc/petsc-as/

6.2 Numerical Results and V&V

6.2.1 The Method of Manufactured Solutions or Forced Solutions

As a first test to check the numerical accuracy of the flow solver, we use the method of manufactured solutions (MMS) or forced solutions [154]. The basic idea behind this method is to simply manufacture an exact solution to the governing equations without being concerned about its physical realism. This solution also defines the boundary conditions to be applied in any forms, *i.e.*, Dirichlet, Neumann, Robin, etc., and the initial conditions. In the MMS, the governing equations are modified through the addition of a source term so that the manufactured solution is an exact solution to the governing equations with this additional source term. The particular form of the source term depends on the manufactured solution selected. This form is found by applying the governing equations operators to the manufactured solution to obtain an analytic formula for the source terms. The source terms are then added to the original equation set to balance it. Then the discrete solutions produced by the code can be compared to the manufactured solution to determine the discretization error. A comparison of the discretization error on a series of uniformly refined meshes will either verify that the observed order of accuracy matches the theoretical order of accuracy, or it will not. In the latter case, it may indicate a coding mistake or improper formulation. A useful set of guidelines for the effective design and application of the MMS are given by Roache [154] and Knupp and Salari [102].

In general, when constructing manufactured solutions, the following guidelines should be observed [102]. First, manufactured solutions should be composed of smooth analytic functions like polynomials, trigonometric or exponential functions. Second, the solution should be general enough that it exercises every term in the governing equations. Third, the solution should have a sufficient number of non-trivial derivatives. Finally, solution derivatives should be bounded by a small constant, this ensures that the solution is not a strongly varying function of space and/or time.

The MMS is much easier and more general than looking for analytical solutions to real problems. When this systematic procedure is used, we are testing for

- All the transformations used (*i.e.*, transformation of the governing equations to generalized curvilinear coordinates).
- The order of the discretization (spatial and temporal).
- The matrix solution procedure.
- Correctness of the numerical discretization.
- Interpolation between the overlapping grids.

Hereafter, we use the MMS to check the numerical accuracy of the flow solver. Here, the incompressible Navier-Stokes equations around a circle in a square with slip walls boundary conditions are solved. In two space dimension we use the following trigonometric functions as the manufactured solutions

$$
\begin{aligned}
u &= \frac{1}{2}\cos(\pi\omega_0 x)\cos(\pi\omega_1 y)\cos(\pi\omega_3 t) + \frac{1}{2} \\
v &= \frac{1}{2}\sin(\pi\omega_0 x)\sin(\pi\omega_1 y)\cos(\pi\omega_3 t) + \frac{1}{2} \\
p &= \cos(\pi\omega_0 x)\cos(\pi\omega_1 y)\cos(\pi\omega_3 t) + \frac{1}{2}
\end{aligned}
\tag{6.1}
$$

when $\omega_0 = \omega_1$ it follows that $\nabla \cdot \mathbf{u} = 0$ (the solution is divergence free).

In table 6.1, the results of this convergence test are presented. In this table, the maximum error in $\mathbf{u}$, p and $\nabla \cdot \mathbf{u}$ are shown. The estimated convergence rate p is also presented. In figure 6.1, the overlapping grids and the solutions for three refinement levels are illustrated.

Grid	h_1/h_g	$\|u - u_{exact}\|_\infty$	$\|v - v_{exact}\|_\infty$	$\|p - p_{exact}\|_\infty$	$\|\nabla \cdot \mathbf{u}\|_\infty$
$\mathbb{G}_1 : 32 \times 32 \cup 34 \times 8$	1	0.0205	0.0235	0.0473	0.0958
$\mathbb{G}_2 : 64 \times 64 \cup 68 \times 16$	2	0.00487	0.0037	0.0128	0.0200
$\mathbb{G}_3 : 128 \times 128 \cup 136 \times 32$	4	0.00120	0.000698	0.00453	0.00635
Order of convergence p		2.24	2.72	2.06	2.47

Table 6.1: *Maximum error at t=1.0 and $\nu = 0.1$ for a trigonometric analytic solution ($\omega_0 = \omega_1 = \omega_3 = 1$). The estimated convergence rate p is also shown. The column entitled as h_1/h_g denotes the ratio of the grid spacing on grid 1 to the spacing on grid g.*

A Navier-Stokes flow solver uses a numerical algorithm that will provide a theoretical order of convergence; however, the boundary conditions, numerical models, non-linearities in the solution, presence of shocks, grid refinement (or coarsening) and perhaps other factors, will reduce this order so that the observed order of convergence p will likely be different than the theoretical one [154, 170]. As outlined by Roache [154], if a grid refinement r is performed with constant refinement ratio r (not necessarily r=2) between all the grids, the observed order of convergence p can be obtained directly from three grid solutions as follows

$$
p = \frac{\ln\left(\dfrac{f_3 - f_2}{f_2 - f_1}\right)}{\ln(r)} \tag{6.2}
$$

where f is the approximate solution or the value of an observed quantity, with the sub-index $_1$ being the solution on the finest grid, $_2$ in the intermediate grid and $_3$ in the coarser grid. If one generates a finer or coarser grid and is unsure of the value of grid refinement ratio r used, one can compute the equivalent effective grid refinement ratio $r_{effective}$ as

$$
r_{effective} = \left(\frac{\mathcal{N}_1}{\mathcal{N}_2}\right)^{\frac{1}{\mathbb{D}}} \tag{6.3}
$$

where $\mathcal{N}$ is the total number of grid points used for the grid and $\mathbb{D}$ is the dimension of the flow domain; here again, the sub-index $_1$ corresponds to the solution on the finest grid, the sub-index $_2$ the solution in the intermediate grid and the sub-index $_3$ the solution in the coarser grid.

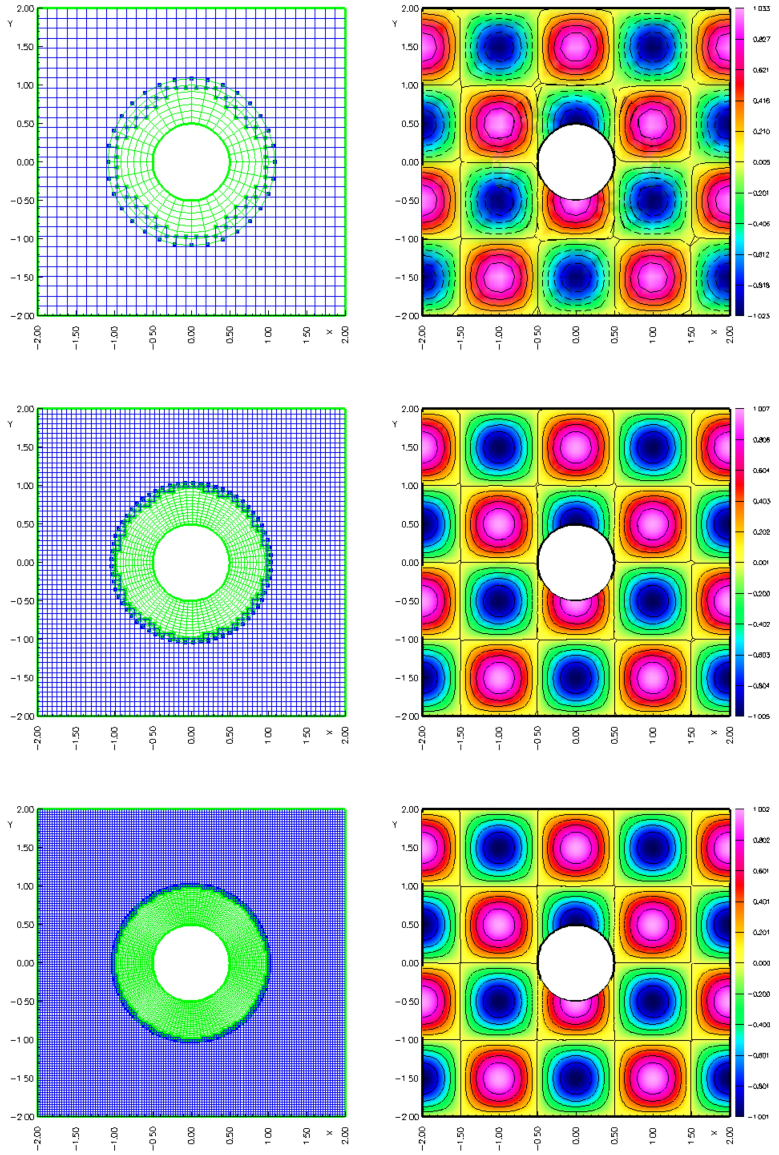

Figure 6.1: *Forced solutions of the incompressible Navier-Stokes equations around a circle in a square with slip wall boundaries and* $\omega_0 = \omega_1 = \omega_3 = 1$. *Top-to-bottom left column, grid system from coarser grid to finer grid. Top-to-bottom right column, corresponding grid level velocity u contours. Notice how the quality of the solution improves as the grid is refined.*

The results presented in table 6.1, show that although the method is converging at the expected convergence rate p (second order accuracy), the errors are significantly reduced when the grid is refined. Note that as the overlapping grids are refined, the positions of the interpolation points will change since the effective overlap decrease. As a result, the reduction in the error is not always as uniform as that from a single grid [76].

6.2.2 Flow Past a Stationary Cylinder at Various Reynolds Number Values

Hereafter, the results from the computation of a flow past a stationary cylinder in an overlapping grid system are shown. Simulations were performed at Reynolds numbers equal to 20, 40, 100, 200 and 400. For the cases of Re equal to 20, 40, 100 and 200, the obtained results were compared with other numerical and experimental data published in the literature. The case of Re equal to 400 is used as a benchmarking case in order to compare the performance of different direct and iterative solution methods.

In figure 6.2, the domain used for the cases where Re is equal to 20, 40 , 100 and 200 is illustrated. Here, the cylinder is located at the origin and has a radius of 0.5. The outer rectangular computational domain extends from $[x_a, x_b] \times [y_a, y_b] = [-5.0, 20.0] \times [-5.0, 5.0]$ and the inner cylindrical computational domain extends from $[x_{origin}, y_{origin}] \times [radius_{inner}, radius_{outer}] = [0.0, 0.0] \times [0.5, 1.0]$ (see figure 6.2). The initial conditions for all cases are those of a uniform flow with $(u, v, p) = (1.0, 0.0, 1.0)$ all over the domain. The top and bottom boundaries of the rectangular domain are slip walls. The left boundary of the rectangular domain is inflow and the right boundary is outflow. The cylinder's wall has a no-slip boundary condition. The Reynolds number based on the cylinder diameter, the kinematic viscosity and the inflow velocity $(u, v) = (1.0, 0.0)$ is controlled by changing the kinematic viscosity.

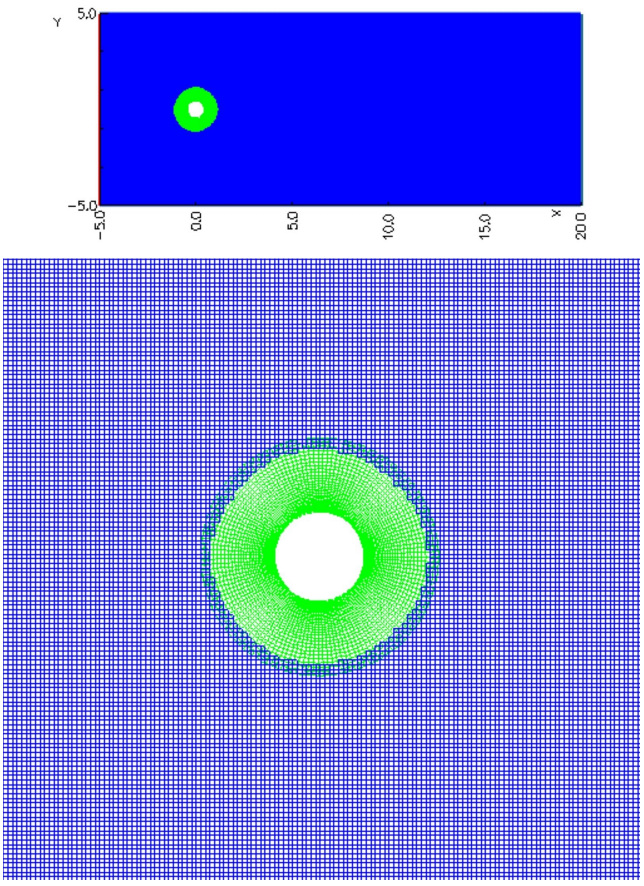

Figure 6.2: *Domain and overlapping grid system of the unsteady flow past a cylinder case. Top view: overall domain. Bottom view: close-up of the grid around the cylinder.*

For the cases where Re is equal to 20 and 40, the wake behind the cylinder shows a steady symmetric behavior as shown in figure 6.3, these solutions are consistent with the well established result that the wake behind the cylinder consist of a steady recirculation region of two symmetrically placed vortices on each side of the wake and is stable to perturbations below a Reynolds number value approximately equals to 46 ± 1 [90, 150, 180, 211]. In table 6.2, a comparison between the values obtained and other numerical and experimental results is presented. Here, despite the fact that the results found in the literature vary by as much as 10% from one another, it is found that the current results compare well with the other numerical simulations and experiments, falling within the range of the reported values.

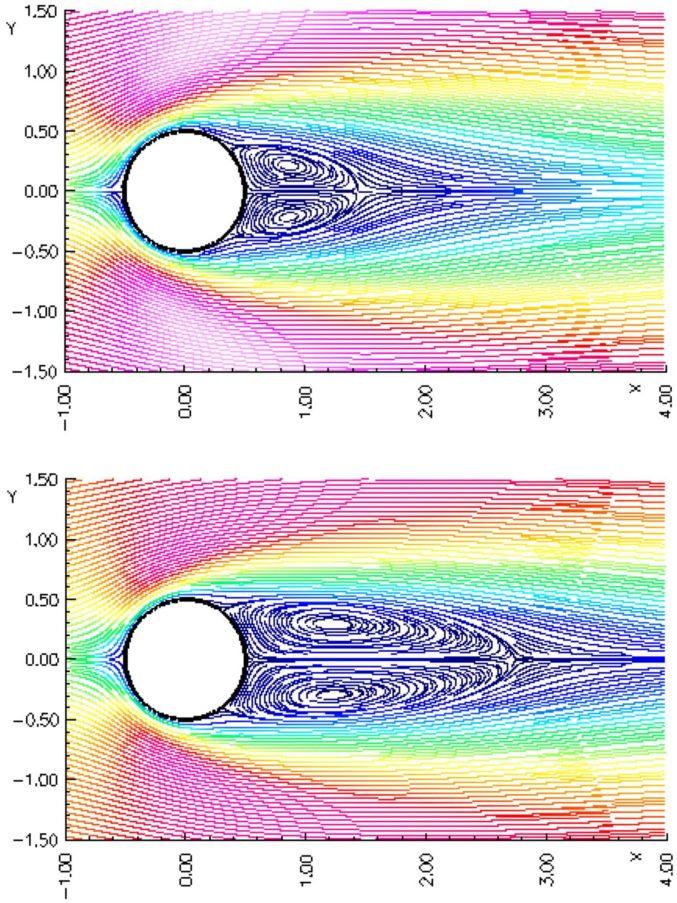

Figure 6.3: *Streamlines for* $Re = 20$ *(top figure) and* $Re = 40$ *(bottom figure) for a nondimensional time* $t = 400$.

It is generally accepted that the wake behind a cylinder first becomes unstable at a critical Reynolds number value of about $Re = 46 \pm 1$ [90, 150, 180, 211], as predicted by the linear theory of stability. Above this critical Re value, a small asymmetric perturbation in the near wake starts to grow in time and leads to an unsteady wake, known as von Karman vortex street, which is indeed what we found for the simulations where Re is equal to 100 and 200 (see figure 6.4). Figure 6.5, shows the variation of lift and drag coefficient with nondimensional time for the case where $Re = 100$. In this unsteady flow regime, the present results are compared

Reference	c_d for $Re = 20$	L for $Re = 20$	c_d for $Re = 40$	L for $Re = 40$
Tritton [194]^{E}	2.22	-	1.48	-
Coutanceau and Bouard [42]^{E}	-	0.73	-	1.89
Russel and Wang [157]^{N}	2.13	0.94	1.60	2.29
Calhoun [30]^{N}	2.19	0.91	1.62	2.18
Mittal *et al.* [215]^{N}	2.03	0.92	1.52	2.27
Fornberg [54]^{N}	2.00	0.91	1.50	2.24
Present results	2.2013	0.929	1.6208	2.216
^{E} Results from experiments				
^{N} Results from numerical computations				

Table 6.2: *Drag coefficient (c_d) and length of wake bubble (L) for $Re = 20$ and $Re = 40$.*

in table 6.3 against other numerical data. For $Re = 100$, the computed drag coefficient and lift coefficient were very close to those reported by Russel and Wang [157]; for the case of $Re = 200$, the present results compare favorably with those obtained by Braza *et al.* [24]. In general, it was found that our results compare well with the other numerical simulations and experiments. In table 6.4, a summary of the overlapping grid system used for all the previous cases is presented.

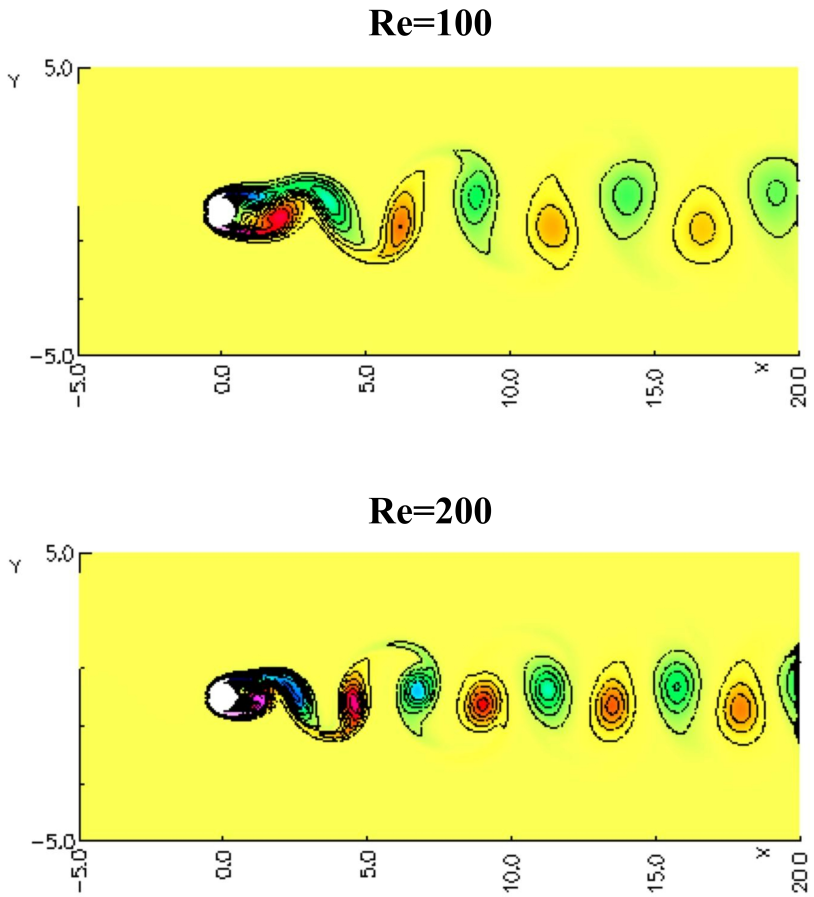

Figure 6.4: *Vorticity contours for* $Re = 100$ *and* $Re = 200$ *for a nondimensional time* $t = 500$.

The discretization of the Navier-Stokes equations leads to a large sparse non-symmetric non-linear system of equations that must be solved in order to obtain the approximate numerical solution to the initial-boundary-value-problem (IBVP). The numerical solution of the resulting system of non-linear equations is a major computational task in CFD and its accurate, robust and efficient solution is essential, especially if we want to solve larger systems (*i.e.*, finer grids).

There are several methods for solving the system of equations arising from the discretization of the governing equations, each raising the issue of the efficiency of the solution and its complexity. The increase in computing power and the introduction of parallel computing has driven the latest advances in algorithm development for the solution of large sparse systems of equations. Most of the research these days is focused on the efficient solution of these systems of equations on complex and large domains in parallel environments. For small-sized problems and even moderate-sized

Reference	c_d for $Re = 100$	c_l for $Re = 100$	c_d for $Re = 200$	c_l for $Re = 200$
Russel and Wang [157]	1.38 ± 0.007	$\pm$ 0.322	1.29 ± 0.022	$\pm$ 0.50
Calhoun [30]	1.35 ± 0.014	$\pm$ 0.30	1.17 ± 0.058	$\pm$ 0.67
Braza *et al.* [24]	1.36 ± 0.015	$\pm$ 0.25	1.40 ± 0.05	$\pm$ 0.75
Choi *et al.* [39]	1.34 ± 0.011	$\pm$ 0.315	1.36 ± 0.048	$\pm$ 0.64
Liu *et al.* [111]	1.35 ± 0.012	$\pm$ 0.339	1.31 ± 0.049	$\pm$ 0.69
Present results	1.3898 ± 0.012	$\pm$ 0.3330	1.4087 ± 0.048	$\pm$ 0.7250

Table 6.3: *Drag coefficient (c_d) and lift coefficient (c_l) for $Re = 100$ and $Re = 200$.*

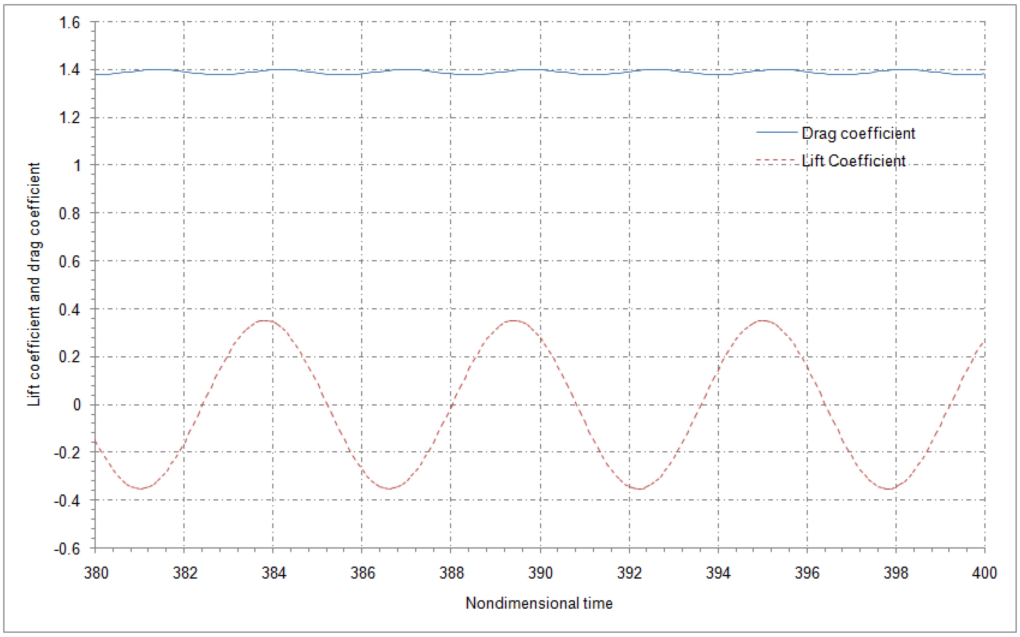

Figure 6.5: *Time dependent lift and drag coefficient for* $Re = 100$.

Number of grids	2
$\mathbb{G}_1$ dimensions (background grid)	500×200
$\mathbb{G}_2$ dimensions (annular grid)	140×80
Total number of grids points ($\mathbb{G}_1 \cup G_2$)	111200
$\mathbb{G}_1 \cup \mathbb{G}_2$, including ghost points	114752
Total number of unused points	1842
Total number of interpolation points	329
Position of the first node normal to the cylinder wall	$0.0001 \times diameter_{cyl}$

Table 6.4: *Summary of the overlapping grid system used for the cases where* $Re = 20$, $Re = 40$, $Re = 100$ *and* $Re = 200$.

problems, direct solvers are very efficient in solving the system of equations arising from the discretization, but they are not efficient for solving large problems [12, 158]. For large sparse systems of equations, Krylov iterative methods, in combination with a suitable preconditioner are the alternatives to direct solvers.

Newton-Krylov iterative methods for solving large non-linear systems have been used in CFD since the late 1980s [129, 199] and are considered an attractive and powerful approach to solve large problems due to their property of semi-quadratic convergence when starting from a good initial guess [45, 59, 158]. In Newton-Krylov methods, one applies a linearization method combined with a preconditioned Krylov subspace algorithm for solving the linear problem resulting from the linearization iteration. To enhance the efficiency and robustness of Newton-Krylov methods, it is necessary to apply preconditioning. Preconditioning is simply a means of transforming the original linear system into one which has the same solution, but which is relatively better conditioned and therefore is likely to be easier to solve with an iterative solver. The choice

of a preconditioner involves the selection of a matrix Q, called the preconditioning matrix or preconditioner, such that the preconditioned system

$$Q^{-1}Au = Q^{-1}b$$

is better conditioned than the original system, $Au = b$. Clearly, one requirement for Q is that it be easily invertible (*i.e.* the linear system having Q as the coefficient matrix can be solved with much less effort than solving the original system $Au = b$). In general, the reliability of iterative techniques, depends much more on the quality of the preconditioner than on the particular Krylov subspace solver or accelerator used. In practice, finding the best preconditioner for a given problem or class of matrices associated with a problem involves extensive testing.

Several authors [28, 29, 36, 153, 169] have studied the effect of various preconditioning methods on the convergence of Newton-Krylov iterative solvers. Their studies suggest that incomplete lower-upper ILU(k) factorization is a very efficient preconditioning strategy for a variety of Newton-Krylov solvers. The parameter k in ILU(k) denotes the level of fill-in that is allowed in the factorization, k equal to zero means no fill-in is permitted during ILU decomposition. In ILU(0) the factorized matrix and the original preconditioning matrix built from direct neighbors have the same graph (*i.e.*, same location for non zero-elements). Choosing k larger than zero would allow some additional fill-in in the factorized matrix which normally increases the accuracy of factorization and quality of the preconditioner. However, increasing the fill-in level would be at the expense of memory usage and increasing computing cost.

In the following simulation, we proceed to set the Reynolds number to 400 and we use the overlapping grid system described in table 6.5. In this case, the outer rectangular computational domain extends from $[x_a, x_b] \times [y_a, y_b] = [-2.5, 15.0] \times [-3.5, 3.5]$ and the inner cylindrical computational domain extends from $[x_{origin}, y_{origin}] \times [radius_{inner}, radius_{outer}] = [0.0, 0.0] \times [0.5, 1.0]$. The initial conditions and boundary conditions are the same as for the previous cases.

As mentioned before, this simulation will be used as a benchmarking computation in order to compare the performance of different direct and iterative solution methods. Here, we also compare the convergence behavior of various fill-in level k in ILU(k) preconditioning and others well known preconditioners such as additive Schwarz (AS), block Jacobi (BJ) and successive over relaxation (SOR) for different Krylov iterative solvers (see [11, 12, 45, 59, 158] for a detailed discussion on Krylov subspace methods and preconditioners).

Number of grids	2
$\mathbb{G}_1$ dimensions (background grid)	350×140
$\mathbb{G}_2$ dimensions (annular grid)	140×80
Total number of grids points ($\mathbb{G}_1 \cup G_2$)	60200
$\mathbb{G}_1 \cup \mathbb{G}_2$, including ghost points	114752
Total number of unused points	1120
Total number of interpolation points	295
Position of the first node normal to the cylinder wall	$0.0001 \times diameter_{cyl}$

Table 6.5: *Summary of the overlapping grid system used for the benchmarking computations.*

In table 6.6, the performance of various direct and iterative solvers used during theses benchmarking computations is compared. The implementation of the different direct and iterative solvers is based on the PETSc library, which was interfaced with Overture.

The computations carried out and presented in table 6.6, show that the GMRES + ILU(0) solver converges faster than the other methods in terms of CPU time for this specific problem. The timing for each case was carried out by measuring the CPU time from the beginning of the simulation until the moment when the wake instability behind the cylinder has its onset (see figure 6.6). We can also see that among the different preconditioner used (ILU, AS, BJ and SOR), the ILU(0) preconditioner lead to the fastest convergence closely followed by the AS preconditioner. The BICGSTAB solver with the additive Schwarz (AS) preconditioner, also shows a good convergence performance in terms of CPU time, although it is not as fast as the GMRES + ILU(0). From these results, it is also evident that the use of direct solvers for large sparse matrices becomes time and cost prohibitive; here, the direct solver is almost 15 slower than the fastest GMRES or BICGSTAB method (in terms of total CPU time) and about 5 times slower than the worst of the iterative solvers used for this benchmarking computation. All the computations were executed using a reverse-Cuthill-McKee (RCM) matrix reordering algorithm, which clusters the non-zero terms along the diagonal reducing in this way the bandwidth of the sparse matrix [16].

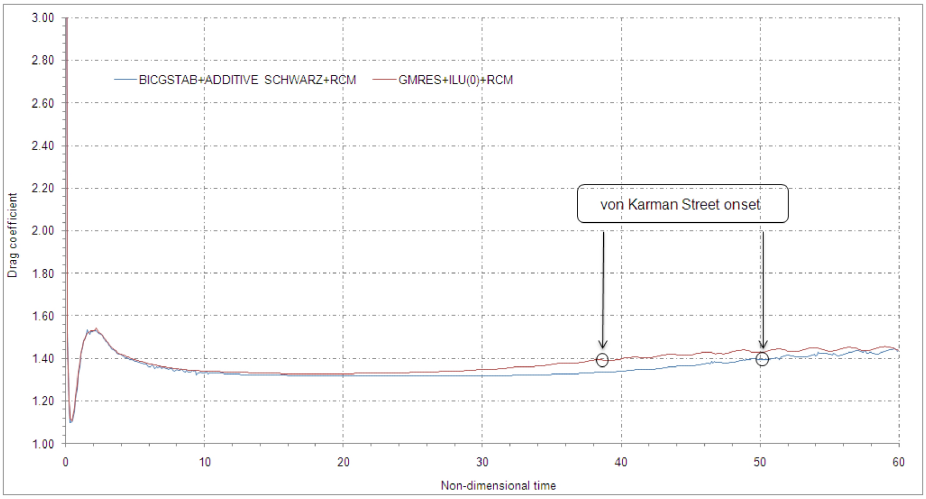

Figure 6.6: *Von Karman street onset (stopping criteria for solver benchmarking).*

These benchmarking computations clearly illustrate a case where the preconditioned GMRES solver performs very well, followed closely in performance by the preconditioned BICGSTAB solver. These two preconditioned iterative solvers will be used as the basic solvers to carry out further benchmarking computations with overlapping moving grids.

6.2.3 Comparison of Fixed Body Solution vs. Moving Body Solution

In this case, the numerical solution of a cylinder moving in quiescent air is compared against the numerical solution of the equivalent case of a flow past a fixed cylinder. Both cases were simulated using a similar computational domain (see figure 6.7), with same grid dimensions and

Case	Method	Iterations	Total CPU Time (seconds)[1]
B1	DIRECT SOLVER (YALE IMPLEMENTATION)	33830	47780
B2	GMRES + ILU(0) + RCM	3000	3180
B3	GMRES + ILU(5) + RCM	5400	5060
B4	GMRES + ILU(10) + RCM	8000	9420
B5	GMRES + ADDITIVE SCHWARZ + RCM	3050	3250
B6	GMRES + SOR + RCM	9000	10500
B7	GMRES NO PRECONDITIONER + RCM	2075	16500
B8	BICGSTAB + ILU(0) + RCM	4000	4220
B9	BICGSTAB + ILU(5) + RCM	8400	8800
B10	BICGSTAB + ADDITIVE SCHWARZ + RCM	3400	3550
B11	BICG + ILU(0) + RCM	3100	5130
B12	BICG + BLOCK JACOBI + RCM	3800	3930
B13	CG + SOR + RCM	6600	6960
B14	CI + BLOCK JACOBI + RCM	Diverged	Diverged
B15	RI + SOR + RCM	Diverged	Diverged
B16	TFQMR + RCM	Diverged	Diverged

GMRES = Generalized Minimal Residual method
BICGSTAB = Stabilized Biconjugate Gradient Squared method
BICG = Biconjugate Gradient Squared method
CG = Conjugate Gradient method
CI = Chebychev iterative method
RI = Richardson iterative method
TFQMR =Transpose Free Quasi Minimal Residual method
[1] INTEL EM64T ×2 @ 2.4 GHz., 4.0 GB RAM., OS LINUX OPENSUSE 64 BITS.

Table 6.6: *Comparison of the performance of different direct and iterative solvers.*

grid spacing. The outer rectangular computational domain extends from $[x_a, x_b] \times [y_a, y_b] = [-3.0, 10.0] \times [-3.0, 3.0]$. For the fixed body simulation, the inner cylindrical computational domain is centred in $[x_{origin}, y_{origin}] = [1.0, 0.0]$, with respective inner and outer radius of $[radius_{inner}, radius_{outer}] = [0.5, 1.0]$. In the case of the moving body, the inner cylindrical computational domain is initially located in $[x_{origin}, y_{origin}] = [8.0, 0.0]$ and then it start to move from right to left with an instantaneous horizontal velocity of $(u, v) = (1.0, 0.0)$. The Reynolds number for both cases, based on the cylinder diameter, the kinematic viscosity and the inflow velocity $(u, v) = (1.0, 0.0)$ or the cylinder translational velocity $(u, v) = (1.0, 0.0)$ is $Re = 500$.

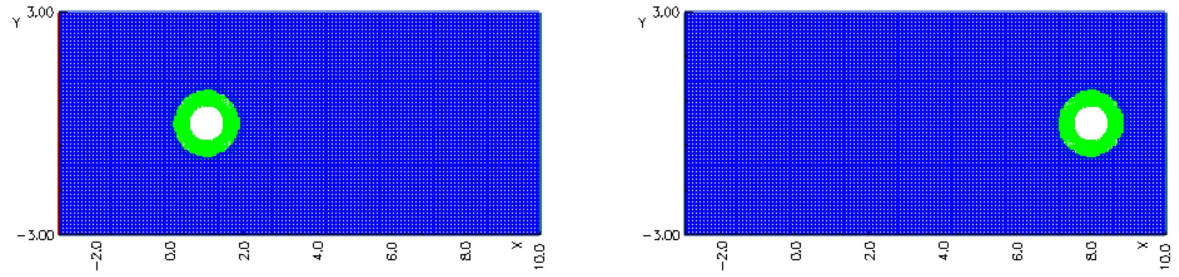

Figure 6.7: *Left: computational domain for the fixed cylinder case. Right: computational domain for the moving cylinder case (the cylinder is in the initial position and it moves from right to left).*

The pressure coefficient c_p, lift coefficient c_l and drag coefficient c_d are computed and compared for a nondimensional time of $t = 7.0$. In table 6.7, the computed c_l and c_d for both cases are shown. In figure 6.8, the c_p profiles for both cases are illustrated. In general, the match between the moving case and the stationary case is very good, which validates the moving overlapping grids framework.

Case	c_d	c_l
Fixed body	1.199795	0.000039
Moving body	1.191358	0.000038

Table 6.7: *Comparison of c_d and c_l coefficients for both cases at $t = 7.0$.*

6.2.4 Comparison to other Numerical and Experimental Results

One useful method of code validation is comparing the results of two or more flow solvers. A reasonably close agreement is encouraging, but is not sufficient to ensure the degree of accuracy, hence deeper validation must be done. Highest encouragement comes when the results from two or more codes closely agree, but they differ significantly in their approaches and algorithms (*i.e.*, finite-volume density-based method versus a finite-difference pressure-based method). Hereafter, we compare the numerical and experimental results obtained by other authors with the results obtained with the proposed flow solver.

Pedro *et al.* [140], numerically studied the propulsive efficiency of a flapping hydrofoil at a Reynolds number of 1100. They used a cell-centred pressure-based finite-volume flow solver with

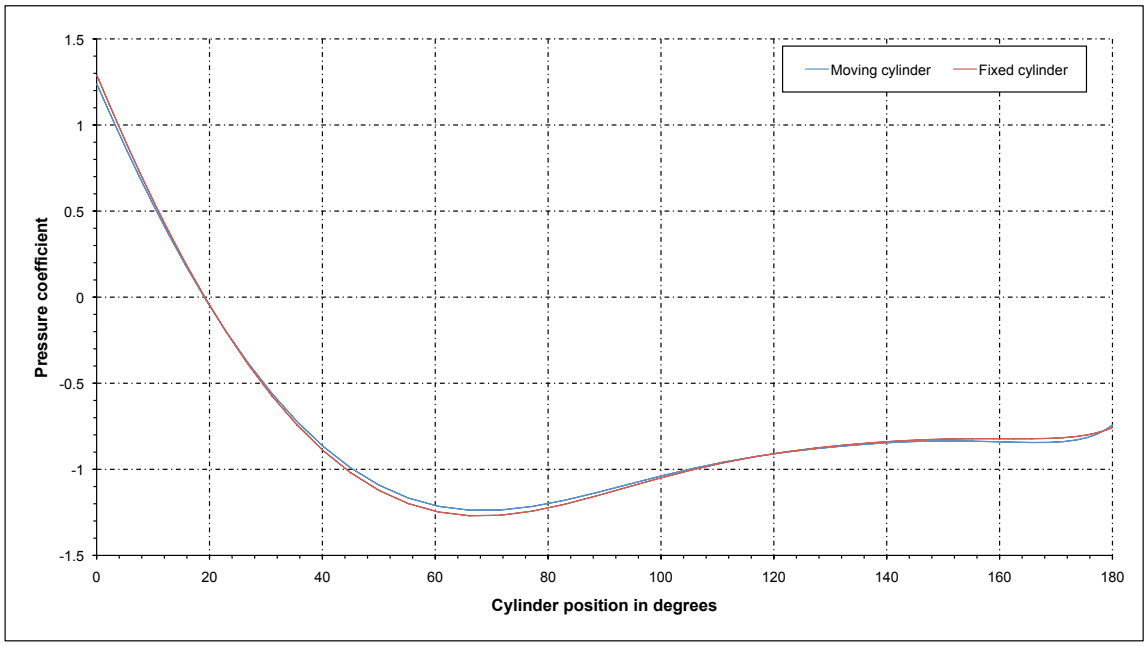

Figure 6.8: *Moving cylinder vs. fixed cylinder, pressure coefficient c_p comparison at a nondimensional time $t = 7.0$.*

Case number	$f_\alpha(Hz)$	k	$\alpha_a(^\circ)$
P1	0.0	0.0	0.0
P2	0.6366	2.0	5.0
P3	1.2732	4.0	5.0
P4	1.9098	6.0	5.0
P5	2.5464	8.0	5.0
P6	3.1830	10.0	5.0
P7	3.8196	12.0	5.0
P8	4.4562	14.0	5.0
P9	5.0928	16.0	5.0
P10	5.7294	18.0	5.0

Table 6.8: *Kinematics parameters for the pitching airfoil case.*

an explicit time-stepping on structured grids. The flow field was discretized by using central differences and to account for the mesh movement, an arbitrary Lagrangian-Eulerian (ALE) formulation was used [83]. In their work, Pedro *et al.* [140] studied airfoils undergoing pure pitching motion and combined heaving-and-pitching motions.

The first scenario considered by Pedro *et al.* [140], was the case of pure pitching. In table 6.8, the parameters governing the pitching motion are shown, where f_α is the pitching frequency, k the reduced frequency and α_a is the maximum pitch angle in degrees. In this case, it is expected that thrust will increase with an increase in either the maximum pitching angle or the frequency of oscillation. The summary of results is presented in tabular format in table 6.9. As it can be

seen in table 6.9, the average thrust coefficient ($\overline{c_t}$) and maximum lift coefficient ($\widehat{c_l}$) for cases P1 to P6, all compare favorably with the results obtained by Pedro *et al.* [140] and less favorably for cases P8, P9 and P10; this difference between both studies may be attributed to dynamic stall phenomena, grid quality issues and the highly irregular nature of the flow at such high oscillating frequencies.

	Pedro *et al.* [140]		Present results	
Case number	$\overline{c_t}$	$\widehat{c_l}$	$\overline{c_t}$	$\widehat{c_l}$
P1	-0.0581	0.0000	-0.1036	0.0000
P2	-0.1132	0.7107	-0.1280	0.6689
P3	-0.0904	2.3600	-0.1021	2.3389
P4	-0.0168	5.4341	-0.0204	5.2650
P5	0.0964	9.8144	0.0386	9.3430
P6	0.2174	15.5948	0.1779	14.9362
P7	0.4543	23.4162	0.4163	20.9512
P8	0.8624	34.1262	0.7356	29.3190
P9	1.2855	45.5064	1.0937	39.2653
P10	1.7467	57.5248	1.4418	49.3107

Table 6.9: *Average thrust coefficient $\overline{c_t}$ and maximum lift coefficient $\widehat{c_l}$ comparison for the pitching airfoil case.*

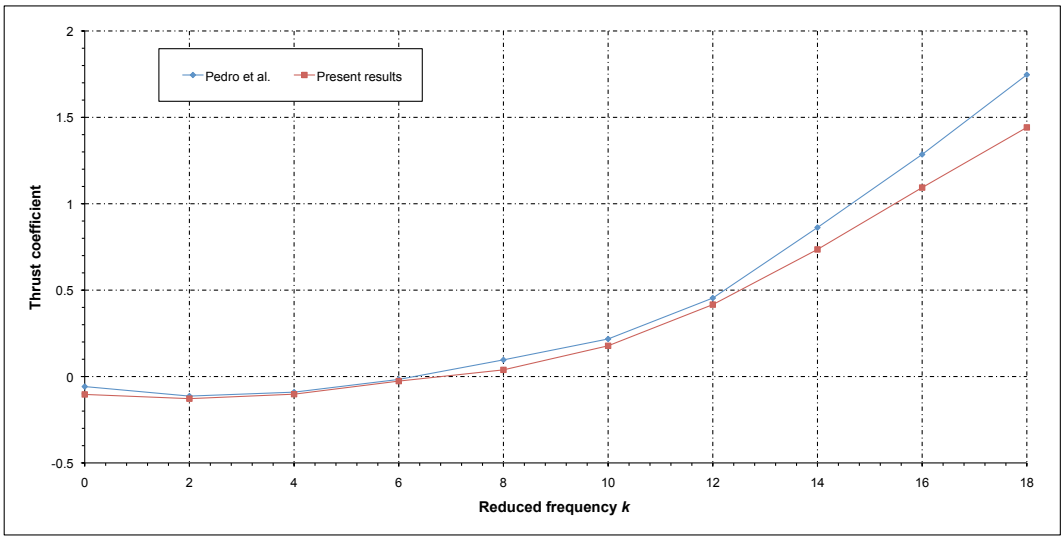

Figure 6.9: *Comparison of average thrust coefficient $\overline{c_t}$ results for the pitching airfoil case (negative values indicate drag production).*

The second scenario considered by Pedro *et al.* [140], is that of a combined heaving-and-pitching motion. The goal is to study the effect that the motion variables have on the values of thrust and lift coefficients. In table 6.10, the parameters governing the airfoil heaving-and-pitching kinematics are presented, where f is the frequency (for both, heaving-and-pitching motion), h_a is the maximum heaving amplitude, α_a is the maximum pitching angle in degrees, φ is the phase angle

between the pitching motion and the heaving motion in degrees, k is the reduced frequency and St is the Strouhal number.

Case number	f(Hz)	h_a	$\alpha_a(^\circ)$	$\varphi(^\circ)$	k	St
F1	0.225	1.0	5.0	90.0	0.7096	0.45
F2	0.225	1.0	10.0	90.0	0.7096	0.45
F3	0.225	1.0	15.0	90.0	0.7096	0.45
F4	0.225	1.0	20.0	90.0	0.7096	0.45
F5	0.225	1.0	25.0	90.0	0.7096	0.45
F6	0.225	1.0	30.0	90.0	0.7096	0.45
F7	0.225	1.0	40.0	90.0	0.7096	0.45
F8	0.255	1.0	50.0	90.0	0.7096	0.45

Table 6.10: *Kinematics parameters for the heaving-and-pitching airfoil case.*

Once again the results are presented in tabular form in table 6.11. In this table, it can be observed that the results obtained compare favorably with those obtained by Pedro *et. al.* [140], except for cases F6, F7 and F8 where the computed values are underpredicted with respect to the values obtained by Pedro *et al.* [140], nevertheless the trend is similar to that of the study carried on by Pedro *et al.* [140] (see figure 6.10). Once again, this differences can be attributed to the dynamic stall phenomena and grid quality issues. In flapping airfoils studies, the understanding of the vortical pattern created by the oscillating airfoil, responsible for the drag production in certain cases, but also for the thrust production in other cases, is a crucial issue. In figure 6.11 and figure 6.12, the comparison between the vorticity contours obtained by Pedro *et al.* [140] and the vorticity contours obtained in the current study for the heaving-and-pitching cases F1 and F6 (see table 6.10) are presented; as it can be seen the qualitative agreement is very satisfactory.

	Pedro *et al.* [140]		Present results	
Case number	$\overline{c_t}$	$\widehat{c_l}$	$\overline{c_t}$	$\widehat{c_l}$
F1	0.4324	8.3333	0.4245	8.0828
F2	0.6511	7.4834	0.6576	7.1699
F3	0.8226	6.6307	0.8360	6.5435
F4	0.9337	5.8176	0.9389	6.1133
F5	1.0046	5.0558	0.9601	5.6080
F6	1.0166	4.3721	0.9771	4.5964
F7	0.7404	3.2278	0.6964	3.6271
F8	0.2953	2.7345	0.2211	3.1785

Table 6.11: *Average thrust coefficient $\overline{c_t}$ and maximum lift coefficient $\widehat{c_l}$ comparison for the heaving-and-pitching airfoil case.*

Wang [207], motivated by the interest in the unsteady aerodynamics of insects flight, devised a computational tool to solve the Navier-Stokes equations around two dimensional moving airfoils. Wang, used a fourth order essentially compact finite difference scheme (EC4) with explicit time-stepping for solving the incompressible Navier-Stokes equations on structured grids; the method

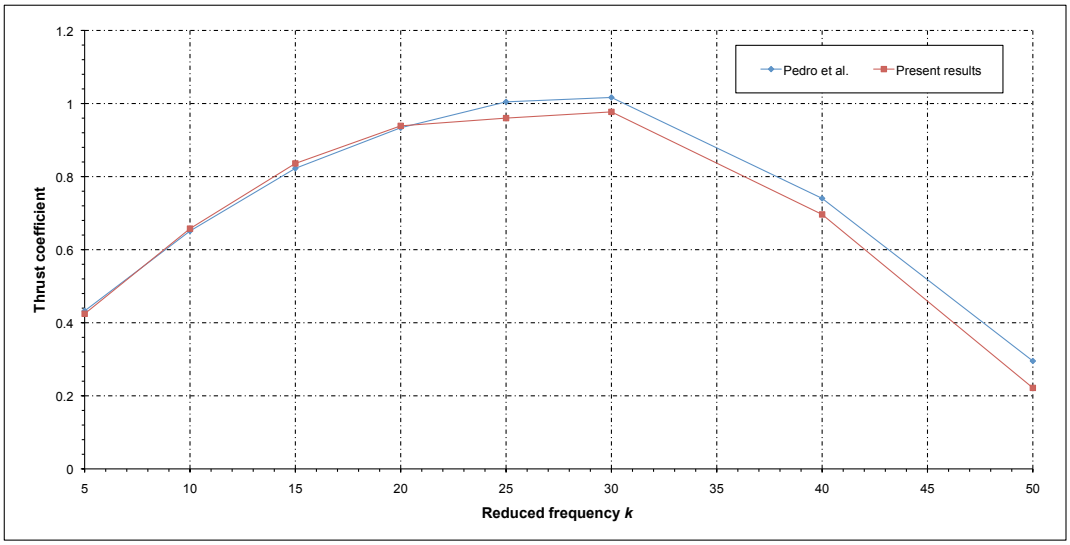

Figure 6.10: *Comparison of average thrust coefficient $\overline{c_t}$ results for the flapping airfoil case (negative values indicate drag production).*

was originally develop by Weinan and Liu [209]. The scheme uses the vorticity-stream function formulation. The governing equations are solved in the inertial frame of reference (hence the flapping motion is implemented by moving the airfoil and the computational domain at the same rate). The focus of this work was to study the frequency selection in forward flapping flight.

k	4.0	2.0	1.0	0.5
$\overline{c_t}$ Wang [207]	0.18	-0.06	-0.24	-0.08
$\overline{c_t}$ Present results	0.1094	-0.0528	-0.2164	-0.0714

Table 6.12: *Average thrust coefficient $\overline{c_t}$ comparison between the present results and the results obtained by Wang [207].*

To validate the code we computed the flow past a heaving ellipse and compare the results to those obtained by Wang [207]. In this numerical experiment the Reynolds number is set to $Re = 1000$, the forward flight velocity to $u = 1.0$, the airfoil chord to $c = 2.0$ and the Strouhal number to St=0.16 (a number based on forward dragonfly flight with $f = 40$ Hz, amplitude $A = 2.0$ cm and forward velocity $u = 5.0$ ms^{-1} [134]), as outlined by Wang [207]. Then we proceed to compute the average thrust coefficient $\overline{c_t}$ for different values of reduced frequency k=0.5, k=1.0, k=2.0 and k=4.0. In table 6.12, we compare the results obtained with those obtained by Wang [207]. As it can be seen, the agreement of the computed values with those obtained by Wang [207] for the cases where $k = 0.5$, $k = 1.0$ and $k = 2.0$ is acceptable, although the agreement is less acceptable for the case where $k = 4.0$, where the computed value is underpredicted. Despite this, the present computations show a similar trend in comparison with the results obtained by Wang [207].

Ramamurti and Sandberg [151], used a finite element incompressible flow solver based on unstructured grids for the study of the unsteady flow past oscillating airfoils. They simulated the viscous flow past a NACA 0012 airfoil at various pitching frequencies and combination of heaving-and-

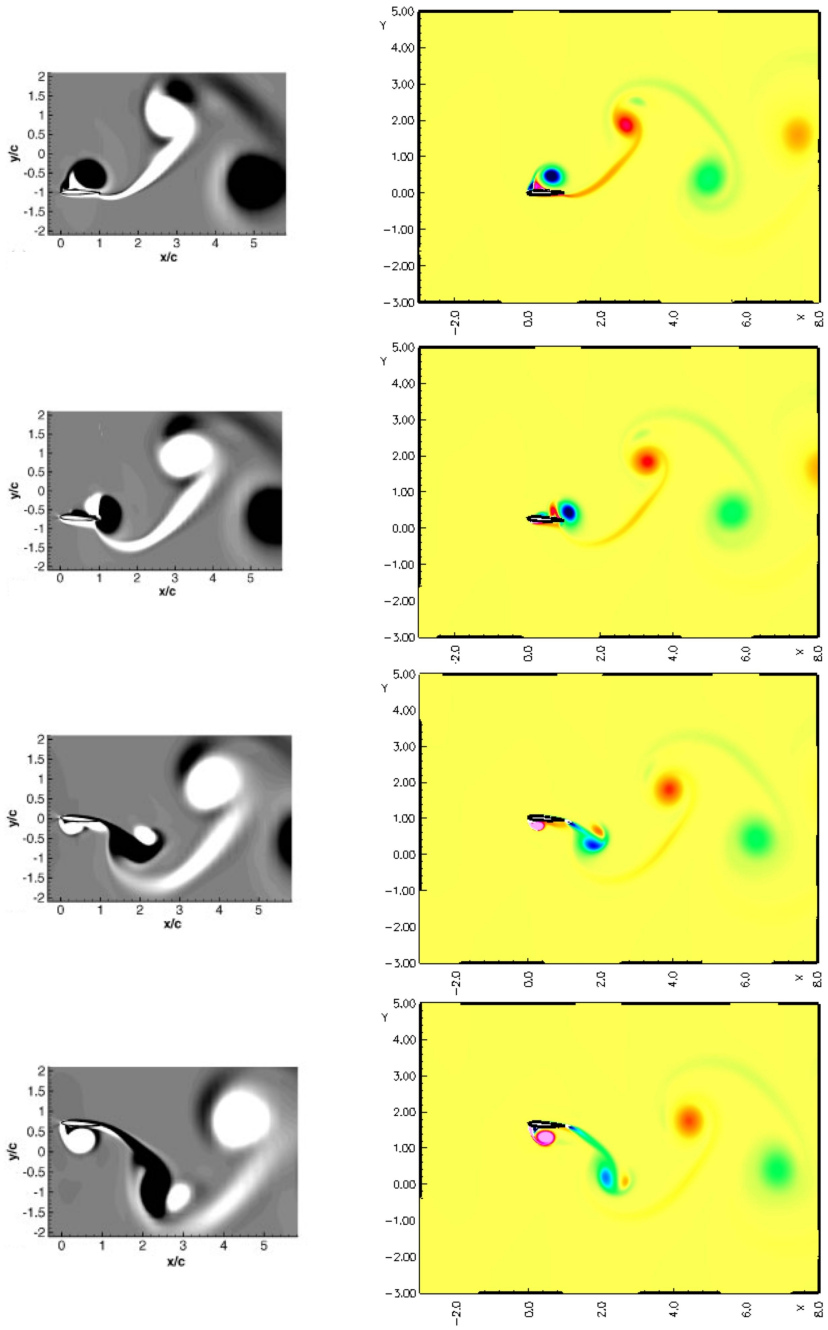

Figure 6.11: *Comparison of vorticity contours for the heaving-and-pitching airfoil case (case F1 in table 6.10). Left column: vorticity contours obtained by Pedro et al. [140]. Right column: present results. The first row is the beginning of one period, the second row is 1/8 of the period, the third row is 1/4 of the period and the last row is 3/8 of the period.*

pitching motion. To tackle the problem of moving bodies, they solved the governing equations using an ALE formulation. To assess the accuracy of the flow solver, they compared their results to those from the experiments carried by Koochesfahani [103]. In figure 6.13, the average thrust coefficient $\overline{c_t}$ values for the pitching airfoil case as described by Koochesfahani [103], are

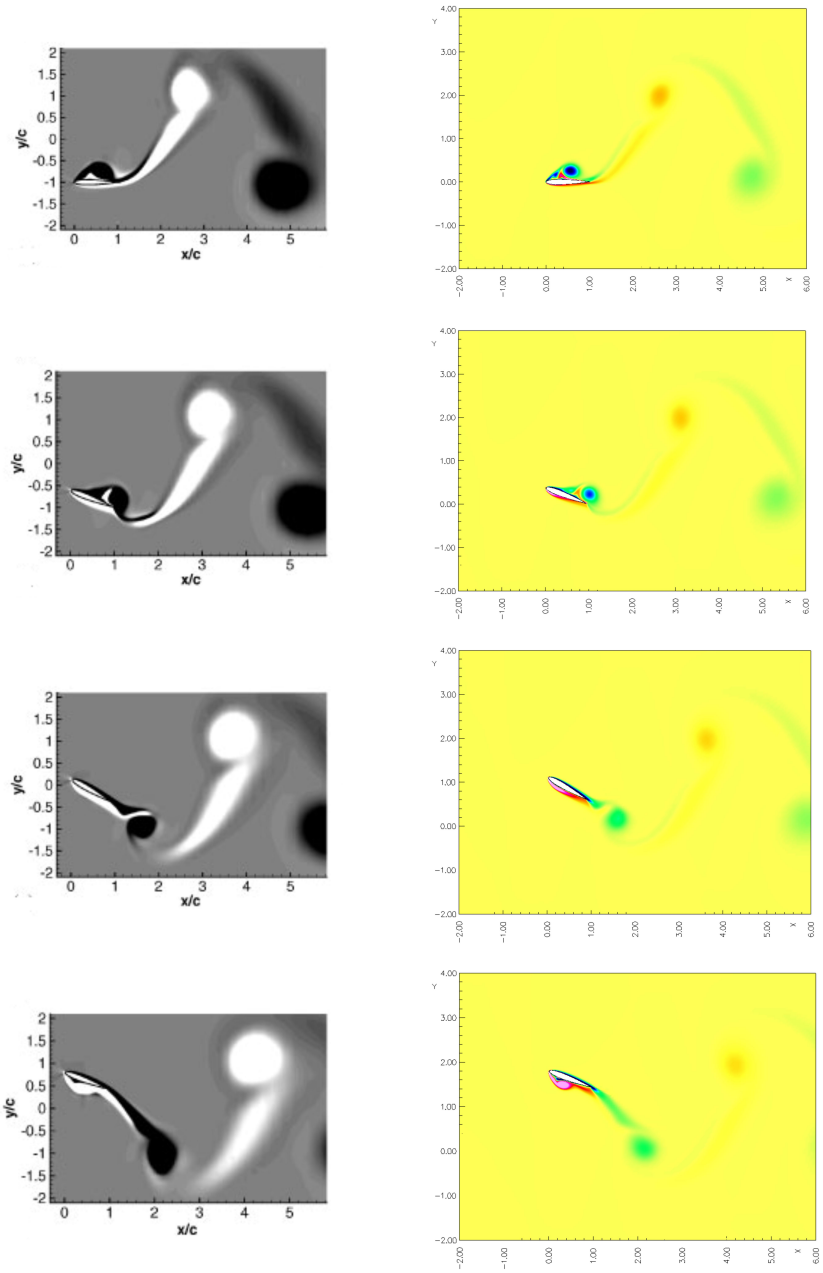

Figure 6.12: *Comparison of vorticity contours for the heaving-and-pitching airfoil case (case F6 in table 6.10). Left column: vorticity contours obtained by Pedro et al. [140]. Right column: present results. The first row is the beginning of one period, the second row is 1/8 of the period, the third row is 1/4 of the period and the last row is 3/8 of the period.*

illustrated. In this figure, the results obtained by Ramurti and Sandberg [151], Koochesfahani [103] and the current results are shown. As it can be seen in figure 6.13, the agreement between the numerical results obtained by Ramamurti and Sandberg [151] and the current results is acceptable and any difference in the solution can be attributed to the discretization scheme and

the grid resolution. In figure 6.13, the agreement between the numerical results (Ramamurti and Sandberg [151] and the current results) and the experimental results (Koochesfahani [103]) for values of reduced frequency above $k = 4$ is discouraging. The difference in these prediction, may be attributed to the method and experimental setup used by Koochesfahani [103] when measuring the forces.

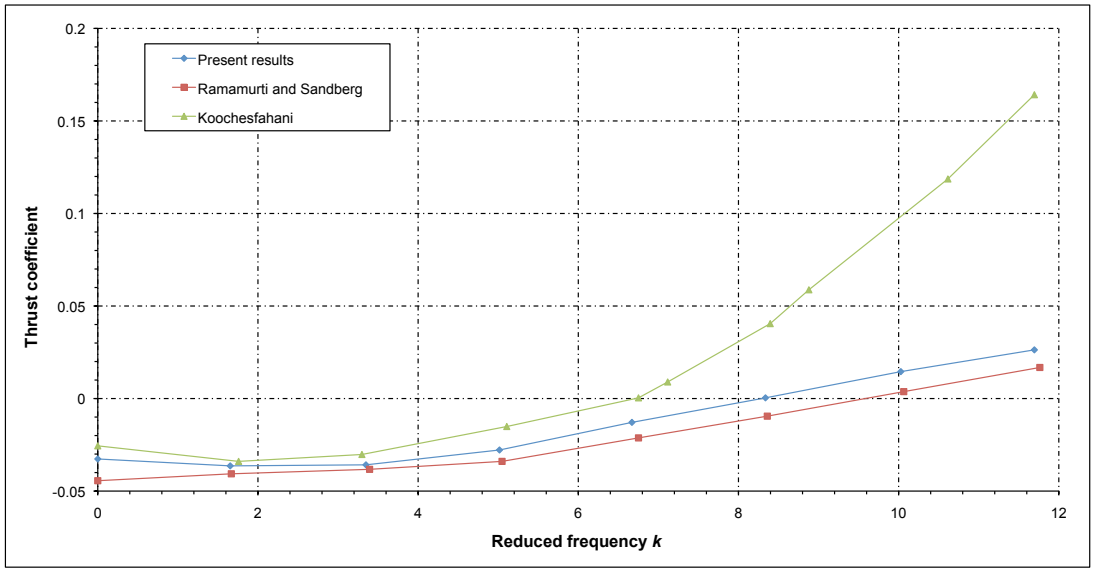

Figure 6.13: *Comparison of variation of thrust coefficient with reduced frequency (negative values indicate drag production).*

In his experimental investigation, Koochesfahani [103] computed the drag or thrust generated by the oscillating airfoil by measuring the momentum deficit or surplus downstream of the body. Usually the assumptions are made that at the cross-section where velocities are measured the flow is parallel, the pressure is freestream and the time-fluctuating quantities are small. Hence, the thrust can be computed from the following integral

$$T = \rho_\infty \int_{-\infty}^{+\infty} u\,(y)\,[u\,(y) - U_\infty]\,dy$$

If the velocity measurements are made sufficiently far downstream of the airfoil, such that the wake eddies are essentially diffused, then this method yields reasonable results, but if the measurements are made in a region where the eddies are still coherent, then the assumptions will not hold. It is not clear from Koochesfahani's article how far downstream the measurements were made for the thrust calculation, but velocity profiles presented in his article were made at one chord-length downstream of the trailing edge and at that distance the eddies are still present and coherent. Similar discrepancies with this experiment have been also documented by Jones and Platzer [97] and Liu and Kawachi [113].

Ramamurti and Sandberg [151], further validate their code by comparing their numerical results with the experimental results obtained by Anderson *et al.* [7]. The first case that was selected by them was that of a heaving-and-pitching airfoil at Reynolds number $Re = 1100$, Strouhal number

$St = 0.3$, maximum heaving amplitude $h_a/c = 1.0$, maximum pitching angle $\alpha_a = 15°$ and phase angle $\varphi = 90°$. In table 6.13, a comparison between the present results, the numerical results presented by Ramamurti and Sandberg [151] and the experimental results obtained by Anderson *et al.* [7] is shown. As it can be seen the agreement between both numerical results results is very satisfactory, even despite the fact that in both computations the results are overpredicted in comparison to the experimental results.

$\overline{c_t}^1$	$\overline{c_t}^2$	$\overline{c_t}^3$
1.3045	1.35	1.20
[1] Present results [2] Ramamurti and Sandberg [151] [3] Anderson *et al.* [7]		

Table 6.13: *Comparison of average thrust coefficient* $\overline{c_t}$.

Guglielmini and Blondeaux [62], computed the dynamics of the vortex structures generated by a foil in steady forward motion, plus a combination of harmonic heaving and pitching oscillations, by means of the numerical solution of the vorticity stream function equations. In their work, they compare their numerical solution with the experimental results obtained by Anderson *et al.* [7]. Qualitatively, Guglielmini and Blondeaux [62] found that the numerical vorticity field was in good agreement with the experimental vorticity visualizations obtained by Anderson *et al.* [7]. From the quantitative point of view, Guglielmini and Blondeaux [62] found that the forces measured experimentally were not in good agreement with the forces computed numerically and they attributed the discrepancy between the numerical and experimental values of c_t to the inaccurate procedure used by Anderson *et al.* [7] to evaluate c_t and to the two-dimensional approximation used by the numerical simulations.

In table 6.14., the average thrust coefficient $\overline{c_t}$ values for a heaving-and-pitching airfoil obtained by Guglielmini and Blondeaux [62] (numerical results), Anderson *et al.* [7] (experimental results) and in the present dissertation are shown. The experiments were conducted at Reynolds number $Re = 1100$ and Strouhal number $St = 0.32$. The maximum heaving amplitude h_a/c, maximum pitching angle α_a (in degrees) and phase angle φ (in degrees), are shown in table 6.14. As it can be seen, the agreement between the present results and the experimental results is acceptable. The difference between the present results and the results obtained by Guglielmini and Blondeaux [62], can be attributed to grid quality issues and some approximations of the flapping parameters done by them.

As a concluding remark, it can be said that from the qualitative and quantitative point of view, the agreement between all the numerical studies is quite satisfactory and any difference between these studies and the present results can be attributed to the discretization method, solution method, problem setup (initial conditions, boundary conditions, motion kinematics parameters and so on) and grid quality issues.

Case number	St	h_a/c	α_a	φ	$\overline{c_t}^1$	$\overline{c_t}^2$	$\overline{c_t}^3$
G1	0.32	0.75	30	90	0.6770	0.48	0.70
G2	0.32	0.75	30	75	0.4528	0.54	0.34
G3	0.32	0.75	30	105	0.7213	0.49	0.77
[1] Present results [2] Guglielmini and Blondeaux [62] [3] Anderson *et al.* [7]							

Table 6.14: *Comparison of average thrust coefficient $\overline{c_t}$.*

6.2.5 Comparison of Sequential Vs. Parallel Computations

Several simulations were performed in order to determine the correctness of the MPI[3] parallelization of the code on distributed memory parallel computers, by comparing the simulations performed in the parallel environment with the one performed in the serial environment. These simulations were also used to determine the relative speedup gained by the code parallelization. It is important to mention that the distributed memory parallel simulations were only performed for fixed bodies as currently the capability to simulate moving bodies in parallel is not fully implemented and still needs to be debugged and validated. Here, the unsteady flow past a cylinder at $Re = 400$ (as in section 2.2) is simulated in a distributed memory parallel computer (see table 6.21). The only difference with the serial environment case is that in this case, the grid (see table 6.5) is refined by a factor of 2.

Case	**Number of processors**	$\overline{c_d}$ $(100 < t < 120)$	**CPU Time (seconds)**	**Speedup**
P1	1[1]	1.7033	57560	-
P2	1[2]	1.7033	68670	0.83
P3	2	1.7033	25900	2.21
P4	4	1.7033	14480	3.97
P5	8	1.7033	7572	7.60
P6	16	1.7033	4255	13.52
P7	32	1.7033	3120	18.44
[1] Serial environment [2] Parallel environment				

Table 6.15: *Parallel computations benchmarking results.*

In table 6.15, average drag coefficient $\overline{c_d}$ values for various parallel simulations with different number of processors are presented, it can be seen that there is no difference between any of the benchmarking cases, indicating that the parallel implementation generates identical results for each one of the parallelization levels and the serial implementation. In figure 6.14 the vorticity magnitude contours around the cylinder are displayed, again it can be seen that there is no discernible difference between the solutions, giving confidence in the parallel code implementation.

Figure 6.15 shows the speedup factor gained by using the parallel code. For the parallel envi-

[3] http://www-unix.mcs.anl.gov/mpi/

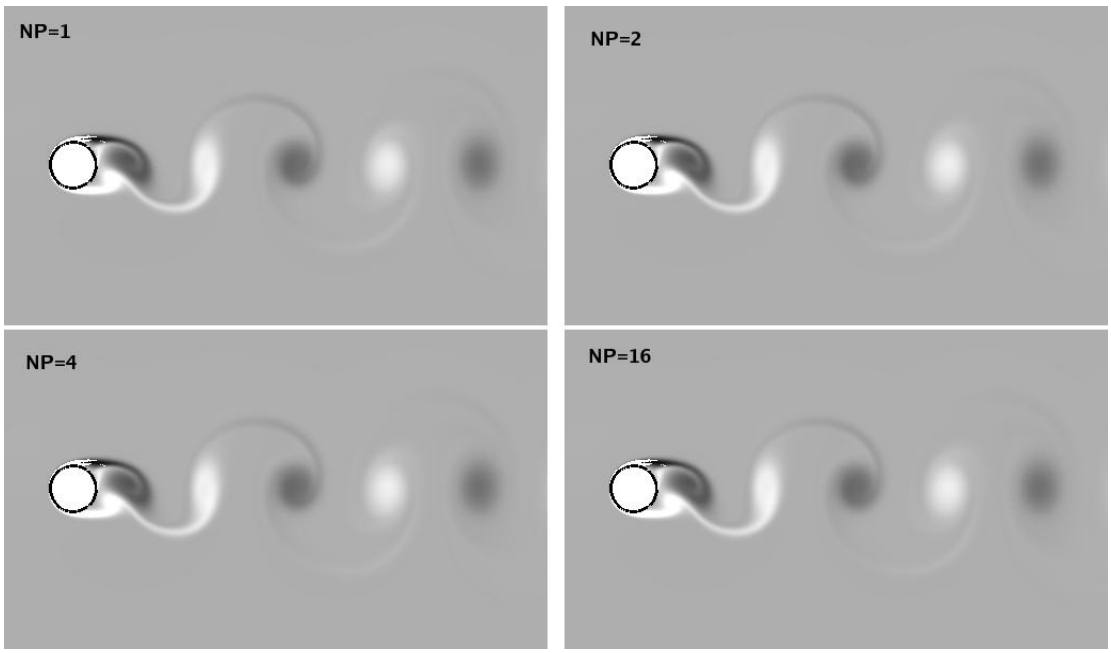

Figure 6.14: *Comparison of vorticity contours between the serial case and the parallel case for a nondimensional time $t = 100$. In the figure, NP stands for number of processors.*

ronment case using a single processor, the speedup factor (or in this case the speedown factor) is about 0.83, indicating that the parallelization added approximately 17% overhead to the serial code. A flattening of the curve in figure 6.15 with increasing processors would indicate that the maximum theoretical speedup has been reached, according with Amdahl's law [35]. At this point, inter-processor communication time will dominate over computation time.

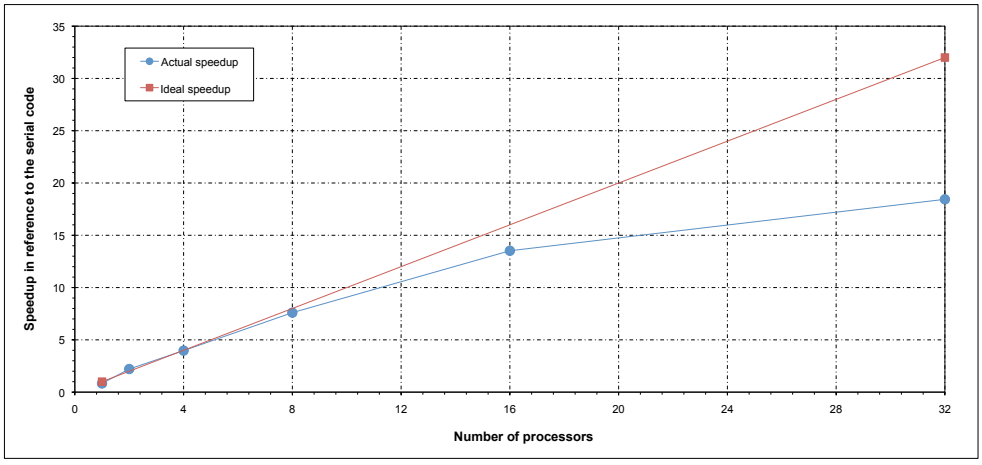

Figure 6.15: *Parallel speed up.*

Despite the fact that the distributed memory parallel implementation of the code still can not be used for moving body simulations, it was effectively used to obtain solutions in fine grids, which

were used later as initial conditions for the moving problem in coarser grids. Nevertheless, certain level of parallelization was reached when running moving body simulations by using OpenMP[4] on shared memory parallel computers, but we were limited by the number of threads (processors) and physical memory available.

6.3 Grid Refinement Study

A consistent numerical analysis will provide a solution which approaches the actual results as the grid resolution gets closer to zero. Thus, the discretized equations will approach the solution of the continuum equations. One significant issue in numerical computations is what level of grid resolution is appropriate. This is a function of the flow conditions, type of analysis, geometry, numerical methods, computational resources and other variables. One is often left to start with a grid resolution and then conduct a series of grid refinements to assess the effect of grid resolution, this is known as a grid refinement study or grid dependency study. In general, a grid refinement study, is a method used for determining the ordered discretization error in numerical simulations and involves performing the simulations on two or more successively finer grids. The method results in an error band on the computational solution which indicates the possible difference between the discrete and continuum value.

Roache [154] suggests a grid convergence index (GCI) to provide a consistent manner in reporting the results of grid refinement studies and perhaps provide an error band on the grid convergence of the solution. The GCI can be computed using two levels of grid; however, three levels are recommended in order to better estimate the order of convergence and to check that the solutions are within the asymptotic range of convergence. The basic idea behind the GCI is to approximately relate the results from any grid refinement study to the expected results from a grid doubling using a second-order method. The GCI is based upon a grid refinement error estimator derived from the theory of the generalized Richardson extrapolation. The object is to provide a measure of uncertainty of the grid convergence. The GCI is a measure of the percentage the computed value is away from the value of the asymptotic solution. It indicates an error band on how far the solution is from the asymptotic range. It also shows how much the solution would change with a further refinement of the grid. A small value of GCI indicates that the computation is well within the asymptotic range.

In practice, wherein the exact solution is not known, we perform at least three grid solutions and calculate two GCI, from the fine grid to the intermediate grid (GCI_{12}) and from the intermediate grid to the coarse grid (GCI_{23}). Then, the GCI on the fine grid is expressed as

$$\mathrm{GCI}_{fine} = F_s \frac{|\varepsilon|}{r^p - 1} \tag{6.4}$$

where ε is defined as

$$\varepsilon = \frac{f_2 - f_1}{f_1} \tag{6.5}$$

In eq. 6.5, f is the approximate solution or the value of an observed quantity and the sub-index $_1$

[4] http://openmp.org/wp/

represents the solution on the finest grid. In eq. 6.4, F_s is a factor of safety. The factor of safety is recommended to be $F_s = 3.0$ for comparisons of two grids and $F_s = 1.25$ for comparisons over three or more grids [154, 170].

It is important that each grid level yields solutions that are in or close to the asymptotic range of convergence for the computed solution. Then the constancy of

$$\mathrm{GCI}_{23} = r^p \mathrm{GCI}_{12} \tag{6.6}$$

indicates that the asymptotic range has been reached. If the desired accuracy level is known and results from the grid resolution study are available, one can then estimate the grid resolution required to obtain the desired level of accuracy,

$$r^* = \left(\frac{\mathrm{GCI}^*}{\mathrm{GCI}_{23}} \right)^{1/p} \tag{6.7}$$

where the superscript * denotes the desired level.

Finally, one must recognize the difference between a numerical result which approaches an asymptotic numerical value and one which approaches the true solution. It is expected that, as the grid is refined and resolution improves, the computed solution will not change much and will approach the asymptotic value (*i.e.*, the true solution). There still may be error between this asymptotic value and the true physical solution to the equations.

The method aforementioned will be used in the following sections in order to conduct the grid refinement study and to determine the most suitable grid in terms of computing time and solution accuracy.

6.3.1 Quantitative Study - Force Measurements

It is a very well known fact that the effect of mesh size is an important factor to consider when assessing the quality of a numerical solution. Here, we conduct a grid refinement or mesh dependency study in order to determine the most suitable grid in terms of computing time and solution accuracy from a quantitative point of view.

To conduct this study, we used the GCI method previously outlined. Here, a heaving airfoil is considered with different grid sizes, layouts and clustering. Several simulations were run at a Reynolds number equal to $Re = 1500$, with a maximum heaving amplitude of $h_a = 0.25$ and Strouhal number equal to $St = 0.5$ (thrust production regime). In each grid, there are typically up to 20 normal points in the direction normal to the airfoil surface (boundary layer area), mesh clustering is also used towards the leading and trailing edge, since we expect the vortices to be generated in these areas. All oscillating flow calculations were started from a fully converged stationary airfoil solution. In our calculations, unsteadiness is observed to disappear typically after 6 cycles of airfoil motion and further calculations show negligible non-periodicity. Thrust coefficient c_t and lift coefficient c_l are computed for each grid and their instantaneous and time average values are compared.

In table 6.16, the parameters for the three grids used for the GCI study are presented. Table 6.16

Grid $\mathbb{G}_g$	Background grid dimensions (BG)	Wake grid dimensions (WG)	Airfoil grid dimensions (AG)	$\frac{h_{\mathrm{AG}_1}}{h_{\mathrm{AG}_g}}$	Position of first node normal to the airfoil surface
$\mathbb{G}_1$	240×120	400×150	600×240	1	$0.000025 \times c$
$\mathbb{G}_2$	200×100	400×150	300×120	2	$0.00005 \times c$
$\mathbb{G}_3$	200×100	360×120	150×60	4	$0.0001 \times c$

Table 6.16: *Description of grids used for the grid refinement study (for force measurements).*

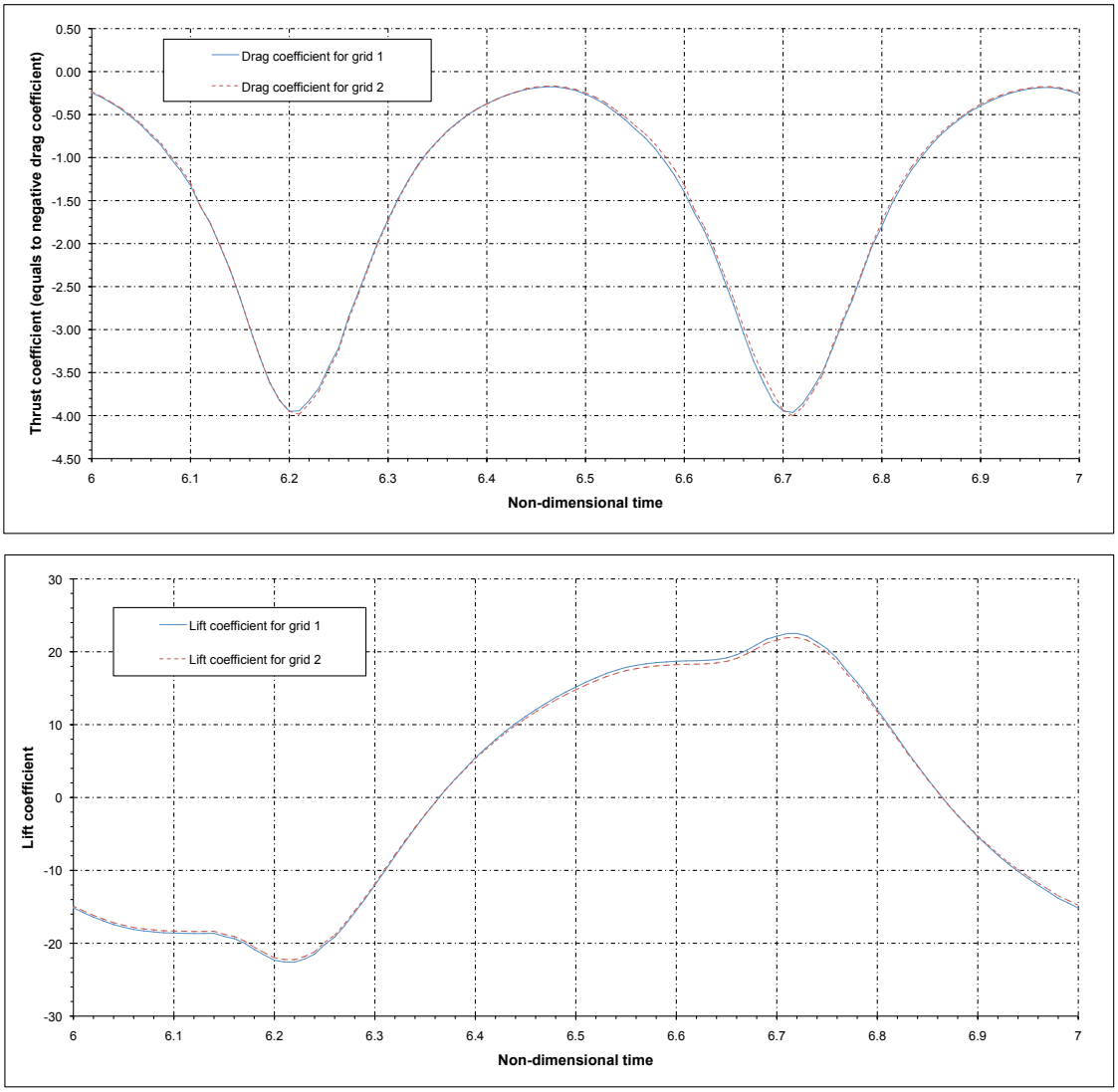

Figure 6.16: *Top: instantaneous drag coefficient c_d (negative values indicate thrust generation). Bottom: instantaneous lift coefficient c_l. Both quantities are shown for an interval equal to $6 < t < 7$.*

shows the grid dimensions, grid spacing refinement ratio (from the finest grid to the coarser grid), position of the first node normal to the airfoil surface and total number of nodes. The airfoil grid (AG) was used as a reference grid to conduct the GCI study and a grid spacing refinement ratio of $r = 2$ was used. Each simulation was checked for acceptable iterative convergence.

In table 6.17, the observed values of the average thrust coefficient $\overline{c_t}$ for each grid are presented; these values are used to calculate the observed order of convergence according to eq. 6.2 and the observed quantity value at zero grid spacing $f_{h=0}$ according to eq. 6.8. In figure 6.16, the instantaneous values of thrust coefficient c_t for $6 < t < 7$ for grids $\mathbb{G}_1$ and $\mathbb{G}_2$ are plotted. Also, the instantaneous values of lift coefficient c_l are shown in figure 6.16; here we can see how c_l varies symmetrically about the zero mean, so that the average vertical force is zero as expected from the symmetric heaving motion. The little bumps in the c_l curve result from the dynamics of the leading-edge vortices.

Grid $\mathbb{G}_g$	$h_{\mathrm{AG}_1}/h_{\mathrm{AG}_g}$	$\overline{c_t}$
$\mathbb{G}_1$	1	1.5267
$\mathbb{G}_2$	2	1.5195
$\mathbb{G}_3$	4	1.4905
Order of convergence p		2.009986
$f_{h=0}$		1 .529078

Table 6.17: *Observed values of the average thrust coefficient $\overline{c_t}$ for each grid. The observed order of convergence and the equivalent zero grid spacing values are also shown.*

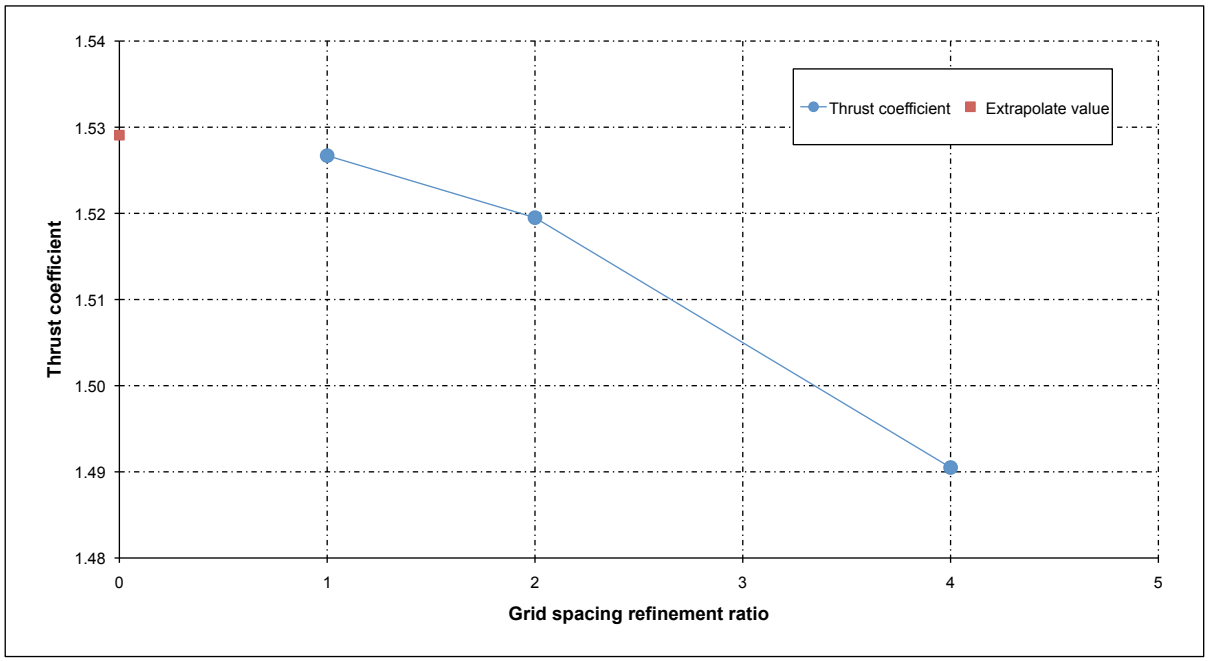

Figure 6.17: *Plot of observed quantity values (average thrust coefficient $\overline{c_t}$) for each grid. The equivalent zero grid spacing value is also plotted.*

We now apply Richardson's extrapolation

$$f_{h=0} \approx f_1 + \frac{f_1 - f_2}{r^p - 1} \tag{6.8}$$

using the two finest grids in order to obtain an estimate of the value of the observed quantities at an equivalent zero grid spacing ($f_{h=0}$) [154, 170]. Richardson extrapolation, eq. 6.8, will provide a fourth-order estimate of $f_{h=0}$ if f_1 and f_2 are computed using second order methods, otherwise it gives a third-order estimate [154, 170]. In figure 6.17, the observed values and $f_{h=0}$ are plotted.

The GCI for the fine grid solution was computed by means of eq. 6.4, here we set $Fs = 1.25$ as suggested by Roache [154] and Slater and Dudek [170]. Hereafter we proceed to calculate the

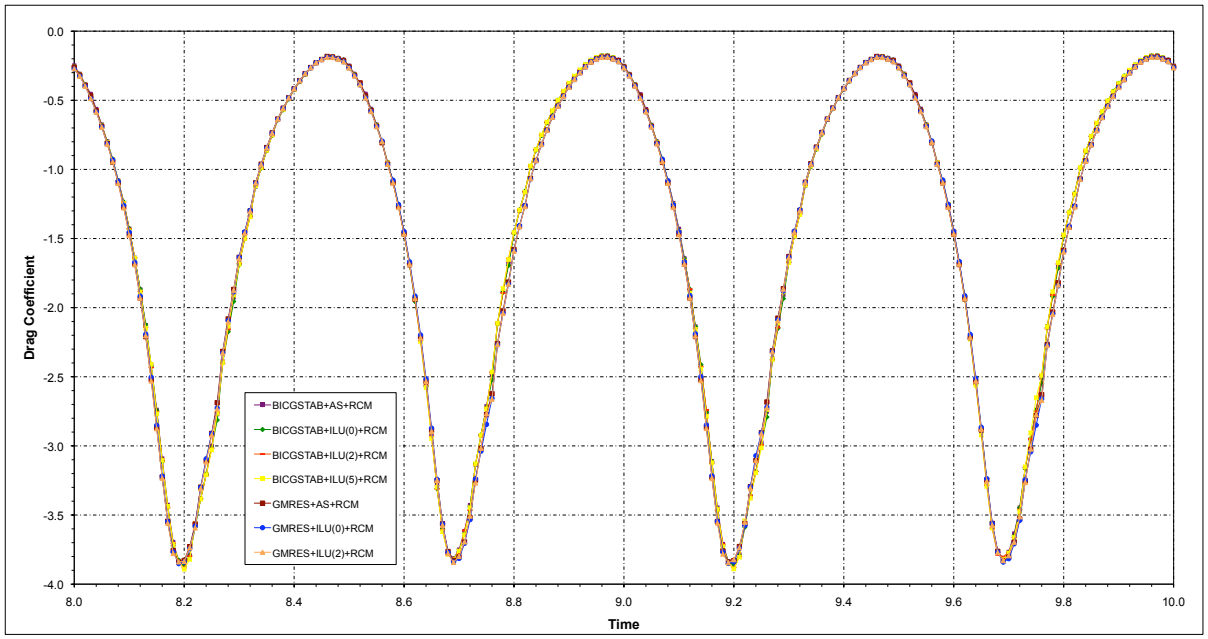

Figure 6.18: *Instantaneous drag coefficient c_d iterative convergence comparison for the heaving airfoil benchmarking case (negative values indicate thrust production).*

GCI_{fine} for grids 1 and 2 and grids 2 and 3 as follows

$$\text{GCI}_{12} = 1.25 \times \frac{(1.5195 - 1.5267)}{1.5267} \times \frac{1}{2^{(2.009986)} - 1} \times 100\% = 0.194699\%$$
$$\text{GCI}_{23} = 1.25 \times \frac{(1.4905 - 1.5195)}{1.5195} \times \frac{1}{2^{(2.009986)} - 1} \times 100\% = 0.787922\%$$

By using eq. 6.6, we now proceed to check that the solutions are in the asymptotic range of convergence. For the three grids we have

$$\frac{0.787922}{0.194699 \times 2^{2.009986}} = 1.004739$$

which is approximately one and indicates that the solution for the three grids are well within the asymptotic range of convergence.

Hereafter, we continue the solver benchmarking study. From the benchmarking case of an unsteady flow past a cylinder (see section 2.2), we found that the preconditioned GMRES solver, followed by the preconditioned BICGSTAB solver showed the best performance (see table 6.6). Here, we proceed to test both solvers, but for the case of a moving body. The case is exactly the same as the one used for the grid refinement study previously carried and the grid used for this benchmarking study is $\mathbb{G}_3$ (see table 6.16). In table 6.18, the results for the benchmarking computations are shown. As it can be seen, cases B3 and B7 show a small difference in terms of CPU time, however, the GMRES method with AS preconditioner seem to be the most promising solution method; hence, it will be used as the basic solution method for all the computations that will be carried on in the following chapters. In figure 6.18, the iterative convergence of the

Case	Method	Iterations	Total CPU Time (seconds)
B1	GMRES + ILU(0) + RCM	37120	22930
B2	GMRES + ILU(2) + RCM	37098	26960
B3	GMRES + ADDITIVE SCHWARZ + RCM	37096	19180
B4	BICGSTAB + ILU(0) + RCM	36742	22430
B5	BICGSTAB + ILU(2) + RCM	36780	22330
B6	BICGSTAB + ILU(5) + RCM	36754	32070
B7	BICGSTAB + ADDITIVE SCHWARZ + RCM	36762	19770

Table 6.18: *Comparison of the performance of different direct and iterative solvers.*

instantaneous drag coefficient c_d for $8 < t < 10$ is shown. In this figure, each case shown in table 6.18 is plotted. As it can be seen, the curves almost exactly collapse into one another, indicating this that each method is converging to the same solution.

6.3.2 Qualitative Validation - Wake Structures Resolution

In the previous section, we performed a quantitative study to determine the best suited grid for force measurement on the airfoil surface in terms of computing time and solution accuracy. In this section, we perform a grid refinement study but from the qualitative point of view (wake structures resolution). The grids used are summarized in table 6.19 and as for the quantitative study, there are up to 20 points in the direction normal to the airfoil surface and mesh clustering is used towards the leading and trailing edge. In figures 6.19 to 6.23, the wake structures resolution for each grid shown in table 6.19 is illustrated. From these results, it can be stated that the wake structures becomes approximately grid independent starting at the grid level case $\mathbb{G}_3$ towards $\mathbb{G}_1$ (see table 6.19).

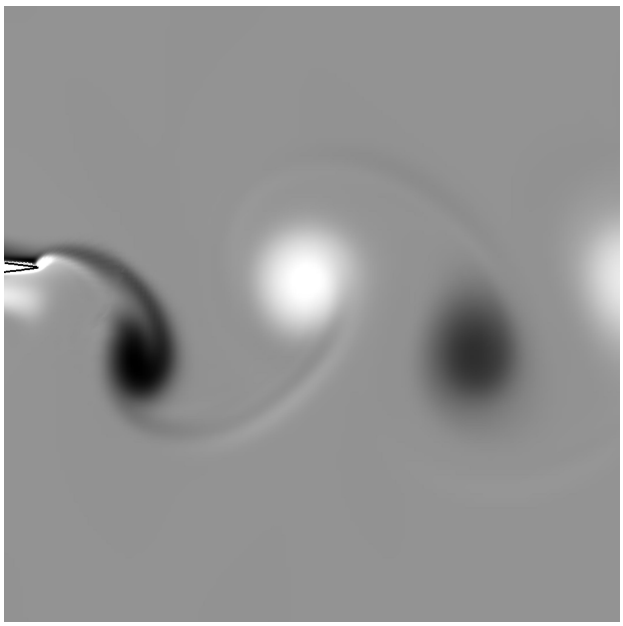

Figure 6.19: *Grid refinement study of the wake structures resolution for the heaving airfoil benchmarking case. Vorticity contours corresponding to $\mathbb{G}_5$ (see table 6.19) are shown.*

6.3.3 Summary of the Quantitative and Qualitative Grid Refinement Study

A quantitative and qualitative grid refinement study has been conducted and hereafter we shortly summarize the results obtained. From the grid convergence index study, it is found that grids $\mathbb{G}_1$ and $\mathbb{G}_2$ (see figure 6.17) are well in the asymptotic range of convergence. Based on the GCI index values found, we could say that the average thrust coefficient $\overline{c_t}$ is estimated to be 1.529078 with an error band of 0.194699% for grid $\mathbb{G}_1$ (see table 6.16) and within an error band of 0.787922% for grid $\mathbb{G}_2$ (see table 6.16), both well within the asymptotic range of convergence. From the qualitative study, it is found that for $\mathbb{G}_3$, $\mathbb{G}_2$ and $\mathbb{G}_1$ (see table 6.19), the wake structure becomes grid independent. In table 6.20, some of the grids used for the qualitative and quantitative

Grid $\mathbb{G}_g$	Background grid dimensions (BG)	Wake grid dimensions (WG)	Airfoil grid dimensions (AG)	$\frac{h_{\mathrm{AG}_1}}{h_{\mathrm{AG}_g}}$	Position of first node normal to the airfoil surface
$\mathbb{G}_1$	240×120	400×150	600×240	1	$0.000025 \times c$
$\mathbb{G}_2$	240×120	400×150	300×120	2	$0.00005 \times c$
$\mathbb{G}_3$	200×100	360×120	300×120	2	$0.00005 \times c$
$\mathbb{G}_4$	200×100	360×120	150×60	4	$0.0001 \times c$
$\mathbb{G}_5$	200×100	360×120	150×60	4	$0.0002 \times c$

Table 6.19: *Description of grids used for the grid refinement study (wake structures resolution).*

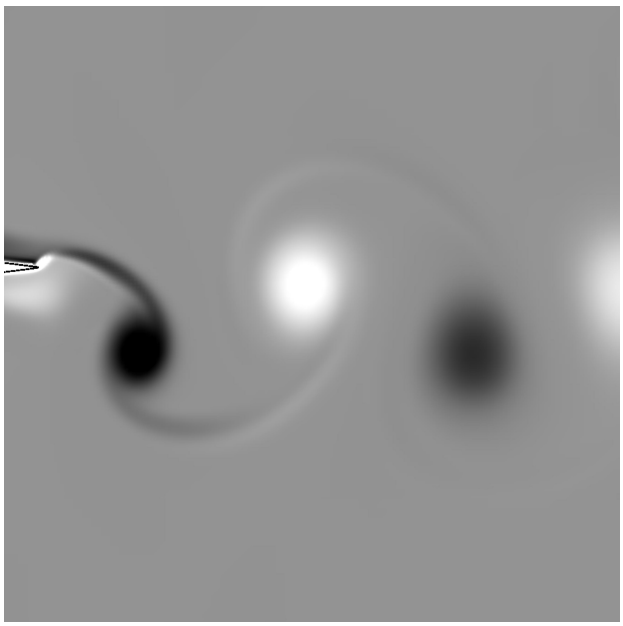

Figure 6.20: *Grid refinement study of the wake structures resolution for the heaving airfoil benchmarking case. Vorticity contours corresponding to* $\mathbb{G}_4$ *(see table 6.19) are shown.*

Figure 6.21: *Grid refinement study of the wake structures resolution for the heaving airfoil benchmarking case. Vorticity contours corresponding to* $\mathbb{G}_3$ *(see table 6.19) are shown.*

grid refinement study are summarized. Grids $\mathbb{G}_1$, $\mathbb{G}_2$, $\mathbb{G}_3$ and $\mathbb{G}_4$ from the quantitative (force measurements) and qualitative (wake structure resolution) standpoint provide grid independent results, but taking into account the computational resources available, CPU time restrictions and solution accuracy, grid $\mathbb{G}_3$ will be used as the base grid to perform all the $2\mathbb{D}$ computations.

Grid $\mathbb{G}_g$	Background grid dimensions (BG)	Wake grid dimensions (WG)	Airfoil grid dimensions (AG)	Position of first node normal to the airfoil surface	Total number of nodes $\mathbb{G}_1 \cup \mathbb{G}_1 \cup \mathbb{G}_1$
$\mathbb{G}_1$	240×120	400×150	600×240	$0.000025 \times c$	232800
$\mathbb{G}_2$	240×120	400×150	300×120	$0.00005 \times c$	124800
$\mathbb{G}_3$	200×100	400×150	300×120	$0.00005 \times c$	116000
$\mathbb{G}_4$	200×100	360×120	300×120	$0.00005 \times c$	99200
$\mathbb{G}_5$	240×120	400×150	150×60	$0.0001 \times c$	97800
$\mathbb{G}_6$	200×100	360×120	150×60	$0.0001 \times c$	72200
$\mathbb{G}_7$	200×100	360×120	150×60	$0.0002 \times c$	72200
$\mathbb{G}_8$	200×100	360×120	150×60	$0.0004 \times c$	72200

Table 6.20: *Description of grids used for the grid refinement study.*

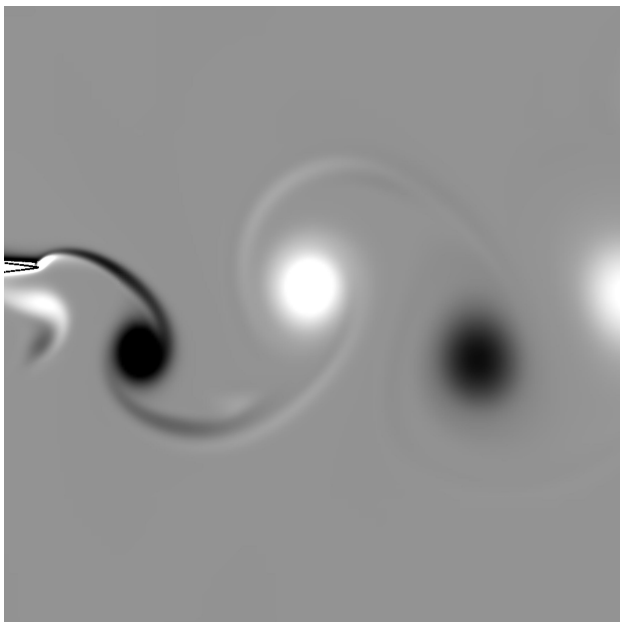

Figure 6.22: *Grid refinement study of the wake structures resolution for the heaving airfoil benchmarking case. Vorticity contours corresponding to* $\mathbb{G}_2$ *(see table 6.19) are shown.*

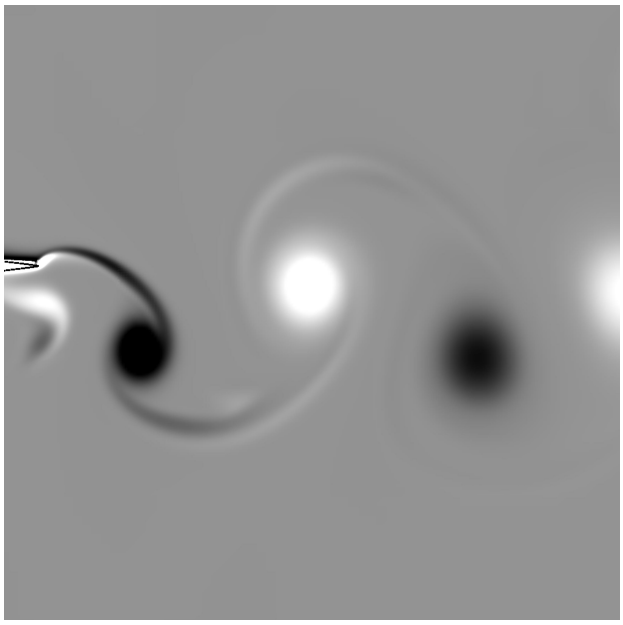

Figure 6.23: *Grid refinement study of the wake structures resolution for the heaving airfoil benchmarking case. Vorticity contours corresponding to* $\mathbb{G}_1$ *(see table 6.19) are shown.*

The overlapping grid system layout used for the qualitative and the quantitative grid refinement study is shown in figure 6.24. In both studies, the background grid (BG) extends from $[x_a, x_b] \times [y_a, y_b] = [-3.0, 9.0] \times [-2.75, 3.25]$. The wake grid (WG) extends from $[x_a, x_b] \times [y_a, y_b] = [-1.0, 7.0] \times [-1.25, 1.75]$. The airfoil grid (AG) which is an hyperbolic grid [75], is marched a distance equal to $0.5 \times c$ from the airfoil surface, with the airfoil leading edge centred at

Benchmarking case description	Computational resources
Unsteady flow past a cylinder (serial environment)	AMD64 ×2 @ 1.6 GHz., 4.0 GB RAM., OS LINUX OPENSUSE 64 BITS.
Heaving airfoil	INTEL EM64T ×2 @ 2.4 GHz., 4.0 GB RAM., OS LINUX OPENSUSE 64 BITS.
Unsteady flow past a cylinder (parallel environment)	NEC Xeon EM64T Cluster with 200 nodes @ 3.2 GHz. 160 nodes @ 1 GB RAM + 40 nodes @ 2 GB RAM. Infiniband node-node interconnection. OS LINUX TAO 64 BITS. http://www.hlrs.de/hw-access/platforms/cacau/

Table 6.21: *Computational resources used in each benchmarking case.*

$[x_{origin}, y_{origin}] = [0.0, 0.0]$ in the initial position and where the airfoil chord c is equal to 1.0. The overlapping grid system layout dimensions are chosen in such a way that the vertical distance to the top and bottom boundaries of the BG and WG, when the airfoil is in the mean position of the flapping period, is equal to $3.0 \times c$ and $1.5 \times c$ respectively. In the case of a bigger or smaller domain, the grid dimensions are scaled in order to keep the same grid spacing as for this domain. The initial conditions used in each heaving airfoil simulation are those of the fully converged solution of the corresponding fixed airfoil case, obtained with a finer grid (with a corresponding overlapping grid system spacing refinement ratio of $r = 2$). The left boundary in figure 6.24, corresponds to an inflow boundary condition with $(u, v) = (1.0, 0.0)$ and the top, bottom and right boundaries are outflow boundaries. The airfoil has no-slip boundary condition. Finally, we mention in table 6.21 the computational resources used to run each benchmarking case.

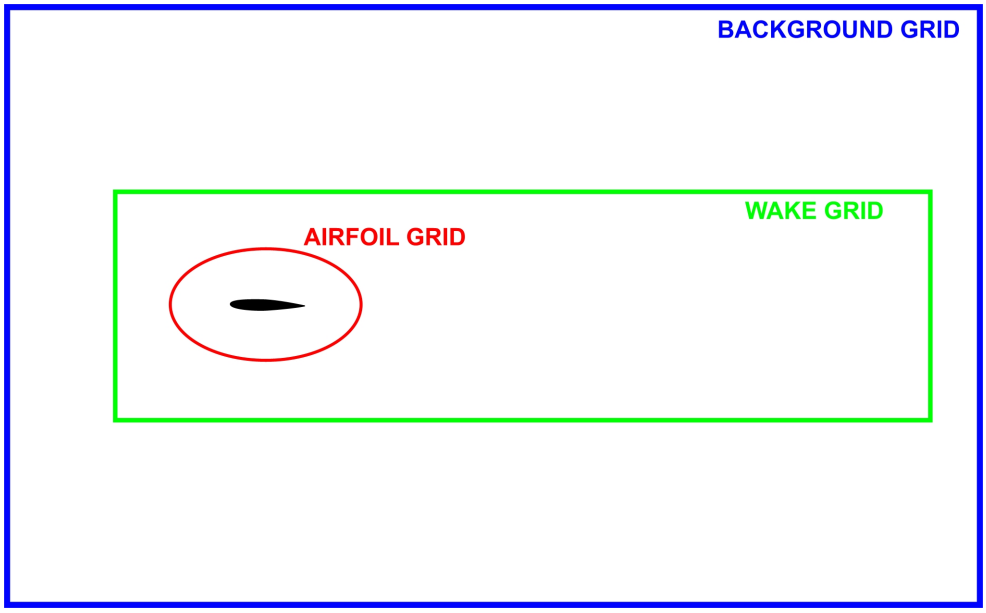

Figure 6.24: *Overlapping grid system layout.*

6.4 Closing Remarks

From the results presented in this chapter, it can be said that from the qualitative and quantitative point of view, the agreement between all the numerical studies is very satisfactory, fact that gives us confidence on the numerical accuracy of the computational tools implemented. Any difference between all the numerical studies carried out in this chapter can be attributed to the differences on the discretization method, solution method, problem setup (initial conditions, boundary conditions, motion kinematics parameters and so on) and grid quality issues.

In this chapter, it was also conducted a qualitative (wake structure resolution) and quantitative (force measurements) grid refinement study in order to determine the grid best suited for the numerical computations to be carried out. The *"optimal"* grid was chosen based on the GCI index, the wake structure resolution, CPU time restrictions and solution accuracy.

Chapter 7

Wake Structures and Aerodynamic Performance of Flapping Airfoils

In the previous chapters, we reviewed the aerodynamics of low Reynolds number flapping flight, where we covered topics such as dynamic stall, vortex shedding, thrust generation due to flapping flight, flapping flight in terms of Reynolds number, Strouhal number and reduced frequency, and flapping flight performance and kinematics. We also presented the method of overlapping grids used to efficiently tackle the problem of moving/deforming bodies and the numerical method used to solve the governing equations on overlapping grids. In this chapter, we present several two dimensional results for heaving and coupled heaving-and-pitching motions. The interest here is to determine the values of flapping frequency and flapping amplitude best suited for flapping flight, in terms of maximum efficiency and thrust production. We also study the influence of airfoil cambering and airfoil flexibility on lift and thrust generation.

7.1 Introduction

As discussed in Chapter 2, Section 2.3, the aerodynamic forces on a body are related to the wake produced by the body. The flow over a stationary airfoil or over an airfoil flapping at a specific frequency and amplitude is constituted by a von Karman vortex street, consisting of alternating rows of clockwise (top) and counter-clockwise (bottom) rotating vortices (see figure 7.1). The time-averaged velocity profile produced by such a configuration is a momentum deficit profile, in that the mean velocity in the wake is lower than the free-stream value. Vortex pairs form mushroom-like structures which are tilted upstream (see figure 7.1). The velocity profile and the upstream-tilted vortex structures are signatures that the body is producing drag.

If the rows of vortices are exchanged so that we have counterclockwise (top) and clockwise (bottom) rotating vortex rows, the configuration is known as a reverse Karman vortex street. The resulting time-averaged velocity profile is now a momentum surfeit (jet) profile, such that the mean velocity is higher than the free-stream. The vortex pairs now form mushroom-like structures that are tilted downstream (see figure 7.2). The momentum surfeit velocity profile and the downstream-tilted vortex structures indicate that the body is producing net thrust.

At the point where thrust due to flapping and the airfoil drag exactly balance, we might expect to see a neutral wake, where the vortices are collinear and mushroom-like vortex pairs are not

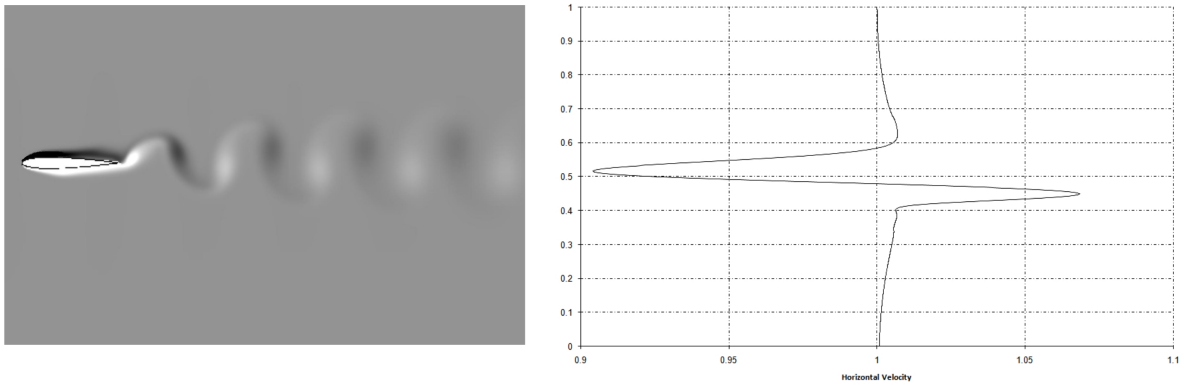

Figure 7.1: *Left: wake structure behind a heaving NACA 0012 airfoil (vorticity contours). Right: horizontal velocity profile measured at a distance equal to 5 times the airfoil chord away from the trailing edge. Flapping parameters:* $St = 0.1, h_a = 0.05, Re = 1100$. *This configuration is indicative of drag production.*

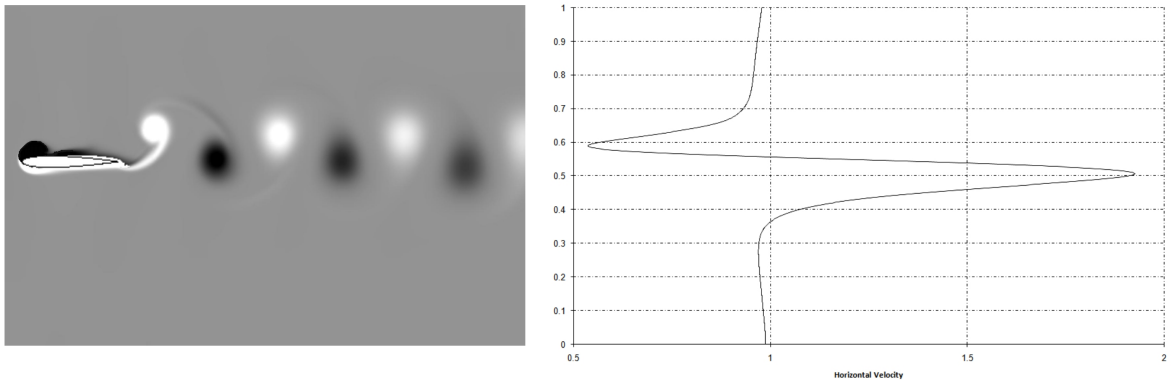

Figure 7.2: *Left: wake structure behind a heaving NACA 0012 airfoil (vorticity contours). Right: horizontal velocity profile measured at a distance equal to 5 times the airfoil chord away from the trailing edge. Flapping parameters:* $St = 0.3, h_a = 0.15, Re = 1100$. *This configuration is indicative of thrust production.*

tilted, as shown in figure 7.3.

Taylor *et al.* [182] performed a study of published wing beat frequencies and amplitudes and cruise speeds, across a range of birds, bats and insects, to determine Strouhal numbers in cruising flight. They found 75% of the 42 species considered to fall within a narrow range of $0.19 < St < 0.41$, with a mean value of $St = 0.29$. Triantafyllou *et al.* [193], provided a graph of measured Strouhal numbers for a range of fishes, sharks and cetaceans, with all falling largely within the Strouhal number range of $0.25 < St < 0.35$. In the results presented by Nudds *et al.* [136], they found that their calculations were consistent with the hypothesis that birds have converged upon a narrow optimum range of St in cruising flight. The best estimates of St for the empirical data given by them fell within the range of $0.2 < St < 0.4$ associated with high propulsive efficiency in other theoretical and experimental studies [152, 193].

Hereafter, we study the dependency of the wake structure and aerodynamic performance on the flapping and geometric parameters such as, flapping frequency, flapping amplitude and airfoil

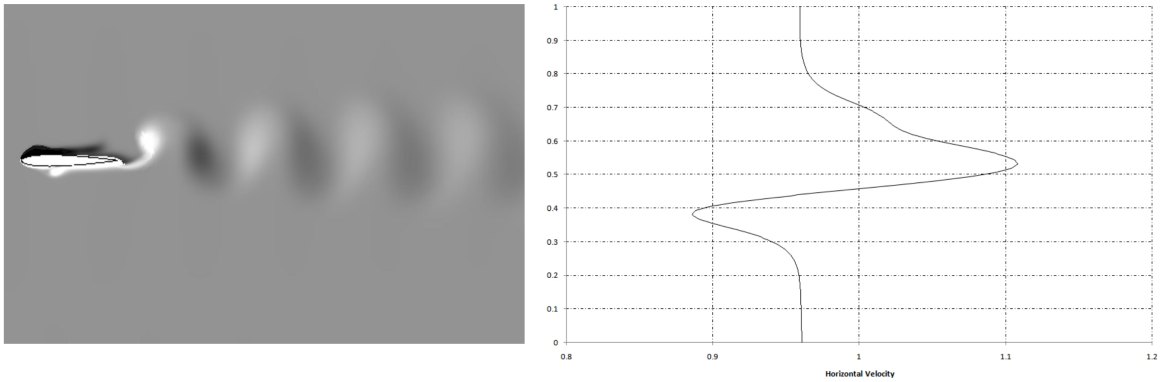

Figure 7.3: *Left: wake structure behind a heaving NACA 0012 airfoil (vorticity contours). Right: horizontal velocity profile measured at a distance equal to 5 times the airfoil chord away from the trailing edge. Flapping parameters:* $St = 0.15, h_a = 0.25, Re = 1100$. *This configuration is indicative of a net balance between thrust production and drag generation (neutral wake).*

geometry (between others), for airfoils undergoing pure heaving motion or coupled heaving-and-pitching motion (flapping motion).

7.2 Heaving Airfoil Wake Signature and Aerodynamic Performance

In order to characterize the wake signature of a heaving airfoil and its dependency on the flapping parameters, we conduct an extensive numerical experiment, where we simulate a NACA 0012 airfoil, undergoing heaving motion as per equation 2.21. The Reynolds number based on the airfoil chord was chosen to be equal to $Re = 1100$. In table 7.1 we present the flapping parameters for this numerical experiment.

Airfoil type	Re	St	h_a	f_h
NACA 0012	1100	$0.1 < St < 1.4$	$0.05 < h_a < 1$	$0.025 < f_h < 10$

Table 7.1: *Flapping parameters for the pure heaving case.*

But before discussing the results of the parametric study, let us first take a look at the heaving motion (eq. 2.21) and at the evolution of lift and thrust coefficients with time for a single case. In figure 7.4 the heaving kinematics is plotted, where one period consists of an up and down stroke. Since the airfoil is symmetric and is oscillating symmetrically about the mean horizontal line, we expect a symmetrical lift evolution (but this does not necessary hold for certain values of h_a and f_h, as presented later). The bumps appearing in the lift coefficient curve corresponds to the dynamics of the leading edge vortex, which will be studied in a later section. The frequency of the thrust coefficient is twice that of the lift coefficient because thrust is generated in both the up and down strokes.

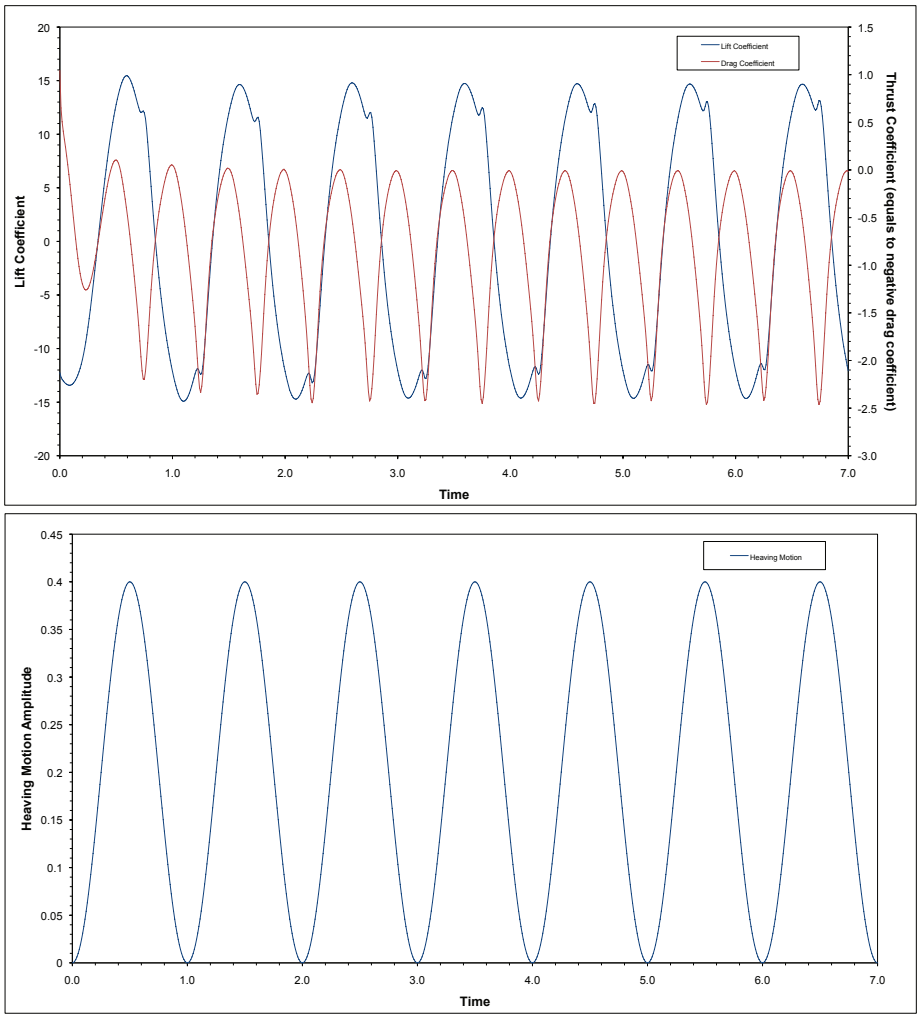

Figure 7.4: *Top figure: time dependent drag and lift coefficients (where negative values of drag coefficient indicate thrust production). Bottom figure: heaving kinematics. Flapping parameters:* $Re = 1100, St = 0.4, h_a = 0.2$ ($\overline{c_t} = 0.8834$, $\overline{c_l} = 0.0098$).

In figure 7.5, we plot the values of average thrust coefficient $\overline{c_t}$ and propulsion efficiency η obtained for Strouhal number values ranging from $0.1 < St < 0.7$, at a fixed heaving frequency $f_h = 1\,(k = 3.14159)$. In this figure, we can clearly identify three regimes: 1) one corresponding to drag production for Strouhal number values approximately $St < 0.18$, 2) a second regime for Strouhal values between $0.18 < St < 0.2$, corresponding to a very narrow transition area where no or very little thrust (or drag) is produced, and 3) a final regime corresponding to thrust generation for Strouhal number values $St > 0.2$, with a maximum propulsive efficiency value obtained at $St = 0.3$.

In the results shown in figure 7.5, the Strouhal number was varied by adjusting the heaving amplitude h_a. But our parameter space (f_h, h_a) is not only limited to one fixed value of f_h, both h_a and f_h can be changed independently, hence, our test matrix becomes quite large. Clearly, the previous experiment does not cover the whole parameter space, but it is a good indicator as far as wake signature characterization is concerned; a more extensive parameter search is however

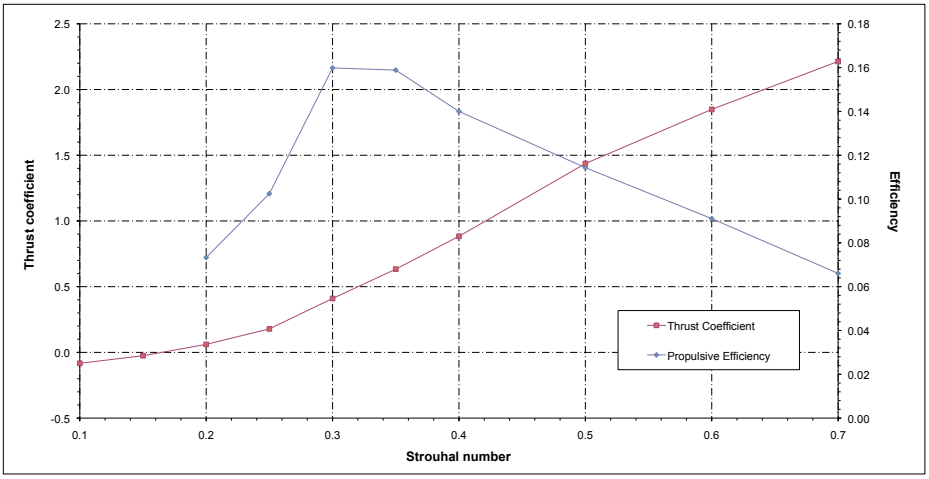

Figure 7.5: *Variation of thrust coefficient and propulsive efficiency with the Strouhal number. Flapping parameters:* $Re = 1100$, $f_h = 1$. *Notice that the propulsive efficiency is only shown for positive thrust.*

necessary to identify the regime(s) with the largest efficiency.

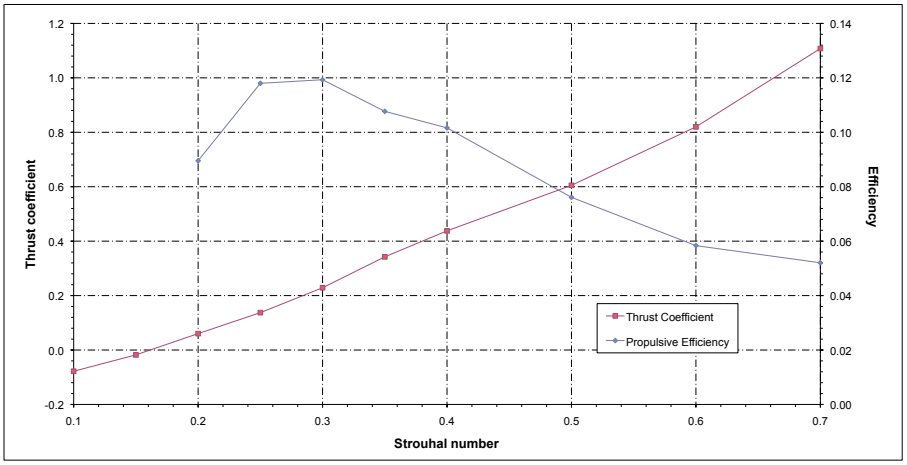

Figure 7.6: *Variation of thrust coefficient and propulsive efficiency with the Strouhal number. Flapping parameters:* $Re = 1100$, $f_h = 0.5$. *Notice that the propulsive efficiency is only shown for positive thrust.*

For purposes of comparison, in figure 7.6, we illustrate a similar plot where the heaving frequency is fixed to $f_h = 0.5\,(k = 1.570795)$ and the Strouhal number varies between $0.1 < St < 0.7$. Again, the maximum efficiency is obtained at $St = 0.3$. In this figure, the drag production regime approximately corresponds to $St < 0.15$ and the thrust production regime to $St > 0.2$, with a very narrow transition area between $0.15 < St < 0.20$, where no or little thrust (or drag) is produced. As it can be evidenced from figures 7.5 and 7.6, the heaving frequency have also a direct influence on whether the airfoil produces thrust or drag and hence on the wake topology.

Let us now fix the Strouhal number value to $St = 0.3$ (corresponding to the maximum efficiency value in figure 7.5) and vary the values of the heaving amplitude h_a and heaving frequency f_h, as

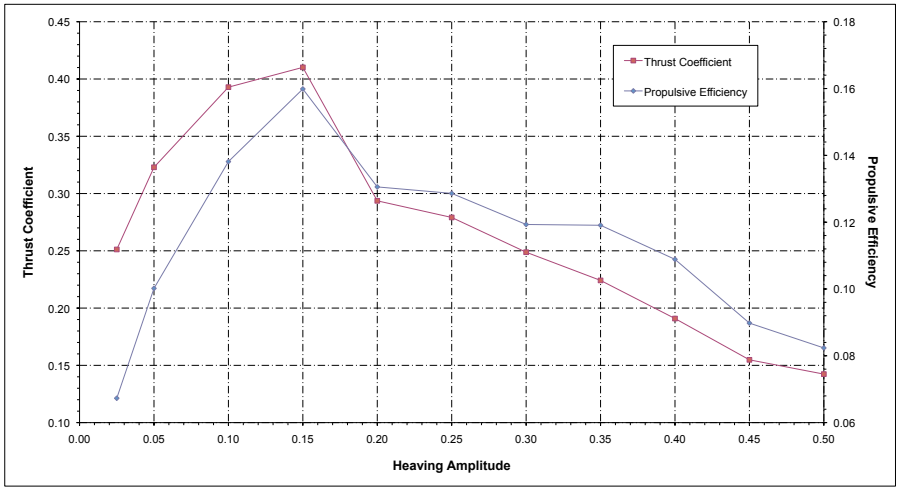

Figure 7.7: *Variation of thrust coefficient and propulsive efficiency with heaving amplitude. Flapping parameters:* $Re = 1100, St = 0.3$.

shown in figure 7.7. In this figure, the maximum efficiency value corresponds to a heaving amplitude value of $h_a = 0.15$ (same value as the case shown in figure 7.5). But this is not necessarily the same for other configurations in parameter space (f_h, h_a), as shown in figure 7.8, where we fix the Strouhal number $St = 0.4$ (thrust production regime) and we change the values of the heaving amplitude h_a and the heaving frequency f_h accordingly. Here, the maximum propulsive efficiency value corresponds to a heaving amplitude value $h_a = 0.30$ and a heaving frequency value $f_h = 0.666667\,(k \approx 2.094)$, which are not the same values for the case presented in figure 7.5.

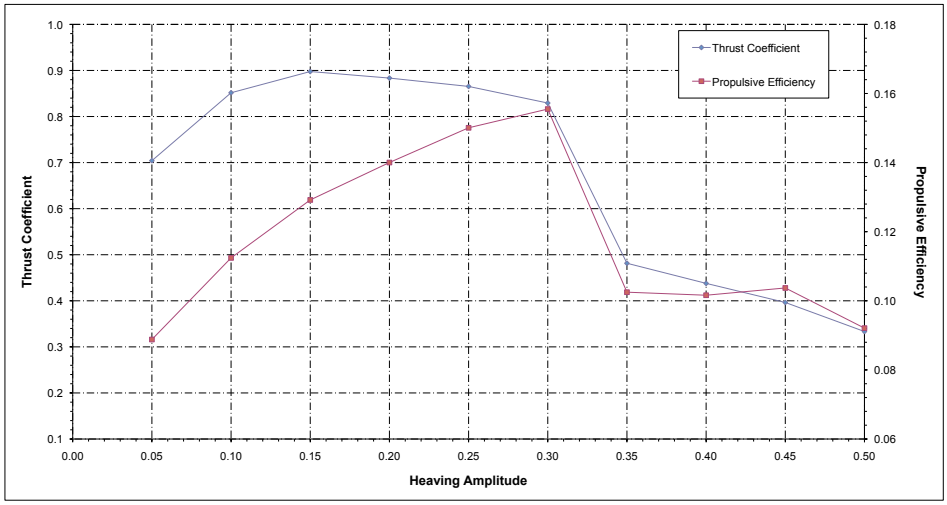

Figure 7.8: *Variation of thrust coefficient and propulsive efficiency with heaving amplitude. Flapping parameters:* $Re = 1100, St = 0.4$.

From the above discussion, we can conclude that the Strouhal number (based on the trailing edge excursion or heaving amplitude h_a) seems to be enough for wake signature characterization, but is not sufficient insofar as maximum efficiency is concerned. Both heaving amplitude h_a and

heaving frequency f_h (hence the Strouhal number St and the reduced frequency k), should be adjusted separately.

In figure 7.9, the previous results are extended as to characterize the wake topology for a wider range of heaving amplitude and heaving frequency combinations, the plot obtained is similar to that used by Triantafyllou *et al.* [191], Jones *et al.* [97] and Lua *et al.* [115]. In this figure, the numerically simulated wakes are classified according to the observed vortex positions. Lines of constant St are included demonstrating the approximate dependence of the numerical results on the Strouhal number. The wakes are classified as drag producing, neutral, thrust producing, deflected wake and jet-switching wake.

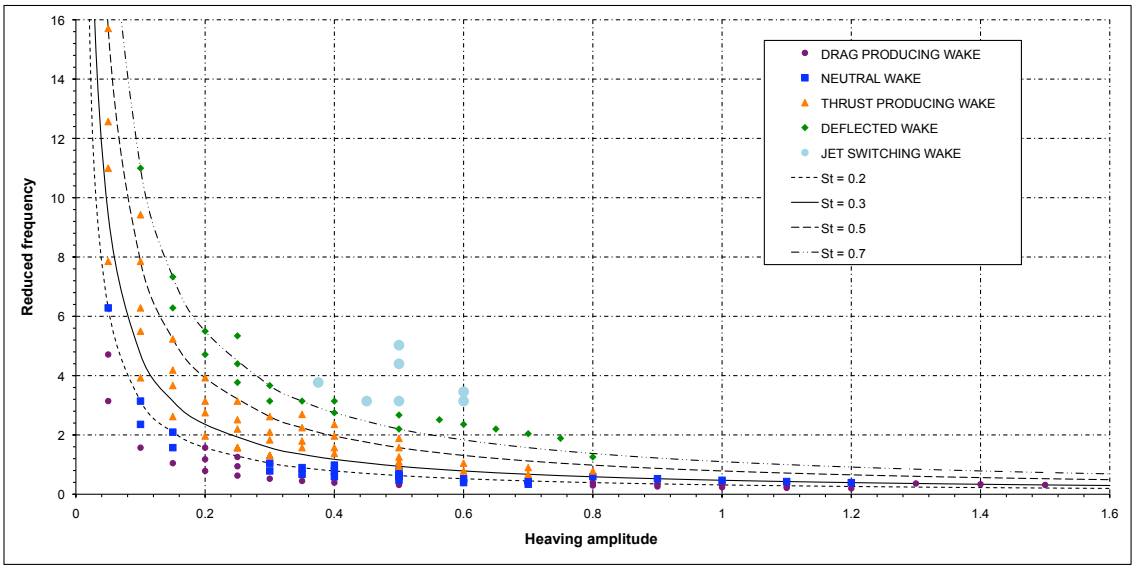

Figure 7.9: *Plot showing the wake structure classification. Lines of constant Strouhal number value are included to demonstrate the approximate dependence of the wake topology on the Strouhal number.*

In figure 7.9, it is observed that for low values of St ($St < 0.2$), a drag producing wake is generated. For values approximately between $0.2 < St < 0.25$ a neutral wake is encountered and for values higher than $St = 0.3$, we are in the presence of a thrust producing wake. In this figure, it is also observed that for Strouhal numbers $St > 0.5$, the wake start to become deflected, becoming asymmetric and in some cases aperiodic, as described later. It is important to note that these classifications are based purely on the observed vortex positions. Quantitative measurements, as we will be presenting later, suggest that these classifications are fairly conservative, with thrust generation occurring at Strouhal numbers as low as $St = 0.2$.

The deflected wake topology (see figure 7.10), apparently was first observed experimentally by Bratt [23], but he did not comment on these deflected wakes. In fact, it seems that they were never again reported until 1998, when Jones *et al.* [94] studied them in greater detail, both experimentally and computationally. Numerically, the deflection angle of the wake is determined by the starting direction of the heaving motion, *i.e.*, if the heaving motion starts from the bottommost position, the wake is deflected upwards and if the heaving motion starts from the topmost position, the wake is deflected downwards, as shown in figure 7.10.

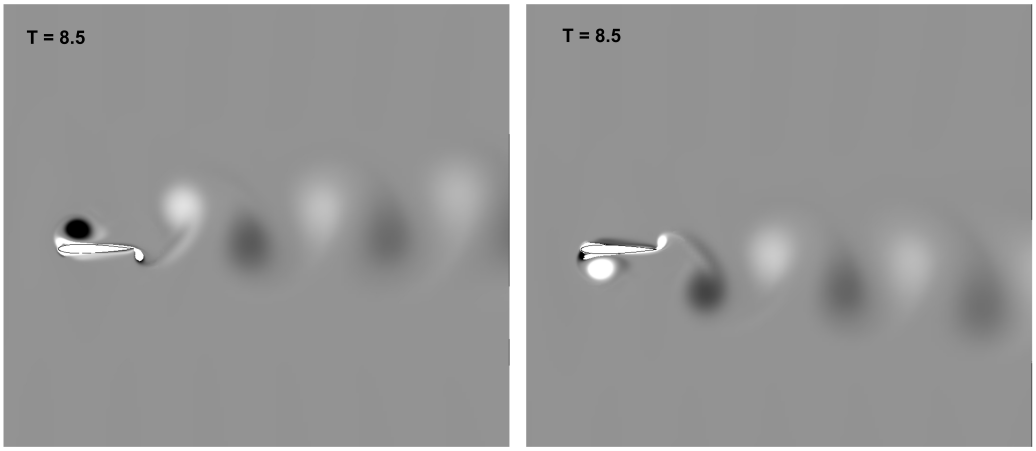

Figure 7.10: *Deflected wake (vorticity contours). In the left figure the wake is deflected upwards (the motion was started from the bottommost position), while in the right figure the wake is deflected downwards (the motion was started from the topmost position). Flapping parameters for both cases:* $Re = 1100$, $St = 0.5$, $h_a = 0.3$.

In a numerical study performed by Emblemsvag *et al.* [49], it was reported that as the Strouhal number increases, the vortices tend to shed in pairs (vortex dipoles) and form a deflected wake. This same observation was also confirmed in the present study. In figure 7.11, we plot the horizontal velocity profile as the Strouhal number increases (which reflects the wake deflection). The time average velocity profiles were measured at a distance equals to 6 times the airfoil chord away from the trailing edge.

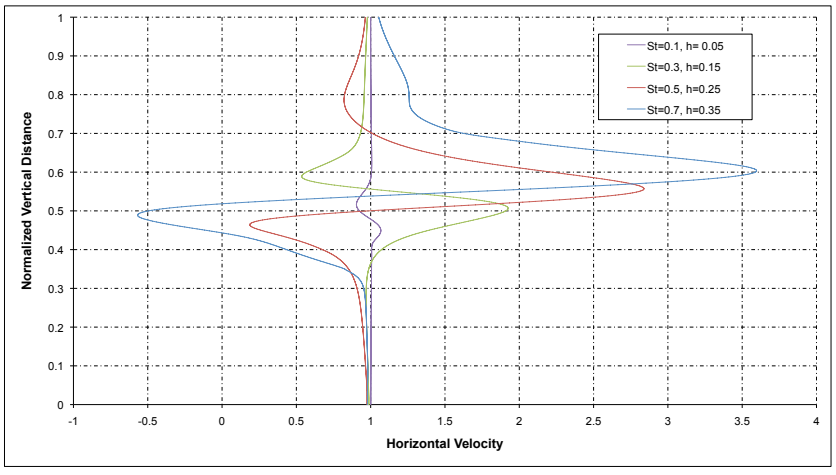

Figure 7.11: *Horizontal velocity profile evolution as function of the Strouhal number.*

Besides the obvious wake deflection, there are several other interesting features of this solution. At this high Strouhal numbers a substantial thrust coefficient is produced, with fairly low propul-

sive efficiency. What is interesting is that an average lift is also produced due to an induced angle of attack created by the presence of the deflected wake (see figure 7.12). In general, the deflected wake topologies were found to be highly reproducible. The simulations were run for a long time and it was observed that for Strouhal number values between $0.5 < St < 0.7$, they do appear to be converging toward a periodic asymptote.

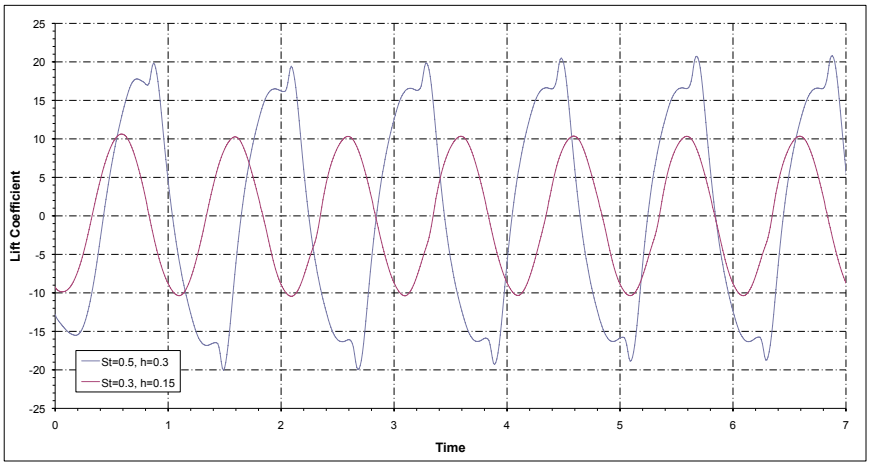

Figure 7.12: *Time dependent lift coefficient for two different heaving cases. Notice that for the case* $St = 0.5\,(h = 0.3)$*, the lift coefficient evolution is not symmetric about the horizontal mean line, this is due to the angle of attack induced by the deflected wake, the bumps on the lift curve are due to the dynamic stall* ($\overline{c_l} = 0.10923$)*. For the case* $St = 0.3\,(h = 0.15)$ *the lift evolution is symmetrical about the horizontal mean line* ($\overline{c_l} = 0.00242$).

An interesting phenomenon related to the deflected wake is the aperiodic jet-switching phenomenon, where the direction in which the wake is deflected is observed to switch from upward to downward deflection (or downward to upward, depending on the starting direction of the heaving motion). This switching was reported by Jones *et al.* [94], for high-frequency heaving in water tunnel experiments, but they were not able to reproduce it using their inviscid unsteady panel code. Lewin and Haj-Hariri [108], were apparently the firsts to numerically simulate the jet-switching phenomenon. In the present study, we were able to simulate the jet-switching phenomenon but for values different from those used by Lewin and Haj-Hariri; the fact that we were not able to reproduce the same results of Lewin and Haj-Hariri is probably due to the differences with the grid and the numerical method. It is worth to mention that from our simulations we were not able to determine if the switching is random or periodic, basically due to the fact that the simulations have to be run for long times due to the aperiodic nature of the phenomenon, as noted by Heathcote and Gursul [69]. In their work, they presented experimental evidence that in fact, the jet-switching phenomenon is quasiperiodic. They found the period of switching to be two orders of magnitude greater than the heaving period. These large periods are clearly challenging for numerical simulations, and may be the reason for the lack of numerical simulations displaying the phenomenon. At the high Strouhal number values where the deflected wake is encountered, the vortices shed from the trailing edge (TEV) come very close to one another and start to interact with each other forming a vortex pair, which apparently seems to be the reason for the phenomenon. The leading edge vortex (LEV) has also an influence on the occurrence of this phenomenon, but in general the TEV seems to be the dominating factor.

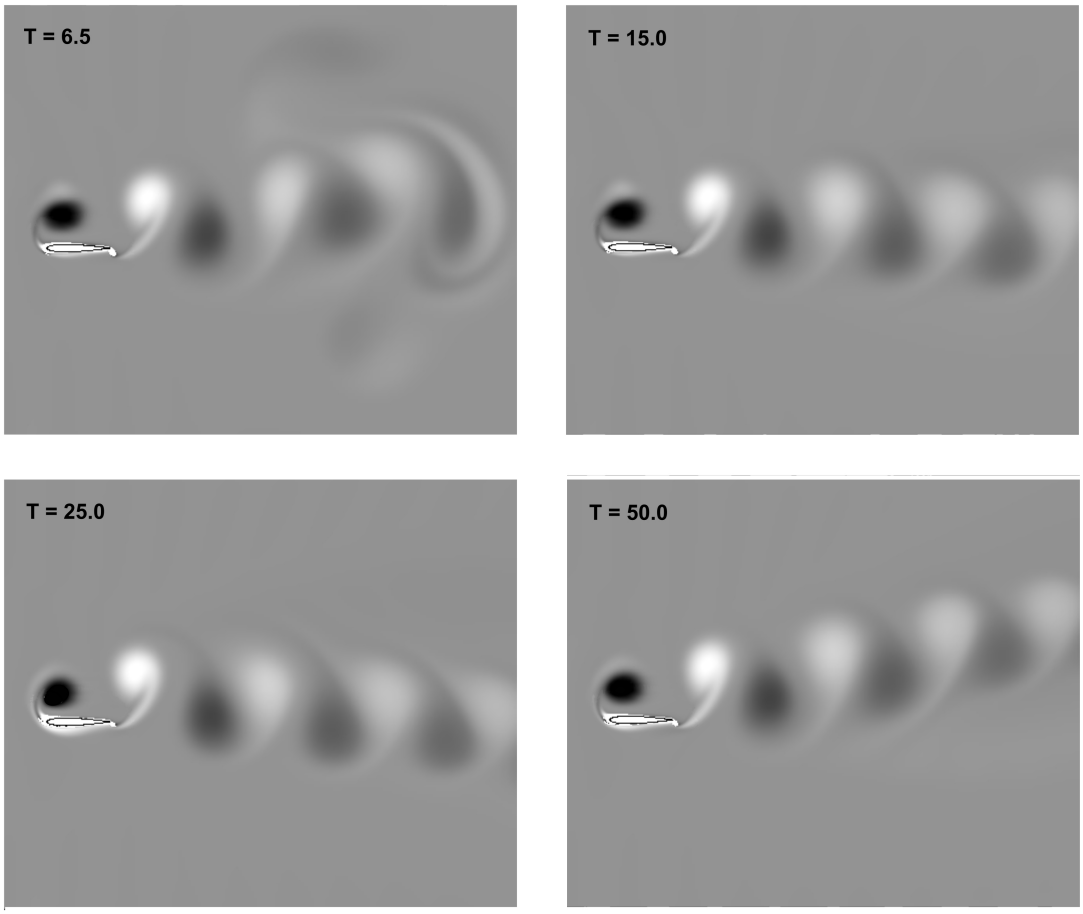

Figure 7.13: *Jet-switching phenomenon (vorticity contours). Notice how the wake first goes to the upward position, then goes to the downward position and then returns to the upward position. Also notice that the wake deflection angles are different for the upward and downward deflection. Flapping parameters:* $Re = 1100$, $St = 0.9$, $h_a = 0.45$.

In order to fully study our parameter space (f_h, h_a) and to isolate the optimal aerodynamic performance, we construct a contour map of propulsive efficiency against Strouhal number and heaving amplitude (nearly 100 simulations), as shown in figure 7.15. In this figure, we can clearly identify two regions of maximum efficiency, one corresponding to $0.25 < St < 0.3$ and a second one corresponding to $0.35 < St < 0.4$. Cross-referencing this figure with its corresponding contour map of thrust coefficient (figure 7.16), we observe that these two regions correspond to a *"low thrust"* and a *"high thrust"* areas, respectively. Usually, the first one is not significant for practical applications because the thrust coefficient is often very small, whereas the second peak is associated with higher thrust coefficient (in our case the coefficient value of the *"high thrust"* area is approximately twice the value of the *"low thrust"* area). In figure 7.17, we present the contour map of input power coefficient; in this figure we can observe that both the *"low thrust"* and *"high thrust"* areas are in the range of low input power coefficient. Thrust coefficients values higher that the values corresponding to the highest efficiencies are in fact obtained, but at the cost of very high input power coefficient (as seen in figure 7.17) and hence the propulsive efficiency is very low (figure 7.15), making this range of high thrust production unattractive for practical

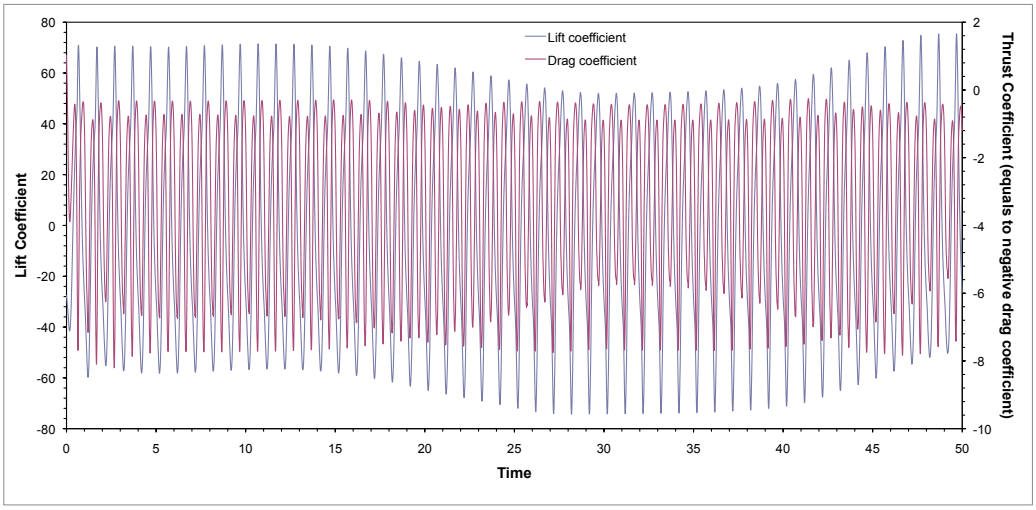

Figure 7.14: *Lift coefficient and drag coefficient time histories for the jet-switching wake. Flapping parameters:* $Re = 1100$, $St = 0.9$, $h_a = 0.45$. *Negative values of drag coefficient indicate thrust production.*

application.

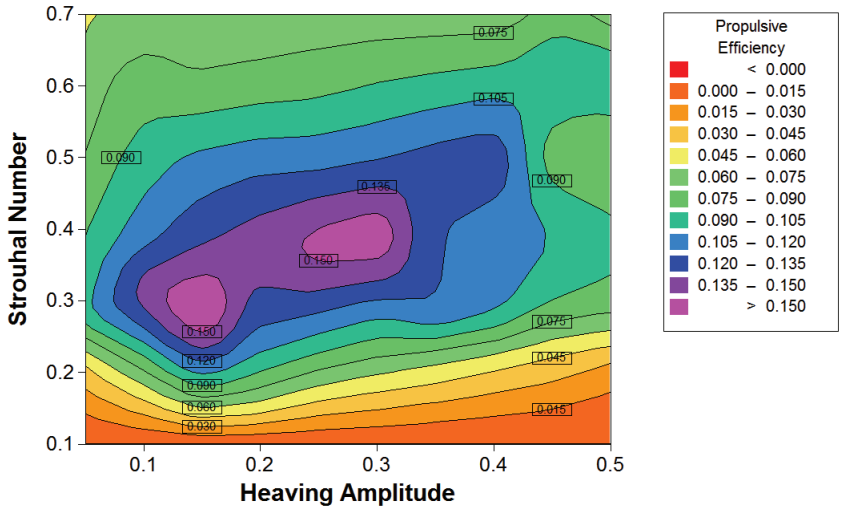

Figure 7.15: *Contour map of propulsive efficiency vs. Strouhal number and heaving amplitude.*

All these qualitative and quantitative results agree with the hypothesis that : *"flying and swimming animals cruise at a Strouhal number tuned for high power efficiency"* [182]. The enhanced efficiency range was found to be between Strouhal number values corresponding to $0.2 < St < 0.4$, which agrees with the observations of Nudds *et al.* [136], Rohr and Fish [155], Taylor *et al.* [182] and Triantafyllou *et al.* [193]. From this extensive numerical experiment, we can also confirm that the Strouhal number (based on the total trailing edge excursion or heaving amplitude) is enough for wake topology characterization, but cannot be used as a single parameter to describe

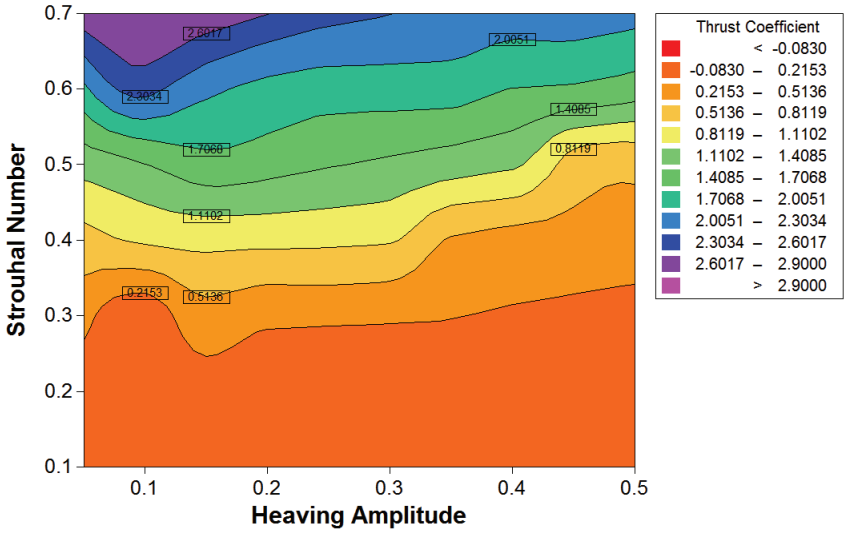

Figure 7.16: *Contour map of thrust coefficient vs. Strouhal number and heaving amplitude.*

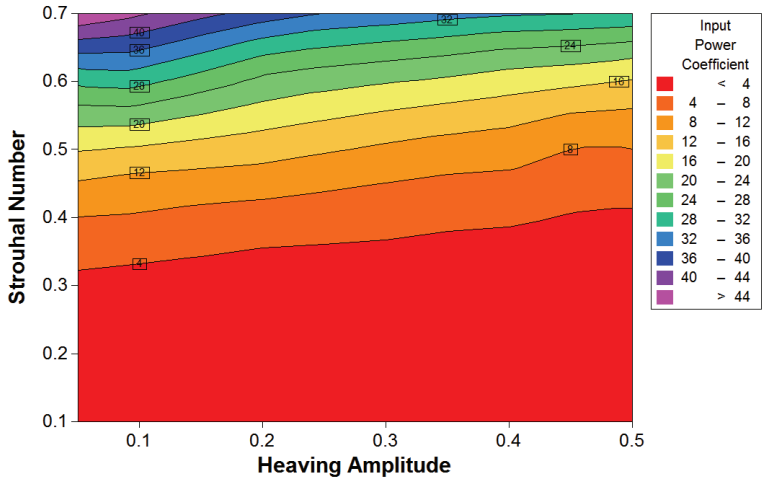

Figure 7.17: *Contour map of input power coefficient vs. Strouhal number and heaving amplitude.*

the flow in terms of maximum efficiency. The reduced frequency k is also necessary to have a complete description of the aerodynamic performance.

7.3 Leading Edge Vortex Shedding and Frequency Dependence

From figure 7.15, it can be observed that very different behaviors on the aerodynamic performance can be obtained between high heaving frequencies (low heaving amplitudes) and low heaving frequencies (high heaving amplitudes). The explanation for this frequency dependence will be studied hereafter.

7.3. LEADING EDGE VORTEX SHEDDING AND FREQUENCY DEPENDENCE

Let us proceed to compute the instantaneous distribution of lift coefficient $c_l\,(x/c)$ and thrust coefficient $c_t\,(x/c)$ on the airfoil surface due solely to the pressure. These quantities are computed as follows

$$\begin{aligned} c_l\,(x/c) &= -c_p cos\,(\theta_s) \\ c_t\,(x/c) &= -c_p sin\,(\theta_s) \end{aligned} \tag{7.1}$$

where c_p is the local pressure coefficient on the airfoil surface, and θ_s is the local angle of the airfoil surface with respect to the horizontal axis. The instantaneous distribution of moment coefficient $c_m\,(x/c)$ and input power coefficient $c_P\,(x/c)$ on the airfoil surface are computed as follows

$$\begin{aligned} c_m\,(x/c) &= -c_l\,(x/c)\left(\frac{x-x_p}{c}\right) - c_t\,(x/c)\left(\frac{y-y_p}{c}\right) \\ c_P\,(x/c) &= -c_l\,(x/c)\left(\frac{\dot{y}(t)}{U_\infty c}\right) - c_m\,(x/c)\left(\frac{\dot{\alpha}(t)}{U_\infty}\right) \end{aligned} \tag{7.2}$$

Note that for pure heaving motion the pitch rate is $\dot{\alpha}(t) = 0$, so that the input power coefficient distribution is dependent only on the lift coefficient distribution and the vertical motion of the airfoil.

In figure 7.18, we show the pressure field around a heaving airfoil for eight different instants in time during period of the signal (which corresponds to the upstroke phase). As it can be seen in this figure, there is a vortex that separates from the leading edge and is convected downstream over the airfoil surface; obviously, the low pressure in the vortex core have an effect on the aerodynamic performance that will be studied hereafter. Halfway through the period a vortex is formed at the leading edge and proceeds along the lower side of the airfoil, eventually detaching at the trailing edge.

By using equations 7.1 and 7.2, we proceed to compute the instantaneous lift coefficient, thrust coefficient, moment coefficient and input power coefficient. In figures 7.19 and 7.20 we show the pressure coefficient distribution on the airfoil surface and the regions on the airfoil surface responsible for the greatest development of thrust at four different instants during the upstroke. The flapping parameters for this case are $Re = 1100$, $St = 0.35$, $h_a = 0.40$ and $f = 0.4375$ ($k = 1.3744325$).

In figure 7.18, the flow separates at the leading edge and forms a leading edge vortex (LEV), which is convected downwards; this LEV is a spike of locally low pressure. The vortex is convected downstream over the airfoil and diffuses as it moves towards the trailing edge. As long as the vortex remains upstream of the point of maximum thickness of the airfoil, it contributes towards thrust, however as it travels aft of this point its contribution changes sharply and drag is produced.

In figures 7.21, 7.22 and 7.23, similar results to those of the previous case are illustrated. In this case the flapping parameters are $Re = 1100$, $St = 0.35$, $h_a = 0.15$, $f = 1.166667$ ($k = 3.665166$). In figure 7.21, the pressure field for eight different instants during the upstroke are shown; also in this case a LEV forms and separates, but this time it is not convected all the way downstream. In this case, the vortex remains more time in the nose region of the airfoil and its contribution to the thrust is higher.

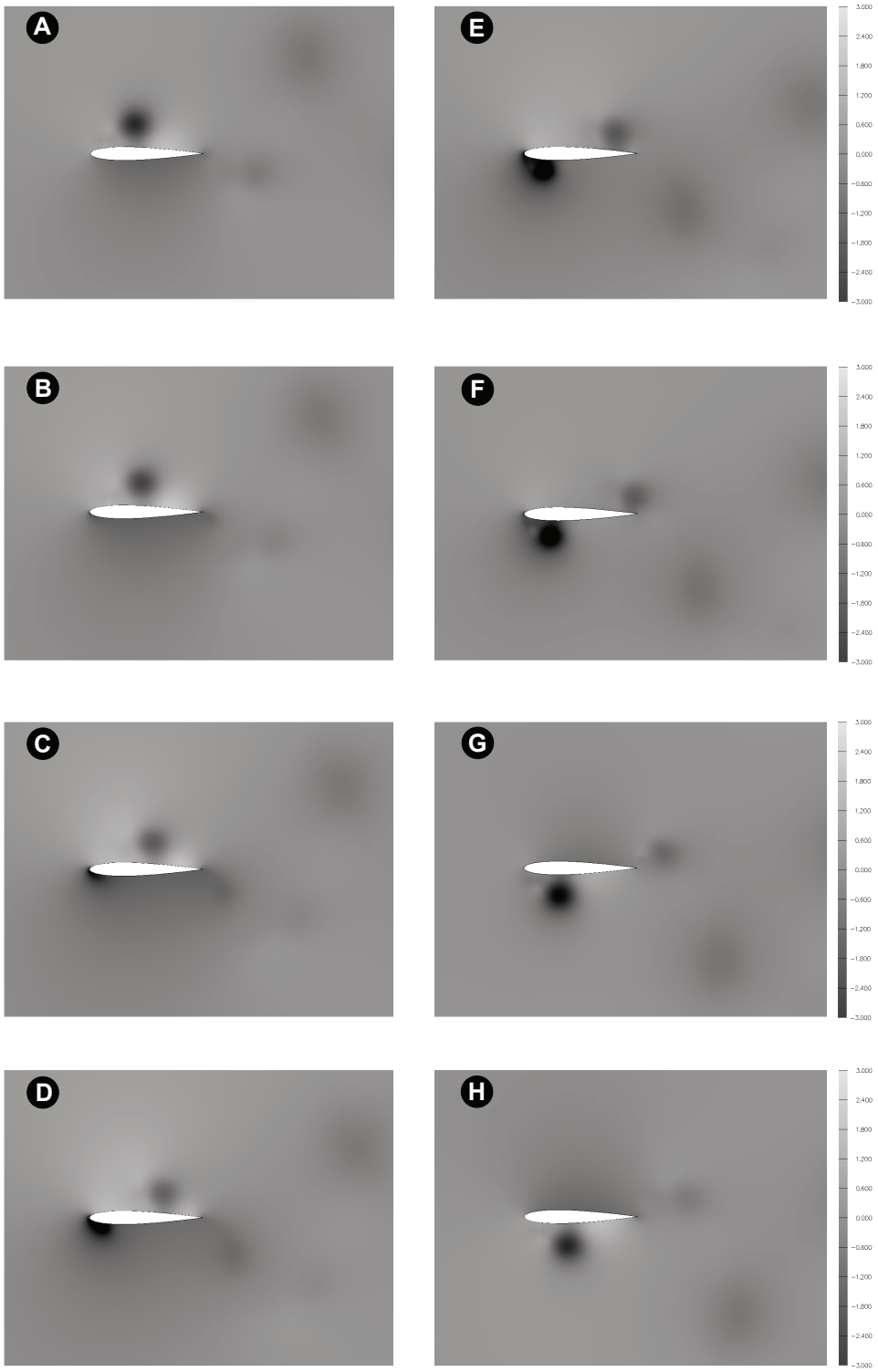

Figure 7.18: *Pressure field during upstroke. Flapping parameters:* $Re = 1100$, $St = 0.35$, $h_a = 0.40$, $f = 0.4375$ ($k = 1.3744325$). *The sequence is from A (bottommost position) to H (topmost position), where: A)* $t = 13.725$, *B)* $t = 13.860$, *C) 14.0, D) 14.168, E) 14.345, F) 14.476, G) 14.652, H) 14.828.*

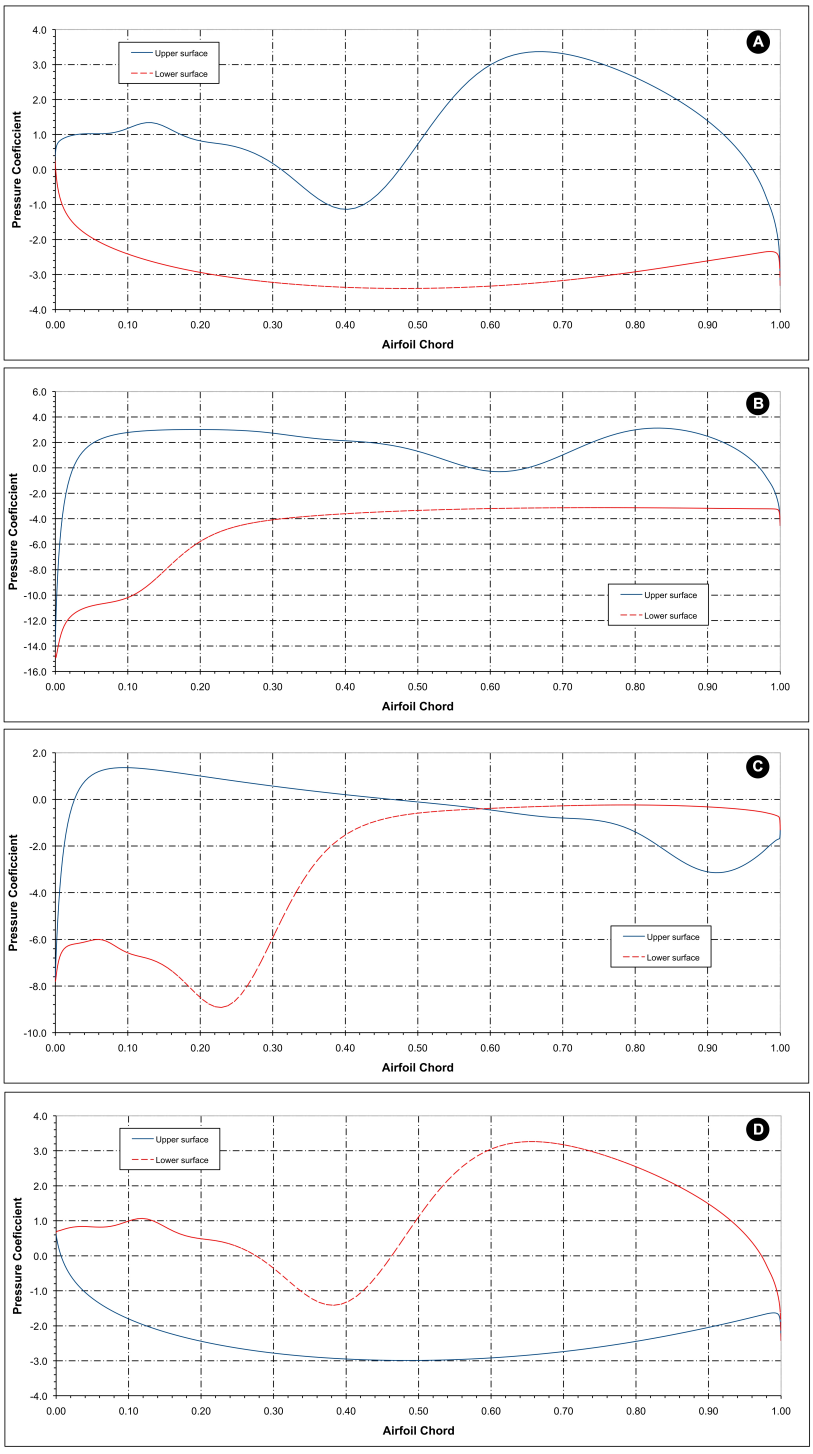

Figure 7.19: *Pressure coefficient distribution on the airfoil surface. Flapping parameters:* $Re = 1100$, $St = 0.35$, $h_a = 0.40$, $f = 0.4375$ $(k = 1.3744325)$. *The sequence is from A (bottommost position) to D (topmost position). The pressure coefficient* c_p *was measured at the following instants: A)* $t = 13.728$, *B)* $t = 14.124$, *C) 14.476, D) 14.828.*

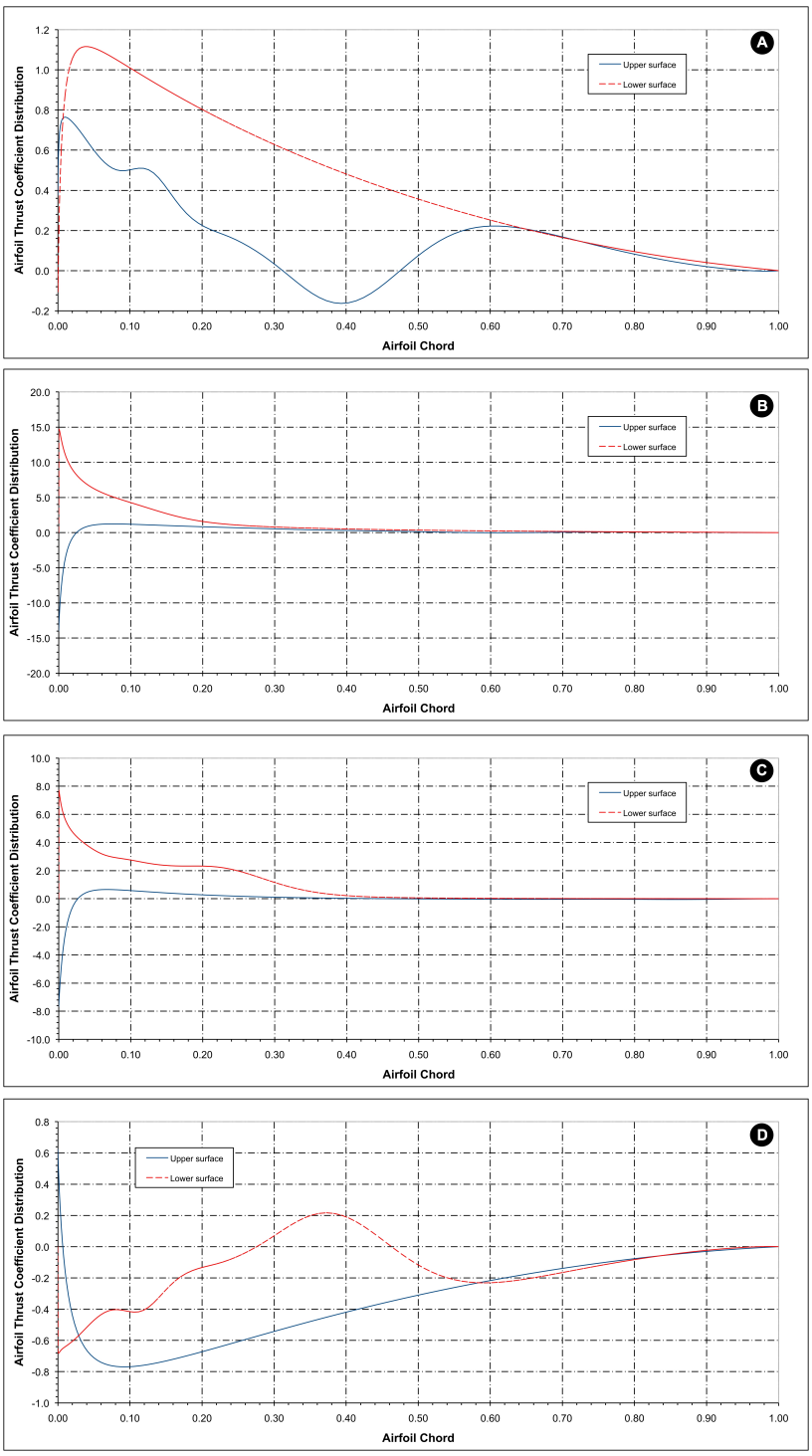

Figure 7.20: *Thrust coefficient distribution on the airfoil surface. Flapping parameters:* $Re = 1100$, $St = 0.35$, $h_a = 0.40$, $f = 0.4375$ $(k = 1.3744325)$. *The sequence is from A (bottommost position) to D (topmost position). The thrust coefficient distribution* $c_t(x/c)$ *was measured at the following instants: A)* $t = 13.728$, *B)* $t = 14.124$, *C) 14.476, D) 14.828.*

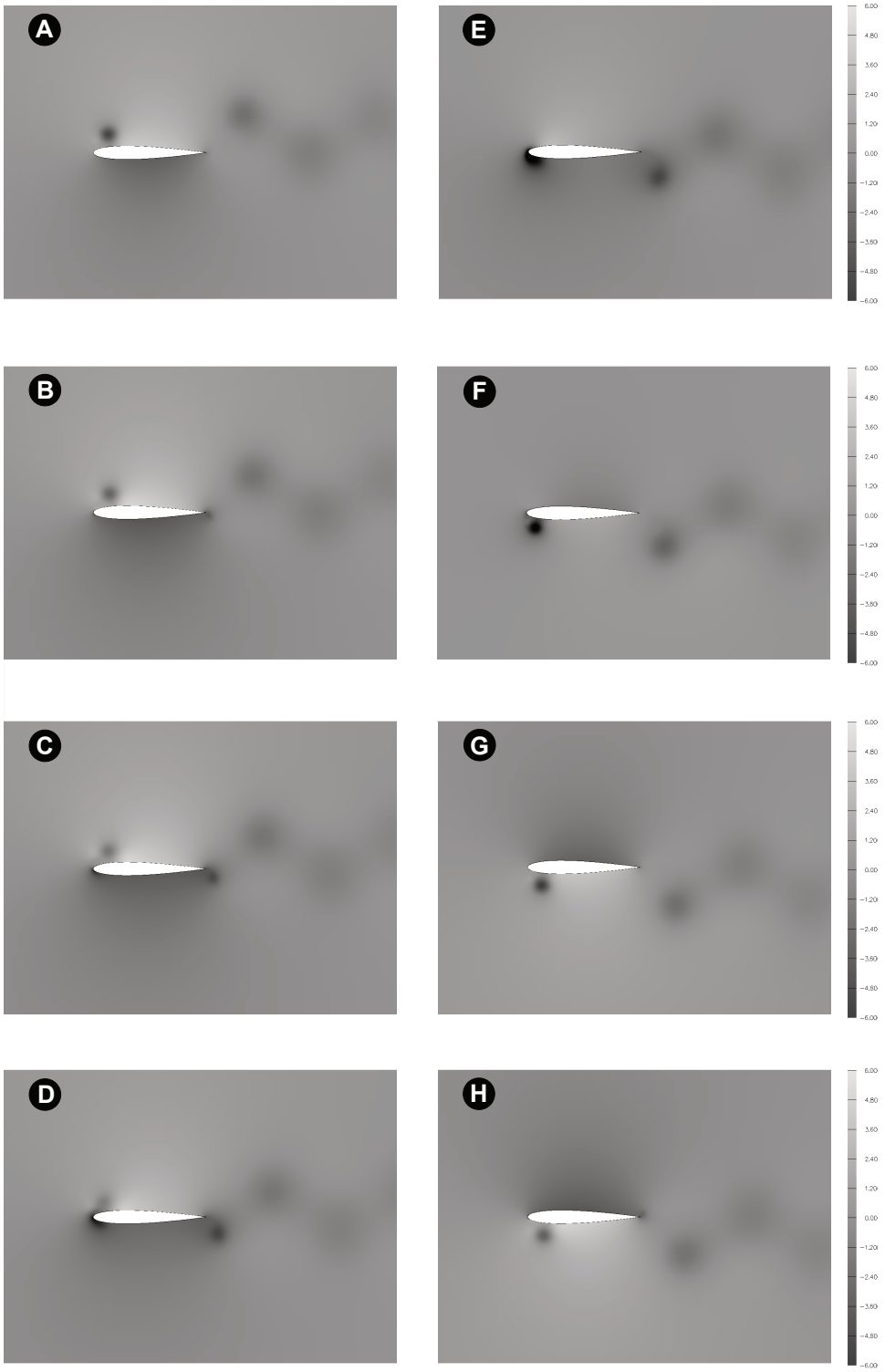

Figure 7.21: *Pressure field during upstroke. Flapping parameters:* $Re = 1100$, $St = 0.35$, $h_a = 0.15$, $f = 1.166667$ ($k = 3.665166$). *The sequence is from A (bottommost position) to H (topmost position), where: A)* $t = 5.120$, *B)* $t = 5.184$, *C) 5.248, D) 5.317, E) 5.376, F) 5.456, G) 5.536, H) 5.6.*

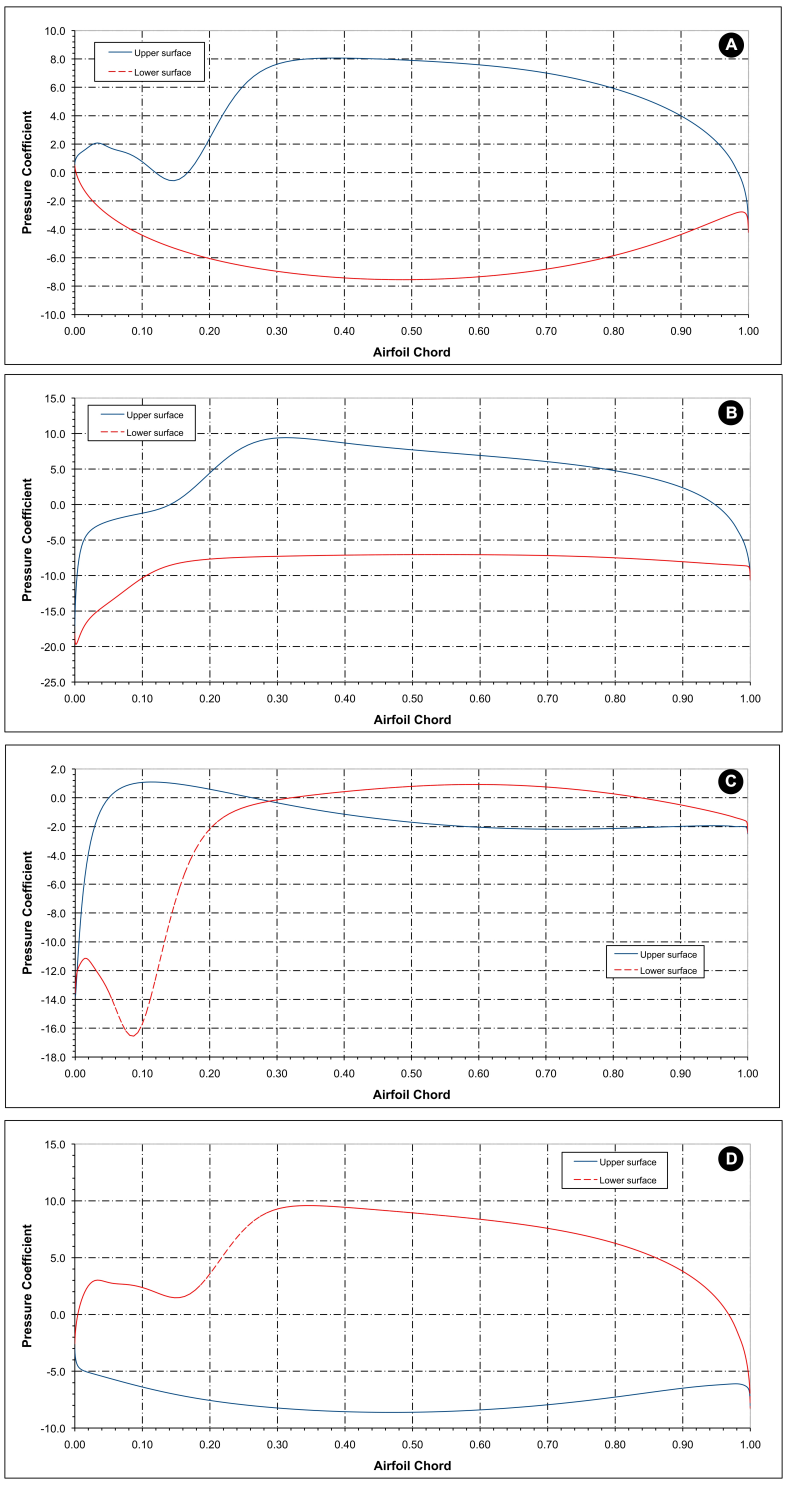

Figure 7.22: *Pressure coefficient distribution on airfoil surface. Flapping parameters:* $Re = 1100$, $St = 0.35$, $h_a = 0.15$, $f = 1.166667$ ($k = 3.665166$). *The sequence is from A (bottommost position) to D (topmost position). The pressure coefficient* c_p *was measured at the following instants: A)* $t = 5.120$, *B)* $t = 5.280$, *C) 5.456, D) 5.6.*

By comparing figures 7.18 and 7.21, it is evident that the effect of the LEV separation becomes

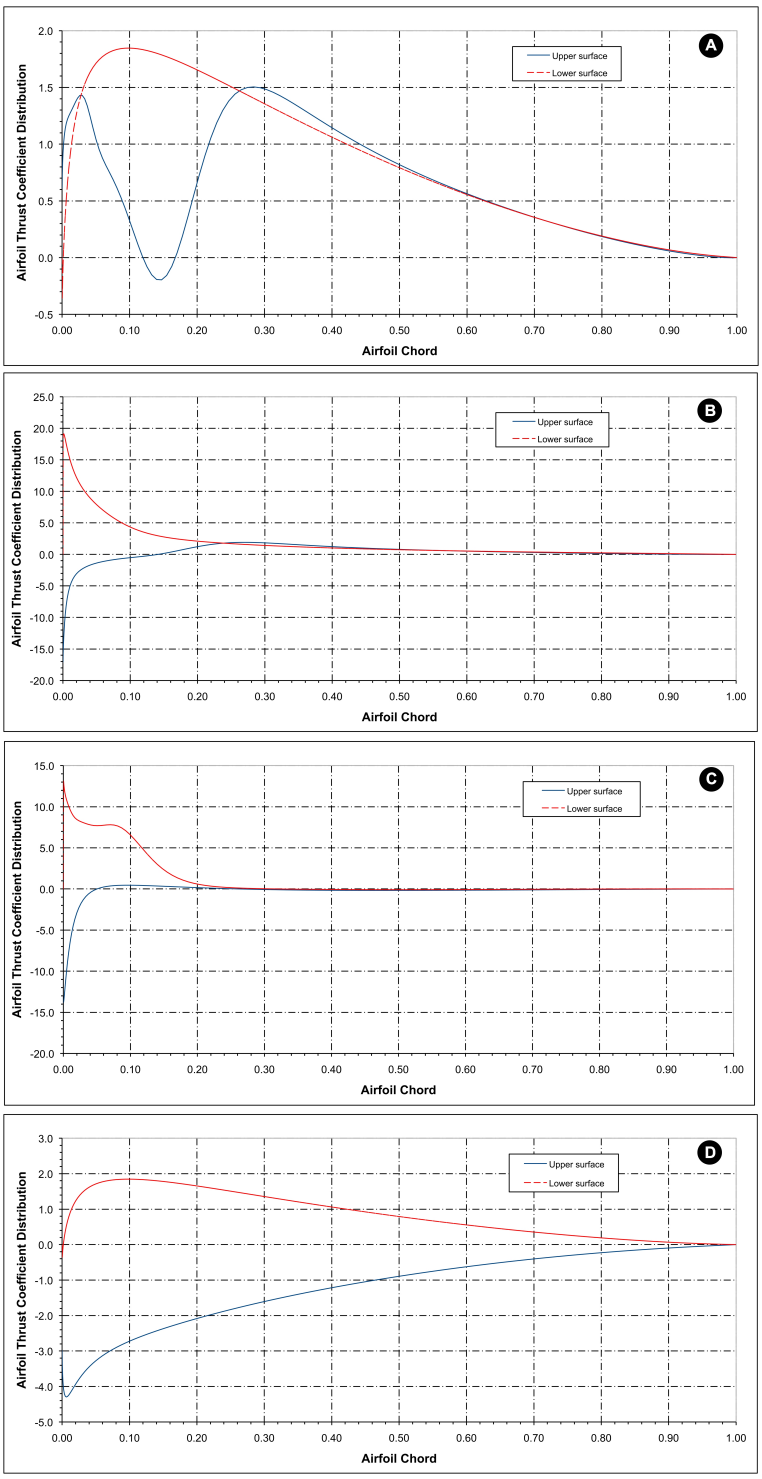

Figure 7.23: *Thrust coefficient distribution on airfoil surface. Flapping parameters:* $Re = 1100$, $St = 0.35$, $h_a = 0.15$, $f = 1.166667$ $(k = 3.665166)$. *The sequence is from A (bottommost position) to D (topmost position). The thrust coefficient distribution* $c_t(x/c)$ *was measured at the following instants: A)* $t = 5.120$, *B)* $t = 5.280$, *C) 5.456, D) 5.6.*

much less pronounced as the flapping frequency f (and hence the reduced frequency k) is increased. Both the size of the vortex and its effect on the airfoil surface pressure distribution are reduced as the frequency k is increased (while holding St constant). Thus leading edge separation introduces a frequency dependence into the results. This provides a mechanism of optimal selection of heaving frequency (in the sense of maximum propulsive efficiency), as discussed by Wang [207].

As the reduced frequency k increases, there is less time for the vortex to form and to travel downstream along the airfoil past the point of maximum thickness, where it contributes to drag rather than to thrust generation, hence lowering the propulsive efficiency. The LEV also has a smaller relative effect on the airfoil surface pressure distribution at higher k, because the pressure extremes developed around the airfoil during the motion cycle are greater with increasing k (note the increased pressure coefficient by comparing figures 7.22 and 7.19). This is consistent with the peak lift coefficient increasing roughly linearly with k, as shown in figure 7.24.

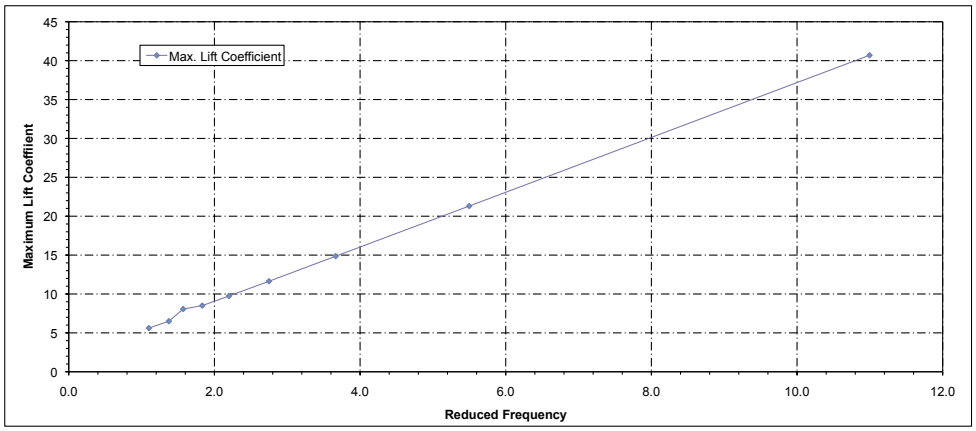

Figure 7.24: *Peak lift coefficient versus reduced frequency k. Flapping parameters: $Re = 1100$, $St = 0.35$.*

It is worth noting that a large majority of the thrust is generated at the nose of the airfoil, particularly at low k, consistent with the observations of Tuck [195] and Lee *et al.* [106]. From these results, it is also evident that the wake roll-up has only a secondary effect on the thrust generated and this is in agreement with the findings of Hall and Hall [64], where the effect of wake roll-up on the forces generated by a heaving flat plate was studied. They showed in [64] that the time histories of the lift coefficient for the prescribed-wake and free-wake models were virtually identical, with only minor differences in peak values. At the frequency tested, the non-linear roll-up of the wake has only a minor effect on the forces felt on the airfoil surface.

Clearly, the separation and convection of the LEV is crucial in determining the aerodynamic behavior of the airfoil. The effect of the LEV on the aerodynamic performance is particularly pronounced for pure heaving airfoils. For coupled heaving-and-pitching motions, where the orientation of the airfoil surface is controlled by the relative amplitudes and phases of the motion, the LEV may create positive thrust for much longer portions of the flapping cycle, thus contributing towards the propulsive efficiency. Coupled heaving-and-pitching motion is explored in the next section.

7.4 Flapping airfoils (Coupled Heaving-and-Pitching Motion)

This section is devoted to the study of airfoils undergoing flapping motion (coupled heaving-and-pitching motion). The main goal is to study the effect of the flapping variables (heaving amplitude h_a, pitching amplitude α_a, heaving frequency f_h, pitching frequency f_α and phase angle φ) on the aerodynamic behavior.

As in the pure heaving case, let us first take a look at the flapping motion (eq. 2.21 and eq. 2.22) and at the evolution of lift and thrust coefficients with time for a single flapping case. In figure 7.25 the heaving-and-pitching kinematics is plotted, in this figure one period consists of an up and down stroke. As in the heaving case, since the airfoil is symmetric and is oscillating symmetrically about the mean horizontal line, we expect a symmetrical lift evolution. The frequency of the thrust coefficient is twice that of the lift coefficient because thrust is generated in both the up and down strokes.

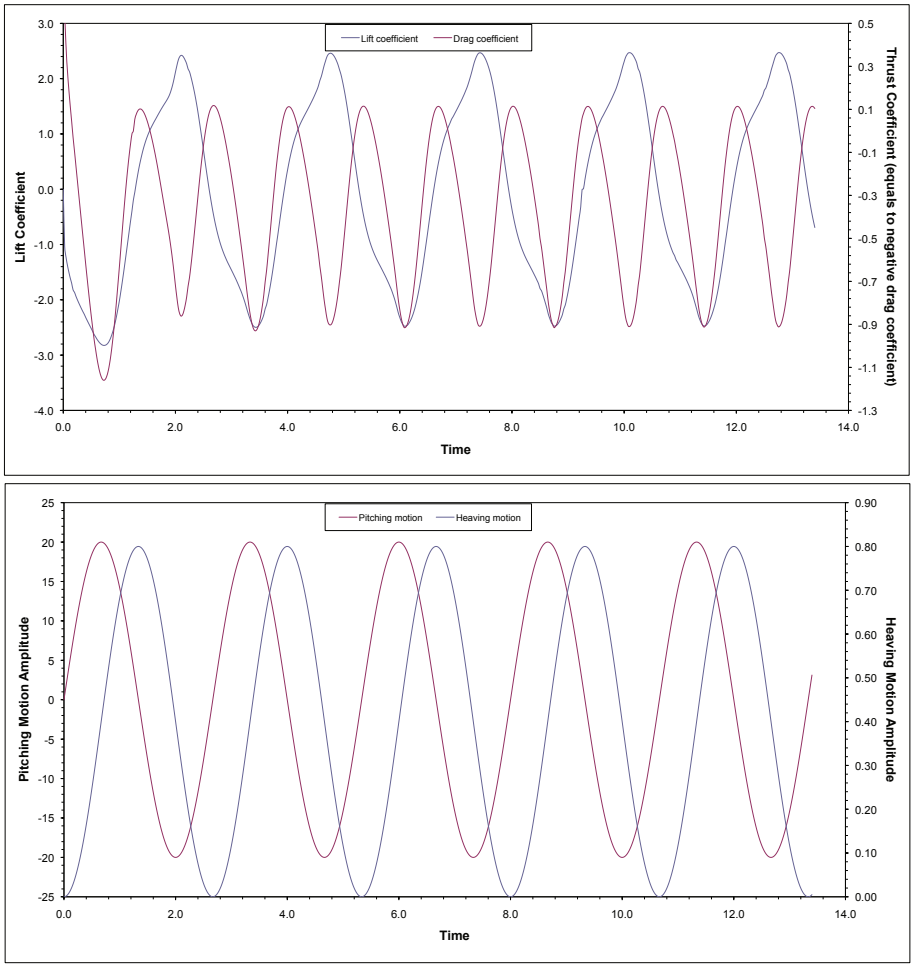

Figure 7.25: *Top figure: time dependent thrust and lift coefficients (where negative values of drag coefficient indicate thrust production). Bottom figure: Heaving and pitching kinematics. Flapping parameters:* $Re = 1100, St = 0.3, h_a = 0.4$, $\alpha_a = 20°$, $\varphi = 90°$ ($\overline{c_t} = 0.354186$, $\overline{c_l} = -0.001725$).

7.4.1 Effect of Maximum Pitching Angle

Hereafter, we conduct a parametric study to determine the best maximum pitching angle α_a for operating with the highest efficiency. In this section, we vary the maximum pitching angle α_a from 5 to 40 degrees, and we fix the pitching and heaving frequency $f_h = f_\alpha = f$, heaving amplitude h_a and phase angle φ. The Strouhal number was set to $St = 0.3$, which corresponds to a maximum efficiency peak in figure 7.15.

Case number	f(Hz)	h_a	$\alpha_a(^\circ)$	$\varphi(^\circ)$	k	St
2DF1-1	0.15	1.0	5.0	90.0	0.7096	0.3
2DF1-2	0.15	1.0	10.0	90.0	0.7096	0.3
2DF1-3	0.15	1.0	15.0	90.0	0.7096	0.3
2DF1-4	0.15	1.0	20.0	90.0	0.7096	0.3
2DF1-5	0.15	1.0	25.0	90.0	0.7096	0.3
2DF1-6	0.15	1.0	30.0	90.0	0.7096	0.3
2DF1-7	0.15	1.0	35.0	90.0	0.7096	0.3
2DF1-8	0.15	1.0	40.0	90.0	0.7096	0.3

Table 7.2: *Kinematics parameters for the study of the effect of maximum pitching angle on flapping airfoils aerodynamic performance. Flapping parameters:* $Re = 1100$, $St = 0.3$, $h_a = 1.0$.

Figure 7.26 shows the average thrust coefficient and propulsive efficiency as function of the maximum pitching angle α_a. In this figure, a maximum in the average thrust coefficient can be identified between $15 < \alpha_a < 25$ degrees, whereas a maximum in the propulsive efficiency can be identified around $\alpha_a = 30$ degrees.

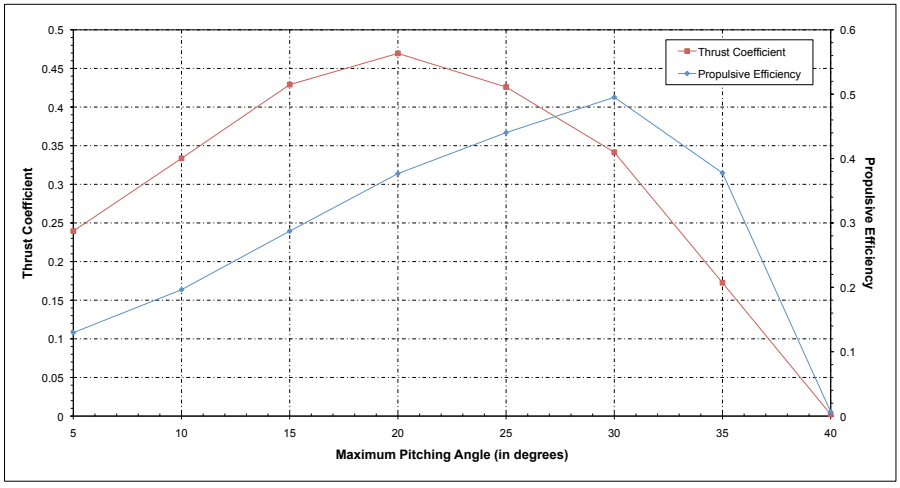

Figure 7.26: *Average thrust coefficient and propulsive efficiency versus maximum pitching angle. Flapping parameters:* $Re = 1100$, $St = 0.3$, $h_a = 1.0$.

To choose a point of operation, one has to balance the need of high thrust without sacrificing propulsive efficiency. Maximum efficiency is obtained at a maximum pitching angle $\alpha_a = 30$ degrees, while maximum thrust is obtained at a maximum pitching angle $\alpha_a = 20$ degrees. There-

fore, a maximum pitching angle value between $20 < \alpha_a < 30$ degrees seems to be a reasonably good choice for operating at high thrust coefficient and propulsive efficiency values.

7.4.2 Effect of Heaving Amplitude

Let us now study the effect of heaving amplitude on the aerodynamic performance. The parameters used for this study are shown in table 7.3, where we fix the Strouhal number value to $St = 0.3$ and we vary the heaving amplitude h_a and the flapping frequency $f_h = f_\alpha = f$, the results are shown for five different values of maximum pitching amplitude (15, 20, 25, 30 and 35 degrees).

Case number	$\alpha_a(^\circ)$	$\varphi(^\circ)$	h_a	f (Hz)	St
2DF2-1	15.0	90.0	$0.1 < h_a < 1.4$	$0.107143 < f < 1.50$	0.3
2DF2-2	20.0	90.0	$0.2 < h_a < 1.4$	$0.107143 < f < 0.75$	0.3
2DF2-3	25.0	90.0	$0.2 < h_a < 1.4$	$0.107143 < f < 0.75$	0.3
2DF2-4	30.0	90.0	$0.2 < h_a < 1.6$	$0.093750 < f < 0.75$	0.3
2DF2-5	35.0	90.0	$0.2 < h_a < 1.6$	$0.093750 < f < 0.75$	0.3

Table 7.3: *Flapping parameters for the study of the effect of heaving amplitude h_a on the aerodynamic performance.*

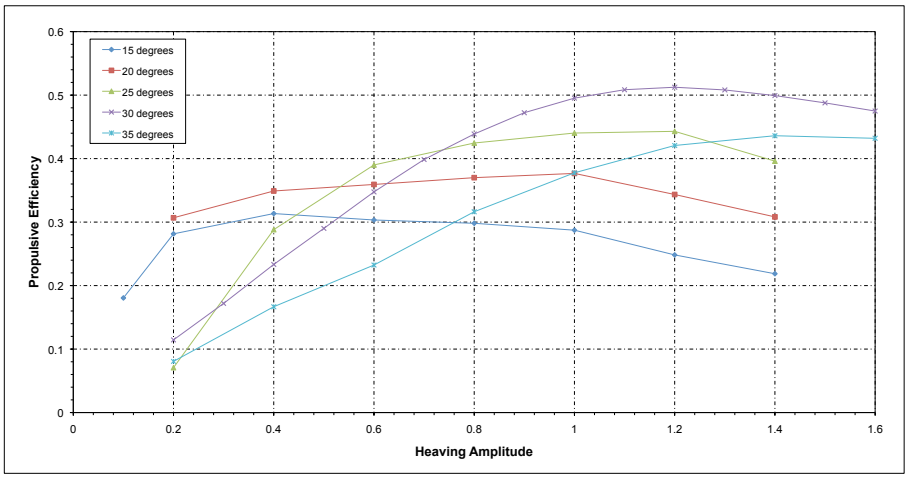

Figure 7.27: *Propulsive efficiency versus heaving amplitude. Flapping parameters: $Re = 1100$, $St = 0.3$.*

As observed in figure 7.27, the propulsive efficiency increases with the heaving amplitude, until it reaches a maximum value where from which point on it start to decrease. For the cases shown in figure 7.27, the maximum propulsive efficiency is obtained around $\alpha_a = 30$ degrees and $h_a = 1.2$. Figures 7.28 and 7.29 show the corresponding thrust coefficient and input power coefficient variations with the heaving amplitude. In figures 7.28 we can observe that at low heaving amplitude values we obtain high thrust coefficients, but at the cost of very high input power coefficient values, making this range of amplitudes unattractive for practical applications. For this case, the best operating point (in terms of propulsive efficiency) is found for heaving amplitude values between $0.8 < h_a < 1.2$.

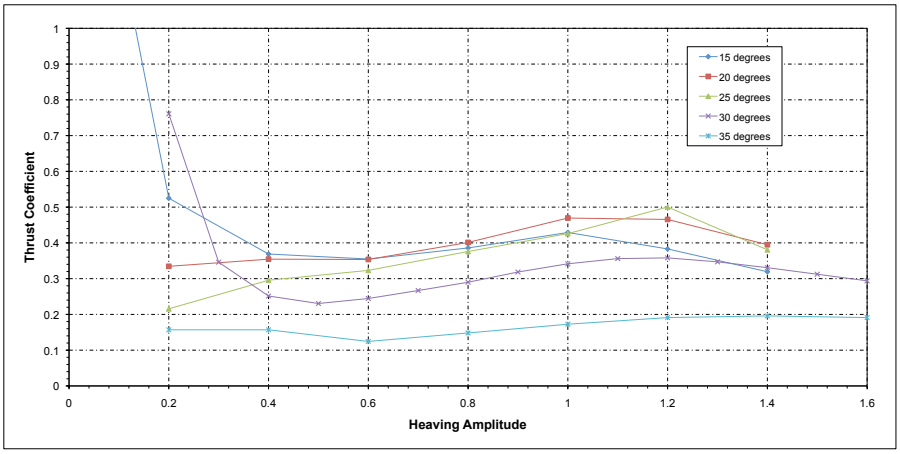

Figure 7.28: *Average thrust coefficient versus heaving amplitude. Flapping parameters:* $Re = 1100$, $St = 0.3$.

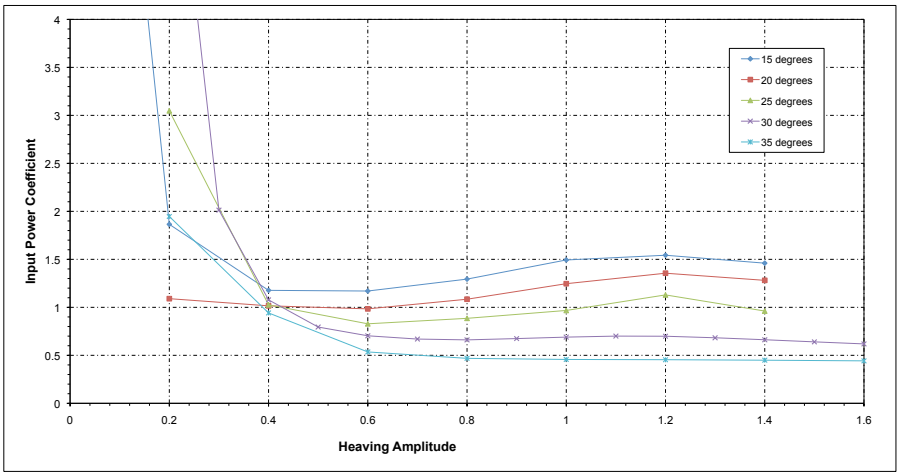

Figure 7.29: *Average input power coefficient versus heaving amplitude. Flapping parameters:* $Re = 1100$, $St = 0.3$.

7.4.3 Effect of Strouhal Number

As observed in the study of pure heaving motion, the Strouhal number is a crucial parameter in the thrust generation mechanism, and even the slightest change in the amplitude or frequency can alter quite significantly thrust and efficiency. In this case, we fix the maximum pitching angle to $\alpha_a = 20$ degrees and we vary the heaving amplitude h_a and the flapping frequency $f_h = f_\alpha = f$, the results are shown for Strouhal number values equal to 0.2, 0.3 and 0.4.

In figure 7.30 the results for the propulsive efficiency are plotted for three values of Strouhal number. In this figure, the value of Strouhal number $St = 0.2$, corresponds to low values of thrust coefficient (almost in the neutral regime), while the other two values correspond to high thrust coefficient values (see figure 7.31). As the Strouhal number rises above $St = 0.2$, the maximum propulsive efficiency values are reached very fast with the increase of the heaving amplitude, until reaching a maximum value where they begin to decrease. The increase in the average thrust

Case number	$\alpha_a(^\circ)$	$\varphi(^\circ)$	h_a	f (Hz)	St
2DF3-1	20.0	90.0	$0.2 < h_a < 1.4$	$0.071428 < f < 0.50$	0.2
2DF3-2	20.0	90.0	$0.2 < h_a < 1.4$	$0.107143 < f < 0.75$	0.3
2DF3-3	20.0	90.0	$0.2 < h_a < 1.4$	$0.142857 < f < 1.00$	0.4

Table 7.4: *Flapping parameters for the study of the effect of Strouhal number St on the aerodynamic performance.*

coefficient and input power coefficient with the Strouhal number (see figures 7.31 and 7.32) can be attributed to the increased pressure differences due to the higher heaving velocity of the airfoil as St increases, thereby resulting in higher lift and thrust coefficients. In this case, the best operating point (in terms of propulsive efficiency) corresponds to a Strouhal number value equal to $St = 0.3$, with a corresponding heaving amplitude value between $0.8 < h_a < 1.2$.

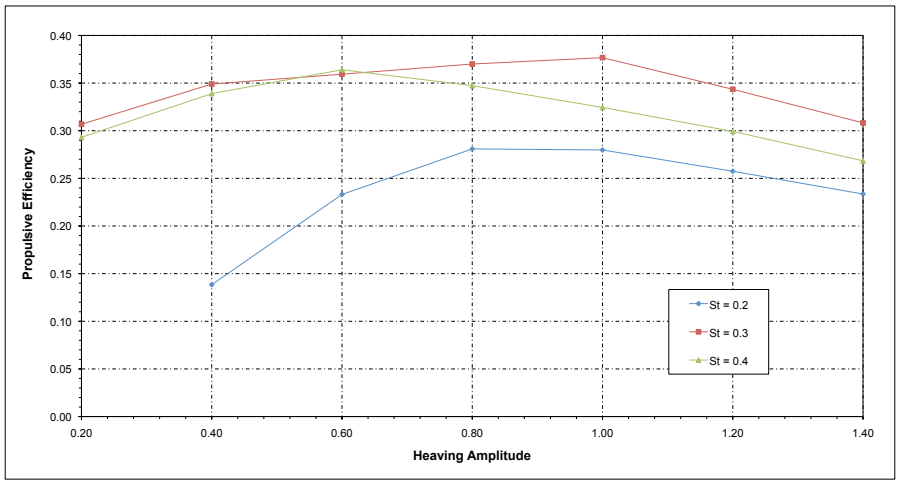

Figure 7.30: *Propulsive efficiency versus heaving amplitude for three different Strouhal number values. Flapping parameters:* $Re = 1100$, $\alpha_a = 20.0$.

7.4.4 Effect of Phase Angle

While a change in the heaving amplitude or flapping frequency has a predictable effect on the aerodynamic performance due to increased pressure differences, the effect of the phase angle is not very evident. In all the previous simulations, the phase angle value was set to $\varphi = 90$ degrees. Hereafter, we vary the phase angle between $80 < \varphi < 115$ degrees, and we fix the Strouhal number to $St = 0.25$, the heaving amplitude to $h_a = 0.5$ and the maximum pitching angle to $\alpha_a = 30$ degrees.

By varying the phase angle φ, the airfoil angle of attack is non-zero at the top and bottom positions. If the phase angle is greater than 90 degrees, at the lowest position of the heaving motion the airfoil will be pitched upwards, while if the phase angle is less than 90 degrees, at the same position the airfoil will be pitched downwards.

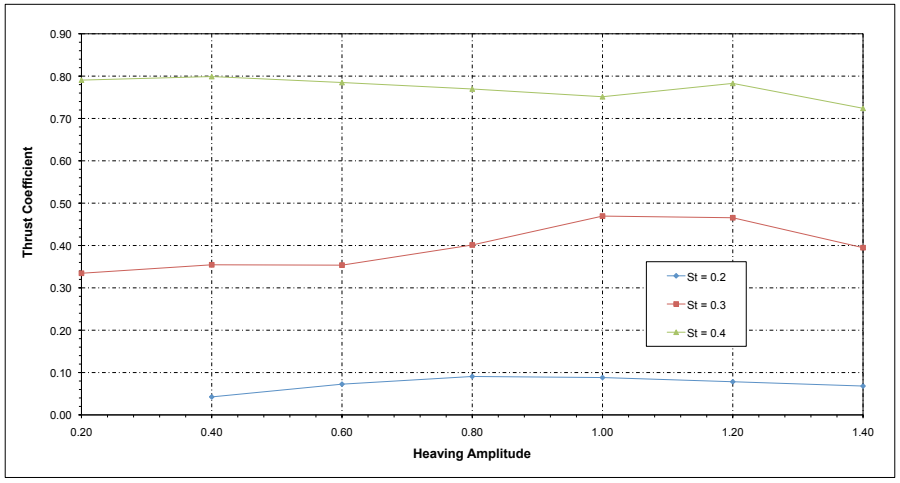

Figure 7.31: *Thrust coefficient versus heaving amplitude for three different Strouhal number values. Flapping parameters:* $Re = 1100$, $\alpha_a = 20.0$.

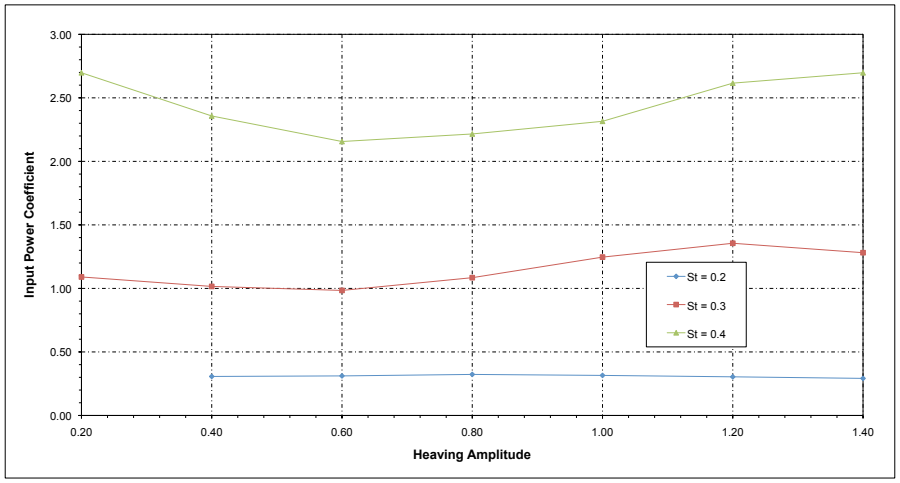

Figure 7.32: *Input power coefficient versus heaving amplitude for three different Strouhal number values. Flapping parameters:* $Re = 1100$, $\alpha_a = 20.0$.

In figure 7.33, the results for different cases simulated as per table 7.5 are plotted. In this figure, we can observe that for values of phase angle $\varphi > 90$ degrees the propulsive efficiency is enhanced reaching a maximum approximately at $\varphi = 100$ degrees and then it starts to decrease. For values greater than approximately 110 degrees, the propulsive efficiency is less than that generated at $\varphi = 90$ degrees. In this case, the best operating range is for phase angle values between $90 < \varphi < 110$ degrees.

In figure 7.34 we illustrate two different cases, one corresponding to a phase angle value equal to $\varphi = 100$ degrees and a second one corresponding to $\varphi = 90$ degrees, in this figure it can be observe that the phase angle has an effect on the LEV generation and shedding mechanism, clearly this influence the aerodynamic performance.

Case number	α_a (in degrees)	h_a	$\varphi(^\circ)$	St
2DF4-1	30.0	0.5	80.0	0.25
2DF4-2	30.0	0.5	85.0	0.25
2DF4-3	30.0	0.5	90.0	0.25
2DF4-4	30.0	0.5	95.0	0.25
2DF4-5	30.0	0.5	100.0	0.25
2DF4-6	30.0	0.5	105.0	0.25
2DF4-7	30.0	0.5	110.0	0.25
2DF4-8	30.0	0.5	115.0	0.25

Table 7.5: *Flapping parameters for the study of the effect of phase angle φ on the aerodynamic performance.*

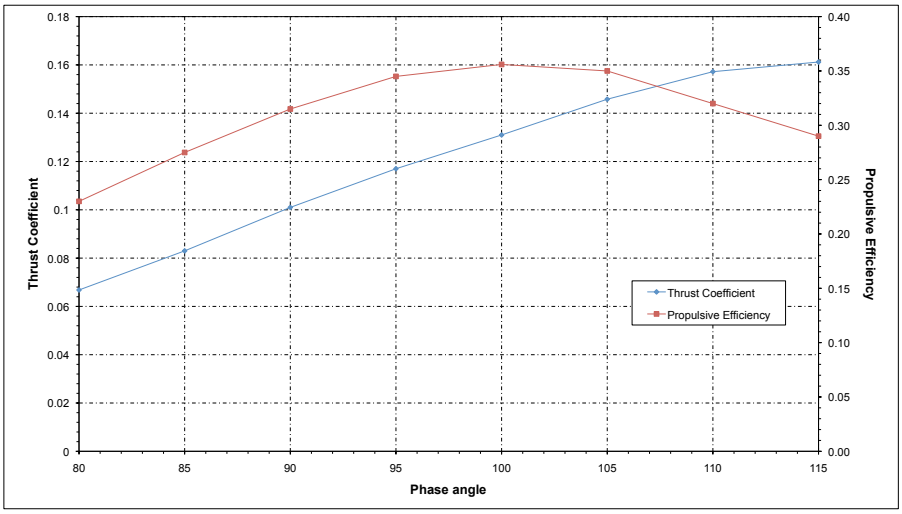

Figure 7.33: *Propulsive efficiency and average thrust coefficient in function of the phase angle φ. Flapping parameters: $Re = 1100$, $\alpha_a = 30.0$, $h_a = 0.5$, $St = 0.25$.*

7.4.5 Propulsive Efficiency, Thrust Coefficient and Input Power Coefficient Contour Maps

From the previous parametric studies, we can draw the following observations for optimal aerodynamic performance (in terms of propulsive efficiency, thrust generation and input power requirements):

- As for the pure heaving case, the best Strouhal number operating range is between $0.3 < St < 0.4$.
- The best operating pitching angle is between $20 < \alpha_a < 30$ degrees.
- Highest efficiency values with reasonable thrust coefficients are obtained for large amplitude values approximately between $0.8 < h_a < 1.2$.
- High propulsive efficiency accompanied by a high thrust coefficient of the order of one was obtained for $St = 0.4$.

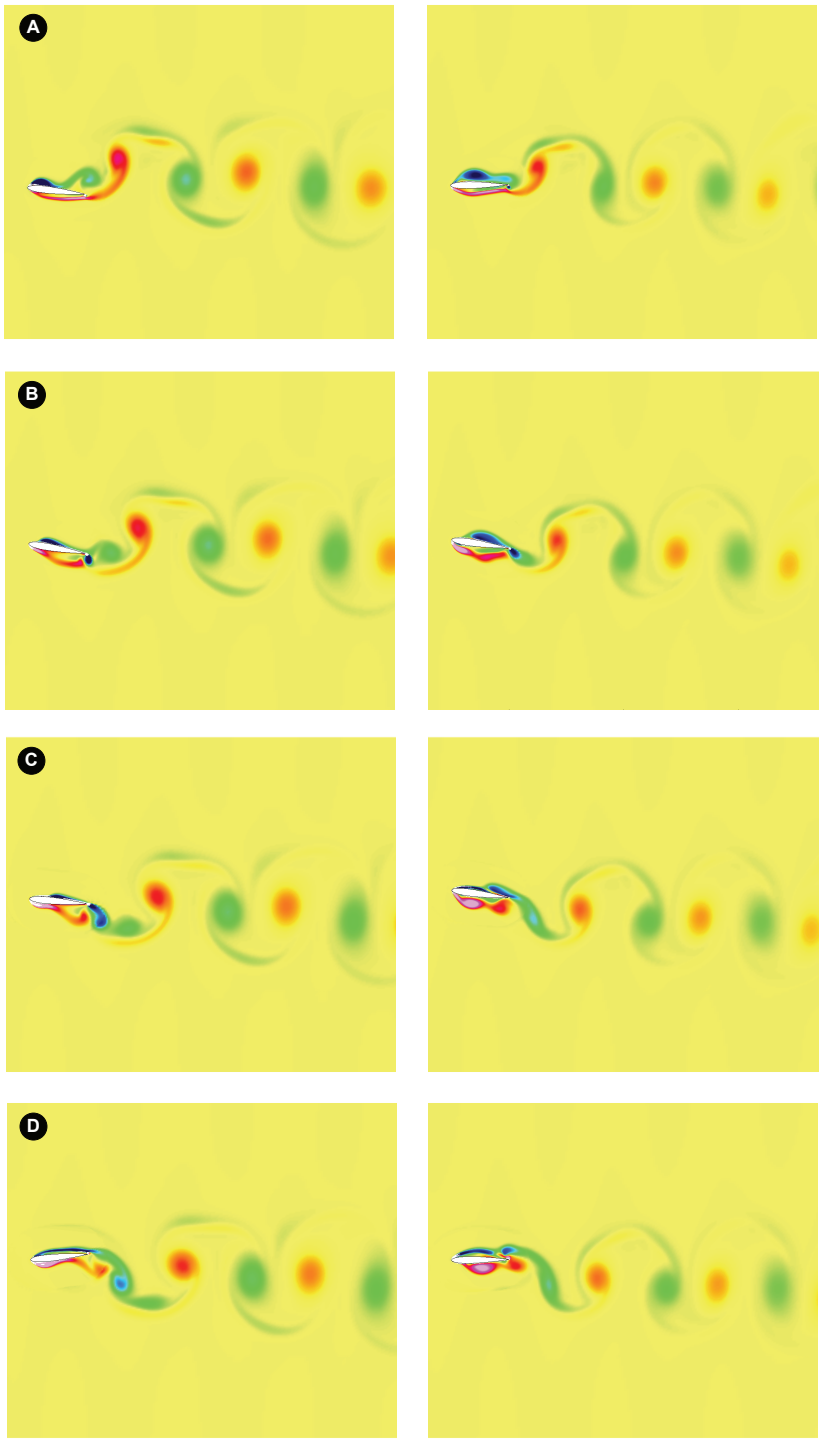

Figure 7.34: *Comparison of the vorticity field for two different flapping cases during upstroke. Left column: flapping airfoil with a phase angle equal to* $\varphi = 100°$*. Right column: flapping airfoil with a phase angle equal to* $\varphi = 90°$*. Flapping parameters:* $Re = 1100$*,* $St = 0.25$*,* $h_a = 0.25$*,* $\alpha_a = 10°$*. The sequence is shown for four instants during the upstroke motion, where: A)* $t = 8.0$ *B) t=8.35 C) t=8.70 D)* $t = 9.0$

- The maximum measured propulsive efficiency in the present set of numerical experiments was 64 percent.
- Phase angles in the range $90 < \varphi < 110$ give best propulsive efficiency without sacrificing thrust generation.

All the previous results are better summarized by plotting the solutions on contour maps for the propulsive efficiency, thrust coefficient and input power coefficient (nearly 80 simulation for each case presented). In figures 7.35, 7.36 and 7.37, we present the propulsive efficiency, thrust coefficient and input power coefficient contour maps for $St = 0.2$ as function of the heaving amplitude h_a and the maximum pitching angle α_a. In figure 7.35 we observe that at approximately $\alpha_a = 25$ degrees, the graph is divided in two parts. For values greater than $\alpha_a = 25$ degrees, little drag or no thrust is produced, while for values less than $\alpha_a = 25$ degrees little thrust is produce, this can be observed by looking at figure 7.36, where the thrust coefficient contour map is plotted. Overall, this case corresponds to a neutral wake.

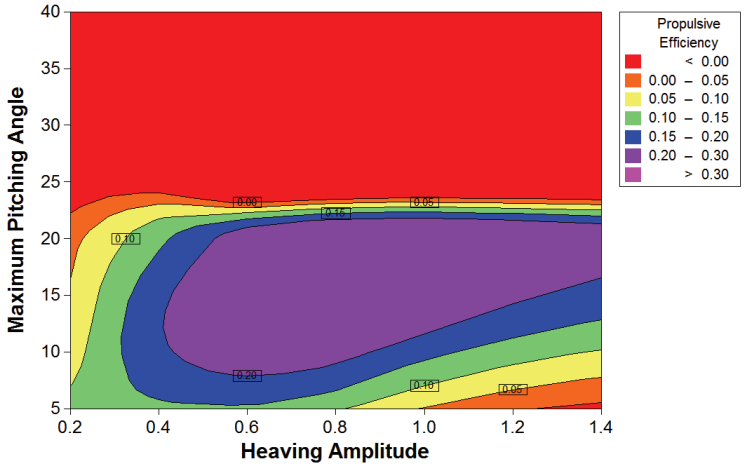

Figure 7.35: *Contour map of propulsive efficiency vs. maximum pitching angle and heaving amplitude. Flapping parameters:* $Re = 1100$, $St = 0.2$

In figures 7.38, 7.39 and 7.40, we present the propulsive efficiency, thrust coefficient and input power coefficient contour maps for $St = 0.3$ as function of the heaving amplitude h_a and the maximum pitching angle α_a. By looking at figure 7.38 and comparing with figure 7.35, the efficiency rapidly increases from relatively low values to relatively high values as the Strouhal number is increased. In this figure, the maximum efficiency values are achieved at heaving amplitudes between $0.8 < h_a < 1.4$ and maximum pitching angles between $25 < \alpha_a < 35$ degrees. In figure 7.40, we observe that the input power coefficient is very high for low amplitudes and maximum pitching angle α_a above 15 degrees. This behavior is expected to occur at low amplitudes, due to the fact that a lot of vorticity is produced near the leading edge and is converted into high input power requirements. By inspecting figure 7.39, we notice that this is a thrust producing case in almost all the range of heaving amplitudes and maximum pitching angles simulated.

In figures 7.41, 7.42 and 7.43, the propulsive efficiency, thrust coefficient and input power coefficient contour maps for $St = 0.4$ as function of the heaving amplitude h_a and the maximum

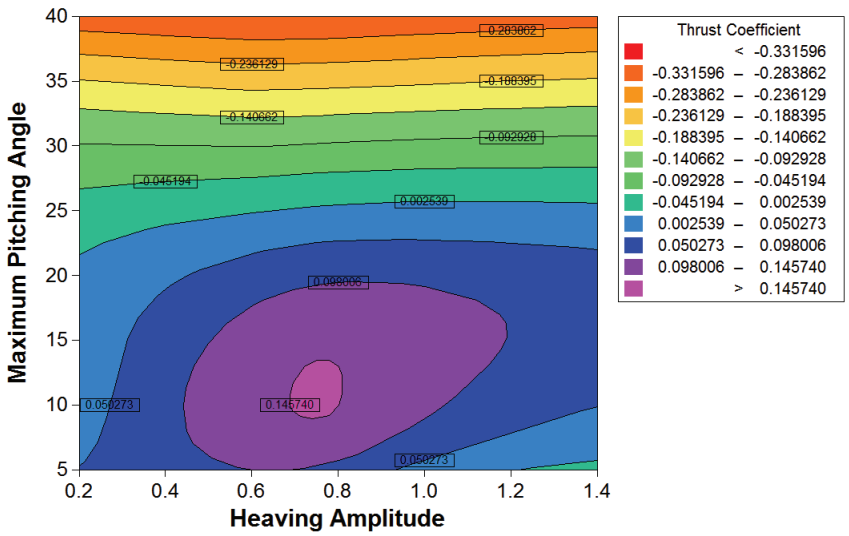

Figure 7.36: *Contour map of thrust coefficient vs. maximum pitching angle and heaving amplitude. Flapping parameters:* $Re = 1100$, $St = 0.2$.

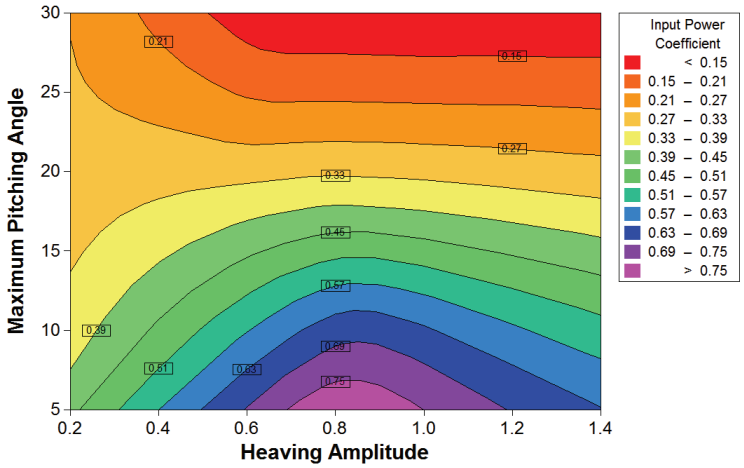

Figure 7.37: *Contour map of input power coefficient vs. maximum pitching angle and heaving amplitude. Flapping parameters:* $Re = 1100$, $St = 0.2$.

pitching angle α_a are plotted. From figure 7.41, we observe that the maximum efficiency values are achieved at high heaving amplitudes between $1.0 < h_a < 1.4$ and maximum pitching angles between $25 < \alpha_a < 40$ degrees. Looking at figure 7.42, we can also identify two *"high thrust"* areas, one corresponding to *"low efficiency"* (for values of heaving amplitude between $0.4 < h_a < 0.6$) and the second one corresponding to *"high efficiency"* (for values of heaving amplitude between $1.0 < h_a < 1.4$). As for the previous case ($St = 0.3$), we observe high input power coefficient values at low amplitudes (see figure 7.43). By examining figure 7.42, it can be inferred that this is a thrust producing case in almost all the range of heaving amplitudes and maximum pitching angles simulated.

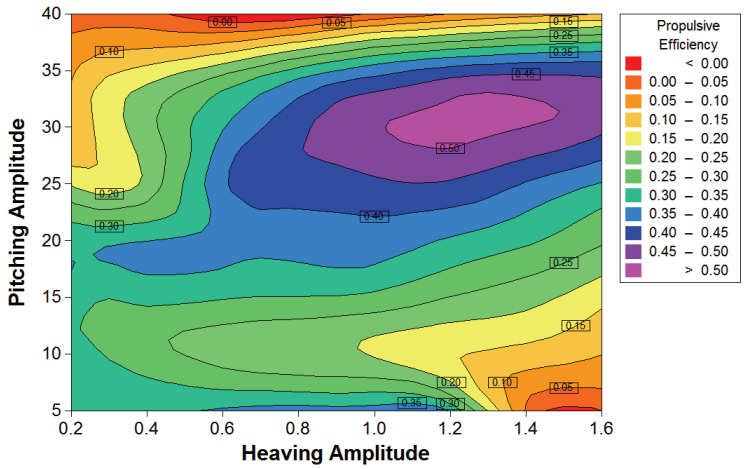

Figure 7.38: *Contour map of propulsive efficiency vs. maximum pitching angle and heaving amplitude. Flapping parameters:* $Re = 1100$, $St = 0.3$

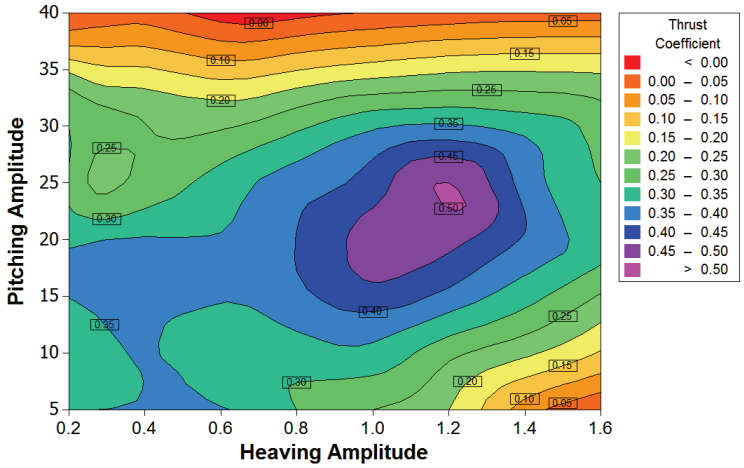

Figure 7.39: *Contour map of thrust coefficient vs. maximum pitching angle and heaving amplitude. Flapping parameters:* $Re = 1100$, $St = 0.3$.

In figures 7.44 and 7.45, we present the vorticity field for several flapping cases. In figure 7.44, where $St = 0.2$, we can observe that the wake topology does not shows the typical mushroom-shaped structure for typical thrust production regimes; in fact, the cases shown produce drag or little thrust. It can be also observed that as we increase the maximum pitching angle, the strength of the LEV is lowered. In figure 7.45 (corresponding to $St = 0.4$), we can now identify the typical mushroom structure for thrust producing wakes; as in the previous case, as we increase the maximum pitching angle the strength of the LEV is weakened and this translates into lower input power requirements.

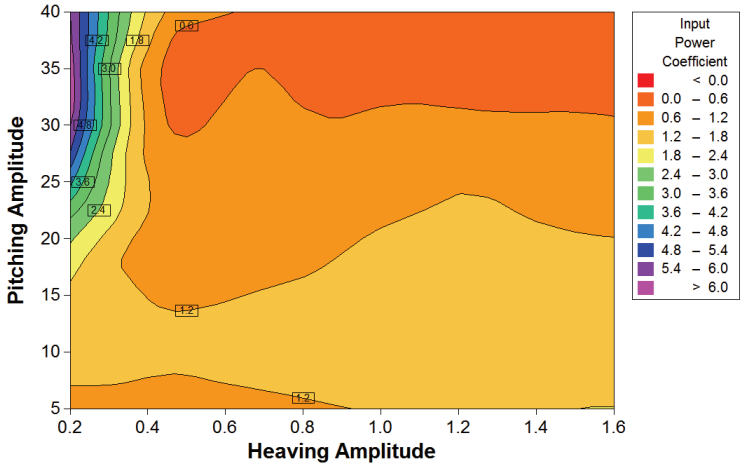

Figure 7.40: *Contour map of input power coefficient vs. maximum pitching angle and heaving amplitude. Flapping parameters:* $Re = 1100$, $St = 0.3$.

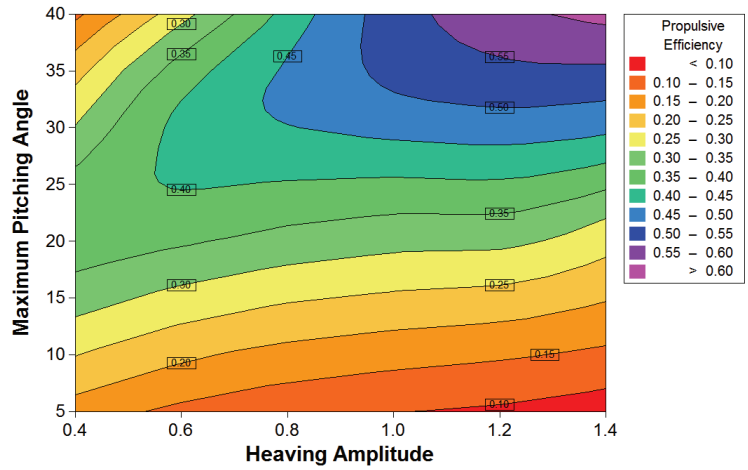

Figure 7.41: *Contour map of propulsive efficiency vs. maximum pitching angle and heaving amplitude. Flapping parameters:* $Re = 1100$, $St = 0.4$

7.5 Heaving Airfoil Vs. Flapping Airfoil

In the previous sections, we studied heaving and flapping airfoils aerodynamic performance. From the results obtained, it is evident that flapping airfoils provide in general better aerodynamic performance than purely heaving airfoils. In figures 7.46, 7.47 and 7.48, we compare the aerodynamic performance for a heaving airfoil and a flapping airfoil. The cases are compared for a Strouhal number value equal to $St = 0.3$.

As it can be seen in figure 7.46, for heaving amplitudes less than 0.2, the pure heaving motion provides better propulsive efficiency than the flapping motion. On the other hand, for values higher than 0.2 the flapping motion gives the best propulsive efficiency, with its value increasing very fast with the heaving amplitude. This behavior is expected to occur for flapping airfoils

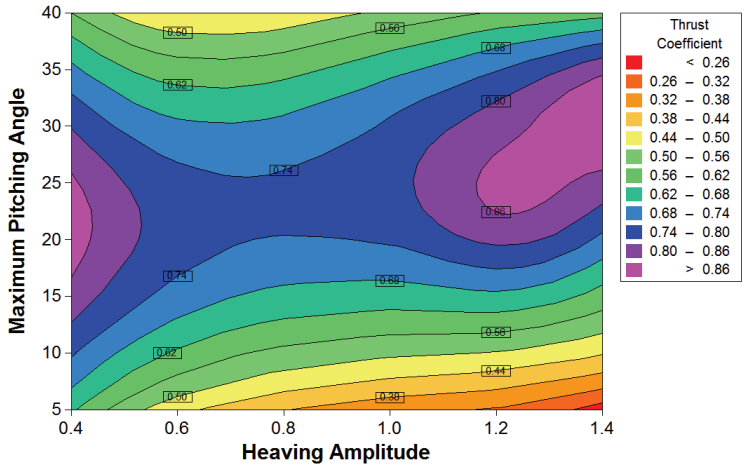

Figure 7.42: *Contour map of thrust coefficient vs. maximum pitching angle and heaving amplitude. Flapping parameters:* $Re = 1100$, $St = 0.4$.

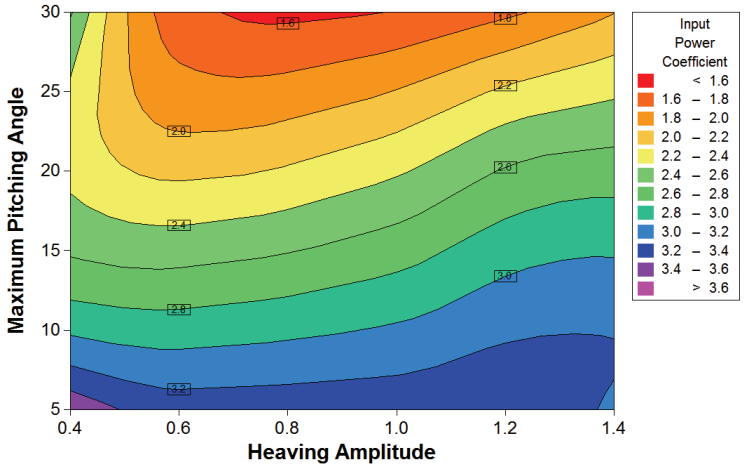

Figure 7.43: *Contour map of input power coefficient vs. maximum pitching angle and heaving amplitude. Flapping parameters:* $Re = 1100$, $St = 0.4$.

heaving with low heaving amplitudes, due to the fact that vorticity is produced near the leading edge and converted into high input power coefficients, as seen in figure 7.48. Commenting on the thrust coefficient (figure 7.47), we can observe that for both kinds of kinematics the maximum thrust coefficient is almost identical when operating at high efficiency values, but with the difference that for the flapping case, the high thrust coefficient values are maintained over a wider range of heaving amplitudes.

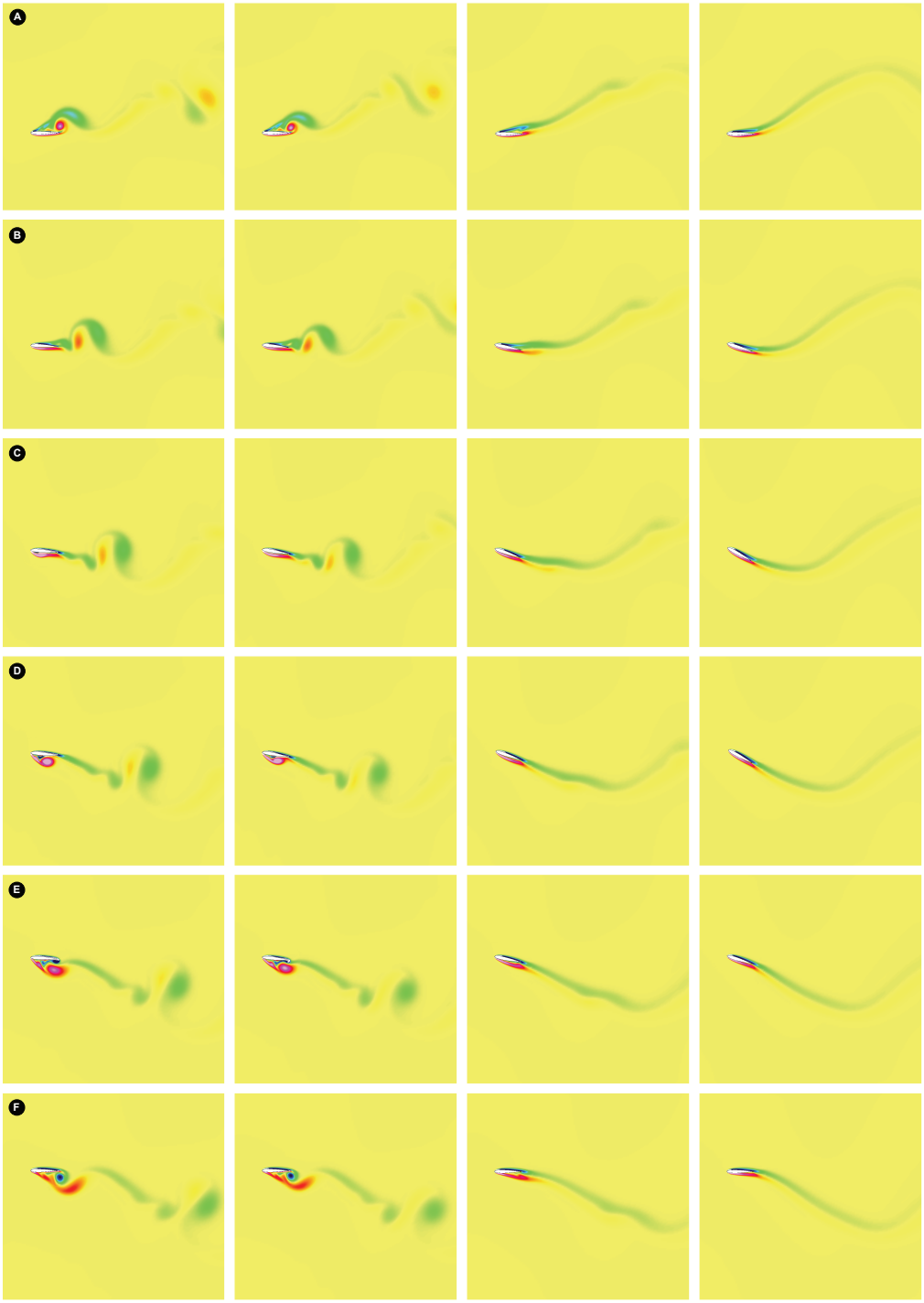

Figure 7.44: *Comparison of the vorticity field for four different flapping cases during upstroke. First column:* $Re = 1100$, $St = 0.2$, $h_a = 1.0$, $\alpha_a = 5°$, $\varphi = 90°$. *Second column:* $Re = 1100$, $St = 0.2$, $h_a = 1.0$, $\alpha_a = 10°$, $\varphi = 90°$. *Third column:* $Re = 1100$, $St = 0.2$, $h_a = 1.0$, $\alpha_a = 20°$, $\varphi = 90°$. *Fourth column:* $Re = 1100$, $St = 0.2$, $h_a = 1.0$, $\alpha_a = 30°$, $\varphi = 90°$. *The sequence is shown for six instants during the upstroke motion, where: A)* $t = 30.0$ *B)* $t = 31.0$ *C)* $t = 32.0$ *D)* $t = 33.0$ *E)* $t = 34.0$ *F)* $t = 35.0$.

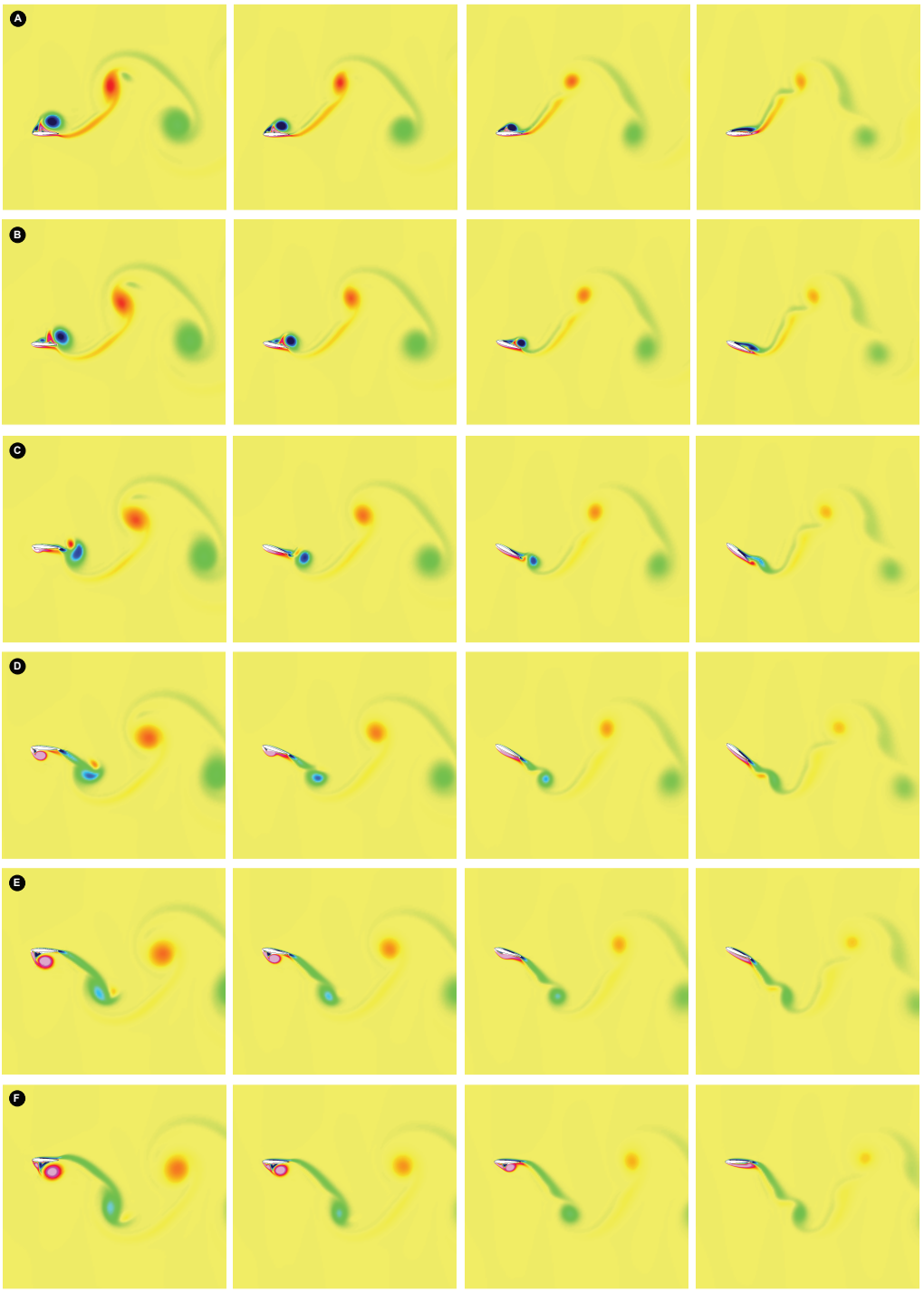

Figure 7.45: *Comparison of the vorticity field for four different flapping cases during upstroke. First column:* $Re = 1100$, $St = 0.4$, $h_a = 1.0$, $\alpha_a = 5°$, $\varphi = 90°$. *Second column:* $Re = 1100$, $St = 0.4$, $h_a = 1.0$, $\alpha_a = 20°$, $\varphi = 90°$. *Third column:* $Re = 1100$, $St = 0.4$, $h_a = 1.0$, $\alpha_a = 30°$, $\varphi = 90°$. *Fourth column:* $Re = 1100$, $St = 0.4$, $h_a = 1.0$, $\alpha_a = 40°$, $\varphi = 90°$. *The sequence is shown for six instants during the upstroke motion, where: A)* $t = 20.0$ *B)* $t = 20.5$ *C)* $t = 21.0$ *D)* $t = 21.5$ *E)* $t = 22.0$ *F)* $t = 22.5$.

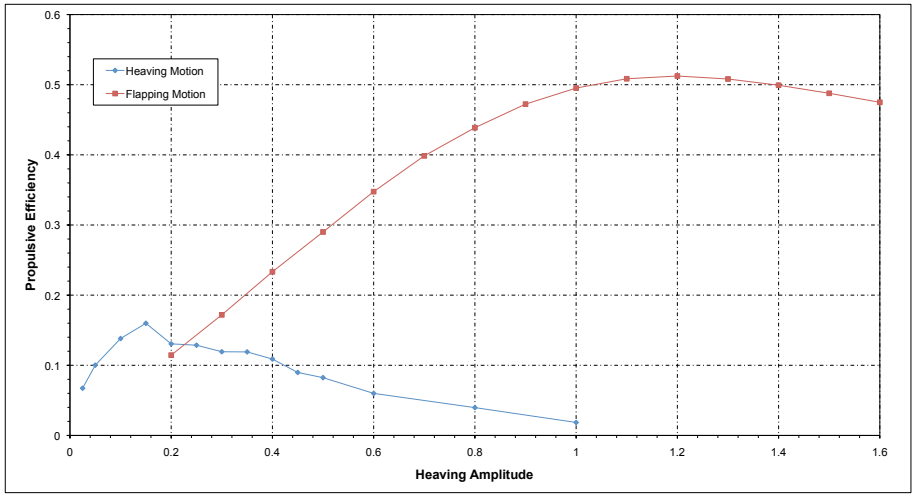

Figure 7.46: *Heaving and flapping motions propulsive efficiency comparison. Flapping parameters:* $Re = 1100$, $St = 0.3$, $\alpha_a = 30.0$.

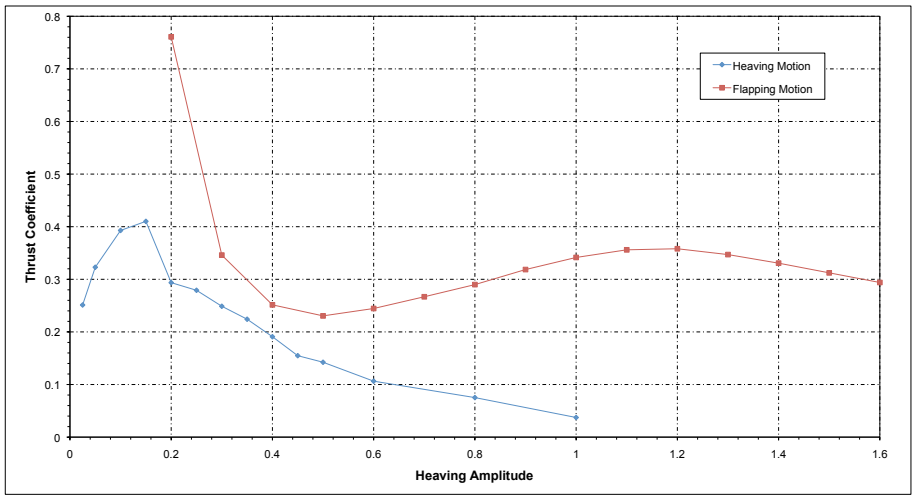

Figure 7.47: *Heaving and flapping motions thrust coefficient comparison. Flapping parameters:* $Re = 1100$, $St = 0.3$, $\alpha_a = 30.0$.

7.6 Effect of Flexibility on the Aerodynamic Performance

Despite the growing interest in modeling and understanding the physics of flexible flapping wings, this field of study remains largely unexplored. In this section, we briefly investigate the effect of chord-wise flexure amplitude on the aerodynamic performance of a NACA 0012 airfoil undergoing pure heaving motion. Here we impose the airfoil deformation as follows

$$y = \frac{h_{flex}}{c} x^2 cos\left(2\pi f_{flex} t + \varphi\right), \tag{7.3}$$

as proposed by Miao and Ho [124]. In eq. 7.3, h_{flex} is the flexure amplitude, c the airfoil chord, x are the airfoil horizontal coordinates, y are the airfoil new vertical coordinates, f_{flex} is the

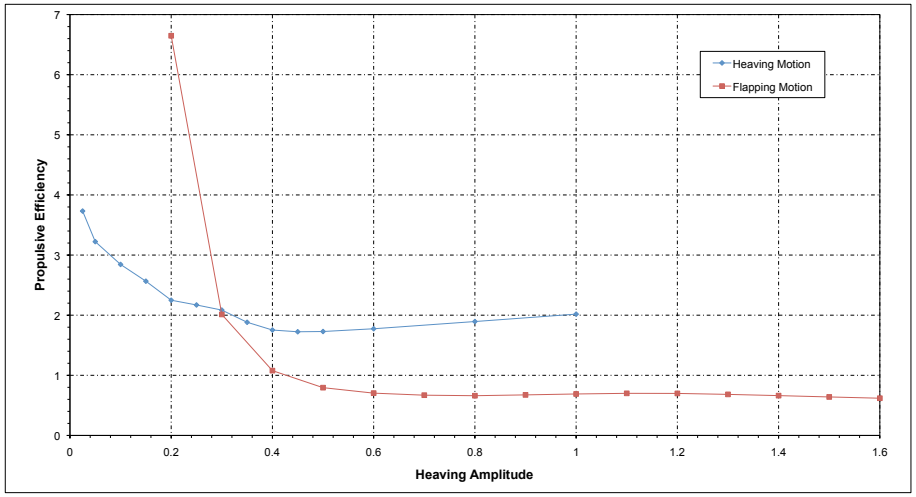

Figure 7.48: *Heaving and flapping motions input power coefficient comparison. Flapping parameters:* $Re = 1100$, $St = 0.3$, $\alpha_a = 30.0$.

flexure frequency (which is assumed to be the same as the heaving frequency) and φ is the phase angle. In order to investigate the effect of the flexure amplitude on the aerodynamic performance of heaving wings, we consider six different flexure amplitudes h_{flex}, as shown in table 7.6. The value $h_{flex} = 0.0$ corresponds to the rigid airfoil case.

Case number	h_{flex}	$\varphi(^\circ)$	h_a	St
2DF5-1	0.1	90.0	0.25	0.3
2DF5-2	0.2	90.0	0.25	0.3
2DF5-3	0.3	90.0	0.25	0.3
2DF5-4	0.4	90.0	0.25	0.3
2DF5-5	0.5	90.0	0.25	0.3
2DF5-6	0.6	90.0	0.25	0.3

Table 7.6: *Flapping parameters for the study of the flexible heaving airfoil.*

Figure 7.49, shows the variation of propulsive efficiency and average thrust coefficient with respect to the flexure amplitude. Comparing the results with the rigid body case, an enhancement in the propulsive efficiency and thrust coefficient is observed, until a value of $h_{flex} = 0.4$ from which point on the propulsive efficiency starts to decrease. This observation is in agreement with the results of Heatcote and Gursul [67, 68], where they found that adding a degree of flexibility increase both thrust and propulsive efficiency. To explain this improvement in the aerodynamics performance, let us suppose that the airfoil deformation can be approximated as an imposed pitch angle, and as studied in previous section; if we increase the maximum pitching angle during flapping motion, the aerodynamic performance is enhanced. In figure 7.50 we illustrate a deforming case, where we set the maximum flexure amplitude to $h_{flex} = 0.3$, the heaving amplitude to $h_a = 0.25$ and the Strouhal number to $St = 0.3$. For purposes of comparison, we also plot in figure 7.51 the equivalent rigid airfoil case. In figure 7.50, it can be observed that the airfoil deformation helps to convect the LEV more smoothly from the leading edge to the trailing edge; furthermore, the

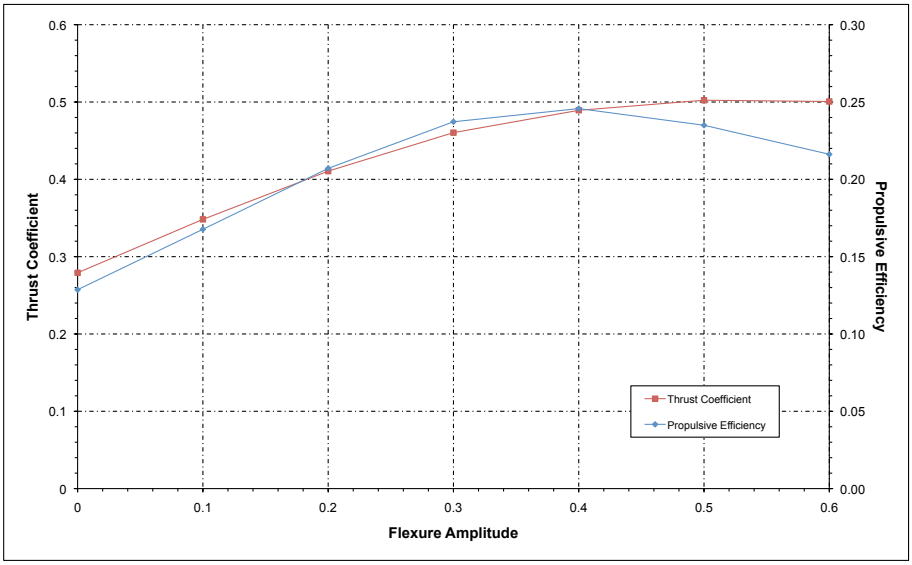

Figure 7.49: *Comparison of propulsive efficiency and average thrust coefficient versus flexure amplitude. Flapping parameters:* $Re = 1100$, $St = 0.3$, $h_a = 0.25$.

strength of the LEV is less than in the equivalent rigid case.

7.7 Effect of Airfoil Cambering on the Aerodynamic Performance

The effect of airfoil cambering on the aerodynamic performance of heaving or flapping airfoils still remains virtually unexplored, even though it is crucial on lift generation. Bird wings actually have camber due to the need of lift generation. The fact that this area have not been sufficiently studied, is related to the fact that most researchers have used symmetrical airfoils in their investigations [62, 89, 94, 96, 108, 140, 151, 197, 207, 216], as they are chiefly interested on thrust generation and propulsion efficiency.

In this section, we conduct a parametric study in order to assess the effect of airfoil cambering on the aerodynamic performance (in particular the lift coefficient c_l) of heaving airfoils. The airfoils used for this study are listed in table 7.7; basically, we used a standard NACA four digits series airfoil, where we simply change the maximum airfoil cambering and its position. Additionally, we also use the high lift low Reynolds number Selig S1223 airfoil. This latter airfoil was chosen because of its similarities to the Seagull and Merganser wings cross-section, as suggested by Liu *et al.* [114]. In table 7.8, the parameters governing the heaving motion are shown. Here, two heaving amplitudes are used, one corresponding to high heaving frequencies and one corresponding to low heaving frequency.

The summary of results is presented in tabular form in tables 7.9 and 7.10, where $\overline{c_t}$ is the average thrust coefficient, $\overline{c_P}$ is the average input power coefficient, $\overline{c_l}$ is the average lift coefficient, η is the propulsive efficiency and $\hat{c}_l$ is the maximum lift coefficient. Inspecting table 7.9 and using the results of the NACA 0012 airfoil as a reference, we observe that the values of $\overline{c_t}$, $\overline{c_P}$, η and $\hat{c}_l$ do not change much as the maximum airfoil cambering and its position are modified; conversely, looking at the values of $\overline{c_l}$, we observe that its values increase as we change the maximum airfoil

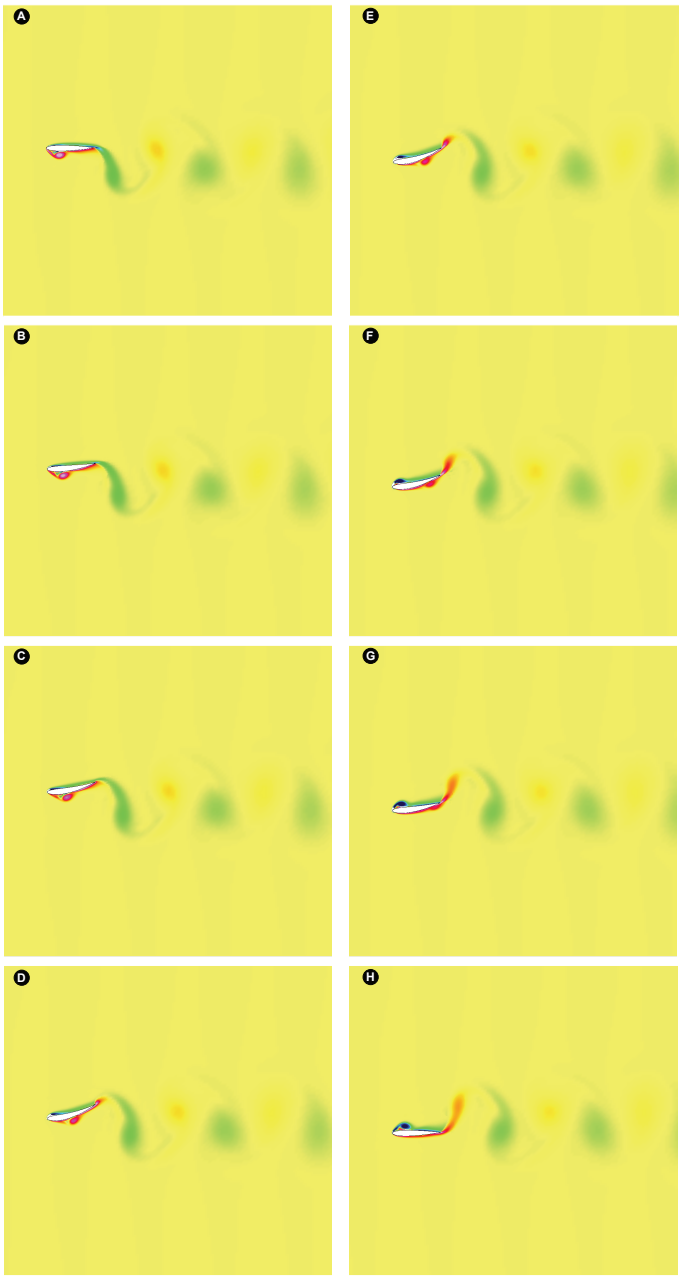

Figure 7.50: *Vorticity field for the flexible airfoil study. The sequence is shown for eight instants during the downstroke motion, where: A) t = 8.33 B) t= 8.45 C) t= 8.55 D) t = 8.70 E) t = 8.80 F) t=8.90 G) t=9.0 H) t=9.16. Flapping parameters:* $Re = 1100$, $h_{flex} = 0.3$, $h_a = 0.25$, $St = 0.3$.

cambering and its position, in fact, we are now producing a positive average lift coefficient $\overline{c_l}$.

In table 7.10, the results for a case similar to the previous one but with a heaving amplitude equal to $h_a = 0.3$ and a heaving frequency equal to $f_h = 0.666667$ are presented. In this table, the same observations as for the previous case hold. Looking closely at the aerodynamic quantities for the S1223 airfoil, we notice that the aerodynamic performance of this airfoil is not close to that of the other airfoils; nevertheless, it still produces thrust and lift.

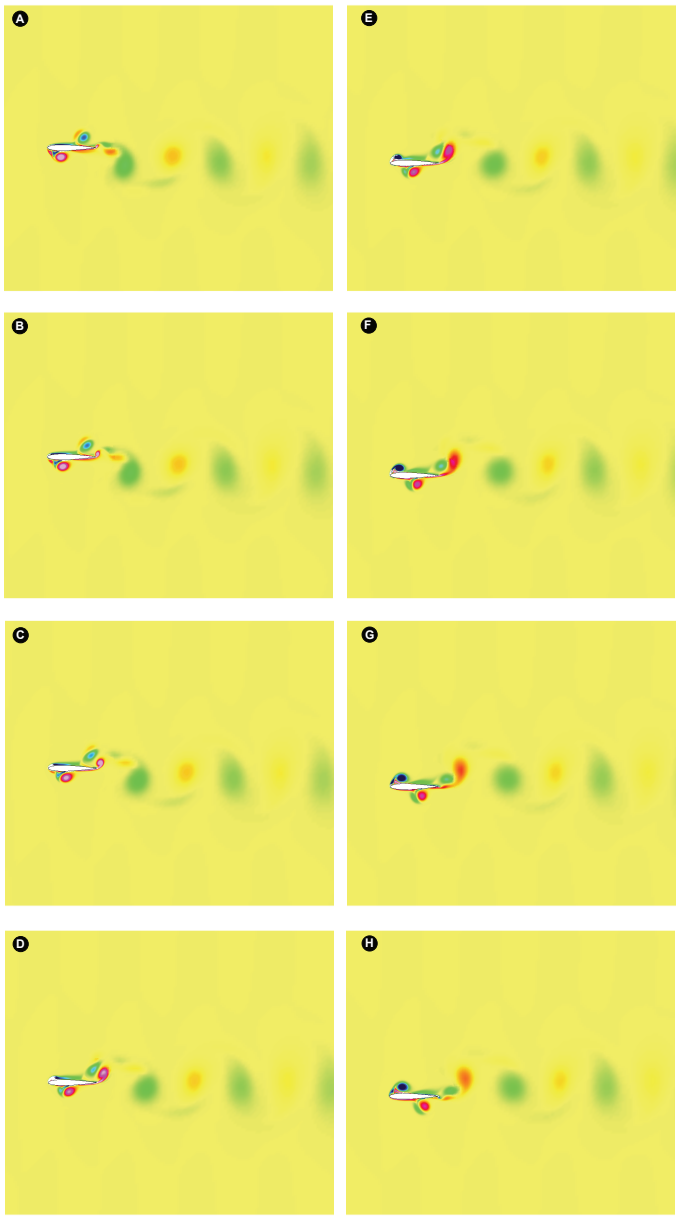

Figure 7.51: *Vorticity field for the flexible airfoil study (rigid airfoil case). The sequence is shown for eight instants during the downstroke motion, where: A) t = 8.33 B) t= 8.45 C) t= 8.55 D) t = 8.70 E) t = 8.80 F) t=8.90 G) t=9.0 H) t=9.16. Flapping parameters:* $Re = 1100$, $h_{flex} = 0.0$, $h_a = 0.25$, $St = 0.3$.

From the previous results and having in mind that if we are interested in flying we need to produce lift, the use of cambering in flapping flight is favorable and it does not cause detriment in the aerodynamic performance. In figure 7.52, a comparison of the vorticity field for a NACA 0012 and a Selig S1223 airfoil is presented. Notice that the vorticity field for the S1223 airfoil is no more symmetric, hence the strength and shedding of the vortices during the upstroke and downstroke are different. The sequence in figure 7.52 is shown for four instants during the upstroke motion. For completeness, in figure 7.53 we present the vorticity field for a NACA 2212 airfoil

7.7. EFFECT OF AIRFOIL CAMBERING ON THE AERODYNAMIC PERFORMANCE

Airfoil Type
NACA 0012
NACA 2212
NACA 2412
NACA 4412
NACA 2612
NACA 4612
NACA 6612
Selig S1223

Table 7.7: *Airfoils used for the study of cambering effect on the aerodynamic performance of heaving airfoils.*

Case number	h_a	f_h	$\varphi(^\circ)$	St
2DF5-1	0.1	1.5	90.0	0.4
2DF5-2	0.3	0.5	90.0	0.4

Table 7.8: *Heaving parameters for the study of airfoil cambering effect on the aerodynamic performance.*

Airfoil type	$\overline{c_t}$	$\overline{c_P}$	η	$\overline{c_l}$	$\hat{c}_l$
NACA 0012	0.8515	7.5748	0.1124	0.009397	28.1761
NACA 2212	0.9326	7.6910	0.1212	0.028386	27.0256
NACA 2412	0.9329	7.7209	0.1208	0.1120	27.1245
NACA 4412	0.9080	7.6433	0.1188	0.4465	27.4484
NACA 2612	0.9286	7.7304	0.1201	0.2475	28.5417
NACA 4612	0.8920	7.6356	0.1168	0.7074	28.8189
NACA 6612	0.8664	7.5440	0.1148	1.1194	28.2583
Selig S1223	0.7944	7.2373	0.1097	1.4181	29.5885

Table 7.9: *Comparison of the aerodynamic performance of eight different airfoils for the study of cambering effect on the aerodynamic behavior. Flapping parameters:* $Re = 1100$, $St = 0.4$, $h_a = 0.1$, $f_h = 2.0$

Airfoil type	$\overline{c_t}$	$\overline{c_P}$	η	$\overline{c_l}$	$\hat{c}_l$
NACA 0012	0.8292	5.3332	0.1554	0.0144	11.3097
NACA 2212	0.8244	5.3221	0.1549	0.2279	11.0896
NACA 2412	0.8486	5.3313	0.1591	0.1074	10.4898
NACA 4412	0.8035	5.2321	0.1535	0.1584	10.7599
NACA 2612	0.8340	5.3293	0.1565	0.3194	11.4170
NACA 4612	0.8067	5.2851	0.1526	0.3963	11.5470
NACA 6612	0.7627	5.1025	0.1494	0.4675	12.7073
Selig S1223	0.4284	4.4070	0.0972	0.2658	11.5889

Table 7.10: *Comparison of the aerodynamic performance of eight different airfoils for the study of cambering effect on the aerodynamic behavior. Flapping parameters:* $Re = 1100$, $St = 0.4$, $h_a = 0.3$, $f_h = 2.0$

and a NACA 4612 airfoil under the same previous flapping conditions. Notice in figure 7.53 that as we added camber, the vorticity shedding is no longer symmetric, also, by examining the wake we can observe that the added airfoil cambering induce a small upward deflection on the wake.

7.8 Summary

In this chapter, we studied the dependency of the wake structure and aerodynamic performance on the flapping and geometric parameters such as flapping frequency, flapping amplitude, airfoil geometry and airfoil chord-wise flexibility (among others), for airfoils undergoing pure heaving motion or flapping motion.

It was found that the Strouhal number (based on the heaving amplitude h_a) seems to be enough for wake signature characterization, but is not sufficient insofar as maximum efficiency is concerned. Both heaving amplitude h_a and heaving frequency f_h (hence the Strouhal number St and the reduced frequency k), should be adjusted separately.

All the qualitative and quantitative results agree with the hypothesis that : *"flying and swimming animals cruise at a Strouhal number tuned for high power efficiency"* [182]. The enhanced efficiency range was found to be between Strouhal number values corresponding to $0.2 < St < 0.4$, which agrees with the observations of Taylor *et al.* [182], Triantafyllou *et al.* [193] and Nudds *et al.* [136].

It was also observed different behaviors on the aerodynamic performance for high flapping frequencies (low heaving amplitudes) and low flapping frequencies (high heaving amplitudes). Firstly, at high flapping frequencies f the LEV does not have sufficient time to grow, whereas at low flapping frequencies f the vortex can become a sizable fraction of the airfoil chord before separating. Thus the impact of the vortex on the pressure at the nose of the airfoil is dependent on the flapping frequency. Secondly, once the vortex separates it is convected downstream over the surface of the airfoil. Due to the low pressure in the vortex core this has the effect of maintaining thrust while the vortex is upstream of the airfoil maximum thickness point (where the airfoil surface is tilted upstream and the vortex low pressure creates an upstream suction force). Once passing this point, the airfoil surface is tilted downstream and the vortex contributes to drag rather than thrust. At high flapping frequencies, the vortex cannot be convected far downstream before the motion cycle creates another leading edge on the opposite side of the airfoil, so the impact is lessened. At low flapping frequencies f however, the vortex travels far downstream over the airfoil surface causing drag for a larger portion of the flapping cycle and therefore lowering the propulsive efficiency.

For flapping motion, where the orientation of the airfoil surface is controlled by the relative amplitudes and phases of the motion, the LEV may create positive thrust for much longer portions of the flapping cycle and thus contribute towards the propulsive efficiency. Also, for flapping motion the average input power coefficient is less than the average input power coefficient for the heaving motion cases, resulting in an improved propulsive efficiency. It was also observed that the best propulsive efficiencies were obtained for high heaving amplitudes (between $0.8 < h_a < 1.2$), in contrast to the heaving motion were the propulsive efficiency is degraded as the heaving amplitude is increased above a value of $h_a = 0.2$.

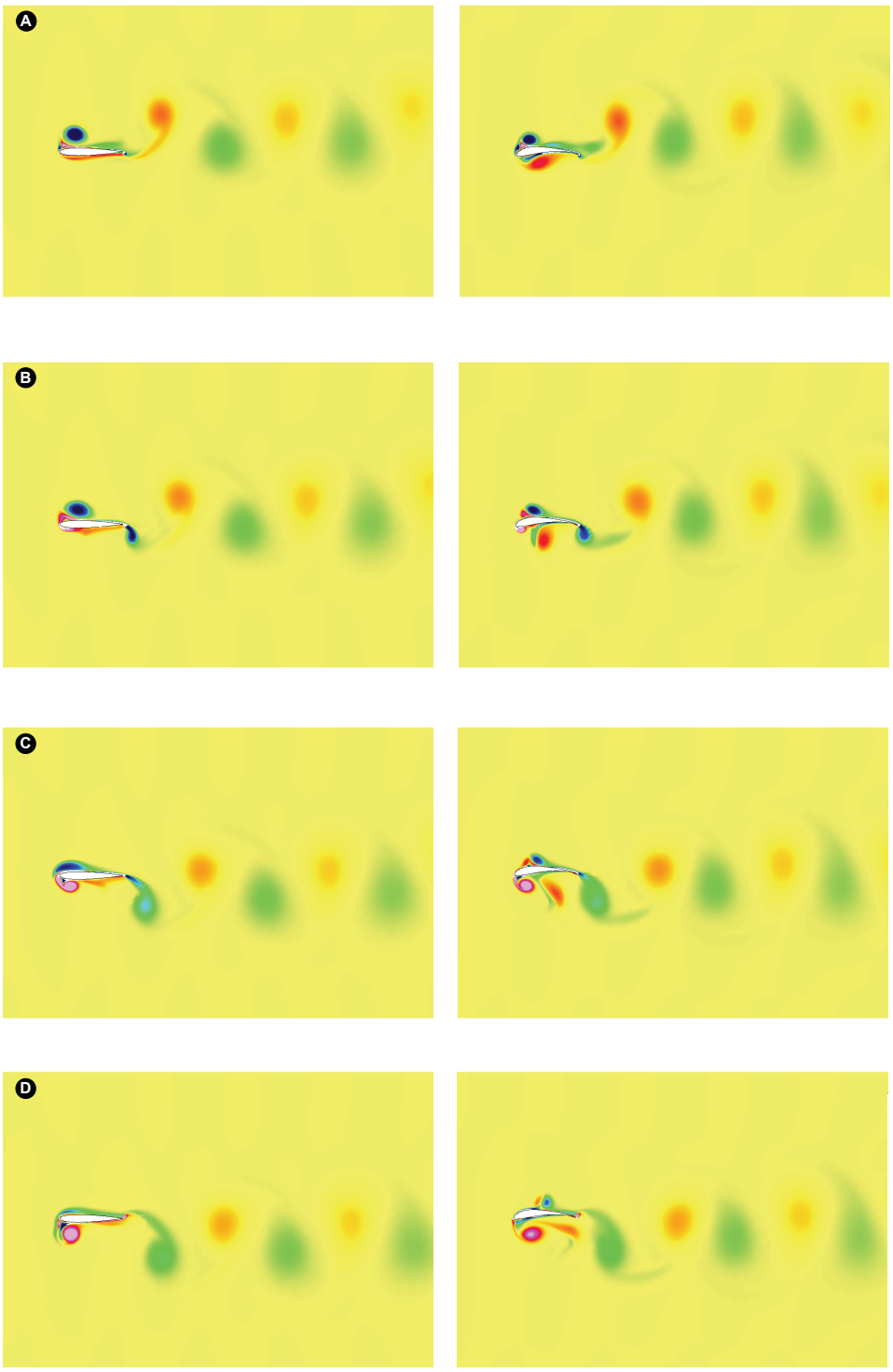

Figure 7.52: *Comparison of the vorticity field for two different airfoils. Left column: NACA 0012 airfoil. Right column: Selig S1223 airfoil. Flapping parameters:* $Re = 1100$, $St = 0.4$, $h_a = 0.3$. *The sequence is shown for four instants during the upstroke motion, where: A)* $t = 9.0$ *B) t=9.25 C) t=9.50 D)* $t = 9.75$.

In this chapter the effect of chord-wise flexibility on the aerodynamic performance of heaving airfoils was also explored. Thrust-indicative wake topologies were observed for the whole range of flexure amplitudes tested. The results also shown that the propulsive efficiency is enhanced for values of $h_{flex} < 0.4$. This observation is in agreement with the results of Heatcote and Gursul [67, 68], where they found that adding a degree of flexibility increases both thrust and propulsive efficiency. They also suggested that birds, bats and insects may benefit aerodynamically from the flexibility of their wings.

Finally, the effect of airfoil cambering on the aerodynamic performance was assessed. It was found that this geometric parameter has a strong influence on the lift coefficient, while it has a small impact on the thrust coefficient and propulsive efficiency. Among all the asymmetric airfoils used, the NACA 6612 airfoil provided the best propulsive efficiency and average lift coefficient, which along with the thrust generation are the crucial factors if we are interested in flapping flight. The S1223 airfoil, which resembles the cross-section of the Seagull and Merganser wings (as observed by Liu *et al.* [114]), provided at high heaving frequencies the biggest average lift coefficient and very similar average thrust coefficient and propulsive efficiency values when compared to the other airfoils. On the other hand, at low heaving frequencies, the aerodynamic performance of the S1223 airfoil was deteriorated in comparison to the other airfoils, while still producing thrust and positive average lift.

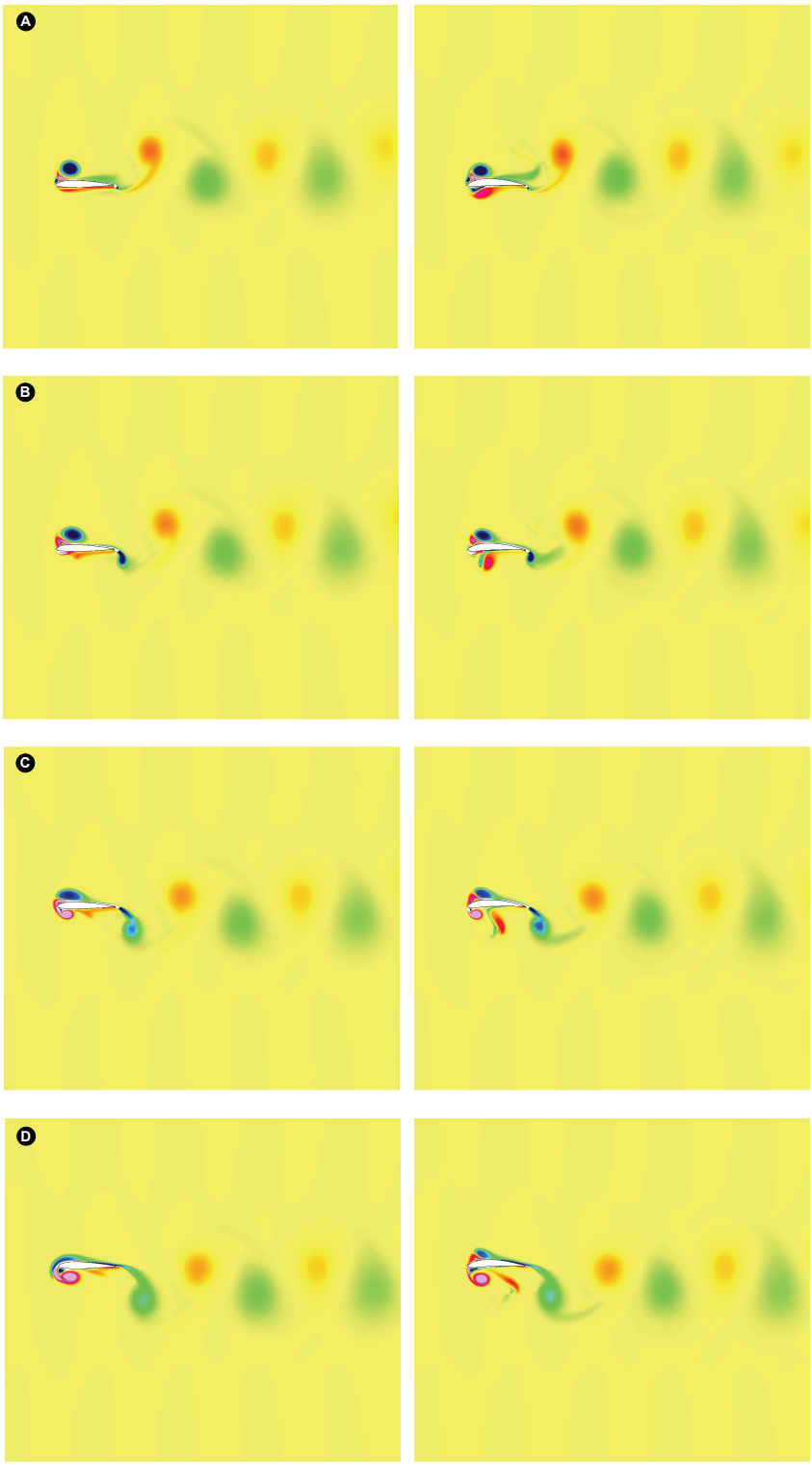

Figure 7.53: *Comparison of the vorticity field for two different airfoils. Left column: NACA 2212 airfoil. Right column: NACA 4612. Flapping parameters:* $Re = 1100$, $St = 0.4$, $h_a = 0.3$. *The sequence is shown for four instants during the upstroke motion, where: A)* $t = 9.0$ *B)* $t=9.25$ *C)* $t=9.50$ *D)* $t = 9.75$.

Chapter 8

Wake Topology and Aerodynamic Performance of Finite-Span Flapping Wings

In this chapter, we extend the two-dimensional results presented in the previous chapter to three-dimensional rigid finite-span flapping wings. We investigate the wake topology behind low aspect ratio flapping wings and their dependence on the Strouhal number and flapping parameters. We also present some results on the aerodynamic performance of flapping wings and establish the best criteria for vortical structures identification.

8.1 Overview

The three-dimensional flow features generated by finite-span wings present a great challenge due to the computational effort required to compute the complex flows generated as a function of the flapping kinematics, flapping frequency, flapping amplitude, wing geometry and Reynolds number; all these factors influence the wake topology and aerodynamic performance of finite-span flapping wings. Hereafter, we study the validity of the Strouhal number St as the fundamental aerodynamic parameter for finite-span flapping wings at low Reynolds number. We also conduct several numerical experiments in order to investigate the wake topology and aerodynamic performance of rigid finite-span wings undergoing pure heaving motion, coupled heaving-and-pitching motion and rolling motion (root-flapping).

8.2 Computational Domain and Grid Setup

The overlapping grid system used for the three-dimensional flapping wings studies, is based in the two-dimensional overlapping grid system used previously, and provides good resolution in the area around the moving wing as well as in the wake region. In figures 8.1 and 8.2 we show the overlapping grid system layout, where the background grid extends $2.5 \times c$ away from the wing leading edge, $6.5 \times c$ away from the wing trailing edge, $1.5 \times c$ away from the wing-tips and $3.0 \times c$ away from the wing upper and lower surface (where c is the mean wing chord). The previous dimensions, are the dimensions for the overlapping grid system layout when the wing is in the

mean position of the flapping motion and where h_a is assumed to be equal to 0.25.

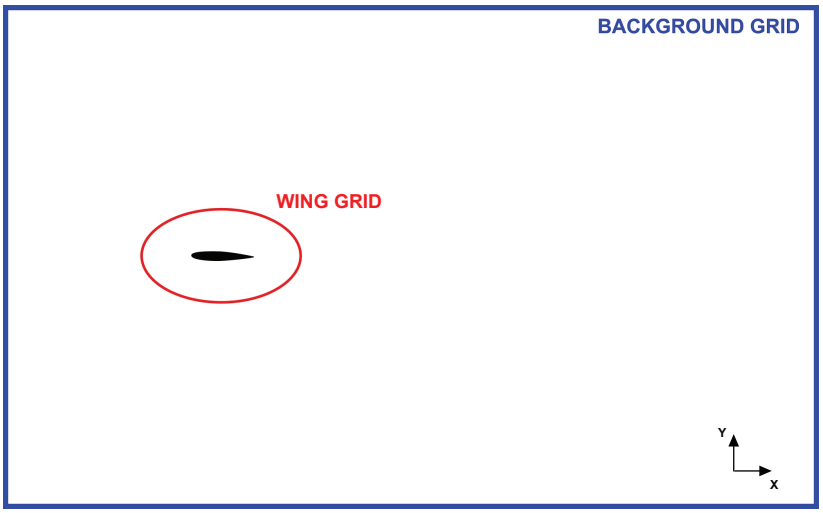

Figure 8.1: *Three-dimensional computational domain layout in the xy plane.*

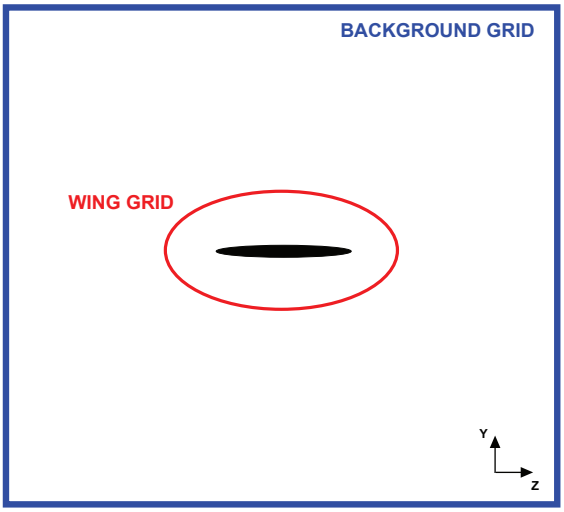

Figure 8.2: *Three-dimensional computational domain layout in the zy plane.*

The nominal grid size employed for the background grid is $160 \times 100 \times 60$ (in the x, y and z directions respectively). For the wing grid, the grid size is about $160 \times 80 \times 60$; this grid size is based on a rectangular wing of aspect ratio equal AR $= 1$ and wing chord equal to $c = 1$. In the case of a bigger or smaller domain, the grid dimensions are scaled in order to keep the same grid spacing as for this domain. The first node normal to the wing surface is placed at a distance equal to $0.0001 \times c$ (which still is in the asymptotic range of convergence as seen in Chapter 6, Section 3), and we clustered up to 14 points in the direction normal to the airfoil surface. In order to better resolve the wing wake, we use grid stretching in the x, y and z directions of the background grid (see figure 8.3). All the flapping wings calculations were started from a fully converged stationary

wing solution. In figure 8.1 the top, bottom and right boundaries are outflow boundaries and the left boundary is inflow with $(u, v, w) = (1.0, 0.0, 0.0)$, whereas in figure 8.2 all the boundaries correspond to outflow. On the wing surface we impose a no-slip boundary condition.

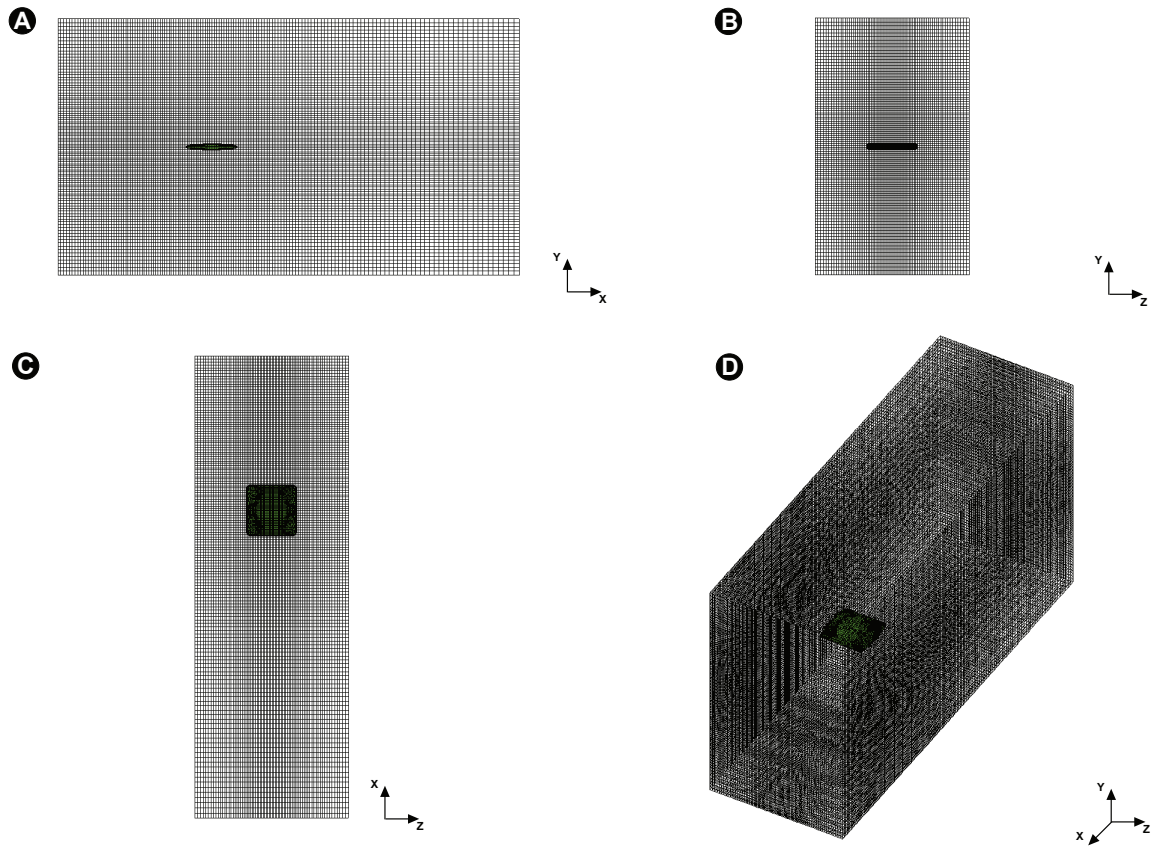

Figure 8.3: *Typical grid system employed in the current three-dimensional study. A) Side view. B) Front view. C) Top view. D) Perspective view.*

8.3 Vortex Identification

A key aspect in the study of the wake topology of finite-span wings is the proper identification of vortical structures. Hereafter, we test four well known criteria which are based on the velocity gradient tensor $\nabla \mathbf{u}$ [40, 63, 91, 178].

1. $\mid \omega \mid$-criterion, where $\mid \omega \mid$ is the norm of the vorticity vector $\mid \omega \mid = \nabla \times \mathbf{u}$. This criterion identifies a flow region as a vortex when $\mid \omega \mid$ reaches a specified threshold.

2. Q-criterion, where Q is the second invariant of $\nabla \mathbf{u}$. It defines a region as a vortex if every point in this region has $Q > 0$. The second invariant Q is defined as

$$\mathrm{Q} = 0.5\left(\| \Omega \|^2 - \| \mathrm{S} \|^2\right) \tag{8.1}$$

where $\Omega = 0.5(\nabla\mathbf{u} - \nabla\mathbf{u}^{\mathrm{T}})$ is the rate-of-strain tensor and $\mathrm{S} = 0.5(\nabla\mathbf{u} + \nabla\mathbf{u}^{\mathrm{T}})$ is the vorticity tensor, which are the asymmetric and symmetric components of $\nabla\mathbf{u}$, respectively; and $\| \mathbf{A} \| = \sqrt{\mathrm{tr}(\mathbf{A}\mathbf{A}^{\mathrm{T}})}$ is the Euclidean norm of a given tensor $\mathbf{A}$ (in our case Ω and S). Q indicates the local competition between the rotation rate and the deformation (or strain) rate, thus $\mathrm{Q} > 0$ means that the local rotational effect dominates [91].

3. λ_2-criterion, where λ_2 is the intermediate eigenvalue of the symmetric tensor $\Omega^2 + \mathrm{S}^2$, which relates the pressure P with the following relation

$$\Omega^2 + \mathrm{S}^2 = -\frac{\nabla\left(\nabla P\right)}{\rho} \tag{8.2}$$

This criterion defines a region as a vortex if every point in this region has $\lambda_2 < 0$, since $\lambda_2 < 0$ implies that the plane perpendicular to the local vortex axis has the local pressure minimum [40, 91].

4. N_k-criterion or kinematic vorticity number, which measure *"the quality of the rotation"*, instead of the local rotation rate given by $\| \Omega \|$. N_k is defined as

$$N_k = \frac{\| \Omega \|}{\| \mathrm{S} \|} \tag{8.3}$$

Thus, N_k is a pointwise measure of $| \omega |$ non-dimensionalized by the norm of the rate-of-strain tensor S, which gives the quality of the rotation regardless of the vorticity magnitude. For example, $N_k = \infty$ and $N_k = 0$ correspond to solid-body rotation and irrotational motion respectively, regardless of the $| \omega |$ value [91].

In figures 8.4, 8.5, 8.6, and 8.7 the vortical structures obtained using these four criteria are presented. For this case, a rectangular wing with aspect ratio equal to AR = 1 and elliptical cross-section (with a corresponding major axis $a = 0.5$ and minor axis $b = 0.0625$) is used. The wing is undergoing pure heaving motion as per equation eq. 2.21, at a Strouhal number equal to $St = 0.5$ and heaving amplitude equal to $h_a = 0.25$. The Reynolds number based on the wing chord is $Re = 500$. The results are shown for a non-dimensional time $t = 7.0$. The thresholds of each criteria were selected carefully so that the isosurfaces show approximately the same topological vortical structures.

It can be seen from figure 8.4, that the $| \omega |$-criterion, although capable of capturing the general vortical structures, has the disadvantage of also showing the shear layers near the wing surface and between the vortices. The Q, λ_2 and N_k criteria, show the vortical structure details more clearly and provide nearly identical structures; this is due to their mathematical and physical similarities [40, 63, 91, 178].

Additionally, we also use the pressure as a criteria for vortex identification. Local pressure minima are related to the presence of vortex structures, however, there is an inherent scale difference

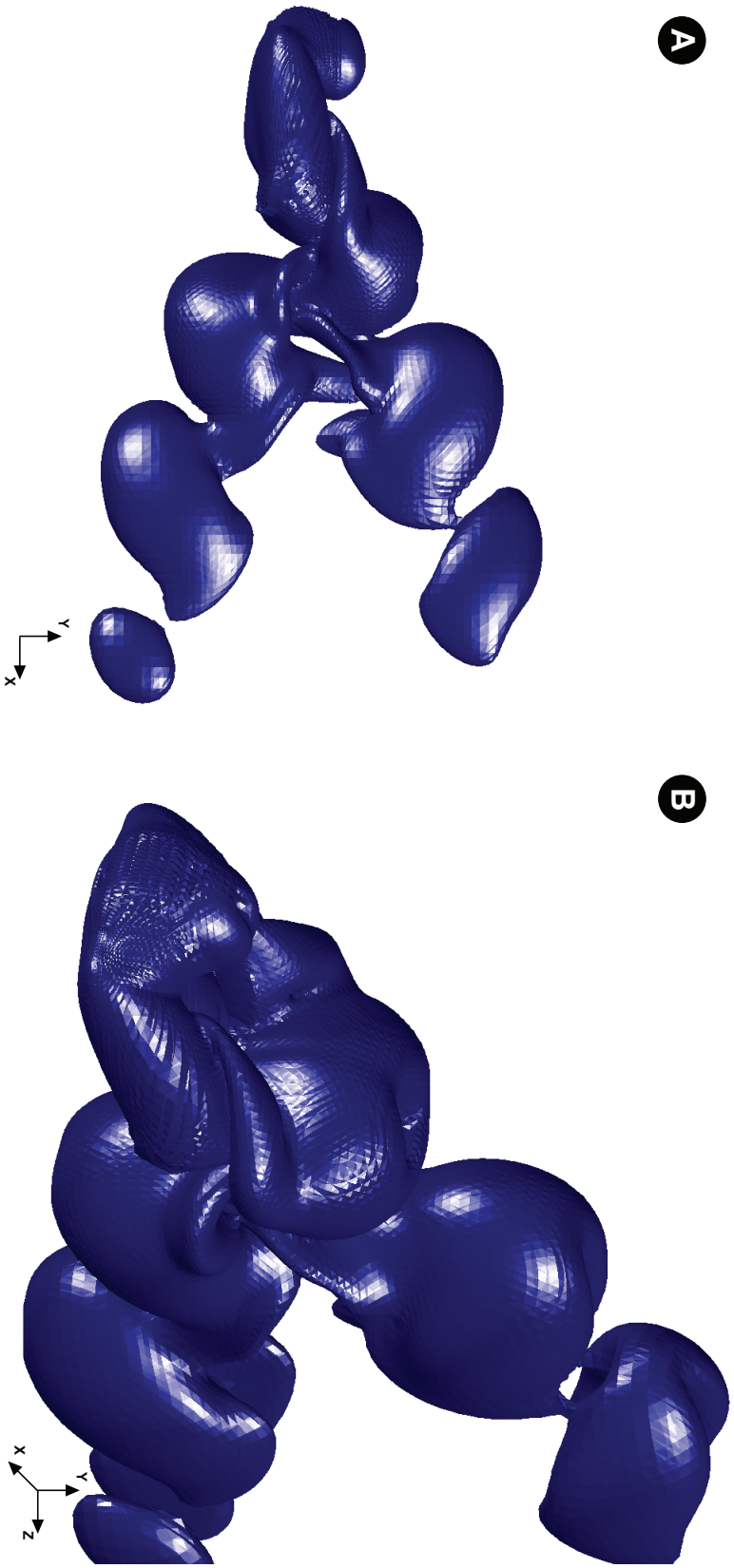

Figure 8.4: *Isosurfaces of* $|\omega|$*-criterion at the beginning of the upstroke (t=7.0). Flapping parameters:* $St = 0.5, h_a = 0.25, Re = 500$.

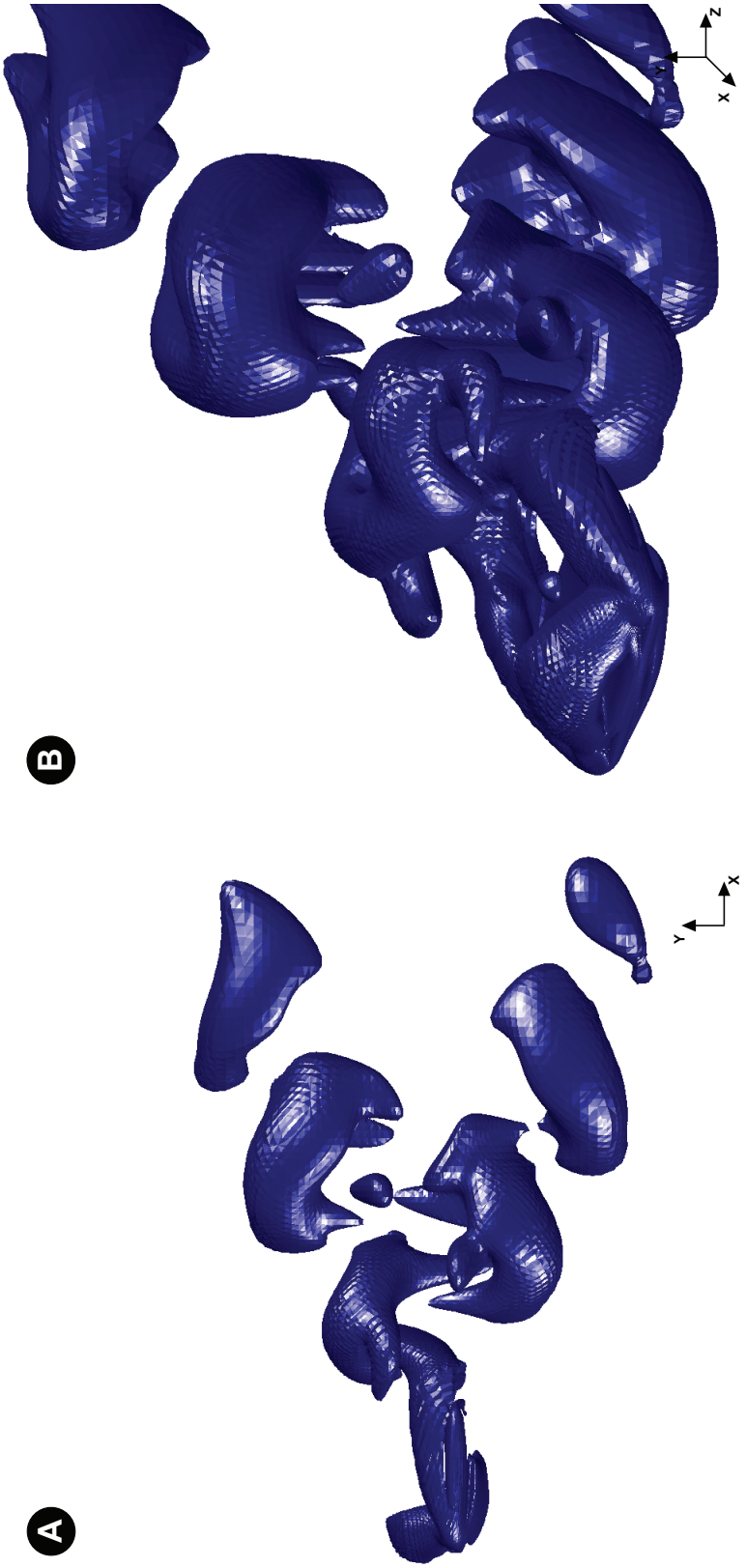

Figure 8.5: *Isosurfaces of Q-criterion at the beginning of the upstroke (t=7.0). Flapping parameters:* $St = 0.5, h_a = 0.25, Re = 500$.

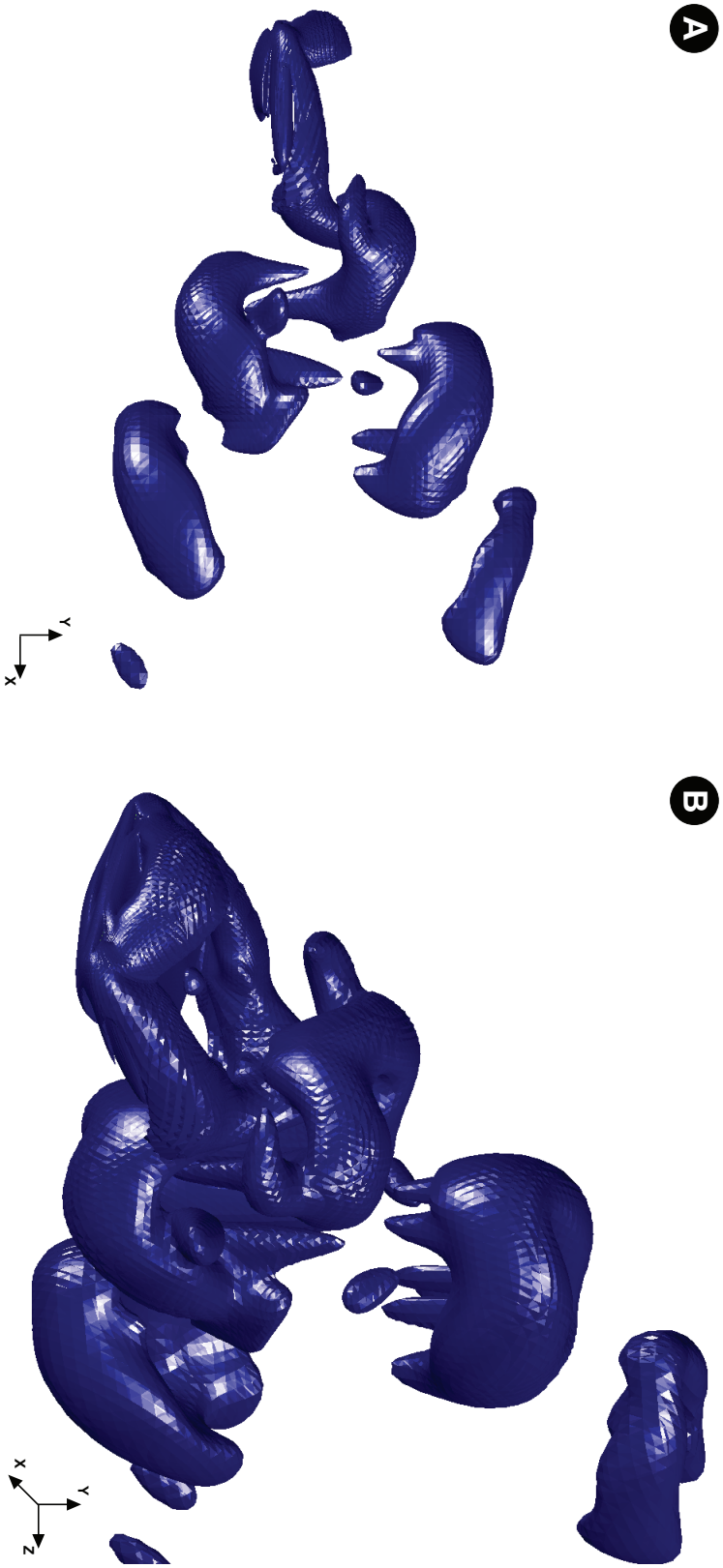

Figure 8.6: *Isosurfaces of λ_2-criterion at the beginning of the upstroke (t=7.0). Flapping parameters: $St = 0.5, h_a = 0.25, Re = 500$.*

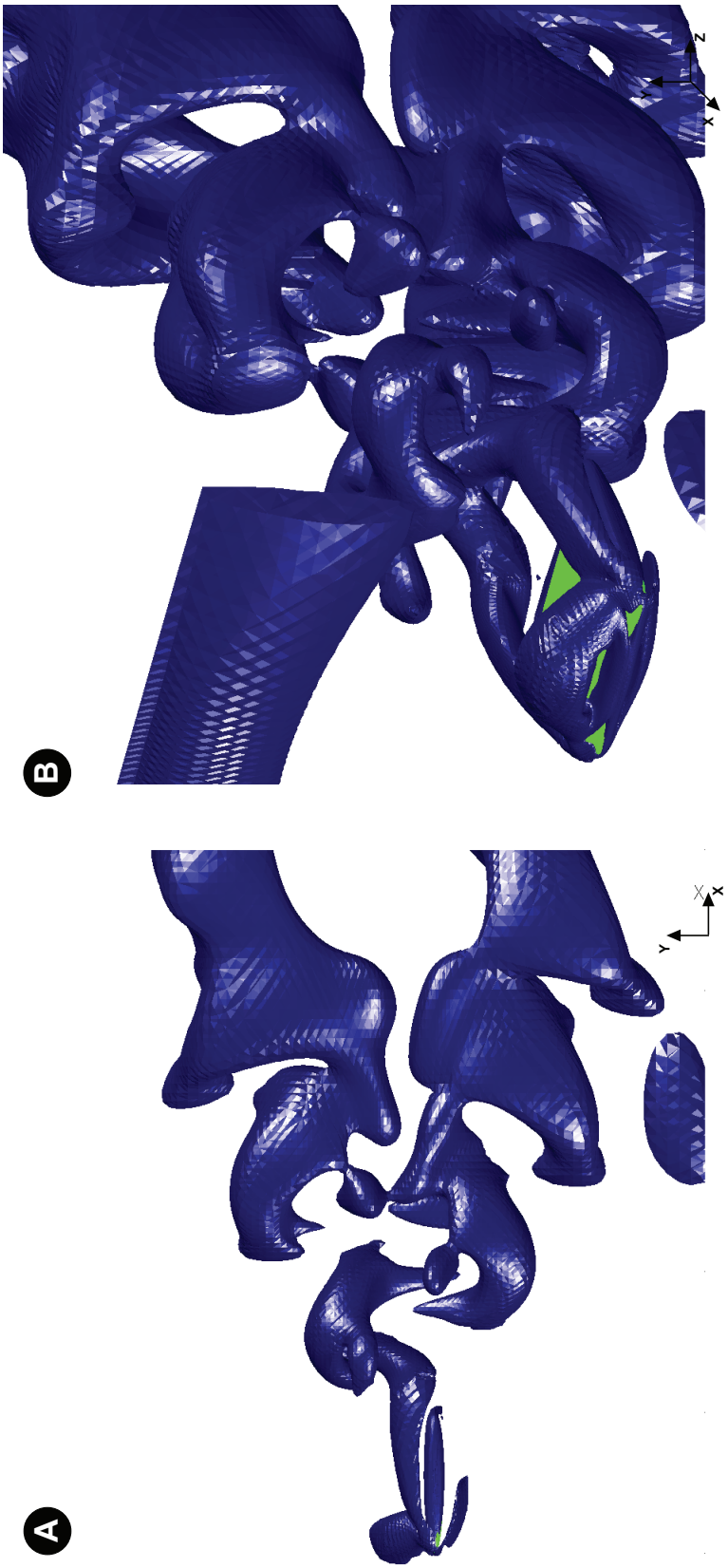

Figure 8.7: *Isosurfaces of N_k-criterion at the beginning of the upstroke (t=7.0). Flapping parameters: $St = 0.5, h_a = 0.25, Re = 500$.*

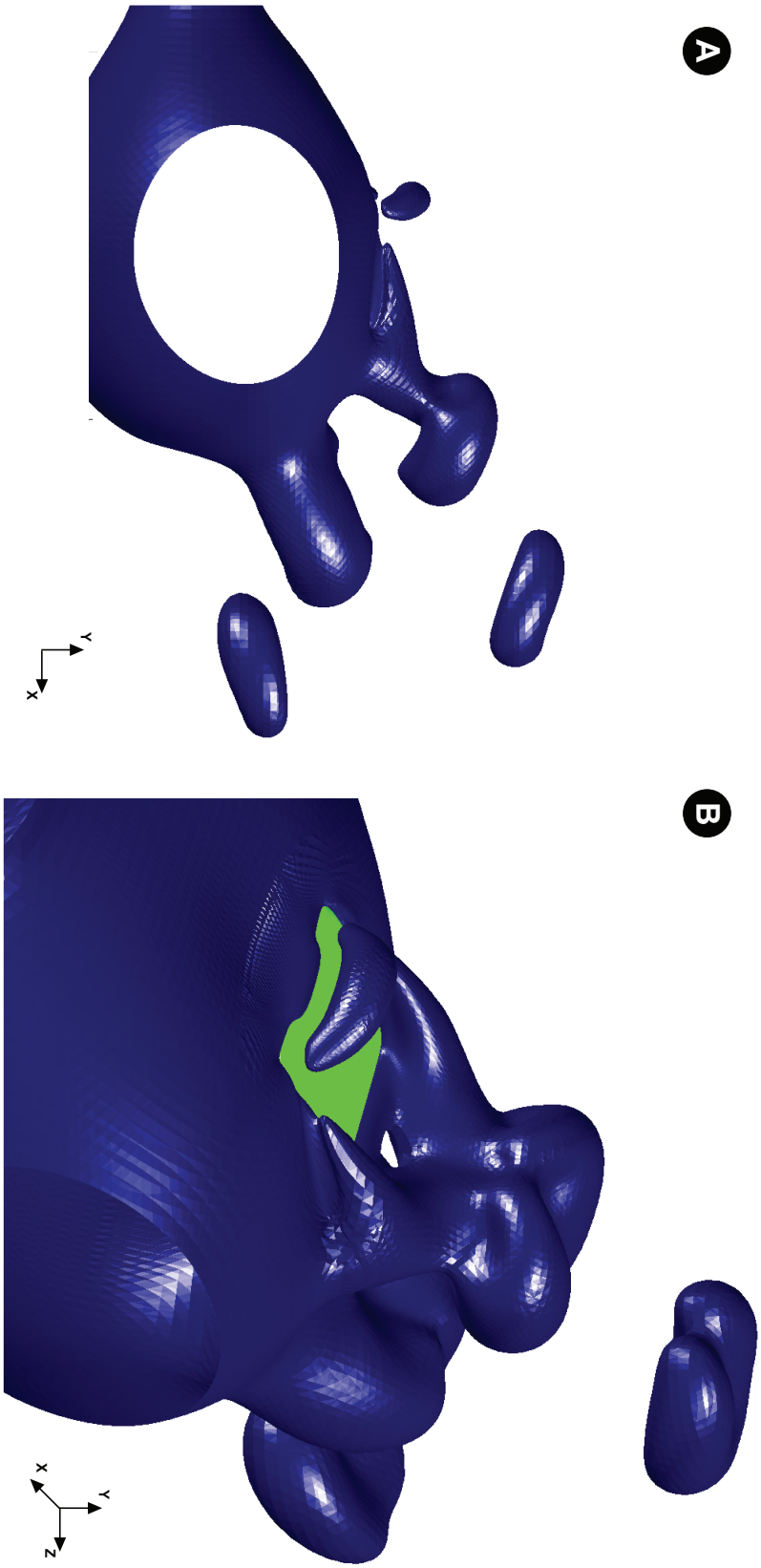

Figure 8.8: *Isosurfaces of pressure field at the beginning of the upstroke (t=7.0). Flapping parameters:* $St = 0.5, h_a = 0.25, Re = 500$.

between a vortex core and the associate low-pressure region and this makes the definition of vortices using isosurface of pressure a difficult task [91]. In figure 8.8, the isopressure surfaces are illustrated; in this figure, in order to capture the same vortical structures as in the velocity gradient tensor criteria, we choose low pressure values for the isosurfaces to be plotted. As it can be seen, in the far wake of the wing it is possible to capture vortex rings, while in the region close to the wing and in the upstream direction, the isopressure values extends farther away, making it difficult to identify the vortical structures.

Comparing all the previous criteria and since the Q-criterion offers more information about the local flow field, *i.e.*, $\mathrm{Q} < 0$ means the local deformation (strain) rate dominates over the rotation rate, we choose this as the main criterion for the wake topology characterization. Also, from a computational point of view it is found that the Q-criterion is less computational expensive than the λ_2 and N_k criteria.

8.4 Heaving Wing

In this section, we carry out a parametric study to asses the effect of Strouhal number St and reduced frequency k on the aerodynamic performance of a rigid finite-span heaving wing. A rectangular wing with aspect ratio equal to $\mathrm{AR} = 1$ and elliptical cross-section (with a corresponding major axis $a = 0.5$ and minor axis $b = 0.0625$) is used. The wing aspect ratio is defined as follows

$$\mathrm{AR} = \frac{S^2}{A} \tag{8.4}$$

where S is the wing span (wingtip-to-wingtip distance) and A is the wing area. For a rectangular wing, eq. 8.4 is simplified to

$$\mathrm{AR} = \frac{S}{c} \tag{8.5}$$

where c is the wing chord. High aspect ratios indicates long narrow wings, whereas a low aspect ratio indicates short, stubby wings.

The wing is undergoing pure heaving motion, wherein the wing cross-section center heaves in the vertical direction as per equation eq. 2.21. In table 8.1, we present the kinematics parameters governing this numerical experiment, where h_a is the heaving amplitude, f_h is the heaving frequency, St is the Strouhal number and k the reduced frequency. The numerical experiment is conducted at a Reynolds number equal to $Re = 500$.

The summary of results is presented in tabular form in table 8.2, where $\overline{c_t}$ is the average thrust coefficient and $\hat{c}_l$ is the maximum lift coefficient. Inspecting table 8.2, we can observe that as we increase St and k, the average thrust coefficient $\overline{c_t}$ and maximum lift coefficient $\hat{c}_l$ also increase (similar behavior as for the two-dimensional case). From the results obtained, for values of Strouhal number less than $St < 0.25$ we are in the drag production regime, for values of St between $0.25 < St < 0.35$ we produce little or no drag (or thrust), whereas for values of St higher than $St > 0.35$ we are in the thrust production regime. As in the two-dimensional cases, we observe two different behaviors of the aerodynamic forces for high and low reduced frequencies

Case number	Re	h_a	f_h	St	k
3DH1-1	500	0.15	0.5	0.15	1.57079
3DH1-2	500	0.075	1.0	0.15	3.14159
3DH1-3	500	0.25	0.5	0.25	1.57079
3DH1-4	500	0.125	1.0	0.25	3.14159
3DH1-5	500	0.35	0.5	0.35	1.57079
3DH1-6	500	0.175	1.0	0.35	3.14159
3DH1-7	500	0.5	0.5	0.5	1.57079
3DH1-8	500	0.25	1.0	0.5	3.14159

Table 8.1: *Kinematics parameters for the pure heaving wing case.*

k. Hence, it seems that for flapping wings, the flapping frequency also plays an important role in the vortex generation and shedding and, henceforth, on the aerodynamic forces.

Case number	St	k	$\overline{c_t}$	$\hat{c_l}$
3DH1-1	0.15	1.57079	-0.1832	1.5863
3DH1-2	0.15	3.14159	-0.1369	3.0548
3DH1-3	0.25	1.57079	-0.1407	2.9257
3DH1-4	0.25	3.14159	-0.0951	5.1792
3DH1-5	0.35	1.57079	-0.0766	4.5644
3DH1-6	0.35	3.14159	-0.0101	8.0671
3DH1-7	0.5	1.57079	0.0775	7.8142
3DH1-8	0.5	3.14159	0.1297	13.2126

Table 8.2: *Simulation results for the pure heaving wing case (positive $\overline{c_t}$ values indicate thrust production whereas negative $\overline{c_t}$ values indicate drag production).*

In table 8.3, we compare the results obtained for an infinite-span wing AR $= \infty$ (simulated as a two-dimensional case) against a finite-span wing. As it can be seen, the two-dimensional simulations highly overestimate the thrust and lift coefficients; obviously, this is due to the fact that three-dimensional effects (such as induced drag) are neglected. In figure 8.9, we show the spanwise vorticity contours for the case 3DH1-8 on the spanwise symmetry plane. For the infinite-span case (figure 8.9.A), we can clearly identify a reverse von Karman street which is indicative of thrust production, whereas for the finite-span wing (figure 8.9.C) the wake is a little bit different but we still can observe some common features with the infinite-span wing, such as the presence of the clockwise vortex V3 in the wake, the counter-clockwise vortex V2 shedding from the trailing edge and the clockwise LEV V1 on the top surface of the wing. However, in spite of these similarities, we also observe significant differences between the two wakes, the most important being the way how the wake is diffused in the streamwise direction for the finite-span case. In figure 8.9.D, we show the spanwise vorticity contours in the plane located at a distance equal to $d = 0.4 \times c$ measured from the wing symmetry plane; in this figure we can observe how the intensity of the vortices decrease as we move in the spanwise direction.

Let us now take a detailed look at the three-dimensional wake topologies shown in figures 8.10

Case number	St	k	$\overline{c_t}$ (3$\mathbb{D}$ case)	$\hat{c_l}$ (3$\mathbb{D}$ case)	$\overline{c_t}$ (2$\mathbb{D}$ case)	$\hat{c_l}$ (2$\mathbb{D}$ case)
3DH1-1	0.15	1.57079	-0.1832	1.5863	-0.1555	2.3144
3DH1-2	0.15	3.14159	-0.1369	3.0548	-0.1767	4.7388
3DH1-3	0.25	1.57079	-0.1407	2.9257	-0.1109	4.4637
3DH1-4	0.25	3.14159	-0.0951	5.1792	-0.04667	8.3741
3DH1-5	0.35	1.57079	-0.0766	4.5644	0.1177	7.2945
3DH1-6	0.35	3.14159	0.0101	8.0671	0.1682	13.2557
3DH1-7	0.5	1.57079	0.0775	7.8142	0.2857	13.6409
3DH1-8	0.5	3.14159	0.1297	13.2126	0.4823	25.3062

Table 8.3: *Simulation results for the pure heaving wing case. Comparison of the* 3$\mathbb{D}$ *results versus the* 2$\mathbb{D}$ *results (positive* $\overline{c_t}$ *values indicate thrust production whereas negative* $\overline{c_t}$ *values indicate drag production).*

and 8.11. From the three-dimensional perspective view of the wake topology (figure 8.10.B), it is clear that the wake of this finite-span wing has little resemblance with the two-dimensional case. This plot, as well as the side view, shows that the downstream wake of this wing consist of two sets of complex shaped vortex rings which convect at oblique angles about the centerline of the motion. In the figure we identify two rings R1 and R3 in the upper part of the wake and one ring R2 in the lower part of the wake. The process by which the vortex rings are formed can be explained by examining the vortex formation and shedding close to the wing. Let us consider the vortex V1 and V2, and their two associated wing-tip vortices TVL1 and TVR1 which constitute the four sides of a vortex loop, as shown in figures 8.10 and 8.11. Vortices V1 and V2 are connected by the wing-tip vortices TVL1 and TVR1 forming a vortex loop; as this vortex loop is convected it fully disconnect from the wing, forming in this way a vortex ring. It is also interesting to note that the vortex rings are themselves inclined with respect to the free-stream. It is also of interest that each vortex loop has two sets of thin contrails (C1 in figure 8.10.A), these structures are segments of the wing-tip vortices and as the vortex loops are convected downstream; these contrails become weaker and ultimately disappear as for vortex ring R1.

The wake topology plotted in figures 8.10 and 8.11 corresponds to a thrust production case (refer to case 3DH1-8 in table 8.2), in figures 8.12 and 8.13 we present the wake topology for a drag production case (refer to case 3DH1-2 in table 8.2). It is clear from these figures that the wake topology is very different from the one of the thrust generation case. In this case, as the vortex loops are convected downstream, they do not convert into vortex rings, instead, they keep their original shape and are diffused. In figures 8.12 and 8.13, it can be also observed that the wake height is very compact, opposite to that of the thrust production case. It can be also observed that the vortex loops are inclined in the direction opposite of their travel.

8.5 Flapping Wing

In this section, we present the results of a wing undergoing flapping motion (coupled heaving-and-pitching motion), where the wing cross-section heaves in the vertical direction as per eq. 2.21 and pitches about the spanwise axis at the wing cross-section center according to eq. 2.22. In table 8.4, we present the kinematic parameters governing this numerical experiment, where h_a is the heaving amplitude, α_a is the pitching amplitude, f is the flapping frequency (for both pitching

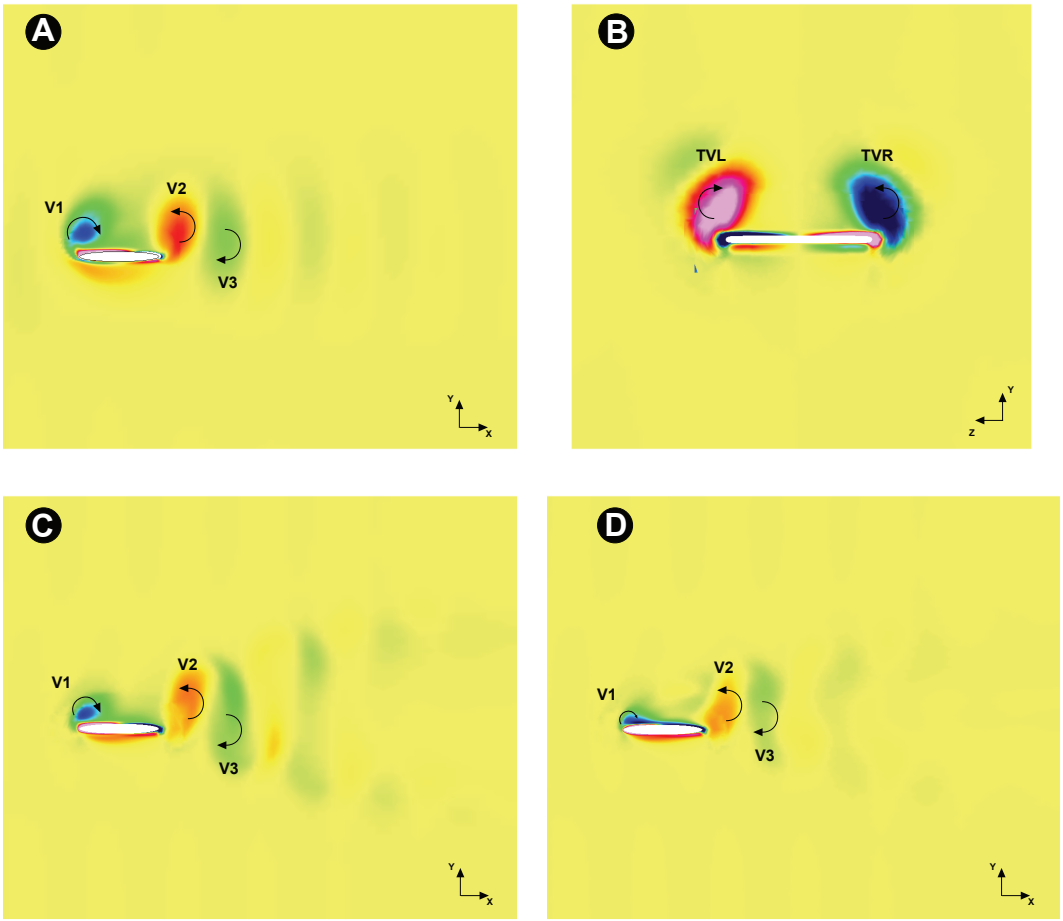

Figure 8.9: *Spanwise vorticity contours at the beginning of the upstroke (t=7.0). Flapping parameters:* $St = 0.5, h_a = 0.25, Re = 500$. *A) Spanwise vorticity contours for the infinite-span wing (2𝔻 case). B) Rear view of the wing-tip vortices for the finite-span wing, where TVL is the left wing-tip vortex and TVR is the right wing tip vortex. C) Spanwise vorticity contours for the finite-span wing at the symmetry plane (3𝔻 case). D) Spanwise vorticity contours for the finite-span wing in the plane located at a distance equal to* $d = 0.4 \times c$ *measured from the wing symmetry plane (3𝔻 case).*

and heaving motion), St is the Strouhal number and k the reduced frequency. The numerical experiment was conducted at a Reynolds number equal to $Re = 500$ and the finite-span wing used is exactly the same as that used in the previous section.

Case number	Re	h_a	$\alpha_a(^\circ)$	f	St	k
3DF1	500	0.25	5	1.0	0.5	3.14159
3DF2	500	0.25	10	1.0	0.5	3.14159
3DF3	500	0.25	20	1.0	0.5	3.14159

Table 8.4: *Kinematic parameters for the flapping wing case.*

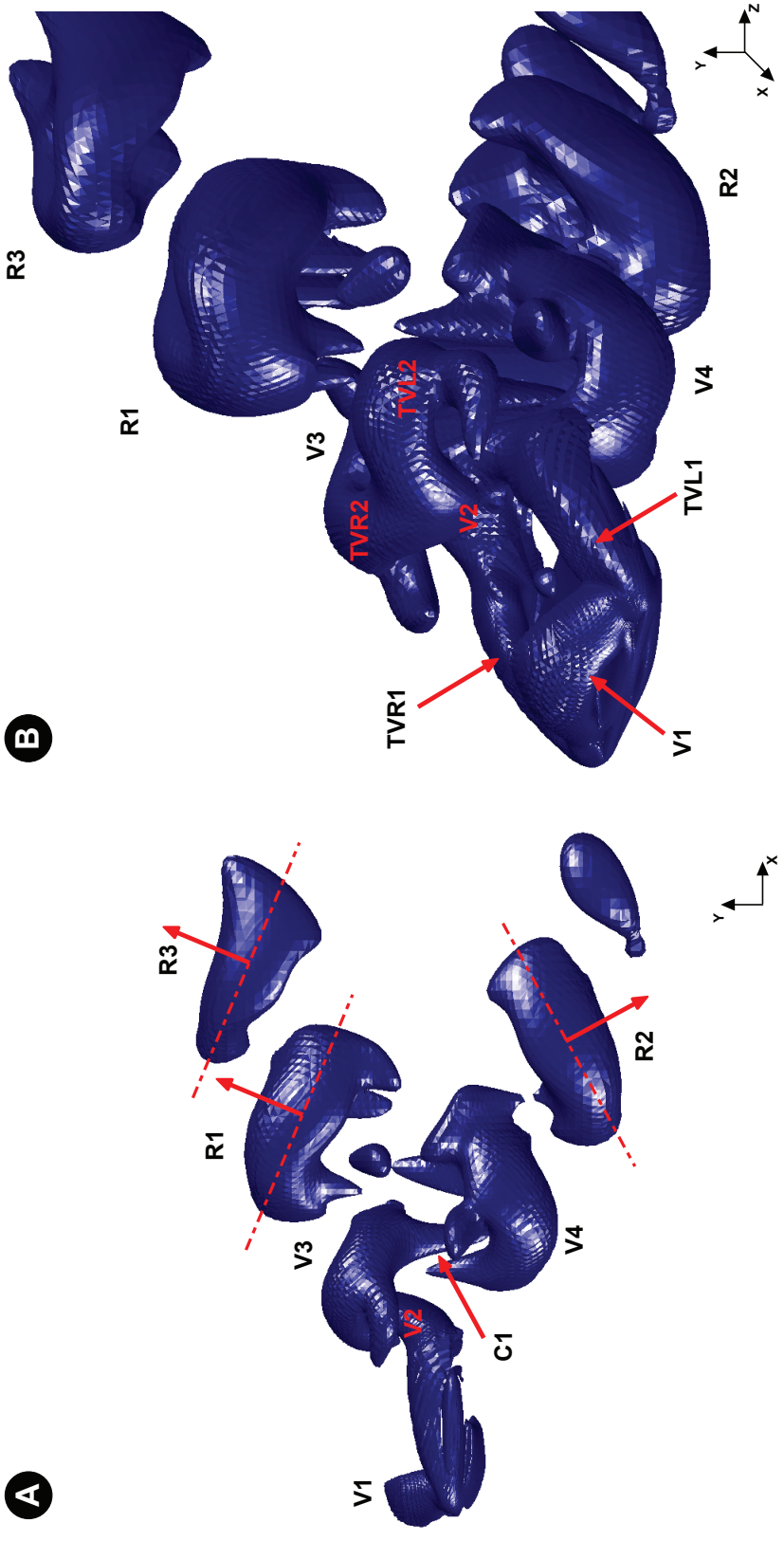

Figure 8.10: *Vortex topology at the beginning of the upstroke (t=7.0). Flapping parameters:* $St = 0.5, h_a = 0.25, Re = 500$. *A) Side view. B) Perspective view.*

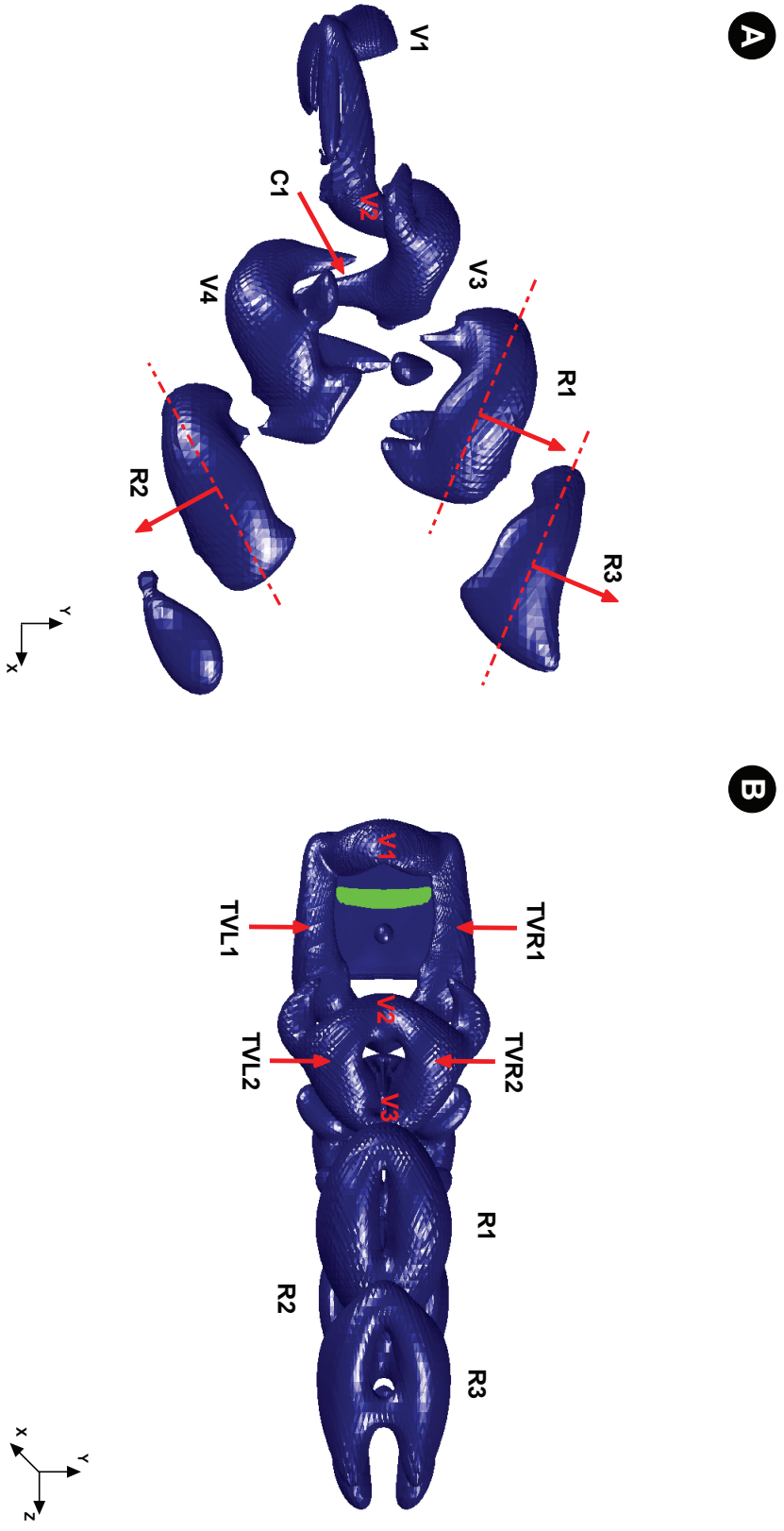

Figure 8.11: *Vortex topology at the beginning of the upstroke (t=7.0). Flapping parameters:* $St = 0.5, h_a = 0.25, Re = 500$. *A) Side view. B) Top view.*

Figure 8.12: *Vortex topology at the beginning of the upstroke (t=7.0). Flapping parameters:* $St = 0.15, h_a = 0.075, Re = 500$. *A) Side view. B) Perspective view.*

Figure 8.13: *Vortex topology at the beginning of the upstroke (t=7.0). Flapping parameters:* $St = 0.15, h_a = 0.075, Re = 500$. *A) Side view. B) Top view.*

In table 8.5, we present the computed average thrust coefficient $\overline{c_t}$ and maximum lift coefficient $\hat{c}_l$ for the cases shown in table 8.4, additionally, for purposes of comparison we also show the results for the case 3DH1-8 (see table 8.4), which corresponds to pure heaving motion.

Case number	St	h_a	$\alpha_a(^\circ)$	f	$\overline{c_t}$	$\hat{c}_l$
3DH1-8	0.5	0.25	0	1.0	0.1297	13.2126
3DF1	0.5	0.25	5	1.0	0.1307	14.7020
3DF2	0.5	0.25	10	1.0	0.1402	10.2724
3DF3	0.5	0.25	20	1.0	0.1591	6.8924

Table 8.5: *Simulation results for the flapping wing case.*

The typical wake topologies for the cases 3DF2 and 3DF3 are shown in figures 8.14 and 8.15, respectively. The figures show the three-dimensional perspective view, side view and top view. As for the heaving case, the key feature observed here is the presence of two interconnected vortex loops that slowly convert into vortex rings as they are convected downstream. Additionally, in figures 8.16 and 8.17 we plot the streamlines for a given instant during the downstroke, notice in these figures the wake evolution and the wing-tip vortices.

8.6 Aspect Ratio Influence on the Aerodynamics Performance

In this section, we consider the effect of aspect-ratio AR on the aerodynamic performance. In this case, the wing is undergoing pure heaving motion as per equation 2.21. The kinematics and geometry parameters are shown in table 8.6, where h_a is the heaving amplitude, AR is the aspect ratio, f_h is the heaving frequency, St is the Strouhal number and k the reduced frequency.

Case number	Re	h_a	f_h	AR	St	k
3DAR1	500	0.175	1.0	1.0	0.35	3.14159
3DAR2	500	0.175	1.0	2.0	0.35	3.14159
3DAR3	500	0.175	1.0	3.0	0.35	3.14159
3DAR4	500	0.175	1.0	4.0	0.35	3.14159

Table 8.6: *Kinematics and geometrical parameters for the heaving wing case with different aspect ratios.*

The main purpose of this numerical experiment, is to assess the effect of AR on the aerodynamic forces of finite-span wings and to study if the assumption of two-dimensionality have some validity for three-dimensional cases. Hereafter, the results are presented in tabular form in table 8.7, where $\overline{c_t}$ is the average thrust coefficient and $\hat{c}_l$ is the maximum lift coefficient. Inspecting table 8.7, we can observe that as we increase the wing AR, the computed values of the aerodynamics forces also increase and this is chiefly to the fact of the high area of high AR wings and to the decrease of three-dimensional effects in long wings. This observation lead us to think that the assumption of two-dimensionality has some validity for birds and insects, where the wings of many species tend to have relatively large aspect ratio.

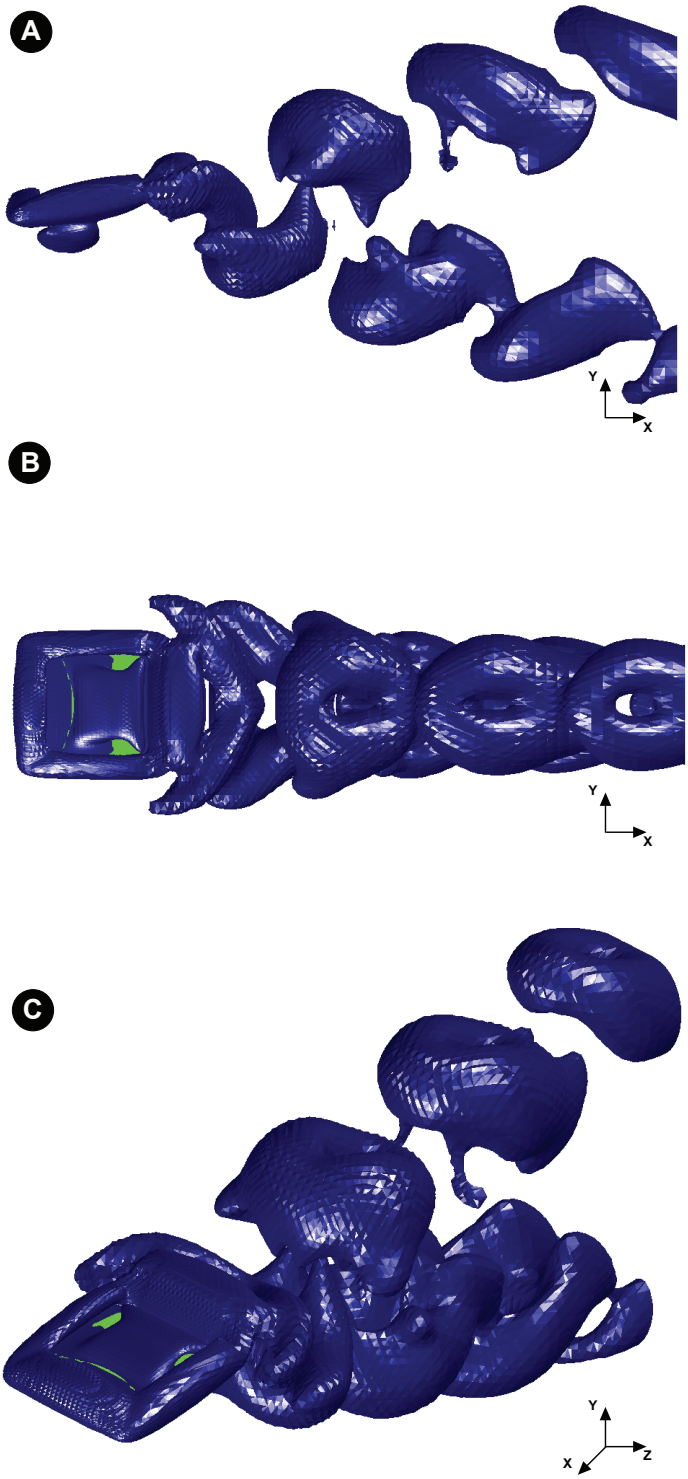

Figure 8.14: *Vortex topology during downstroke (t=6.75). Flapping parameters:* $St = 0.5, h_a = 0.25, \alpha_a = 10°, Re = 500$. *A) Side view. B) Top view. C) Perspective view.*

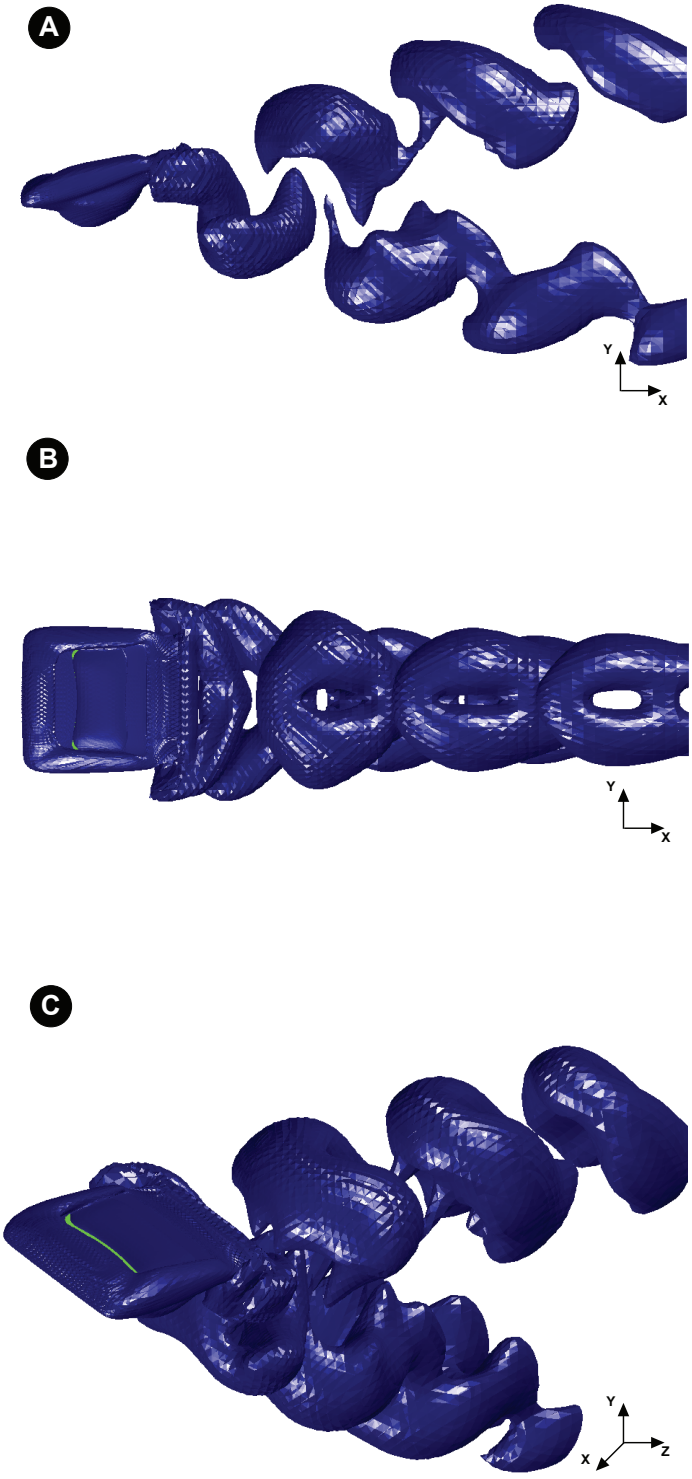

Figure 8.15: *Vortex topology during downstroke (t=6.75). Flapping parameters:* $St = 0.5, h_a = 0.25, \alpha_a = 20°, Re = 500$. *A) Side view. B) Top View. C) Perspective view.*

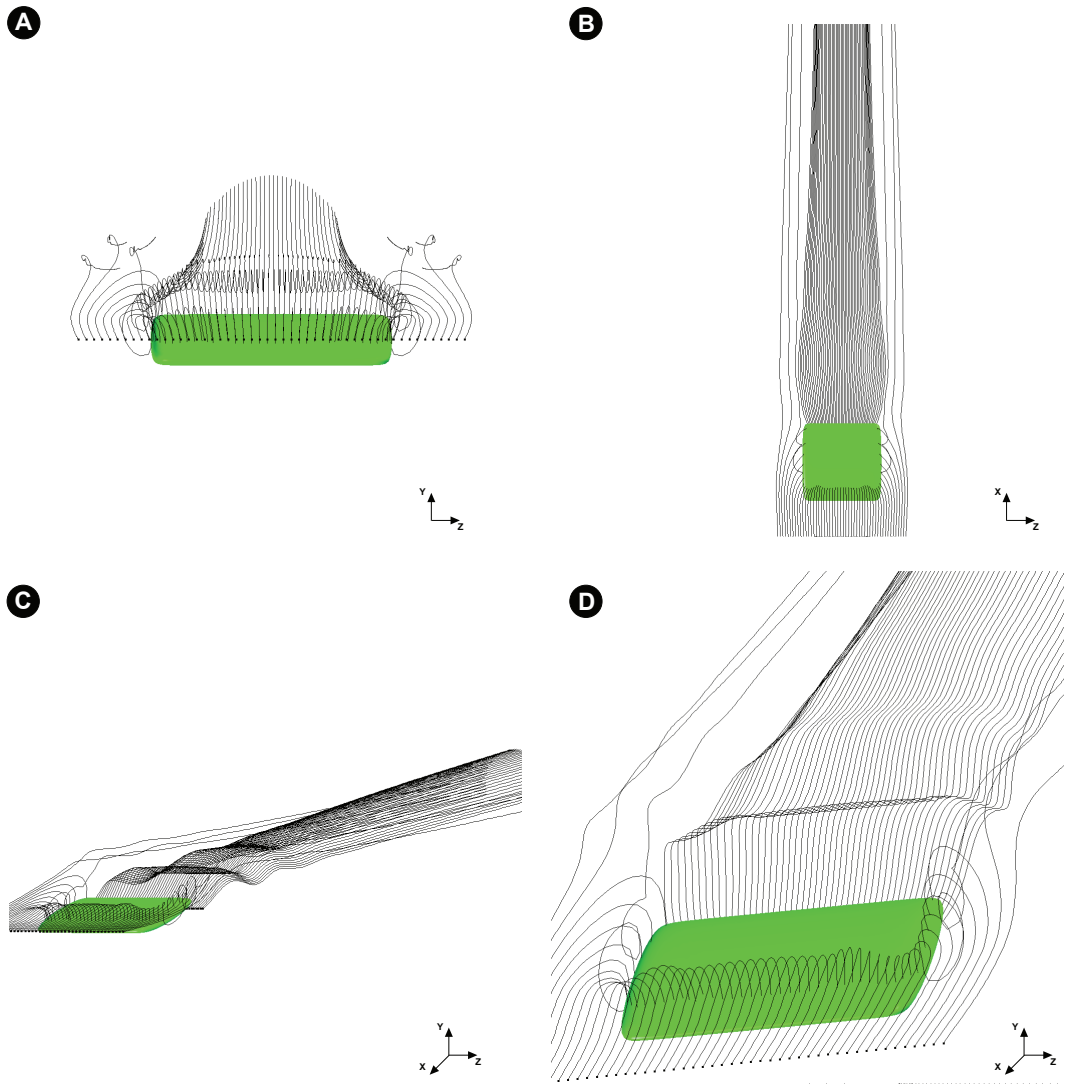

Figure 8.16: *Streamlines visualization during downstroke (t=6.75). Flapping parameters:* $St = 0.5, h_a = 0.25, \alpha_a = 10°, Re = 500$. *A) Front view. B) Top View. C) Perspective view. D) Perspective view.*

Case number	AR	$\overline{c_t}$	$\hat{c_l}$
3DAR1	1	-0.0101	8.0671
3DAR2	2	0.0356	16.5643
3DAR3	3	0.0784	33.8802
3DAR4	4	0.0864	49.5601
3DH1-6	∞	0.1682	13.2557

Table 8.7: *Simulation results for the heaving wing case with different aspect ratios (positive* $\overline{c_t}$ *values indicate thrust production whereas negative* $\overline{c_t}$ *values indicate drag production).*

8.7 Rolling Wing

In this section, we simulate a rectangular wing rolling about its traveling axis (root-flapping motion). The wing aspect ratio for this case is equal to AR = 2 and its cross-section is elliptical

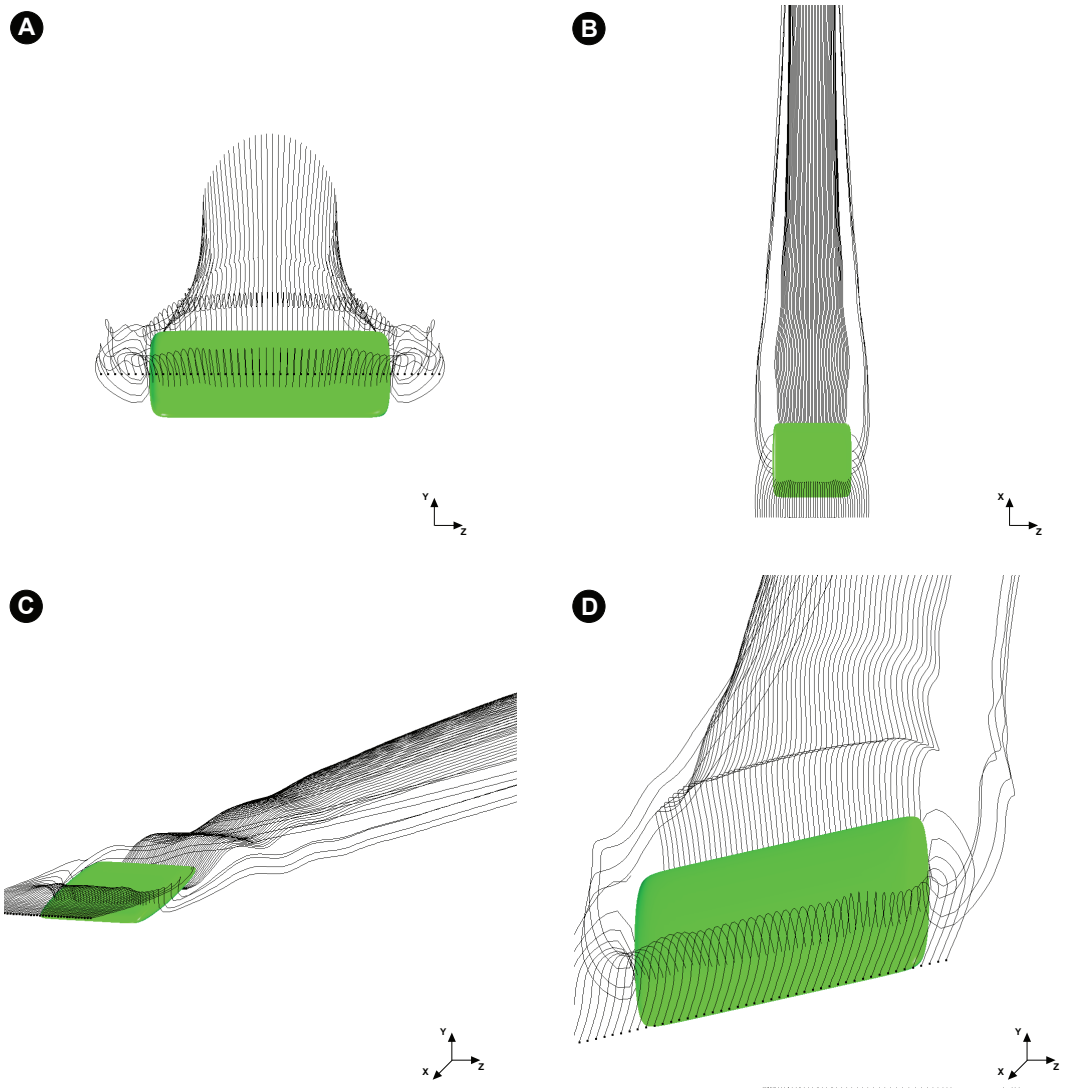

Figure 8.17: *Streamlines visualization during downstroke (t=6.75). Flapping parameters:* $St = 0.5, h_a = 0.25, \alpha_a = 20°, Re = 500$. *A) Front view. B) Top View. C) Perspective view. D) Perspective view.*

(with corresponding major and minor axis equal to $a = 0.25$ and $b = 0.025$). In this case, the wing is hinged at one wing-tip and is rolling about the traveling axis as per eq. 2.22 (where the traveling axis and the hinged point are collinear). The Strouhal number for this case is based in the dorsoventral stroke angle as proposed by Taylor *et al.* [182] and is computed as follows

$$St = \frac{S\ sen\,(\phi/2)\,f_{roll}}{U} \tag{8.6}$$

where S is the wing span, ϕ is the dorsoventral stroke angle or positional angle (see figure 2.21), f_{roll} is the wing rolling frequency and U the forward velocity. The kinematic parameters for this case are shown in table 8.8. The numerical experiment is conducted at a Reynolds number equal to $Re = 500$.

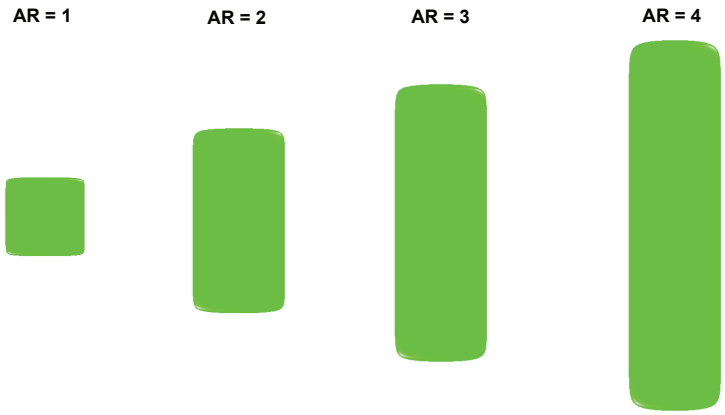

Figure 8.18: *Different wing platforms used for the study of aspect ratio influence on the aerodynamic performance.*

Case number	Re	$\phi(^\circ)$	f_{roll}	St
3DRL1	500	12.5	1.0	0.10
3DRL2	500	30.0	1.0	0.25
3DRL3	500	45.0	1.0	0.38

Table 8.8: *Kinematics parameters for the rolling wing case.*

Usually, flapping wing studies only consider heaving or coupled heaving-and-pitching motions. Hereafter, we carry out this numerical study in order to check whether this mode of motion shows similar features to those of the heaving or coupled heaving-and-pitching motions, we also study the validity of the use of the Strouhal number for wake characterization.

In table 8.9 we show the simulation results for this case, as it can be seen, for values less than $St < 0.25$ we are in the drag production regime, for values approximately equal to $St = 0.25$ we produce little or no drag (or thrust), whereas for values higher that $St > 0.25$ we are in the thrust production regime. From these results, it is clear that this behavior is similar to that of heaving or coupled heaving-and-pitching motions. Comparing these results with the results for pure heaving or coupled heaving-and-flapping motions, we found that the latter motions generate larger vortices and forces than root-flapping motion, presumably because the average velocity is higher across the span, but otherwise the same wake regimes occurs at similar Strouhal numbers.

Case number	St	$\overline{c_t}$	$\hat{c_l}$
3DRL1	0.10	-0.1485	0.5991
3DRL2	0.25	-0.0843	2.3637
3DRL3	0.38	0.0748	4.8749

Table 8.9: *Summary of results for the rolling case with different dorsoventral stroke angles (positive $\overline{c_t}$ values indicate thrust production whereas negative $\overline{c_t}$ values indicate drag production).*

In figures 8.19, 8.20 and 8.21, the three-dimensional wake structures for the cases shown in table 8.8 are illustrated. The salient feature that needs to be pointed out from these figures is the absence of any link or the presence of a very weak link between the root-tip vortex and the leading and trailing edge vortices. We also observe that for the case of $St = 0.10$, there is no connection at all between both wing-tip vortices and the trailing and leading edge vortices; this is clearly due to the fact that the tip vortices generated at this low Strouhal number have very low strength. Additionally, in figures 8.22 and 8.23 we plot the streamlines for four different instants during the downstroke, notice in these figures the wake evolution and the wing-tip vortices at the moving tip.

8.8 Summary

In this chapter, we have presented several results for finite-span flapping wings. The simulations show that the wake of thrust producing, rigid finite-span flapping wings is formed by two sets of interconnected vortex ring loops that slowly convert into vortex rings as they are convected downstream. It was also observed that the vortex rings are themselves inclined with respect to the free-stream; the angle of inclination of the vortex rings is found to be in the direction of their travel and in the streamwise direction for thrust producing configurations; whereas for drag producing configurations the angle of inclination is opposite to the direction of travel of the streamwise flow. It was also noted the presence of thin contrails that link the vortex loops, these structures are segments of the wing-tip vortices and as the vortex loops are convected downstream, these contrails become weaker and ultimately disappear. In general, the observed structures are qualitatively similar to those observed in the experiments by Parker *et al.* [138].

In this chapter, the effect of aspect ratio AR on the aerodynamic forces of finite-span wings was also assessed. It was observed that as we increase the wing AR, the aerodynamic forces also increase and this is chiefly attributed to the large area of high aspect ratio wings and to the decrease of three-dimensional effects in long wings. This observation lead us to think that the assumption of two-dimensionality has some validity for birds and insects, where the wings of many species tend to have relatively large aspect ratio, *e.g.*, large seabirds such as Albatrosses have aspect ratios about 13 to 15, land birds such as Eagles and Vultures have aspect ratios of roughly 6 to 8 and insects such as the Bumblebee and the Crane fly have aspect ratios between 6 to 7 [3].

Finally, besides the heaving and coupled heaving-and-pitching motions, we have also studied the root-flapping motion characteristic of flying animals, which as far as the author is aware, still remains virtually unexplored. From the results obtained it is found that, indeed, root-flapping motion produces wake structures similar to those of heaving or coupled heaving-and-pitching motions, but with the difference that the latter motions generate larger vortices and forces than root-flapping motion, presumably because the average velocity is higher across the span; aside from this, similar wake regimes occurs at similar Strouhal numbers.

Figure 8.19: *Vortex topology for the rolling wing case (t=5.0). Flapping parameters:* $St = 0.10, f_{roll} = 1.0, Re = 500$. *A) Perspective view. B) Top view. In this view the right wing-tip corresponds to the hinged extreme. C) Side view.*

Figure 8.20: *Vortex topology for the rolling wing case (t=5.0). Flapping parameters:* $St = 0.25, f_{roll} = 1.0, Re = 500$. *A) Perspective view. B) Top view. In this view the right wing-tip corresponds to the hinged extreme. C) Side view.*

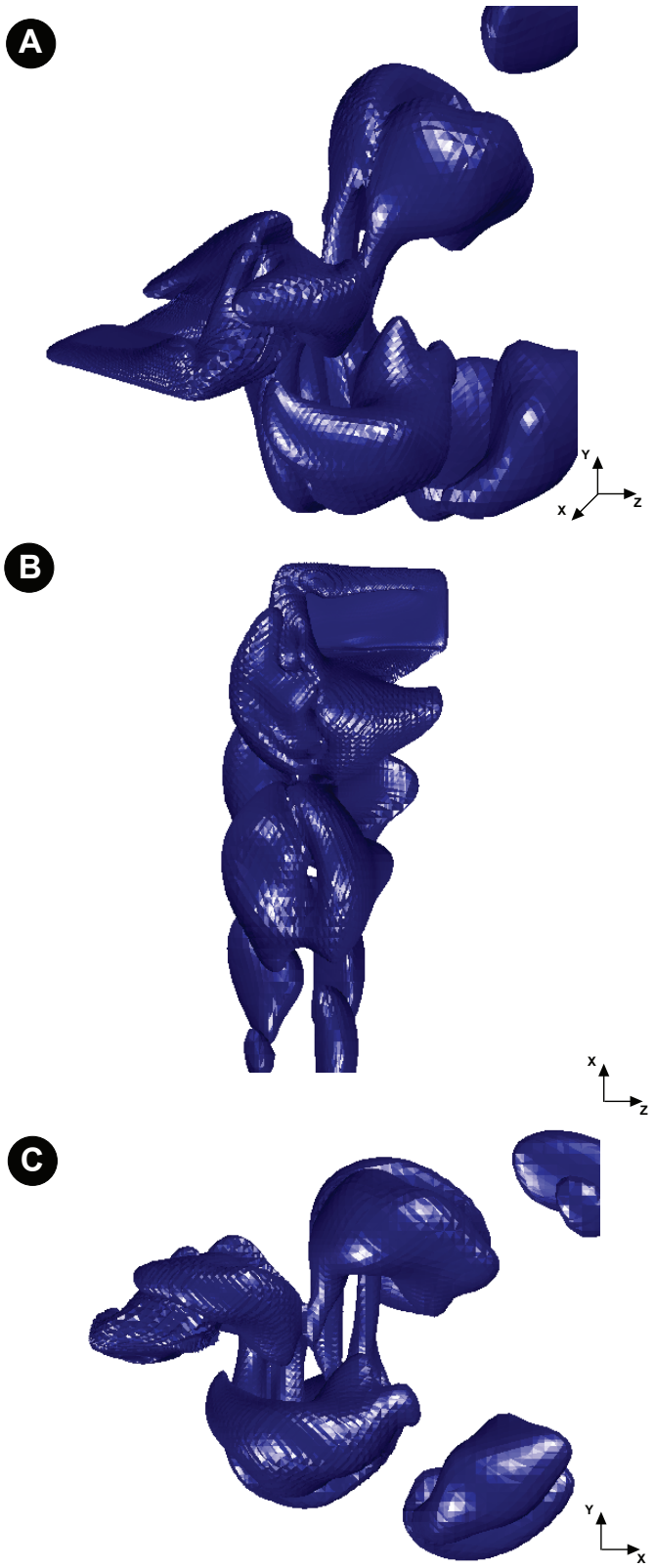

Figure 8.21: *Vortex topology for the rolling wing case (t=5.0). Flapping parameters:* $St = 0.38, f_{roll} = 1.0, Re = 500$. *A) Perspective view. B) Top view. In this view the right wing-tip corresponds to the hinged extreme. C) Side view.*

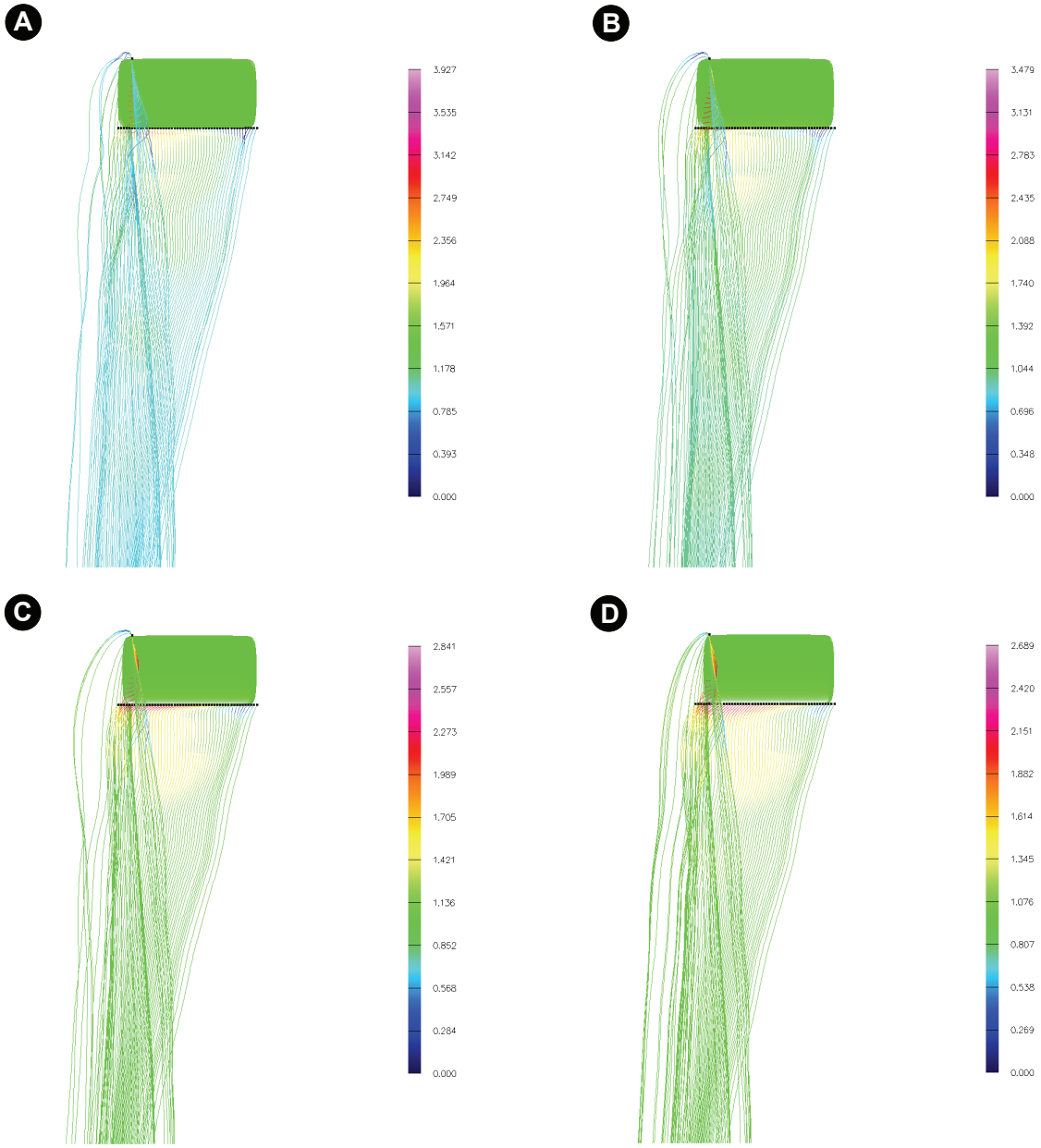

Figure 8.22: *Streamlines visualization during downstroke (top view), the streamlines are colored according to the velocity magnitude values. In this view the right wing-tip corresponds to the hinged extreme. Flapping parameters:* $St = 0.38, f_{roll} = 1.0, Re = 500$. *A) t=5.0 B) t=5.05 C) t=5.10 D) t=5.15.*

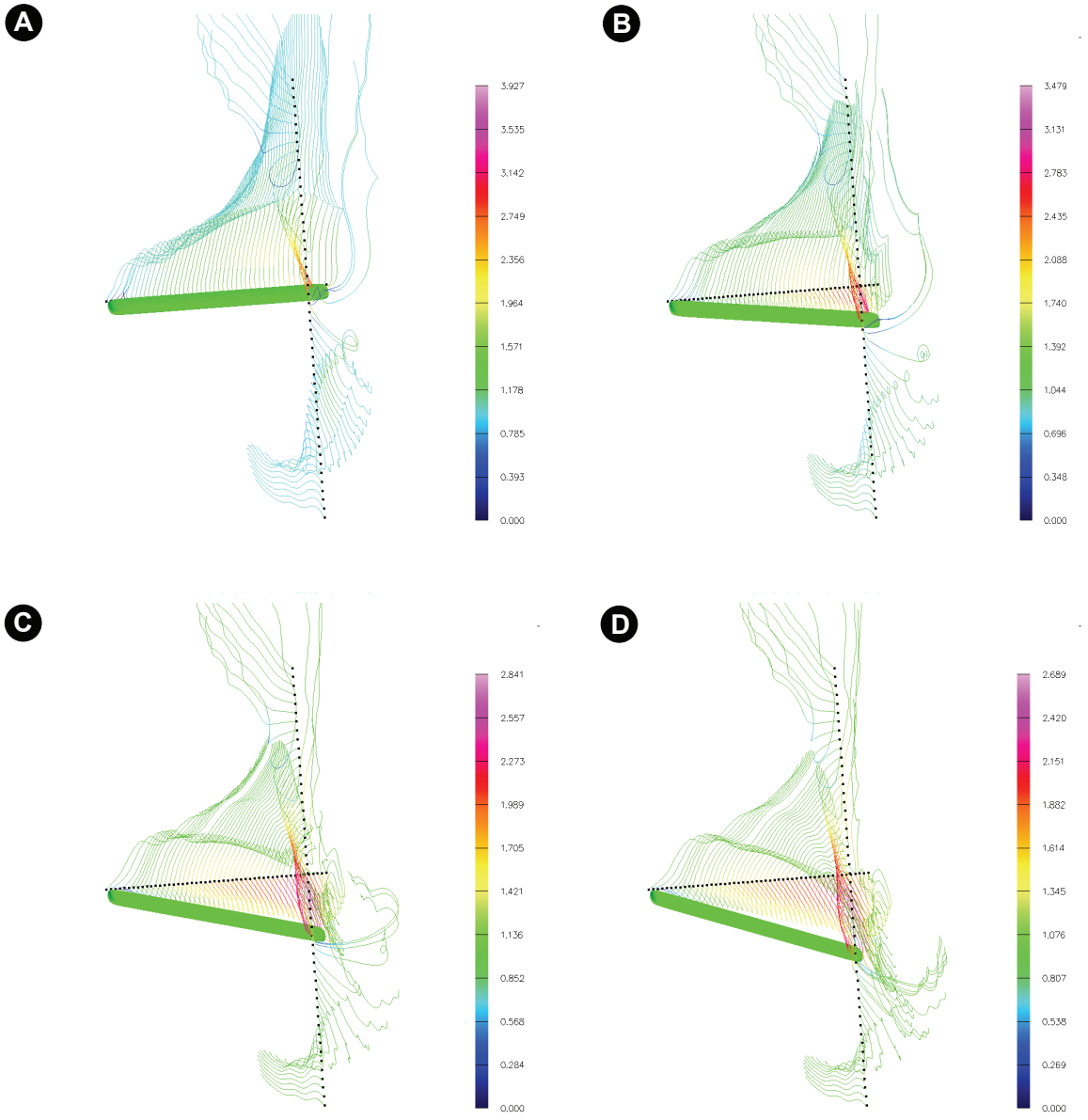

Figure 8.23: *Streamlines visualization during downstroke (perspective view), the streamlines are colored according to the velocity magnitude values. In this view the left wing-tip corresponds to the hinged extreme. Flapping parameters:* $St = 0.38, f_{roll} = 1.0, Re = 500$. *A) t=5.0 B) t=5.05 C) t=5.10 D) t=5.15.*

Chapter 9

Conclusions and Perspectives

9.1 Conclusions

In the last years, significant progress has been made in the understanding of the major aerodynamic features of flapping airfoils/wings by using experimental and computational approaches. Much of the work done by researchers has been focused on two-dimensional flapping airfoils and, as one would expect, the information available on three-dimensional flow features and aerodynamic performance of finite-span span wings is quite limited. This limitation is mainly due to the unsteady nature of the flow and the requirement of high resolution in regions of flow separation, vortex shedding and vortex wake evolution, which, added to the requirement of bigger and finer computational domains render three-dimensional simulations of flapping wings a very expensive problem from a computational point of view. Clearly, the flapping wing mechanism that has evolved in nature presents many interesting challenging problems. Platzer *et al.* [145], addressed the challenges that still remains open or need to be investigated more deeply for a complete understanding of flapping airfoils and wings aerodynamics. In [145] they listed the following areas of research:

1. Most flapping airfoil/wing propulsion computational studies and experiments have been limited to symmetrical NACA airfoils, or to a lesser extent, elliptical airfoils. The effect of airfoil geometry, remains to be systematically explored. Of particular importance is the physics governing the onset of vortex shedding from the leading edge and their interaction with the trailing edge vortices.

2. The three-dimensional flow features generated by finite-span wings present a great challenge due to the effort to visualize, measure and compute the complex flows generated by flapping wings. Of special importance are the spanwise flow features which may include the formation and shedding of spanwise vortices.

3. Flapping wing propulsion in nature involves flexible wings. However, the physics of the flow over flexible flapping wings is yet to be fully understood. Advances have to be made through carefully designed experiments and the development of flow solvers that incorporate efficient and effective fluid-structure coupling.

4. One key application of studying flapping wing aerodynamics is the determination of the conditions for *"optimal"* aerodynamic performance in terms of lift, thrust and propulsive efficiency. Despite substantial research efforts, there is still a lack of full understanding of the physics governing the *"optimal"* aerodynamic performance of flapping wings. Important

parameters that influence the aerodynamics of flapping wings include the shape of the wing, the stiffness of the wing, the type of flapping motion and Reynolds number.

5. Modeling of transition from laminar to turbulent flows is very difficult even for fixed wings applications. Detailed experimental information is needed on the formation of separation bubbles. Although considerable progress has been made in this area for steady airflow flows, very little is known about the behavior of separation bubbles on oscillating airfoils. It is inevitable that advances in separation bubble formation and transition flow modeling are required before the benefits of flapping wing propulsion can be fully exploited.

In this dissertation, we attempted to address most of the previous challenges, in order to contribute to a better comprehension of the mechanism of flapping airfoils/wings propulsion and the associated unsteady aerodynamics, independently of their possible practical applications.

In chapter 7, we explored the wake structures and aerodynamic performance of flapping airfoils. We studied the dependency of the wake structure and aerodynamic performance on the flapping and geometric parameters such as flapping frequency and amplitude, airfoil geometry and airfoil chord-wise flexibility, for airfoils undergoing pure heaving motion or flapping motion. All the qualitative and quantitative results presented in this chapter support the hypothesis that : *"flying and swimming animals cruise at a Strouhal number tuned for high power efficiency"* [182]. The enhanced efficiency range was found to be between Strouhal number values corresponding to $0.2 < St < 0.4$, with a maximum efficiency peak at approximately $St = 0.3$, which agrees with the observations of Nudds *et al.* [136], Rohr and Fish [155], Taylor *et al.* [182] and Triantafyllou *et al.* [193].

In that chapter, we assessed also the validity of the Strouhal number St as the fundamental aerodynamic single parameter insofar as high propulsive efficiency is concerned. It was found that St seems to be enough for wake signature characterization, but is not sufficient insofar as maximum efficiency is concerned. Both heaving amplitude h_a and heaving frequency f_h (hence the Strouhal number St and the reduced frequency k) should be adjusted separately.

The wake topology of a heaving airfoil was also characterized for a wide range of heaving amplitudes and heaving frequencies combinations. The wakes were classified as drag producing, neutral, thrust producing, deflected wake and jet-switching wake. It was found that the deflected wake topologies were highly reproducible and they were mainly found in a range of Strouhal number values between $0.5 < St < 0.8$. At higher St values, we encountered the aperiodic jet-switching wake, but we were not able to determine if the switching is random or periodic, basically due to the fact that the simulations have to be run for extremely long times in order to observe the onset of the jet-switching. It is important to mention that as far as the author is aware, very limited experimental and computational information exist on the jet-switching phenomenon; therefore, much work remains to be accomplished in this area. The reason for the deflected wake and jet-switching phenomenon seems to be the interaction of the vortices shed at the trailing edge at the high St where these phenomena are encountered.

Different behaviors on the aerodynamic performance for high flapping frequencies (low heaving amplitudes) and low flapping frequencies (high heaving amplitudes) were also observed. Firstly, at high flapping frequencies the LEV does not have sufficient time to grow, whereas at low flapping frequencies the vortex has a size which is a sizable fraction of the airfoil chord, before separating.

Thus the impact of the vortex on the pressure at the nose of the airfoil is dependent on the flapping frequency. Secondly, once the vortex separates it is convected downstream over the surface of the airfoil. Due to the low pressure in the vortex core this has the effect of maintaining thrust while the vortex is upstream of the airfoil maximum thickness point (where the airfoil surface is tilted upstream and the vortex low pressure creates an upstream suction force). Once passing this point, the airfoil surface is tilted downstream and the vortex contributes to drag rather than thrust. At high flapping frequencies, the vortex cannot be convected far downstream before the motion cycle creates another leading edge vortex on the opposite side of the airfoil, so the impact is lessened. At low flapping frequencies however, the vortex travels far downstream over the airfoil surface causing drag for a larger portion of the flapping cycle and therefore lowering the propulsive efficiency. For flapping motion, where the orientation of the airfoil surface is controlled by the relative amplitudes and phases of the motion, the LEV may create positive thrust for much longer portions of the flapping cycle and thus contribute towards the propulsive efficiency. Also, for flapping motion the average input power coefficient is less than the average input power coefficient for the heaving motion cases, resulting in an improved propulsive efficiency. It was also observed that the best propulsive efficiencies were obtained for high heaving amplitudes (between $0.8 < h_a < 1.2$), in contrast to the heaving motion were the propulsive efficiency is deteriorated as the heaving amplitude is increased above a value of $h_a = 0.2$.

In chapter 7, we also studied the effect of chord-wise flexibility on the aerodynamic performance of heaving airfoils. Thrust-indicative wake topologies were observed for the whole range of flexure amplitudes tested. The results have also shown that the propulsive efficiency is enhanced for values of $h_{flex} < 0.4$. This observation is in agreement with the results of Heatcote and Gursul [67, 68], where they found that adding a degree of flexibility increases both thrust and propulsive efficiency. They also suggested that birds, bats and insects may benefit aerodynamically from the flexibility of their wings.

Finally, the effect of airfoil cambering on the aerodynamic performance was assessed. It was found that this geometric parameter has a strong influence on the lift coefficient, while it has a small impact on the thrust coefficient and propulsive efficiency. Among all the asymmetric airfoils used, the NACA 6612 airfoil provided the best propulsive efficiency and average lift coefficient, which, along thrust generation, are the crucial factors if we are interested in flapping flight. The S1223 airfoil, which resembles the cross-section of the Seagull and Merganser wings (as observed by Liu *et al.* [114]), provided at high heaving frequencies the biggest average lift coefficient and very similar average thrust coefficient and propulsive efficiency values when compared to the other airfoils. On the other hand, at low heaving frequencies, the aerodynamic performance of the S1223 airfoil was deteriorated in comparison to the other airfoils, but regardless this fact, it stills produce reasonable values of average thrust coefficient and positive average lift coefficient.

In chapter 8, we extended the two-dimensional results presented in chapter 7 to three-dimensional rigid finite-span flapping wings. In this study, we investigated the aerodynamic forces and wake topology behind low aspect ratio flapping wings and their dependence on the Strouhal number and flapping parameters; we also established the best criteria for vortex identification. The simulations show that the wake of thrust producing, rigid finite-span flapping wings is formed by two sets of interconnected vortex ring loops that slowly convert into vortex rings as they are convected downstream. It was also observed that the vortex rings are themselves inclined with respect to the free-stream; the angle of inclination of the vortex rings is found to be in the direction of their motion and in the streamwise direction for thrust producing cases, whereas for drag producing

configurations the angle of inclination is opposite to the direction of travel of the incoming flow. The presence of thin contrails that link the vortex loops was also observed; these structures are segments of the wing-tip vortices and, as the vortex loops are convected downstream, these contrails become weaker and ultimately disappear. It was also observed that the wake topology of drag producing flapping wings, in fact, is very different from that of a thrust producing flapping wing; the main differences are the absence of vortex rings and the compactness of the wake. In general, the observed structures are qualitatively similar to those observed in the experiments of Parker *et al.* [138].

The effect of the wing aspect ratio AR on the aerodynamic forces of finite-span wings was also assessed. It was found that as we increase the wing AR, the aerodynamic forces also increase and this is chiefly due to the large area of high aspect ratio wings and to the relatively minor effect of three-dimensionality in long wings. This observation lead us to think that the assumption of two-dimensionality has some validity for birds and insects, where the wings of many species tend to have relatively large aspect ratio.

Finally, besides the heaving and coupled heaving-and-pitching motions, we also studied the root-flapping motion characteristic of some flying animals, a configuration that still remains almost unexplored. From the results obtained, it was found that root-flapping motion produces wake structures very similar to those of heaving or coupled heaving-and-pitching motions, but with the difference that the latter motions generate larger vortices and forces than root-flapping motion, presumably because the average velocity is higher across the span; otherwise the same wake regimes occurs at similar Strouhal numbers.

From a computational point of view, the biggest breakthrough contribution is the proposed gridding methodology, which allows us to efficiently handle single and multiple fixed or moving/deforming bodies in two and three space dimensions. The overlapping grids method, added to the accurate and stable numerical method used to numerically solve the incompressible Navier-Stokes equations and to the numerical tools implemented, constitutes a very powerful tool to solve fluid dynamics problems with fixed or moving/deforming boundaries in two and three space dimensions.

9.2 Perspectives for future work

There are a number of ways in which the current work could be extended and improved. Firstly, the large number of flapping parameters to be considered, even for simple two-dimensional kinematics (such as heaving frequency and amplitude, pitching frequency and amplitude, phase between pitching and heaving, pivot point location, airfoil geometry and so on) and the significant effect each can have on the observed aerodynamic performance, suggest the use of an automated optimization strategy such as gradient-descent based optimization methods or genetic algorithms. Also, the majority of the simulations were performed at low Reynolds number ($Re < 2000$), the extension of the current work to the higher Reynolds numbers characteristic of large birds and high speed fishes and cetaceans is an area of future research.

The fully parallelization of the moving grids solver using the MPI standard remains an open issue. For the moment the moving grids solver was parallelized using the OpenMP API specification,

which, while being the easiest way to parallelize a code, is highly limited to the number of processors available per computing node. In general, we used computing nodes with 4 processors, which provided us with a modest speedup of about 1.6 times. The full parallelization of the moving grids solver using the MPI specification will allow us to run the solver on large computing clusters; therefore, we should be able to scale the problems and expect to achieve higher speedups.

Always related to the solver, is the use of multigrid algorithms to solve the pressure equation. During the development and implementation of the numerical tools used, we tested a multigrid solver and the speed-up obtained was remarkably good (about 10 times faster than the serial code), but we found several constrains when using geometries with sharp edges or several patches. These constrains were mainly related to the interpolation between the component grids in the overlapping grid system. Hence, the tune-up of the multigrid solver and the solution of the interpolation issues when coarsening the grid during the multigrid algorithm, is a very promising area where much work has to be done.

Finally, three-dimensional numerical simulations were performed in this dissertation, but a large qualitative and quantitative scale survey as that performed for the two-dimensional case remains to be accomplished and, to the author's knowledge, such as extensive study has not yet been conducted.

Appendix A

Companion DVD with selected animations

Selected animations from the results obtained in chapters 7 and 8 are included in the attached DVD. Hereafter we list the included animations:

- Two-dimensional simulations
  - Heaving airfoil at $St = 0.15$, $h_a = 0.30$, $Re = 1100$
  - Heaving airfoil at $St = 0.30$, $h_a = 0.15$, $Re = 1100$
  - Heaving airfoil at $St = 0.30$, $h_a = 0.35$, $Re = 1100$
  - Heaving airfoil at $St = 0.40$, $h_a = 0.40$, $Re = 1100$
  - Heaving airfoil at $St = 0.50$, $h_a = 0.20$, $Re = 1100$
  - Heaving airfoil at $St = 0.50$, $h_a = 0.30$, $Re = 1100$
  - Heaving airfoil at $St = 0.60$, $h_a = 0.10$, $Re = 1100$
  - Heaving airfoil at $St = 0.60$, $h_a = 0.40$, $Re = 1100$
  - Heaving airfoil at $St = 0.90$, $h_a = 0.45$, $Re = 1100$
  - Heaving airfoil at $St = 0.90$, $h_a = 0.375$, $Re = 1100$
  - Heaving airfoil at $St = 1.20$, $h_a = 0.60$, $Re = 1100$

  - Heaving-and-pitching airfoil at $St = 0.20$, $h_a = 0.40$, $\alpha_a = 5°$, $\varphi = 90°$, $Re = 1100$
  - Heaving-and-pitching airfoil at $St = 0.20$, $h_a = 1.00$, $\alpha_a = 10°$, $\varphi = 90°$, $Re = 1100$
  - Heaving-and-pitching airfoil at $St = 0.20$, $h_a = 1.00$, $\alpha_a = 20°$, $\varphi = 90°$, $Re = 1100$
  - Heaving-and-pitching airfoil at $St = 0.20$, $h_a = 0.40$, $\alpha_a = 30°$, $\varphi = 90°$, $Re = 1100$
  - Heaving-and-pitching airfoil at $St = 0.20$, $h_a = 0.60$, $\alpha_a = 40°$, $\varphi = 90°$, $Re = 1100$

  - Heaving-and-pitching airfoil at $St = 0.30$, $h_a = 1.00$, $\alpha_a = 5°$, $\varphi = 90°$, $Re = 1100$
  - Heaving-and-pitching airfoil at $St = 0.30$, $h_a = 1.40$, $\alpha_a = 15°$, $\varphi = 90°$, $Re = 1100$
  - Heaving-and-pitching airfoil at $St = 0.30$, $h_a = 1.00$, $\alpha_a = 25°$, $\varphi = 90°$, $Re = 1100$
  - Heaving-and-pitching airfoil at $St = 0.30$, $h_a = 1.20$, $\alpha_a = 30°$, $\varphi = 90°$, $Re = 1100$

- Heaving-and-pitching airfoil at $St = 0.30$, $h_a = 0.40$, $\alpha_a = 40°$, $\varphi = 90°$, $Re = 1100$

- Heaving-and-pitching airfoil at $St = 0.40$, $h_a = 0.40$, $\alpha_a = 5°$, $\varphi = 90°$, $Re = 1100$
- Heaving-and-pitching airfoil at $St = 0.40$, $h_a = 0.40$, $\alpha_a = 20°$, $\varphi = 90°$, $Re = 1100$
- Heaving-and-pitching airfoil at $St = 0.40$, $h_a = 1.00$, $\alpha_a = 20°$, $\varphi = 90°$, $Re = 1100$
- Heaving-and-pitching airfoil at $St = 0.40$, $h_a = 1.00$, $\alpha_a = 30°$, $\varphi = 90°$, $Re = 1100$
- Heaving-and-pitching airfoil at $St = 0.40$, $h_a = 2.00$, $\alpha_a = 50°$, $\varphi = 90°$, $Re = 1100$

- Heaving-and-pitching airfoil at $St = 0.25$, $h_a = 0.25$, $\alpha_a = 10°$, $\varphi = 85°$, $Re = 1100$
- Heaving-and-pitching airfoil at $St = 0.25$, $h_a = 0.25$, $\alpha_a = 10°$, $\varphi = 95°$, $Re = 1100$
- Heaving-and-pitching airfoil at $St = 0.25$, $h_a = 0.25$, $\alpha_a = 10°$, $\varphi = 105°$, $Re = 1100$

- Heaving NACA 2212 airfoil at $St = 0.30$, $h_a = 0.10$, $Re = 1100$
- Heaving NACA 4412 airfoil at $St = 0.30$, $h_a = 0.30$, $Re = 1100$
- Heaving NACA 6612 airfoil at $St = 0.30$, $h_a = 0.30$, $Re = 1100$
- Heaving SELIG S1223 airfoil at $St = 0.40$, $h_a = 0.10$, $Re = 1100$
- Heaving SELIG S1223 airfoil at $St = 0.40$, $h_a = 0.30$, $Re = 1100$

- Heaving flexible airfoil at $St = 0.30$, $h_a = 0.25$, $h_{flex} = 0.0$, $Re = 1100$
- Heaving flexible airfoil at $St = 0.30$, $h_a = 0.25$, $h_{flex} = 0.1$, $Re = 1100$
- Heaving flexible airfoil at $St = 0.30$, $h_a = 0.25$, $h_{flex} = 0.3$, $Re = 1100$
- Heaving flexible airfoil at $St = 0.30$, $h_a = 0.25$, $h_{flex} = 0.5$, $Re = 1100$

- Three-dimensional simulations

  - Heaving wing at $St = 0.15$, $h_a = 0.075$, $Re = 500$
  - Heaving wing at $St = 0.25$, $h_a = 0.25$, $Re = 500$
  - Heaving wing at $St = 0.35$, $h_a = 0.35$, $Re = 500$
  - Heaving wing at $St = 0.35$, $h_a = 0.175$, $Re = 500$
  - Heaving wing at $St = 0.50$, $h_a = 0.25$, $Re = 500$

  - Heaving-and-pitching wing at $St = 0.25$, $h_a = 0.15$, $\alpha_a = 20°$, $\varphi = 90°$, $Re = 500$

  - Rolling wing (Root-flapping) at $St = 0.10$, $\phi = 12.5°$, $Re = 500$
  - Rolling wing (Root-flapping) at $St = 0.38$, $\phi = 45.0°$, $Re = 500$

These and other simulations can be also viewed at the author's website:
http://www.dicat.unige.it/guerrero

Bibliography

[1] M. Aftosmis. Solution adaptive cartesian grid methods for aerodynamic flows with complex geometries. *Von Karman Institute for Fluid Dynamics, Lecture Series 1997-02*, 1997.

[2] M. Aftosmis, M. Berger, and J. Melton. Robust and efficient cartesian mesh generation for component-based geometry. *AIAA Paper 97-0196-CP*, 1997.

[3] D. E. Alexander and S Vogel. *Nature's Flyers*. Johns Hopkins University Press, 2002.

[4] R. Alexander. *Principles of Animal Locomotion*. Princeton University Press, 2005.

[5] J. D. Anderson. *Computational Fluid Dynamics: The Basics with Applications*. McGraw-Hill, 1995.

[6] J. D. Anderson. *Fundamentals of Aerodynamics*. McGraw-Hill, 2001.

[7] J. M. Anderson, K. Streitlien, D. S. Barrett, and M. S. Triantafyllou. Oscillating foils of high propulsive efficiency. *Journal of Fluids Mechanics*, 360:41–72, 1998.

[8] H. Ashley and M. Landahl. *Aerodynamics of Wings and Bodies*. Dover, 1965.

[9] A. Azuma. *The Biokinetics of Flying and Swimming*. AIAA, 2006.

[10] J. Babcock. Free flight store simulation using beggar. Master's thesis, Department of Aeronautics and Astronautics. Air Force Institute of Technology. Wright-Patterson Air Force Base, Ohio, 2006.

[11] S. Balay, K. Buschelman, V. Eijkhout, W. Gropp, D. Kaushik, M. Knepley, L. McInnes, B. Smith, and H. Zhang. Petsc users manual. Technical report, Argonne National Laboratory, ANL-95/11 - Revision 2.3.3, 2007.

[12] R. Barrett, M. Berry, T. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. van der Vorst. *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*. SIAM, 1993.

[13] D. Belk. The role of overset grids in the development of the general purpose cfd code, in: Surface modeling, grid generation, and related issues in computational fluid dynamics (cfd) solution. Technical report, NASA, NASA CP-3291, 1995.

[14] D. M. Belk. *Unsteady Three-Dimensional Euler Equations Solutions on Dynamic Blocked Grids*. PhD thesis, Mississippi State University, USA, 1986.

[15] J. Benek, P. Buning, and J. Steger. A 3d chimera grid embedding technique. *AIAA Paper 85-1523-CP*, 1985.

[16] M. Benzi, D. Szyld, and A. Van Duiv. Ordering for incomplete factorization preconditioning of nonsymmetric problems. *SIAM Journal on Scientific Computing*, 20:1652–1670, 1999.

[17] M. Berger. On conservation at grid interfaces. Technical report, ICASE, Report No. 84-43, 1984.

[18] M. Berger and J. Oliger. Adaptive mesh refinement for hyperbolic partial differential equations. *Journal of Computational Physics*, 53:482–512, 1984.

[19] M. Bern and D. Eppstein. Mesh generation and optimal triangulation. *Computing in Euclidean Geometry*, pages 23–90, 1992.

[20] A. Betz. Ein beitrag zur erklarung des segelfluges. *Zeitschrift fur Flugtechnik und Motorluftschiffahrt*, 3:269–272, 1912.

[21] A. Biewener. *Animal Locomotion*. Oxford Animal Biology Series, 2003.

[22] W. Birnbaum. Das ebene problem des schlagenden fluels. *Zeitschrift fur Angewandte Mathematik und Mechanik*, 4:277–292, 1924.

[23] J. Bratt. Flow patterns in the wake of an oscillating airfoil. Technical report, Aeronautical Research Council, Report R2773, 1950.

[24] M. Braza, P. Chassaing, and H. Minh. Numerical study and physical analysis of the pressure and velocity fields in the near wake of a circular cylinder. *Journal of Fuid Mechanics*, 165:79–130, 1986.

[25] D. L. Brown, R. Cortez, and M. Minion. Accurate projection methods for the incompressible navier-stokes equations. *Journal of Computational Physics*, 168:464–499, 2001.

[26] D.L. Brown and W. D. Henshaw. Overture: Object-oriented tools for solving cfd and combustion problem. In *High performance computing 1998, Grand challenges in computer simulation, Society for Computer Simulation International*, 1998.

[27] G. Browning. A comparison of three numerical methods for solving differential equations on the sphere. *Monthly Weather Review*, 117:1058–1075, 1989.

[28] X. Cai, W. Gropp, D. Keyes, R. Melvin, and D. Young. Parallel newton-krylov-schwarz algorithms for the transonic full potential equation. *SIAM Journal on Scientific Computing*, 19:246–265, 1998.

[29] X. Cai, W. Gropp, D. Keyes, and M. Tidriri. Newton-krylov-schwarz methods in cfd. In *Proceedings of the International Workshop on the Navier-Stokes Equations, Notes in Numerical Fluid Mechanics*, 1994.

[30] D. Calhoun and Z. Wang. A cartesian grid method for solving the two-dimensional streamfunction-vorticity equations in irregular regions. *Journal of Computational Physics*, 176:231–275, 2002.

[31] L. W. Carr and M. S. Chandrasekhara. Compressibility effects of dynamic stall. *Progress in Aeronautical Sciences*, 32:523–573, 1996.

[32] W. Chan, R. Gomez, P. Buning, and S. S. Rogers. Best practices in overset grid generation. *AIAA Paper 2002-3191-CP*, 2002.

[33] W. M. Chan. Cad interface, strand grid technology and other new developments in chimera grid tools 2.0. In *Proceedings of the 8th Symposium on Overset Composite Grid and Solution Technology, Houston*, 2006.

[34] W. M. Chan, S. Rogers, S. Nash, P. Buning, R. Meakin, D. Boger, and S. Pandya. Chimera grid tools user's manual. Technical report, NASA Advanced Supercomputing (NAS) division, 2007.

[35] R. Chandra, L. Dagum, D. Kohr, D. Mayden, J. McDonald, and R. Menon. *Parallel Programming in OpenMP*. Academic Press.

[36] A. Chapman, Y. Saad, and L. Wigton. High-order ilu preconditioners for cfd problems. *International Journal for Numerical Methods in Fluids*, 33:767–788, 2000.

[37] G. Chesshire and W. Henshaw. Composite overlapping meshes for the solution of partial differential equations. *Journal of Computational Physics*, 90:1–64, 1990.

[38] G. Chesshire and W. D. Henshaw. A scheme for conservative interpolation on overlapping grids. *SIAM Journal on Scientific Computing*, 15:819–845, 1994.

[39] J. Choi, R. Oberoi, J. Edwards, and J. Rosati. An immersed boundary method for complex incompressible flows. *Journal of Computational Physics*, 224:757–784, 2007.

[40] M. Chong, A. Perry, and B. Cantwell. A general classification of three-dimensional flow fields. *Physics of Fluids*, 2:765–777, 1990.

[41] K. Corfeld, R. Strawn, and L. Long. Computational analysis of a prototype martian rotorcraft experiment. *AIAA Paper 2002-2815-CP*, 2002.

[42] M. Coutanceau and R. Bouard. Experimental determination of the main features of the viscous flow in the wake of a circular cylinder in uniform translation. part 1. steady flow. *Journal of Fluid Mechanics*, 79:257–272, 1973.

[43] W. A. Davis. Nano air vehicles: A technological forecast. *USAF, Blue Horizons Paper. Center for Strategy and Technology, Air War College*, 2007.

[44] M. H. Dickinson. Solving the mystery of insect flight. *Scientific American*, June:35–41, 2001.

[45] D. Drikakis and W. Rider. *High-Resolution Methods for Incompressible and Low-Speed Flows*. Springer, 2004.

[46] Robert Dudley. *The Biomechanics of Insect Flight: Form, Function, Evolution*. Princeton University Press, 2002.

[47] J. A. Ekaterinaris and M. F. Platzer. Computational prediction of airfoil dynamic stall. *Progress in Aeronautical Sciences*, 33:759–846, 1997.

[48] C. P. Ellington. The novel aerodynamics of insect flight applications to micro-air vehicles. *The Journal of Experimental Biology*, 202:3439–3448, 1999.

[49] J. Emblemsvag, R. Suzuki, and G. Candler. Numerical simulation of flapping micro air vehicles. *AIAA Paper 2002-3197-CP*, 2002.

[50] M. Farrashkhalvat and J. P. Miles. *Basic Structured Grid Generation with an Introduction to Unstructured Grid Generation*. Butterworth-Heinemann, 2003.

[51] P. Fast and W. D. Henshaw. Time accurate simulation of viscous flow around deforming bodies using overset grids. *AIAA Paper 2001-2604-CP*, 2001.

[52] P. Fast and M. J. Shelley. A moving overset grid method for interface dynamics applied to non-newtonian hele-shaw flow. *Journal of Computational Physics*, 195:117–142, 2004.

[53] C. A. J. Fletcher. *Computational Techniques for Fluid Dynamics: Volume 2: Specific Techniques for Different Flow Categories*. Springer, 1996.

[54] B. Fornberg. A numerical study of steady viscous flow past a circular cylinder. *Journal of Fuid Mechanics*, 98:819–855, 1980.

[55] Ch. Forster, W. A. Wall, and E. Ramm. On the geometric conservation law in transient flow calculations on deforming domains. *International Journal for Numerical Methods in Fluids*, 50:1369–1379, 2005.

[56] C. J. Freitas and S. R. Runnels. Simulation of fluid-structure interaction using patched-overset grids. *Journal of Fluids and Structures*, 13:191–207, 1999.

[57] I. Garrick. Propulsion of a flapping and oscillating airfoil. Technical report, NACA, Technical Report No. 567, 1936.

[58] J. P. Giesing. Nonlinear two-dimensional unsteady potential flow with lift. *Journal of Aircraft*, 5:135–143, 1968.

[59] G. Golub and C. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, 1996.

[60] P. M. Gresho. Incompressible fluid dynamics: some fundamental formulation issues. *Annual Review of Fluid Mechanics*, 23:413–453, 1991.

[61] P. M. Gresho and R. L. Sani. On the pressure boundary conditions for the incompressible navier-stokes equations. *International Journal for Numerical Methods in Fluids*, 7:1111–1145, 1987.

[62] L. Guglielmini and P. Blondeaux. Propulsive efficiency of oscillating foils. *European Journal of Mechanics*, 23:255–278, 2004.

[63] R. Haimes and D. Kenwright. On the velocity gradient tensor and fluid feature extraction. *AIAA Paper 1999-3288-CP*, 1999.

[64] K. C. Hall and S. R. Hall. A rational engineering analysis of the efficiency of flapping flight. *Fixed and Flapping Wing Aerodynamics for Micro Air Vehicles Applications*, pages 249–274, 2000.

[65] S. Hamdi, W. Schiesser, and G. Griffiths. *Method of lines*. Scholarpedia, 2007.

[66] F. Harlow and J. Welch. Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. *Journal of Computational Physics*, 8:2182–2189, 1965.

[67] S. Heathcote and I. Gursul. Flexible flapping airfoil propulsion at low reynolds numbers. *AIAA Paper 2005-1405-CP*, 2005.

[68] S. Heathcote and I. Gursul. Flexible flapping airfoil propulsion at low reynolds numbers. *AIAA Journal*, 45:1066–1079, 2007.

[69] S. Heathcote and I. Gursul. Jet switching phenomenon for a periodically plunging airfoil. *Physics of Fluids*, 19:027104–1 – 027104–12, 2007.

[70] W. Henshaw. Ogen: An overlapping grid generator for overture. Technical report, Lawrence Livermore National Laboratory, Research Report UCRL-MA-132237, 1998.

[71] W. Henshaw. Overblown: A fluid flow solver for overlapping grids, reference guide, version 1.0. Technical report, Lawrence Livermore National Laboratory, Research Report UCRL-MA-132237, 2003.

[72] W. Henshaw. A solver for the incompressible navier-stokes equations. reference manual. Technical report, Lawrence Livermore National Laboratory, Research Report, 2006.

[73] W. Henshaw and D. Schwendeman. An adaptive numerical scheme for high speed reactive flow on overlapping grids. *Journal of Computational Physics*, 191:420–447, 2003.

[74] W. Henshaw and D. Schwendeman. Moving overlapping grids with adaptive mesh refinement for high-speed reactive and non-reactive flow. *Journal of Computational Physics*, 216:744–779, 2006.

[75] W. D. Henshaw. The overture hyperbolic grid generator user guide, version 1.0. Technical report, Lawrence Livermore National Laboratory, Research Report UCRL-MA-134240, 2003.

[76] W. D. Henshaw. On multigrid for overlapping grid. *SIAM Journal on Scientific Computing*, 26:1547–1572, 2005.

[77] W. D. Henshaw and D. W. Schwendeman. Detonation initiation: modeling, computation and mechanisms. In *Proceedings of CHT 4*, 2004.

[78] W.D. Henshaw. A fourth-order accurate method for the incompressible navier-stokes equations on overlapping grids. *Journal of Computational Physics*, 113:13–25, 1994.

[79] W.D. Henshaw and H.-O. Kreiss. Analysis of a difference approximation for the incompressible navier-stokes equations. Technical report, Los Alamos National Laboratory, Research Report LA-UR-95-3536, 1995.

[80] W.D. Henshaw, H.-O. Kreiss, and L.G.M. Reyna. A fourth-order accurate difference approximation for the incompressible navier-stokes equations. *Journal of Computer and Fluids*, 23:575–593, 1994.

[81] W.D. Henshaw and N. A. Petersson. A split-step scheme for the incompressible navier-stokes equations. In *Numerical Solutions of Incompressible Flows*, 2003.

[82] J. L. Hess and A. Smith. Calculation of potential flow about arbitrary bodies. *Progress in Aeronautical Sciences*, 8:1–138, 1966.

[83] C. Hirt, A. Amsden, and J. Cook. An arbitrary lagrangian-eulerian computing method for all flow speeds. *Journal of Computational Physics*, 14, 1974.

[84] K. A. Hoffmann and S.T. Chiang. *Computational Fluid Dynamics, Volume I.* EESBooks, 2004.

[85] K. A. Hoffmann and S.T. Chiang. *Computational Fluid Dynamics, Volume II.* EESBooks, 2004.

[86] T.Y. Hou, T.Y. Wu, and V.G. Stredie. Mathematical modeling and simulation of aquatic and aerial animal locomotion. *Journal of Computational Physics*, 225:1603–1631, 2007.

[87] F. S. Hover, O. Haugsdal, and M. S. Triantafyllou. Effect of angle of attack profiles in flapping foil propulsion. *Journal of Fluids and Structures*, 19:37–47, 2004.

[88] K. Isogai and Y. Shinmoto. Study on aerodynamic mechanism of hovering insects. *AIAA Paper 2001-2470-CP*, 2001.

[89] K. Isogai, Y. Shinmoto, and Y. Watanabe. Effects of dynamic stall on propulsive efficiency and thrust of a flapping airfoil. *AIAA Journal*, 37:1145–1151, 2000.

[90] C. Jackson. A finite element study on the onset of vortex shedding in flow past variously shaped bodies. *Journal of Fluid Mechanics*, 182:23–45, 1987.

[91] J. Jeong and F. Hussain. On the identification of a vortex. *Journal of Fluids Mechanics*, 285:69–94, 1995.

[92] H. Johnston and J. Liu. Finite difference schemes for incompressible flow based on local pressure boundary conditions. *Journal of Computational Physics*, 180:120–154, 2002.

[93] K. D. Jones, C. M. Dohring, and M. F. Platzer. Wake structures behind plunging airfoils: A comparison of numerical and experimental results. *AIAA Paper 96-0078-CP*, 1996.

[94] K. D. Jones, C. M. Dohring, and M. F. Platzer. An experimental and computational investigation of the knoller-betz effect. *AIAA Journal*, 36:1240–1246, 1998.

[95] K. D. Jones, S. J. Duggan, and M. F. Platzer. Flapping-wing propulsion for a micro-air vehicle. *AIAA Paper 2001-0126-CP*, 2001.

[96] K. D. Jones, J. Lai, I. H. Tuncer, and M. F. Platzer. Computational and experimental investigation of flapping-foil propulsion. In *1st International Symposium on Aqua Bio-Mechanisms/International Seminar on Aqua Bio-Mechanisms, Tokai University Pacific Center*, 2000.

[97] K. D. Jones and M. F. Platzer. Numerical computation of flapping-wing propulsion and power extraction. *AIAA Paper 97-0826-CP*, 1997.

[98] J. Katz and A. Plotkin. *Low Speed Aerodynamics.* Cambridge University Press, 2001.

[99] R. Katzmayr. Effect of periodic changes of angles of attack on behavior of airfoils. Technical report, NACA, Technical Report No. 147, 1922.

[100] C. Kiris, S. Rogers, and I. Chang. Computational approach for probing the flow through artificial heart devices. *Journal of Biomedical Engineering*, 119:452–460, 1997.

[101] R. Knoller. Die gesetze des luftwiderstandes. *Flug und Motortechnik Wien*, 3:1–7, 1909.

[102] P. Knupp and K. Salari. Code verification by the method of manufactured solutions. Technical report, Sandia National Laboratories Report, SANDIA REPORT SAND2000-1444, June 2000.

[103] M. Koochesfahani. Vortical patterns in the wake of an oscillation foil. *AIAA Journal*, 27:1200–1205, 1989.

[104] B. Kreiss. Construction of a curvilinear grid. *SIAM J. Sci. Stat. Comput.*, 4 (2):270–279, 1983.

[105] J. Lai and M. F. Platzer. Jet characteristics of a plunging airfoil. *AIAA Journal*, 37:1529–1537, 1999.

[106] J. Lee, C. Kim, and K. H. Kim. Design of flapping airfoil for optimal aerodynamic performance in low-reynolds number flows. *AIAA Journal*, 44:1960–1972, 2006.

[107] F. Lehmann. The mechanisms of lift enhancement in insect flight. *Naturwissenschaften (2004)*, 91:101–122, 2004.

[108] G. C. Lewin and H. Haj-Hariri. Modelling thrust generation of a two-dimensional heaving airfoil in a viscous flow. *Journal of Fluid Mechanics*, 492:339–362, 2003.

[109] M. Lighthill. Aquatic animal propulsion of high hydromechanical efficiency. *Journal of Fluid Mechanics*, 44:265–301, 1970.

[110] O. Lilienthal. *Birdflight as the Basis of Aviation*. Markowski International Publishers, 2001.

[111] C. Liu, X. Zheng, and C. Sung. Preconditioned multigrid methods for unsteady incompressible flows. *Journal of Computational Physics*, 139:33–57, 1998.

[112] H. Liu, C. Ellington, K. Kawachi, C. Van Den Berg, and A. Willmot. A computational fluid dynamic study of hawkmoth hovering. *Journal of Experimental Biology*, 201:461–477, 1998.

[113] H. Liu and K. Kawachi. A numerical study of undulatory swimming. *Journal of Computational Physics*, 155:223–247, 1999.

[114] T. Liu, K. Kuykendoll, R. Rhew, and S. Jones. Avian wings. *AIAA Paper 2004-2186-CP*, 2004.

[115] K. B Lua, T. T Lim, K. S Yeo, and G. Y Oo. Wake-structure formation of a heaving two-dimensional elliptic airfoil. *AIAA Journal .*, 45:1571–1583, 2007.

[116] R. Maple and D. Belk. A new approach to domain decomposition, the beggar code. In *4th International Conference on Numerical Grid Generation in Computational Fluid Dynamics and Related Fields, Wales*, 1994.

[117] W. J. McCroskey, K. W. McAlister, L. W. Carr, and S. L. Pucci. An experimental study of dynamics stall on advanced airfoil section. Technical report, NASA, Report TM-84245, 1982.

[118] J. M. McDonough. *Lectures in Computational Fluid Dynamics of Incompressible flow: Mathematics, Algorithms and Implementations.* Department of Mechanical Engineering and Mathematics. University of Kentucky, 2007.

[119] J. H. McMasters and M. J. Henderson. Low-speed single-element airfoil synthesis. Technical report, NASA SP-2085, 1979.

[120] J. M. McMichael and M. S. Francis. Micro air vehicles - toward a new dimension in flight. Technical report, Defense Advanced Research Projects Agency (DARPA), 1997.

[121] R. Meakin. Moving body overset grid methods for complete aircraft tilt rotor simulations. *AIAA Paper 93-3350-CP*, 1993.

[122] R. Meakin and N. Suhs. Unsteady aerodynamic simulation of multiple bodies in relative motion. *AIAA Paper 89-1996-CP*, 1989.

[123] J. Melton. *Automated Three-Dimensional Cartesian Grid Generation and Euler Flow Solutions for Arbitrary Geometries.* PhD thesis, Univ. CA. Davis, CA, USA, 1996.

[124] J. Miao and M. M. Ho. Effect of flexure on aerodynamic propulsive efficiency of flapping flexible airfoil. *Journal of Fluids and Structures*, 22:401–419, 2006.

[125] R. Michelson and M. Naqvi. Extraterrestrail flight (entompter-based mars surveyor). *Von Karman Institute for Fluid Dynamics, Lecture Series Nov 24-28, 2003.*

[126] P. Moin and J. Kim. On the numerical solution of time-dependent viscous incompressible flows involving solid boundaries. *Journal of Computational Physics*, 35:381–392, 1980.

[127] T. J. Mueller. *Fixed and Flapping Wing Aerodynamics for Micro Air Vehicle Applications.* AIAA (American Institute of Aeronautics and Astronautics), 2002.

[128] M. S. Murman, Y. M. Rizk, and L. B. Schiff. Coupled numerical simulation of the external and engine inlet flows for the f-18 at large incidence. *AIAA Paper 92-2621-CP*, 1992.

[129] A. Nejat and C. Ollivier-Gooch. On preconditioning of newton-gmres algorithm for a higher-order accurate unstructured solver. *Proceedings of the Fourteenth Annual Conference of the Computational Fluid Dynamics Society of Canada*, 2006.

[130] R. Noack. Suggar: A general capability for moving body overset cfd. In *Proceedings of 7th Symposium on Overset Composite Grid and Solution Technology, California*, 2004.

[131] R. Noack. Current status of dirtlib and suggar. In *Proceedings of the 8th Symposium on Overset Composite Grid and Solution Technology, Houston*, 2006.

[132] R. Noack. Integrated overset capability using dirtlib and libsuggar. In *Proceedings of the 9th Symposium on Overset Composite Grid and Solution Technology, Pennsylvania*, 2008.

[133] R. Noack, D. O'Gwynn, and N. Doddamani. Isedover: An integrated simulation environment for overset cfd. In *Proceedings of the 6th Symposium on Overset Composite Grids and Solution Technology, Florida*, 2002.

[134] R. Norberg. Hovering flight of the dragonfly aeschna juncea l., kinematics and dynamics. *Swimming and Flying in Nature*, 2, 1975.

[135] U. M. Norberg. *Vertebrate Flight: Mechanics, Physiology, Morphology, Ecology and Evolution*. Springer, 1990.

[136] R. L. Nudds, G. K. Taylor, and A. R. Thomas. Tuning of strouhal number for high propulsive efficiency accurately predicts how wingbeat frequency and stroke amplitude relate and scale with size and flight speed in birds. *Proc. Biol. Sci.*, 7:2071–2076, 2004.

[137] Steve Owen. *Meshing Research Corner*. http://www.andrew.cmu.edu/user/sowen/mesh.html.

[138] K. Parker, J. Soria, and K. von Ellenrieder. Thrust measurements from a finite-span flapping wing. *AIAA Journal*, 45:58–70, 2007.

[139] D. Pearce, S. Atanley, F. Martin, R. Gomex, G. Beau, and P. Buning. Development of a large scale chimera grid system for the space shuttle launch vehicle. *AIAA Paper 93-0533-CP*, 1993.

[140] G. Pedro, A. Suleman, and N. Djilali. A numerical study of the propulsive efficiency of a flapping hydrofoil. *International Journal for Numerical Methods in Fluids*, 42:493–526, 2003.

[141] N. Petersson. Hole-cutting for three-dimensional overlapping grids. *Journal of Computational Physics*, 21:646–665, 1999.

[142] N. Petersson. Stability of pressure boundary conditions for stokes and navier-stokes equations. *Journal of Computational Physics*, 172:40–70, 2001.

[143] N. Petersson. An algorithm for assembling overlapping grid systems. *Journal of Computational Physics*, 20:646–665, 2002.

[144] N. A. Petersson. A numerical method to calculate the two-dimensional flow around an underwater obstacle. *Journal of Numerical Analysis*, 29:20–31, 1992.

[145] M. Platzer, K. Jones, J. Young, and J. J. Lai. Flapping-wing aerodynamics: Progress and challenges. *AIAA Journal*, 46:2136–2149, 2008.

[146] M. F. Platzer and K. D. Jones. The unsteady aerodynamics of flapping-foil propellers. In *9th International Symposium on Unsteady Aerodynamics, Aeroacoustics and Aeroelasticity of Turbomachines, Ecole Centrale de Lyon*, 2000.

[147] M. F. Platzer, K. D. Jones, and T. G. Lund. Experimental and computational investigation of flapping wing propulsion for micro-air vehicles. In *Symposium of Low-Reynolds Number Vehicles, University of Notre Dame*, 2000.

[148] M. F. Platzer, K. S. Neace, and C. K. Pang. Aerodynamic analysis of flapping wing propulsion. *AIAA Paper 93-0484-CP*, 1993.

[149] N. Prewitt, D. Belk, and W. Shyy. Parallel computing of overset grids for aerodynamic problems with moving objects. *Progress in Aerospace Sciences*, 36:117–172, 2000.

[150] M. Provansal, C. Mathis, and L. Boyer. Benard-von karman instability: Transient and forced regimes. *Journal of Fluid Mechanics*, 182:1–22, 1987.

[151] R. Ramamurti and W. Sandberg. Simulation of flow about flapping airfoils using finite element incompressible flow solver. *AIAA Journal*, 39:253–260, 2001.

[152] D. A. Read, F. S. Hover, and M. S. Triantafyllou. Forces on oscillating foils for propulsion and maneuvering. *Journal of Fluids and Structures*, 17:163–183, 2003.

[153] M. Rehman, C. Vuik, and G. Segal. A comparison of preconditioners for incompressible navier-stokes solvers. *International Journal for Numerical Methods in Fluids*, 57:1731–1751, 2007.

[154] P. Roache. *Verification and Validation in Computational Science and Engineering*. Hermosa Publishers, 1998.

[155] J. Rohr and F. Fish. Strouhal number and optimization of swimming by odontocete cetaceans. *The Journal of Experimental Biology*, 207:1633–1642, 2004.

[156] K. V. Rozhdestvensky and V. A. Ryzhov. Aerohydrodynamics of flapping wing propulsors. *Progress in Aeronautical Sciences*, 39:583–633, 2003.

[157] D. Rusell and Z. Wang. A cartesian grid method for modeling multiple moving objects in 2d incompressible viscous flow. *Journal of Computational Physics*, 191:177–205, 2003.

[158] Y. Saad. *Iterative Methods for Sparse Linear Systems*. SIAM, 2003.

[159] R. L. Sani, J. Shen, O. Pironneau, and P. M. Gresho. Pressure boundary condition for the time-dependent incompressible navier-stokes equations. *International Journal for Numerical Methods in Fluids*, 50:673–682, 2006.

[160] W. E. Schiesser. *The Numerical Method of Lines: Integration of Partial Differential Equations*. Academic Press, 1991.

[161] W. Schmidt. Der wellpropeller, ein neuer antrieb fuer wasser, land, und luft fahrzeuge. *Zflugwiss*, 13:472–479, 1965.

[162] R. Schneiders. *Mesh Generation and Grid Generation on the Web*. http://www-users.informatik.rwth-aachen.de/ roberts/meshgeneration.html.

[163] L. Schouveiler, F. S. Hover, and M. S. Triantafyllou. Performance of flapping foil propulsion. *Journal of Fluids and Structures*, 20:949–959, 2005.

[164] J. Shewchuk. Delaunay refinement algorithms for triangular mesh generation. *Computational Geometry: Theory and Applications*, 22:21–74, 2002.

[165] W. Shyy, M. Berg, and D. Ljungqvist. Flapping and flexible wings for biological and micro air vehicles. *Progress in Aerospace Sciences*, 35:455–505, 1999.

[166] W. Shyy, Y. Lian, J. Tang, H. Liu, P. Trizila, B. Stanford, L. Bernal, C. Cesnik, P. Friedmann, and P. Ifju. Computational aerodynamics of low reynolds number plunging, pitching and flexible wings for mav applications. *AIAA Paper 2008-253-CP*, 2008.

[167] W. Shyy, Y. Lian, J. Tang, D. Viieru, and H. Liu. *Aerodynamics of Low Reynolds Number Flyers*. Cambridge Aerospace Series, 2007.

[168] A. Silverstein and U. Joyner. Experimental verification if the theory of oscillating airfoils. Technical report, NACA, Technical Report No. 673, 1939.

[169] D. Silvester, H. Elman, D. Kay, and A. Wathen. Efficient preconditioning of the linearized navier-stokes equations for incompressible flow. *Journal of Computational and Applied Mathematics*, 128:261–279, 2001.

[170] J. Slater and J. Dudek. The nparc alliance verification and validation archive. Technical report, NASA, Technical Memorandum NASA/TM-2000-209946, April 2000.

[171] M. Smith. Simulating moth wing aerodynamics: Towards the development of flapping wing technology. *AIAA Journal*, 34:1348–1355, 1996.

[172] M. Smith. Advances in overset methodologies for rotorcraft applications. *8th Symposium on Overset Composite Grids and Solution Technology, Houston, USA*, 2006.

[173] M. Smith, P. Wilkin, and M. Williams. The advantages of an unsteady panel method in modeling the aerodynamic forces on rigid flapping wings. *Journal of Experimental Biology*, 199:1073–1083, 1996.

[174] G. Starius. Composite mesh difference methods for elliptic boundary value problems. *Numer. Math.*, 28:243–258, 1977.

[175] G. Starius. On composite mesh difference methods for hyperbolic differentials equations. *Numer. Math.*, 35:241–255, 1980.

[176] J. Steger, F. Dougherty, and J. Benk. A chimera scheme; advances in grid generation. *ASME FED*, 5:59–69, 1983.

[177] V. G. Stredie. *Mathematical Modeling and Simulation of Aquatic and Aerial Animal Locomotion.* PhD thesis, California Institute of Technology, 2004.

[178] D. Sujudi and R. Haimes. Identification of swirling flow in 3-d vector fields. *AIAA Paper 1995-1715-CP*, 1995.

[179] C. H. Tai, K. M. Liew, and Y. Zhao. Numerical simulation of 3d fluid-structure interaction flow using an immersed object method with overlapping grids. *Journal of Computers and Structures*, 85:749–762, 2007.

[180] S. Tang and N. Aubry. On the symmetry breaking instability leading to vortex shedding. *Physics of Fluids*, 9:2550–2561, 1997.

[181] J. C. Tannenhill, D. A. Anderson, and R. H. Pletcher. *Computational Fluid Mechanics and Heat Transfer.* Taylor & Francis, 1997.

[182] G. K. Taylor, R. L. Nudds, and A. R. Thomas. Flying and swimming animals cruise at a strouhal number tuned for high power efficiency. *Letters to Nature*, 425:707–711, 2003.

[183] N. H. Teng. The development of a computer code for the numerical solution of unsteady, inviscid and incompressible flow over and airfoil. Master's thesis, U.S. Naval Postgraduate School, Department of Aeronautics and Astronautics, 1987.

[184] T. Theodorsen. General theory of aerodynamic instability and the mechanism if flutter. Technical report, NACA, Technical Report No. 496, 1935.

[185] J. F. Thompson. *Numerical Grid Generation. Foundations and Applications.* Joe F. Thompson, 1997.

[186] J. F. Thompson, B. K. Soni, and N. P. Weatherill. *Handbook of Grid Generation*. CRC-Press, 1998.

[187] B. Tobalske. Biomechanics of bird flight. *Journal of Experimental Biology*, 210:3135–3146, 2007.

[188] B. Tobalske, T. Hedrick, and A. Biewener. Wing kinematics of avian flight across speeds. *Journal of Avian Biology*, 33:177–184, 2003.

[189] F. Togashi, Y. Ito, M. Murayama, K. Nakahashi, and T. Kato. Flow simulation of flapping wings of an insect using overset unstructured grid. *AIAA Paper 2001-2619-CP*, 2001.

[190] F. Togashi, Y. Ito, K. Nakahashi, and S. Obayashi. Overset unstructured grids method for viscous flow computations. *AIAA Journal*, 44:1617–1623, 2006.

[191] G. S. Triantafyllou, M. S. Triantafyllou, and M. A. Grosenbaugh. Optimal thrust development in oscillating foils with application to fish propulsion. *Journal of Fluids and Structures*, 7:205–224, 1993.

[192] M. S. Triantafyllou and G. S. Triantafyllou. Hydrodynamics of fishlike swimming. *Annual Review of Fluid Mechanics*, 32:33–53, 2000.

[193] M. S. Triantafyllou, G. S. Triantafyllou, and R. Gopalkrishnan. Wake mechanics for thrust generation in oscillating foils. *Physics of Fluids*, 3:2835–2837, 1991.

[194] D. Tritton. Experiments on the flow past a circular cylinder at low reynolds numbers. *Journal of Fluid Mechanics*, 6:547–567, 1959.

[195] E. O. Tuck. The effect of spanwise variations in amplitude on the thrust generating performance of a flapping thin wing. In *Symposium on Swimming and Flying in Nature*, 1974.

[196] I. H. Tuncer. A 2-d unsteady navier-stokes solution method with moving overset grids. *AIAA Journal*, 35:471–476, 1997.

[197] I. H. Tuncer and M. Kaya. Optimization of flapping airfoils for maximum thrust. *AIAA Paper 2003-0420-CP*, 2003.

[198] I. H. Tuncer and M. F. Platzer. Thrust generation due to airfoil flapping. *AIAA Journal*, 34:324–331, 1995.

[199] V. Venkatakrishnan and D. Mavriplis. Implicit solvers for unstructured meshes. *AIAA Paper 91-537-CP*, Jan. 1991.

[200] J. Videler. *Avian Flight*. Oxford Ornithology, 2006.

[201] M. Vinokur. Conservation equations of gas-dynamics in curvilinear coordinate systems. *Journal of Computational Physics*, 14:105–125, 1974.

[202] H. Viviand. Conservative forms of gas dynamics equations. *Rech. Aecrosp.*, 1974-1:65–68, 1974.

[203] E. A. Volkov. A finite difference method for finite and infinite regions with piecewise smooth boundaries. *Doklady*, 168:744–757, 1966.

[204] E. A. Volkov. The method of composite meshes for finite and infinite regions with piecewise smooth boundaries. *Proc. Steklov Inst. Math.*, 96:145–185, 1968.

[205] T. von Karman and J. Burgers. General aerodynamic theory - perfect fluids. *Aerodynamic Theory*, 2, 1935.

[206] Z. Wang. A fully conservative interface algorithm for overlapped grids. *Journal of Computational Physics*, 122:96–106, 1995.

[207] Z. J. Wang. Vortex shedding and frequency selection in flapping flight. *Journal of Fluid Mechanics*, 410:323–341, 2000.

[208] Z. J. Wang and R. Kannan. An overset adaptive cartesian/prism grid method for moving boundary flow problems. *AIAA Paper 2005-0322-CP*, 2005.

[209] E. Weinan and J. Liu. Essentially compact schemes for unsteady viscous incompressible flows. *Journal of Computational Physics*, 126:122–138, 1996.

[210] P. Wesseling. *Principles of Computational Fluid Dynamics*. Springer, 2001.

[211] C. Williamson. Vortex dynamics in the cylinder wake. *Annual Review of Fluid Mechanics*, 29:477–539, 1996.

[212] D. J. Willis, J. Peraire, and J. K. White. A combined pfft-multipole tree code, unsteady panel method with vortex particle wakes. *International Journal for Numerical Methods in Fluids*, 35:1399–1422, 2007.

[213] A. P. Willmott and C. P. Ellington. The mechanics of flight in the hawkmoth manduca sexta. i. kinematics of hovering and forward flight. *Journal of Experimental Biology*, 200:2705–2722, 1997.

[214] J. Wright and W. Shyy. A pressure-based composite grid method for the navier-stokes equations. *Journal of computational physics*, 107:225–238, 1993.

[215] T. Ye, R. Mittal, H. Udaykumar, and W. Shyy. An accurate cartesian grid method for viscous incompressible flows with complex immersed boundaries. *Journal of Computational Physics*, 156:209–240, 1999.

[216] J. Young and J. Lai. Oscillation frequency and amplitude effects on the wake of a plunging airfoil. *AIAA Journal*, 42:2042–2052, 2004.

[217] Y. Zheng and M. Liou. A novel approach of three-dimensional hybrid grid methodology: Part 1. grid generation. *Journal of Computer Methods in Applied Mechanics and Engineering*, 192:4147–4171, 2003.

[218] Y. Zheng and M. Liou. A novel approach of three-dimensional hybrid grid methodology: Part 2. flow solution. *Journal of Computer Methods in Applied Mechanics and Engineering*, 192:4173–4193, 2003.