ABSTRACT

Title of dissertation: A FLUID-STRUCTURE INTERACTION STRATEGY WITH APPLICATION TO LOW REYNOLDS NUMBER FLAPPING FLIGHT

Marcos Vanella
Doctor of Philosophy, 2010

Dissertation directed by: Professor Elias Balaras
Department of Mechanical Engineering

In this work a structured adaptive mesh refinement (S-AMR) strategy for fluid-structure interaction (FSI) problems in laminar and turbulent incompressible flows is developed. The Eulerian computational grid consists of nested grid blocks at different refinement levels. The grid topology and data-structure is managed by using the Paramesh toolkit. The filtered Navier-Stokes equations are evolved in time by means of an explicit second-order projection scheme, where spatial derivatives are approximated with second order central differences on a staggered grid. The level of accuracy of the required variable interpolation operators is studied, and a novel divergence-preserving prolongation scheme for velocities is evolved. A novel direct-forcing embedded-boundary method is developed to enforce boundary conditions on a complex moving body not aligned with the grid lines. In this method, the imposition of no-slip conditions on immersed bodies is done on the Lagrangian markers that represent their wet surfaces, and the resulting force is transferred to the surrounding Eulerian grid points by a moving least squares formulation.

Extensive testing and validation of the resulting strategy is done on a numerous set of problems. For transitional and turbulent flow regimes the large-eddy simulation (LES) approach is used. The grid discontinuities introduced in AMR methods lead to numerical errors in LES, especially if non-dissipative, centered schemes are used. A simple strategy is developed to vary the filter size for filtered variables around grid discontinuities. A strategy based on explicit filtering of the advective term is chosen to effectively reduce the numerical errors across refinement jumps. For all the FSI problems reported, the complete set of equations governing the dynamics of the flow and the structure are simultaneously advanced in time by using a predictor-corrector strategy. Dynamic fluid grid adaptation is implemented to reduce the number of grid points and computation costs. Applications to flapping flight comprise the study of flexibility effects on the aerodynamic performance of a hovering airfoil, and simulation of the flow around an insect model under prescribed kinematics and free longitudinal flight. In the airfoil simulations, it is found that peak performance is located in structural flexibility-inertia regions where non-linear resonances are present.

A FLUID STRUCTURE INTERACTION STRATEGY WITH APPLICATION TO LOW REYNOLDS NUMBER FLAPPING FLIGHT

by

Marcos Vanella

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2010

Advisory Committee:
Elias Balaras, Associate Professor (Chair)
Balakumar Balachandran, Professor
Peter Bernard, Professor
Kenneth Kiger, Associate Professor
Sung Lee, Professor (Dean's Representative)

Dedication

To my dear parents, Leonardo and Susana.

Acknowledgments

I would like first to express my heartfelt gratitude to my advisor, Prof. Elias Balaras for his relentless guidance during the course of my graduate studies at the University of Maryland. His approach towards research, where the direction is defined, but the means are provided by creativity and dedication, has been most stimulating to me. He has been equally supportive in both, good periods and hard times of my graduate life.

I would also like to thank the members of the committee, Prof. Balakumar Balachandran, Prof. Peter Bernard, Prof. Kenneth Kiger and Prof. Sung Lee for carefully reading this dissertation and providing many constructive suggestions and comments. I want to thank Prof. Balachandran for teaching me several courses in the areas of dynamics and vibration, and for his help and discussions on the studies on flexibility of airfoils and rigid body modeling of flapping flight systems. I would also like to thank Prof. Ugo Piomelli for teaching me courses in computational fluid dynamics and turbulence simulations, and for his guidance and help during our work on large eddy simulation for adaptive grids. I thank Prof. Amr Baz for teaching me courses in control theory and vibration control, and for his generous help in class and research activities.

My most sincere gratitude goes also to Prof. Sergio Preidikman, for his invaluable help at the beginning of my career, for his work on the two-link dynamical model of an airfoil, and his help and company on other academic and life endeavors.

I am most indebted to my dear wife Patricia, for her love, unconditional support and goodwill over all these years. She is the base on which this milestone is

founded.

I kindly thank my lab mates Jianming Yang, Senthil Radhakrishnan and Nikolaos Beratlis for their help and support during the beginning of my graduate studies. With the latter individual I have shared countless research discussions and fun situations along the way, which bring great memories. I thank my colleague Patrick Rabenold, the math-guy, who provided the seminal work on adaptive mesh refinement for incompressible flow using the Paramesh© package. I also thank my fellow lab mates Grigorios Panagakos, Khaled Abdelaziz, Don Daniel, Clarence Baney; and the nonlinear duo: Tim Fitzgerald and Marcelo Valdez, with whom I have developed singular but generalized forms of friendship.

Finally, the support provided by the Air Force Research Laboratory throughout the course of this work is greatly appreciated.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

c	Chord length
C_p	Pressure Coefficient
C_f	Skin Friction coefficient
C_D	Drag coefficient
C_L	Lift coefficient
C_{PW}	Power coefficient
e	Impact restitution coefficient
E	Longitudinal velocity spectra
f_0	Natural shedding frequency
g	Field acceleration
K	Resolved turbulent kinetic energy
L, L_R, L_c	Reference length
L_{11}	One dimensional integral length scale
m_i	Mass of Rigid Body i
q_i, $\dot{q}_i$, $\ddot{q}_i$	Generalized coordinate and time derivatives
$Re = \frac{L\,U}{\nu}$	Reynolds number
$St = 1/9(\rho_b/\rho_f)Re$	Stokes number
t	Time
T	Oscillation Period
U, U_R, U_c	Reference velocity
U_∞	Free stream velocity
Γ	Fluid circulation
Γ_Ω	Flow Domain Boundaries
λ	Taylor microscale
μ	Fluid dynamic viscosity
$\nu = \frac{\mu}{\rho}$	Fluid kinematic viscosity
ν_T	Turbulent viscosity
ω	Fluid vorticity field
ω_n	Structural system natural frequency of oscillation
ω_f	Structural system forcing frequency
ω_f/ω_n	Frequency ratio
Ω	Flow domain
ρ, ρ_f	Fluid density
ρ_b	Rigid body density
ρ_b/ρ_f	Mass ratio
τ_{ij}	Subgrid scale stresses (SGS)
S-AMR	Structured Adaptive Mesh Refinement
DNS	Direct Numerical Simulation
LES	Large Eddy Simulation
FSI	Fluid structure interaction

Chapter 1

Introduction

1.1 Motivation and objectives

The recent fusion of decades of advancements in mathematical models, numerical algorithms and computer architecture marked the beginning of a new era in the science of simulation. A simulation that would have taken centuries on a 1947 Mark I computer using Gaussian elimination now takes only a few seconds with parallel multigrid on the IBM Blue Gene/P supercomputer. Computational fluid dynamics (CFD), which is at the forefront of computational mechanics, in utilizing large-scale computational resources to tackle problems of increasing size and complexity, has also greatly benefited from the above developments. Nowadays, even commercially available, general purpose, CFD solvers offer some form of parallel computing capability. The gains, however, in our ability to resolve the energy transfer in convection dominated flows are not as impressive. During the past 30 years direct numerical simulations (DNS), which are three-dimensional, time-dependent computations where all scales of motion down to the Kolmogorov scale are resolved, have only seen moderate increases in the simulated Reynolds numbers. In addition, all these state-of-the-art computations are usually performed on idealized settings (i.e., homogeneous turbulence [115], fully developed channel flows [26]). Applications to more realistic configurations are limited (see for example [30]).

Many problems of interest in physics and engineering include complex geometries with moving and/or deforming boundaries. The latter is amongst the most challenging problems in computational mechanics: the main difficulty arises from the fact that the spatial domain occupied by the fluid changes with time and the location of the boundary is usually an unknown itself that depends on the fluid flow and the motion and/or deformation of the body. There is only a limited number of special cases where established CFD codes can be directly applied to fluid-structure interaction (FSI) problems with a relatively small overhead. The use of moving reference frames [55], or coordinate transformations [72] are characteristic examples. In more complex configurations there are two classes of methods that can be used: i) formulations utilizing moving and/or deforming grids that continuously adapt to the changing location of the body (see, for example [98, 46]); and ii) non-boundary conforming methods, where the requirement for the grid to conform to the body is relaxed and boundary conditions are imposed by using external forcing functions, or local reconstructions (see [68] for a recent review of the different strategies). The latter family of methods has significant advantages in configurations involving multiple bodies undergoing large motions and/or deformations, compared with the former, where the need for constant deformation/regeneration of the grid has an adverse impact on the accuracy and efficiency of the fluid solvers. Applications of either of the above strategies in massively parallel environments are yet to be reported.

The last few years have seen a paradigm shift in high performance computing hardware from machines with few powerful processors that achieved performance by increasing the clock rate, to systems with many relatively simpler processors running

at moderate clock speeds. In the new paradigm, the computing power of a processor node is increased by including multiple processing units with shared memory on the same chip, leading to fine-grained parallelism. This presents a unique opportunity to advance high-fidelity, eddy-resolving CFD solvers, to the next level and develop scalable algorithms for challenging multiphysics, multiscale problems such as fluid-structure interactions.

The main objective of the proposed work is the development of scalable tools and algorithms applicable to fluid-structure interactions in viscous incompressible flows. To achieve this objective, the focus of the dissertation will be on the following specific aims:

1. *Development of a robust immersed-boundary formulation with adaptive mesh refinement (AMR):* immersed boundary methods are well suited for multibody FSI problems. Existing methods however, lack robustness, and are confined to low Reynolds numbers because the global grid refinement, which is usually required to resolve the sharper velocity gradients associated with higher Reynolds numbers, making the computations prohibitively expensive. To overcome these constraints, the following is to be carried out: i) development of a robust immersed boundary method based on a moving least squares (MLS) formulation. This approach is applicable to multibody problems without special treatments. ii) development of an AMR refinement strategy to locally refine the computational mesh in areas of sharp velocity gradients.

2. *Development of strategies to perform large-eddy simulations (LES) of turbu-*

lent and transitional flows within the AMR solver: In DNS with AMR, the solution is smooth on the grid scale and the interpolation errors due to the reconstruction of the fluxes at the non-matching interfaces between coarse and fine grids are small. In LES, on the other hand, the flow field is generally not smooth at the smallest scale (the filter width is not much larger than the grid size), and the numerical errors in the interpolation between grids can be significant. In addition, when a non-uniform filter-width is used, differentiation and filtering do not commute, and additional terms ("commutator errors") appear in the equations of motion. In the present work, the author will investigate the significance of the different errors on the accuracy of the results and develop strategies to eliminate them.

3. *Utilization of the tool in applications to flapping flight:* The developed tools will be applied to a set of two-dimensional and three-dimensional flapping flight problems. A very important question nowadays is how does wing flexibility affect the aerodynamic characteristics of a wing for a given prescribed kinematics. The effects of wing flexibility are evaluated on the performance of a flapping, flexible airfoil. In three-dimensional studies most of the work on flapping flight today involves tethered insects with prescribed kinematics. To demonstrate the capabilities of the proposed method, the author conducts computations of insects in free flight (the wing motion is prescribed and the overall system 'flies' in response to the aerodynamic forces that are produced).

1.2 Prior work

The combination of fluid and structural solvers on FSI simulations requires a strategy to impose boundary conditions on the fluid, and also define the fluid forces acting on the solids. Usually, the fluid is described on an Eulerian reference frame, while it is simpler to use a Lagrangian description for the immersed bodies. A way of overcomming this complication is to use the Arbitrary Lagrangian-Eulerian (ALE) formulation [29] for the fluid description. Here, the grid is required to conform to the immersed bodies, in the sense that, the boundaries of these are also boundaries of the discrete fluid domain. Then, the vertices (or nodes) of the fluid mesh are displaced by using a conveniently defined velocity field, to account for the motion of internal surface boundaries of the fluid domain. Boundary-conforming methods have the disadvantage that, in large motion or deformation regimes, the grid distortion incurred has a negative impact on the accuracy of calculations [8]. It is possible to combine ALE schemes with remeshing, for cases where the mesh has reached an unacceptable level of distortion, but at the overhead of modifying the grid topology and implementing complicated interpolation operations for grid variables.

On the other hand, immersed boundary methods can be used to compute the flow around immersed bodies by solving the Navier-stokes equations of motion on a fixed structured grid, generally not aligned with the body. Depending on the specifics of the formulation, boundary conditions are imposed by appropriately modifying the stencil in the neighborhood of the body [100], or by using a forcing function on the Navier-Stokes equations which can be derived either by using

physical arguments [76], or directly from the discrete problem [37].

In the case of Peskins [76] formulation, the presence of the body in the flow is taken into account through an elastic deformation force, which is transferred to the fluid momentum equation as a pseudo body force. This transfer requires special treatment and it is done by using kernel functions. A weakness of this method is that limiting inelastic body cases result in numerically stiff problems, requiring very small integration time steps. Several modifications to the above method have been suggested. Goldstein *et. al* [41] applied a forcing term governed by a feedback loop, on the momentum equations discretized by a spectral method. They were able to effectively reduce the timestep constraint inherent to Peskins scheme. Saiki and Biringen [86] extended the method of Goldstein et. al [41] to higher-order finite difference schemes and successfully applied it to simulations of low Re flow around rigid cylinders. Peskins immersed boundary method has been applied to different types of low Reynolds number biological flows [76], [77], [78].

A different approach was taken by Glowinski *et al.* ([39], [40]) , who implemented a fictitious domain method, enforcing the fluid velocity to the given value inside the rigid particle by a distributed Lagrange multiplier. Baaijens [8] developed a distributed Lagrange multiplier (DLM), mortar element fictitious domain method where the no slip condition in the boundary of the obstacle is imposed as an equation for the Lagrange multiplier on this boundary. He applied the method to two-dimensional fluid-structure interactions of flexible membranes subjected to a fluctuating channel flow. The DLM fictitious domain method was extended by Yu [116] for the case of fluid flexible body interactions where the constitutive equa-

tion used is the one of an incompressible neo-Hookean solid. In this formulation the DLM is used to enforce the motion of the fluid as the solid not only on the boundary but also inside the deformable solid domain.

Mohd-Yusof [69] and Fadlun *et al.* [37] introduced a non-boundary conforming formulation based on a direct forcing approach. In this case, the forcing is done in the discrete equations defined on the Eulerian reference frame, and it is defined such that the boundary conditions are enforced on the boundary itself. These methods -also called direct-forcing methods- are particularly attractive, especially when combined with finite-difference or finite-volume formulations, since they can be easily implemented in a manner that does not affect the efficiency and stability of the solver. Boundary motion, however, introduces additional complications, and a straightforward extension of the direct-forcing formulations designed for stationary boundaries (see for example [37], [50], [9]) to fluid-structure interaction problems, leads to hydrodynamic forces that lack smoothness and are a potential source of instabilities [101, 113].

Yang and Balaras [113] suggested that the large fluctuations of the hydrodynamic forces on moving immersed bodies were due to the fact that, at any given timestep, some of the Eulerian grid points in the vicinity of the body will not have the correct velocity, pressure or their derivatives, due to their association with the solid in a previous timestep. The problematic cells as well as the appropriate treatment depends on the details of the implementation. Yang and Balaras [113], for example, proposed a field-extension procedure, where the solution is 'extended' into the body in a way that the cells that emerge into the fluid have the proper velocity

and pressure at later timesteps. Mittal *et al.* [67] in their generalized ghost-cell formulation, assigned the proper values at the problematic cells by interpolating from their surroundings. Yang et. al [114] combined the above method with a predictor-corrector time integrator for structural equations, and applied it to two-dimensional rigid body motion problems. A similar scheme was applied by De Tullio and collaborators [99] to DNS and FSI simulation of bileaflet prostatic heart valves under pulsatile flows and physiological conditions.

Uhlmann [101] suggested an alternative direct-forcing scheme, where the force is computed on the Lagrangian markers rather than Eulerian points as it was done in all previous implementations, which resulted in much smoother hydrodynamic forces. He applied the Lagrangian forcing scheme to low Re FSI particle sedimentation problems. A major drawback of immersed boundary schemes coupled to a single logically cartesian grid, specially in moving bondaries, is that the local resolution required in areas where the bodies are present is unavoidably extended to distant regions of the domain. In order to overcome this issue adaptive mesh refinement of the fluid grid can be employed.

Over the past decades a significant amount of work on adaptive meshing has been done in the framework of unstructured grid solution methods. The lack of inherent structure of such grids usually allows for a straightforward implementation of a variety of AMR strategies. A widely used approach is the so called h-refinement, where local refinement is achieved by splitting existing cells into several smaller ones, or by locally introducing additional nodes (see [65] for a review). For problems that involve moving boundaries, however, local mesh motion (r-refinement) is also

necessary to maintain grid quality [98]. A drawback of the above methods, especially in problems with large boundary motions and deformations, is the difficulty to control grid quality, which has an adverse impact on accuracy and stability of the computations.

Two general categories of AMR for structured grids, which can be ideally coupled to the class of non-boundary conforming methods mentioned above, can be identified: i) isotropic splitting of individual cells that can be managed using hierarchical tree [15], or fully unstructured data-structures [45]; ii) grid embedding, where block-structured grids composed of nested rectangular patches are used. The latter approach maintains most of the advantages of structured grid methods and is an attractive platform for introducing AMR capabilities in eddy resolving techniques such as the LES and DNS. Structured adaptive mesh refinement (S-AMR) was initially introduced by Berger and Oliger [20] for the solution of one- and two-dimensional hyperbolic problems, where the sharp discontinuities in the solution were better captured with increasingly refined rectangular grid patches. Since then, the method has been extended to three-dimensions, and it has been demonstrated to be a robust, cost effective approach for a range of hyperbolic problems (see for example [16], [19]). Applications to incompressible flows, however, have been limited, primarily due to complications associated to the enforcement of the divergence-free constraint.

Most algorithms for incompressible flows are based on the extension of the second-order projection method by Bell and collaborators [17] on the grid topology proposed in [20]. In this particular splitting scheme the viscous and advective terms

in the momentum equation are advanced via a Crank-Nicolson scheme, and an unsplit, second-order upwind Godunov method is used to evaluate the nonlinear term at the time-centered location. In most cases time refinement (i.e. different time-steps are used in different AMR levels) is also employed and synchronization of the solution between AMR levels is required to ensure the divergence free constraint. Recent applications to multiphase flows and to prototypical laminar flows can be found [2] and [64] respectively. Applications of S-AMR strategies such as the above to fluid-structure interaction problems have been limited (see for example [42]). Of particular interest is the work of Roma *et. al* [84], who applied the a fully implicit version of the immersed boundary method [76] on the two-dimensional viscous incompressible flow equations. These, in turn, were solved using an implicit projection method inspired by the method given in [17] on the composite grid structure of [20]. Their results obtained by using self adaptation on the contraction of a 2D elastic spherical balloon showed no significant difference between AMR calculations and uniform grid calculations with same resolution as the finest AMR level.

Most of the AMR applications to date for the solution of turbulent and transitional flows have been in DNS and Reynolds Averaged Navier-Stokes (RANS) solutions, but their understanding and use in LES of turbulent flows is reduced. This is due to the fact that in LES the flow field is generally not smooth at the smallest scale (the filter width is not much larger than the grid size), so that numerical errors in the interpolation between grids can be significant. In addition, the subgrid-scale (SGS) eddy viscosity used to represent the effect of the unresolved scales is usually proportional to the filter width (and, in most cases, to the grid size) squared.

Then, a sudden mesh refinement or coarsening may result in a discontinuity in eddy viscosity, which can generate significant errors. Kravchenko *et al.* [51], for example, computed plane channel flows using a method based on B-splines that allowed them to vary the grid size (and hence the filter width) in planes parallel to the wall. They observed errors on the velocity field when the coarse-to-fine grid ratio was 2, and noted that better agreement with a single-grid calculation could be achieved either by using a different coarse-to-fine grid-size ratio, or by providing a smoother transition between meshes. Of particular interest is the work by Pantano *et al.* [75] on a hybrid AMR finite-difference LES of compressible flows using weighted non-oscillatory schemes. They found that the conservation properties of their fine-coarse upwinding scheme behave well in cases where the flow structures passing through the interface are well resolved (*i.e.*, the flow is smooth on the coarse-grid level).

Recently, two studies have focused on evaluating the errors due to variable or discontinuous filters, examining the combined effect of commutator errors and discontinuous eddy viscosity. Cubero and Piomelli [25] performed LES of plane channel flow in which the grid was uniform in the streamwise (x) and spanwise (z) directions, and smoothly stretched in the wall-normal direction y; however, a non-uniform filter was imposed. They found that, as the filter width is increased, the wall stress and the magnitude of the SGS stress decreases, and the smaller scales contain less energy, but the size of the large scales is essentially unaffected. Inverse trends were observed for the decreasing filter-width ratio. They also noted that, in the near-wall region, the transition from one filter-width to the other often included overshoots or undershoots. This effect may be related to the additional closure terms

on the standard LES equations introduced by the variable filter width; according to van der Bos and Geurts [102], this undesirable phenomenon may be decreased by reducing the gradient of the filter width.

Piomelli *et al.* [80] applied the CDP finite-volume unstructured code to plane channel flow in which grid discontinuities were artificially introduced, the fine-coarse grid interface being placed either parallel or normal to the main advection direction. Their findings suggest that, for interfaces parallel to the flow direction, the resolved stresses decrease as the grid is coarsened, while the subgrid-scale (SGS) ones increase proportionately. If the grid interface is placed close to the buffer layer, however, a thicker sublayer results. When the interface is normal to the main advection, they observed a more complex behavior: a sudden grid coarsening by a factor of two in each direction resulted in aliasing error, loss of phase information between eddies, and decrease of the resolved Reynolds stresses. None of the SGS models tested was capable of balancing this decrease. A coarse-to-fine interface was found to have a more benign character.

It is quite clear that the grid discontinuities that are introduced in AMR methods may lead to numerical errors, in some cases significant ones. This is especially true if non-dissipative, centered schemes are used. In Chapter 4, we will study the development of turbulence in the vicinity and downstream of a refinement interface evaluating the errors introduced by the grid discontinuity, and the distance required for a return to equilibrium. The effect of different LES strategies on interpolation and aliasing errors around refinement interfaces will be evaluated.

From the above literature survey it is clear that efficient, high fidelity numerical

tools for the solution of complex large-scale fluid-structure interaction problems on massively parallel machines are yet to be developed. Such tools are critical for conducting the next generation engineering simulations.

1.3 Outline

The remainder of this document is organized as follows: In Chapter 2, the mathematical model for fluid and structural equations is laid out and the developed immersed boundary scheme is explained. Its accuracy and efficiency is evaluated using a set of prototypical problems of increasing complexity. In Chapter 3, the FSI S-AMR strategy is developed. The level of accuracy of the required variable interpolation operators is studied, and a novel divergence-preserving prolongation scheme for velocities is presented. Extensive tests are performed on a variety of problems. In Chapter 4, the effects of grid discontinuities in the case of LES is investigated. A strategy based on explicit filtering of the advective term, which is effective in reducing numerical errors across the jump, is presented. To demonstrate the applicability of the method in complex turbulent flows the case of the flow around a sphere at $Re = 10000$ is presented. In Chapter 5, applications of the scheme developed to two and three dimensional problems relevant to flapping flight are discussed. Finally, in Chapter 6, conclusions and directions for future work are given.

Chapter 2

Immersed boundary reconstruction

In this Chapter, the author defines fluid and structure models by formulating the filtered Navier-Stokes and structural dynamics equations. Then, the differences between Eulerian and Lagrangian forcing as means of imposing fluid boundary conditions in immersed boundary methods are illustrated. A thorough description of the developed Moving Least Squares reconstruction technique for Lagrangian forcing is provided, along with an assessment of its efficiency and accuracy on selected examples.

2.1 Problem Formulation

We consider dynamical systems consisting of fluid flow interacting with moving and possibly deforming immersed bodies. The flow is always incompressible, and transitional or turbulent flow patters may be present. An example is shown in Fig. 2.1, where a set of solid bodies, Φ_1, Φ_2 and Φ_3, interacts with the flow within the domain Ω bounded by Γ_Ω. In this setting, the dynamics of the fluid and structures are described by different sets of equations, which need to be solved as a coupled system. On the side of the fluid, the LES modeling framework is adopted, and the

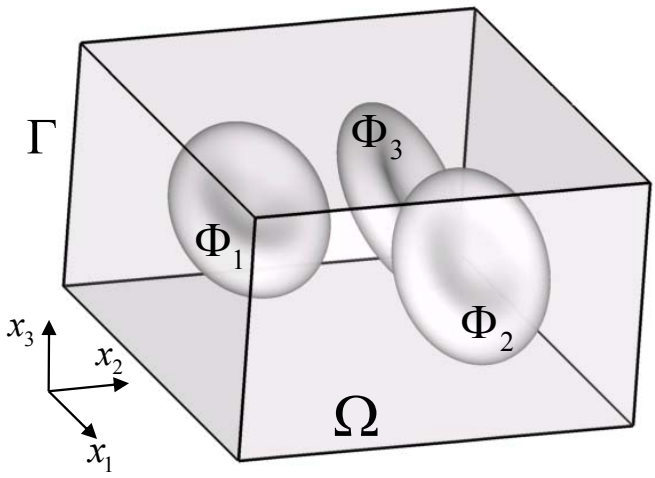


Figure 2.1: An example of a dynamical system, where three solid bodies Φ_1, Φ_2 and Φ_3 interact with the flow in domain Ω, with boundary Γ_Ω.

spatially-filtered Navier-Stokes equations for incompressible flow are solved:

$$\frac{\partial \overline{u}_i}{\partial t} + \frac{\partial \overline{u}_i \overline{u}_j}{\partial x_j} = -\frac{\partial \overline{p}}{\partial x_i} - \frac{\partial \tau_{ij}}{\partial x_j} + \frac{1}{Re}\frac{\partial^2 \overline{u}_i}{\partial x_j \partial x_j} + f_i, \tag{2.1}$$

$$\frac{\partial \overline{u}_i}{\partial x_i} = 0, \tag{2.2}$$

where x_i $(i = 1, 2, 3)$ are the Cartesian coordinates, $\overline{u}_i$ are the resolved velocity components in the corresponding directions, $\overline{p}$ is the resolved pressure, and f_i represents an external body force field. The equations are non-dimensional and $Re = UL/\nu$ is the Reynolds number (U is a reference velocity, L a reference length scale and ν is the kinematic viscosity of the fluid). In LES the large scales are resolved directly as in a DNS, and all scales smaller than the filter size, which is usually proportional to the local grid size, are modeled. In Eq. (2.1) the effect of the unresolved scales appears in the subgrid scale (SGS) stress term, $\tau_{ij} = \overline{u_i u_j} - \overline{u}_i \overline{u}_j$, which needs to be parameterized. We revisit LES in Chapter 4. Unless stated otherwise, for the

remainder of this dissertation we drop the overbars on variables, and assume that they are filtered variables in cases were LES is used.

On the other hand, the motion of a set of rigid bodies within the fluid domain, Ω, is governed by a set of ordinary differential equations (ODEs) of the form [14]:

$$\begin{bmatrix} [\mathbf{I}] & [\mathbf{0}] \\ [\mathbf{0}] & [\mathbf{M}(\mathbf{q}_1)] \end{bmatrix} \begin{Bmatrix} \dot{\mathbf{q}}_1 \\ \dot{\mathbf{q}}_2 \end{Bmatrix} = \begin{Bmatrix} \mathbf{q}_2 \\ \mathbf{F}(\mathbf{q}_1, \mathbf{q}_2, t) \end{Bmatrix}, \tag{2.3}$$

where $\mathbf{q}_1 = [\ q_1 \quad q_2 \quad \dots \quad q_n\ ]^T$ is a vector containing the set of n generalized coordinates of the structural system and $\mathbf{q}_2 = \dot{\mathbf{q}}_1$ is the set of generalized velocities. $[\mathbf{I}]$ is the $n\ x\ n$ identity matrix, $[\mathbf{M}(\mathbf{q}_1)]$ is the nonlinear mass matrix and $\mathbf{F}(\mathbf{q}_1, \mathbf{q}_2, t)$ is a vector containing damping, rotation derived, elastic and externally applied forces. This last set of forces in our case, is composed by the projection of the gravity force and fluid tractions in the direction of the coordinates q_i $(i = 1, \dots, n)$. We note that for deformable elastic bodies, discretization of their governing equations would also lead to first order systems of the form given by Eq. (2.3). Therefore, the solution procedure presented here can be extended to treat deformable bodies.

2.2 Eulerian and Lagrangian forcing on immersed boundary methods

Immersed boundary methods provide the means to impose no-slip boundary conditions on immersed surfaces that do not conform to the Eulerian fluid grid. The velocity field including the boundary effect is reconstructed around these surfaces by applying a force field on the right hand side of the discrete momentum equations. This force field, as noted in Capter 1 can be defined on the Eulerian fluid grid points

([37], [9], [67]), or on the Lagrangian marker points that define the immersed surfaces [101]. For the purpose of illustrating the differences between the two strategies let us assume that u_i $(i = 1, 2, 3)$ is a discrete approximation of the velocity field and write the time-discretized form of momentum equation as:

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = rhs^{n+1/2} + f_i^{n+1/2}, \tag{2.4}$$

where rhs contains all advective, diffusive and, if performing LES, the SGS terms. Also, f_i is the direct-forcing function which is different from zero only at the grid points in the vicinity of the immersed body, and n, $n+1$ refer to the current and next timestep respectively. In the direct forcing scheme proposed in [37] or [9], for example, for every point where $f_i \neq 0$, one can replace u_i^{n+1} in Eq. (2.4) with the desired velocity u_i^d (usually determined by means of interpolation from the surrounding nodes), and find:

$$f_i^{n+1/2} = \frac{u_i^d - u_i^n}{\Delta t} - rhs^{n+1/2}. \tag{2.5}$$

Substituting f_i back into Eq. (2.4) the proper boundary condition, $u_i^{n+1} = u_i^d$, is recovered. In the formulation proposed by Uhlmann [101] on the other hand, the direct forcing function is computed on each Lagrangian marker, rather than on the Eulerian grid nodes, as follows:

$$F_i^{n+1/2} = \frac{U_i^d - U_i^n}{\Delta t} - RHS^{n+1/2}. \tag{2.6}$$

The upper case symbols in Eq. (2.6) denote the same variables as in Eq. (2.5), but at the Lagrangian points on the immersed body. Setting $\widetilde{U} = U_i^n + \Delta t\ RHS^{n+1/2}$, which is practically the Lagrangian counterpart of the predicted velocity, $\tilde{u}_i$, (see

Eq. 2.8), we can rewrite Eq. (2.6) as:

$$F_i^{n+1/2} = \frac{U_i^d - \widetilde{U}_i}{\Delta t} \tag{2.7}$$

Uhlmann [101] computed the volume force from Eq. (2.7) and used the regularized delta functions introduced in [76] as kernels in the transfer of variables between the Eulerian and Lagrangian grids. The overall implementation was tailored to model suspended rigid spherical particles in a laminar and turbulent flows, where it was demonstrated to be very efficient and robust. Direct extension to more complex fluid-structure interaction problems however, hinges upon the requirement to have uniform elements on the surface of the body, as well as on the fact that only integral hydrodynamic forces could be computed. In the following sections, based on the ideas presented in [101], we propose a direct-forcing scheme that utilizes a versatile Moving Least Square (MLS) approximation to build the transfer functions between the Eulerian and Lagrangian grids, and can be applied to arbitrary moving/deforming bodies. We will also propose a method to compute the local traction forces. The overall formulation utilizes very compact stencils and, without compromising accuracy and robustness, gives results that are identical to 'sharp' direct forcing methods.

2.3 Methodologies

The proposed formulation will be discussed in the framework of a finite-difference, fractional-step, Navier-Stokes solver for incompressible flow. The advective, diffusive and SGS terms are advanced explicitly using an Adams-Bashforth

scheme, and all spatial derivatives are discretized using central, second-order, finite-differences on a staggered grid. Details on the basic solver together with applications in a variety of wall-bounded and free-shear flows can be found in [79, 10, 9]. In the following sections we will focus on the proposed direct-forcing scheme and computation of the hydrodynamic forces.

2.3.1 MLS reconstruction

In the framework of the above mentioned splitting scheme we first take a provisional step to compute the intermediate velocities, $\tilde{u}_i$, which do not satisfy the incompressibility constraint and the boundary conditions on the immersed body:

$$\tilde{u}_i = u_i^n + \frac{\Delta t}{2}\left(3H(u_i^n) - H(u_i^{n-1})\right) - \Delta t \frac{\partial p^n}{\partial x_i}, \tag{2.8}$$

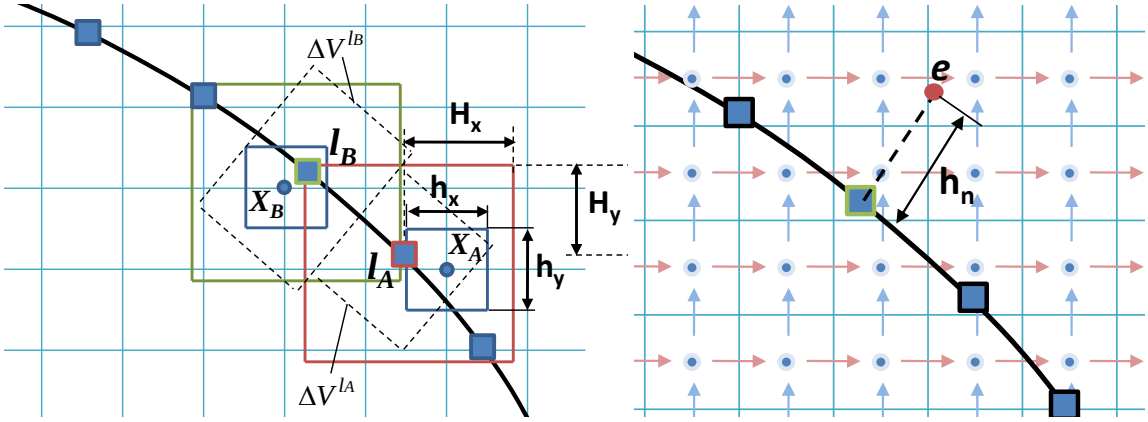
where H is a discrete operator representing the spatially discretized convective, viscous and SGS terms. Next, we will build a direct-forcing function that will enforce the proper boundary conditions on all the Eulerian grid nodes influenced by the immersed body. As in [101] we will compute the forcing function on the Lagrangian markers and then transfer it to the Eulerian grid nodes. Our transfer operators, however, will be constructed using MLS shape functions with compact support [52, 57]. To facilitate this process, for each Lagrangian marker we: i) Identify the closest Eulerian grid node. Referring to Fig. 2.2a, for example, the marker l_a is associated to the grid node x_a, which is in the center of a cell with dimensions h_x and h_y in the x and y directions respectively. Marker l_b is associated to the grid node x_b and so on. Note that more than one Lagrangian markers from the same,

or different immersed bodies, can be associated with the same Eulerian grid node. ii) Define a support-domain around each Lagrangian marker, in which the shape functions will be constructed. In our case the support domain is a rectangular box of size $2H_x \times 2H_y \times 2H_z$ centered at the location of the marker. H_x, H_y and H_z are different for each marker and are proportional to the local Eulerian grid. We found $H_x = 1.2h_x$, $H_y = 1.2h_y$ and $H_z = 1.2h_z$ (see Fig. 2.2a) to be sufficient for all cases considered in this study; iii) Associate a volume, $\Delta V^l = A^l h^l$ (A^l is the area of the body surface associated to marker l, and h^l is a local thickness that depends on local grid size and will be defined in the following paragraphs) to each marker point. In Fig. 2.2a the volumes ΔV^{l_a} and ΔV^{l_b} for markers l_a and l_b respectively are shown. There is no overlapping between successive volumes, ΔV^l, and the sum of all local A^l is equal to the total area of the immersed object surface.

We can now define the transfer operator that will enable the computation of $\widetilde{U}_i$ from the corresponding velocities, $\tilde{u}_i$, given by Eq. (2.8). Using the MLS method, $\widetilde{U}_i$ for each Lagrangian marker, l, can be approximated in its support domain as follows:

$$\widetilde{U}_i(\mathbf{x}) = \sum_{j=1}^{m} p_j(\mathbf{x}) a_j(\mathbf{x}) = \mathbf{p}^T(\mathbf{x})\mathbf{a}(\mathbf{x}), \tag{2.9}$$

where $\mathbf{p}^T(\mathbf{x})$ is the basis functions vector of length m, $\mathbf{a}(\mathbf{x})$ is a vector of coefficients, and $\mathbf{x}$ is the position of the Lagrangian marker. We found that a linear basis, $\mathbf{p}^T(\mathbf{x}) = [\ 1 \quad x \quad y \quad z\ ]$, is a cost-efficient choice and would represent the field variation for all variables up to the accuracy of our spatial discretization scheme.



(a) (b)

Figure 2.2: (a) Definition of the support-domain for two neighboring Lagrangian markers, l_A and l_B, which are color coded for clarity. X_A and X_B denote the closest Eulerian nodes to l_A and l_B respectively. The corresponding volumes ΔV are also shown (dashed line). (b) The normal probe defined by the Lagrangian marker l and point, e is shown together with the support domain used in the MLS approximation.

To obtain the coefficient vector, $\mathbf{a}(\mathbf{x})$, the following weighted $L2$-norm is defined:

$$J = \sum_{k=1}^{ne} W\left(\mathbf{x} - \mathbf{x}^k\right)\left[\mathbf{p}^T(\mathbf{x}^k)\mathbf{a}(\mathbf{x}) - \tilde{u}_i^k\right]^2, \tag{2.10}$$

where $\mathbf{x}^k$ is the position vector of the Eulerian point k in the interpolation stencil, $\tilde{u}_i^k$ is the variable defined in Eq. (2.8) for grid point k, and $W\left(\mathbf{x} - \mathbf{x}^k\right)$ is a given weight function that will be defined below. ne is the total number of grid points in the interpolation stencil, which for the linear basis function above, involves five and seven points in two- and three-dimensions respectively. For simplicity we set the closest point to the Lagrangian marker to be the center point in the stencil.

Minimizing J with respect to $\mathbf{a}(\mathbf{x})$ leads to the following set of equations:

$$\begin{aligned}
\mathbf{A}(\mathbf{x})\ \mathbf{a}(\mathbf{x}) &= \mathbf{B}(\mathbf{x})\ \tilde{\mathbf{u}}_i^k \quad \text{with,} \\
\mathbf{A}(\mathbf{x}) &= \sum_{k=1}^{ne} W(\mathbf{x}-\mathbf{x}^k)\mathbf{p}(\mathbf{x}^k)\mathbf{p}^T(\mathbf{x}^k), \\
\mathbf{B}(\mathbf{x}) &= [W(\mathbf{x}-\mathbf{x}^1)\mathbf{p}(\mathbf{x}^1) \ \cdots \ W(\mathbf{x}-\mathbf{x}^{ne})\mathbf{p}(\mathbf{x}^{ne})], \quad \text{and} \\
\tilde{\mathbf{u}}_i^k &= [\tilde{u}_i^1 \ \cdots \ \tilde{u}_i^{ne}]^T. \qquad (2.11)
\end{aligned}$$

The size of matrix $\mathbf{A}(\mathbf{x})$ depends on the size of the basis vector, $\mathbf{p}(\mathbf{x})$, and it is 3×3 in two-dimensions and 4×4 in three-dimensions, while $\mathbf{B}(\mathbf{x})$ is of size $3 \times ne$ in two-dimensions or $4 \times ne$ in three-dimensions. Combining Eqs. (2.9) and (2.11) one can write $\widetilde{U}_i$ as follows:

$$\widetilde{U}_i(\mathbf{x}) = \sum_{k=1}^{ne} \phi_k^l(\mathbf{x})\tilde{u}_i^k = \mathbf{\Phi}^T(\mathbf{x})\tilde{\mathbf{u}}_i^k \qquad (2.12)$$

where $\mathbf{\Phi}(\mathbf{x}) = \mathbf{p}(\mathbf{x})\ \mathbf{A}(\mathbf{x})^{-1}\ \mathbf{B}(\mathbf{x})$ is a column vector with length ne, containing the shape function values for marker point l. Cubic splines are used for the weight functions, $W(\mathbf{x}-\mathbf{x}^k)$, above, which can be written as:

$$W(\mathbf{x}-\mathbf{x}^k) = \begin{cases} 2/3 - 4\overline{r_k}^2 + 4\overline{r_k}^3 & \text{for} \quad \overline{r_k} \le 0.5 \\ 4/3 - 4\overline{r_k} + 4\overline{r_k}^2 - 4/3\overline{r_k}^3 & \text{for} \quad 0.5 \le \overline{r_k} \le 1.0 \\ 0 & \text{for} \quad \overline{r_k} > 1.0 \end{cases} \qquad (2.13)$$

where $\overline{r_k} = |\mathbf{x}-\mathbf{x}^k|/H_i$. These functions are monotonically decreasing and are sufficiently smooth in the support domain. The resulting shape functions reproduce exactly the linear polynomial contained in their basis and possess the partition of unity property $\sum_{i=1}^{ne}\phi_i(\mathbf{x}) = 1$ [57]. Also, the field approximation is continuous on the global domain as the *MLS* shape functions are compatible.

Eq. (2.12) will give $\widetilde{U}_i$, which can then be substituted into Eq. (2.7) to obtain the volume force F_i on all Lagrangian markers. To transfer F_i to the Eulerian points associated with each marker, l, the same shape functions used in interpolation procedure can be used if properly scaled by a factor c_l, which will be determined later. In such case, the final forces on the Eulerian grid nodes would be:

$$f_i^k = \sum_{l=1}^{nl} c_l \phi_k^l F_i^l, \tag{2.14}$$

where f_i^k is the volume force in the Eulerian point k in the direction i, ϕ_k^l is the shape function previously obtained relating variables between grid point k and marker l, and F_i^l is the force in marker l. Also, nl is the number of Lagrangian markers which are related to the grid point k. To properly rescale the shape functions we require that the total force acting on the fluid is not changed by the transfer:

$$\sum_{k=1}^{nte} f_i^k \Delta V^k = \sum_{l=1}^{ntl} F_i^l \Delta V^l \tag{2.15}$$

where, $\Delta V^k = (h_x \times h_y \times h_z)$ is the volume associated with the Eulerian grid point k, and $\Delta V^l = A^l h^l$ is the volume associated with the marker l, with $h^l = 1/3 \sum_{k=1}^{ne} \phi_k^l (h_x + h_y + h_z)$. nte and ntl is the total number of forced grid points, and total number of Lagrangian markers respectively. As our surface is discretized using triangular elements, the area for marker l is obtained by a simple angle averaging process. Using (2.14) in (2.15) and rearranging the sums in the left hand side in terms of the total number of markers we get

$$\sum_{l=1}^{ntl} \sum_{k=1}^{ne} \phi_k^l \Delta V^k c_l F_i^l = \sum_{l=1}^{ntl} \Delta V^{El} c_l F_i^l = \sum_{l=1}^{ntl} F_i^l \Delta V^l \tag{2.16}$$

where ΔV^{El} is the averaged Eulerian grid volume associated to the Lagrangian marker l. For Eq. (2.16) to hold the scaling factor c_l needs to be set to:

$$c_l = \frac{\Delta V^l}{\Delta V^{El}} = \frac{A^l h^l}{\Delta V^{El}}, \tag{2.17}$$

One can also show that the above scheme guarantees the equivalence of total torque between the Eulerian and Lagrangian meshes:

$$\sum_{k=1}^{nte} \mathbf{x}_k \times \mathbf{f}_k \Delta V^k = \sum_{l=1}^{ntl} \mathbf{X}_l \times \mathbf{F}_k \Delta V^l \tag{2.18}$$

For simplicity we will provide a proof in two-dimensions, but the extension to three-dimensions is straightforward. In particular, the two-dimensional form of Eq. (2.18) can be written as:

$$\sum_{k=1}^{nte} \left(x^k f_y^k - y^k f_x^k \right) \Delta V^k = \sum_{l=1}^{ntl} \left(X^l F_y^l - Y^l F_x^l \right) \Delta V^l, \tag{2.19}$$

or as:

$$\sum_{k=1}^{nte} x^k f_y^k \Delta V^k = \sum_{l=1}^{ntl} X^l F_y^l \Delta V^l, \tag{2.20}$$

$$\sum_{k=1}^{nte} y^k f_x^k \Delta V^k = \sum_{l=1}^{ntl} Y^l F_x^l \Delta V^l, \tag{2.21}$$

Equation (2.19) will hold if both (2.20) and (2.21) hold. In the following we will consider proof of Eq. (2.21), and similar arguments can be used for Eq. (2.20). For each Lagrangian marker, Y^l can be expressed in terms of the shape functions as follows:

$$Y^l = \sum_{k=1}^{ne} \phi_k^l y^k. \tag{2.22}$$

Substituting (2.22) and (2.14) into (2.21) and reordering the sums in the LHS:

$$\sum_{l=1}^{ntl} F_x^l c_l \sum_{k=1}^{ne} \phi_k^l y^k \Delta V^k = \sum_{l=1}^{ntl} F_x^l \sum_{k=1}^{ne} \phi_k^l y^k \Delta V^l. \tag{2.23}$$

Inspection of Eq. (2.23) confirms that the equivalence of total torque will be satisfied, if for each Lagrangian marker, l, the following is true:

$$c_l \sum_{k=1}^{ne} \phi_k^l y^k \Delta V^k = \sum_{k=1}^{ne} \phi_k^l y^k \Delta V^l \tag{2.24}$$

Given that $c_l = \Delta V^l / \sum_{k=1}^{ne} \phi_k^l \Delta v^k$, and assuming ΔV^k is constant on the Eulerian stencil, (2.24) is trivially satisfied. In summary, the proposed transfer operators, conserve momentum on both uniform and stretched grids. For torque to be conserved the cell volume across the stencil should be kept constant for each marker. This is satisfied in case of uniform grids. In other situations, the departure from equivalence for torque will depend on the amount of stretching of the grid. Numerical experiments on two dimensional meshes showed that this difference is small (less than 0.5%) for 10% grid stretching in each direction.

Using the forcing function from Eq. (2.14), we can now correct the intermediate velocity $\tilde{u}_i$ to respect the boundary conditions on the immersed body: $u_i^* = \tilde{u}_i + \Delta t f_i$. The resulting approximate velocity field, u_i^*, which is not divergence free, can be projected into a divergence-free space by applying a correction of the form:

$$u_i^{n+1} = u_i^* - \Delta t \frac{\partial}{\partial x_i} (\delta p) , \tag{2.25}$$

where $\delta p = p^{n+1} - p^n$ is the pressure correction, which satisfies the following Poisson equation:

$$\frac{\partial^2 (\delta p)}{\partial x_i \partial x_i} = \frac{1}{\Delta t} \frac{\partial u_i^*}{\partial x_i}. \tag{2.26}$$

The velocity field, u_i^{n+1}, given by Eq. (2.25) is divergence-free and satisfies the boundary conditions to the order of $O(\Delta t^2)$ [50].

2.3.2 Calculus of surface forces

In non-boundary conforming formulations the fact that the computational grid and the surface of the body are almost never aligned, introduces complications to the computation of hydrodynamic forces generated by the surrounding fluid. In the present formulation for the case of rigid bodies the distributed forcing function given by Eq. (2.14) can be utilized to compute the total hydrodynamic force on a solid object, provided that all interior points are properly treated (see for example [101]). Extension of this approach, however, to the general case of moving and/or deforming bodies is not trivial. In the present study we have developed a methodology where the local hydrodynamic force per unit area on a surface element,

$$f_i^H = \tau_{ji} n_j = \left[-p\delta_{ij} + \mu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \right] n_j, \tag{2.27}$$

is computed directly from the flow field around the body. In Eq. (2.27), f_i^H is the hydrodynamic surface force in x_i direction, τ_{ji} is stress tensor, and n_j is the direction cosine of the normal unit vector in x_j direction. The use of Eq. (2.27) requires knowledge of p and $\partial u_i/\partial x_j$ on the body surface. In the formulation outlined above the boundary is defined in a sharp manner, but the pressure and velocity fields are forced to vary smoothly through the surface of the body. Consequently, the use of the same transfer functions to estimate p and $\partial u_i/\partial x_j$ at the Lagrangian markers would probably underestimate the actual traction forces. This was also verified by a series of numerical experiments we conducted for the case of the flow around an oscillating cylinder below.

To avoid such problems, for each Lagrangian marker, l, on the body we create

a normal probe by locating an external point, e, at a distance, h_n, from the surface (see Fig. 2.2b). The distance h_n is proportional to the local grid spacing and is set to: $h_n = (h_x + h_y + h_z)/3$. To compute the surface pressure at marker, l, we first compute the pressure, p^e, at point e, using the MLS formulation described in the previous section. The support domain in this case is centered around point e as shown in Fig. 2.2b. Next, the value of $\partial p/\partial n$ is obtained from the momentum equation normal to the boundary [113]:

$$\frac{\partial p}{\partial n} = -\frac{D\mathbf{u}}{Dt} \cdot \mathbf{n}, \tag{2.28}$$

where $\mathbf{n}$ is the normal unit vector passing through the marker l, and $\frac{D\mathbf{u}}{Dt}$ is the acceleration of the marker. The value of the pressure at the surface is then obtained from:

$$p^l = p^e - \frac{\partial p}{\partial n} h_n \tag{2.29}$$

The velocity derivatives, $\partial U_i/\partial x_j$, at the location e for each Lagrangian marker, l, are computed by differentiating equation (2.12):

$$\frac{\partial U_i}{\partial x_j} = \sum_{k=1}^{ne} \frac{\partial \phi_k}{\partial x_j} u_i, \tag{2.30}$$

where $\partial \phi_k/\partial x_j$ comes from the solution of an additional system of equations similar to (2.11) [57]. Given the fact that h_n is of the order of the local grid size, and assuming a linear variation of the velocity near the body, the derivatives, $\partial U_i/\partial x_j$, coming from equation (2.30) are good approximation for the derivatives $\partial u_i/\partial x_j$, at the surface. Higher-order reconstruction procedures could also be adopted, albeit at a higher cost. It is shown next that the above procedure reproduces, the forces

on the surface of an immersed body very accurately when compared to boundary-conforming methods at the same grid resolution.

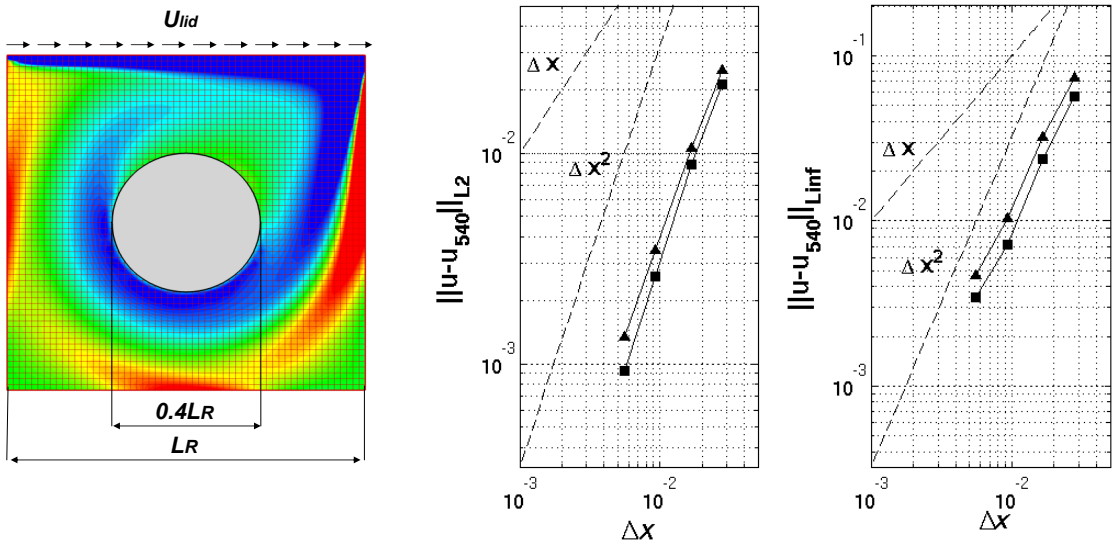
2.4 Results

In this section, the author presents two test problems to demonstrate the accuracy and robustness of the proposed formulation. First, the formal accuracy is examined for the case of the flow around a cylinder submerged in a driven cavity. Then, the flow around a cylinder oscillating in a cross flow is considered. Here, the focus is on the accuracy of the local force distribution on the surface of the cylinder. Three dimensional examples including this immersed boundary formulation are shown in the next Chapter.

2.4.1 Accuracy study

To evaluate the spatial accuracy of proposed algorithm we performed simulations of the flow around a cylinder immersed in a lid-driven cavity. In Figure 2.3(a) the geometry and a typical vorticity distribution are shown. For all cases considered the cylinder diameter was set to $D = 0.4L_R$, and the Reynolds number, $Re = U_{lid}L_R/\nu = 1000$, where L_R is the cavity size and U_{lid} the velocity of the top boundary. The no-slip conditions on the surface of the cylinder were enforced using the proposed MLS reconstruction.

Although an analytical solution for this problem is not available, the overall accuracy of the scheme can be evaluated by comparing the solution among grids at



(a) (b) (c)

Figure 2.3: The flow around a cylinder immersed in a lid-driven cavity. (a) Computational set-up; (b) L_2 norm of the error and (c) L_{inf} norm of the error as a function of the cell size Δx. (▲) u velocity, (■) v velocity

different resolution. To facilitate this comparison on a staggered grid arrangement we considered meshes with 36^2, 60^2, 108^2, 180^2 and 540^2 nodes. With this choice the finest one (540^2) is the reference solution, and the average and maximum errors on each of the coarser grids is computed without the need to interpolate. In Figure 2.3$(b),(c)$, the L_2 and L_{inf} norms of the error are shown as a function the spatial resolution. Both errors decrease with a second-order slope, indicating that the second-order spatial accuracy of the Cartesian solver maintained.

2.4.2 Oscillating cylinder in a cross-flow

The ability of non-boundary conforming methods to properly capture the surface pressure and viscous stress distribution is of paramount importance, especially in fluid-structure interaction problems. To assess the performance of the proposed formulation the author considered the case of a transversely oscillating cylinder in a cross-flow. The dominant parameters are the Reynolds number $Re = U_\infty D/\nu$ (U_∞ is the inflow velocity), the forcing frequency, f_e, and amplitude, a_0, of the oscillation. When f_e varies around the natural shedding frequency, f_0, interesting phenomena occur due to the complex energy transfer between the fluid and the body [43, 44].

Capturing the detailed flow physics for this problem requires an accurate reproduction of the vorticity dynamics on the surface of the body and is a stringent test for non-boundary-conforming schemes. The parametric space we considered is the one used in the experiments by Gu et al. [43], the boundary-conforming simulations of Guilmineau and Queutey [44], and computations by Yang and Balaras [113], where an embedded-boundary method with a direct forcing scheme is used. The motion of the cylinder is given by $y(t) = a_0 sin(2\pi f_e t)$. We considered three cases with $Re = 185$, $a_0 = 0.2D$ and $f_e/f_0 = 1.0,\ 1.1,\ 1.2$ respectively. For all cases the computational domain was set to $50D \times 30D$ in the streamwise and cross-stream directions respectively, with the cylinder located at $10D$ from the inflow boundary. Free-slip conditions are used at the freestream boundaries and a convective condition at the outflow boundary [74]. We considered two grids with 500×450 and 850×750 nodes, where the resulting cell size around the cylinder was $\Delta x = \Delta y = 0.008D$

and $\Delta x = \Delta y = 0.004D$ respectively. A series of tests for flow over a stationary cylinder was first conducted to examine the sensitivity of the results to the grid resolution. The predicted mean and root-mean-square (rms) values of the drag and lift coefficients on the fine grid were $\overline{C}_D = 1.377$, $C_D^{rms} = 0.0296$ and $C_L^{rms} = 0.461$, and the corresponding values on the coarser grid are within 1.5% of the above, demonstrating the grid independency of the results.

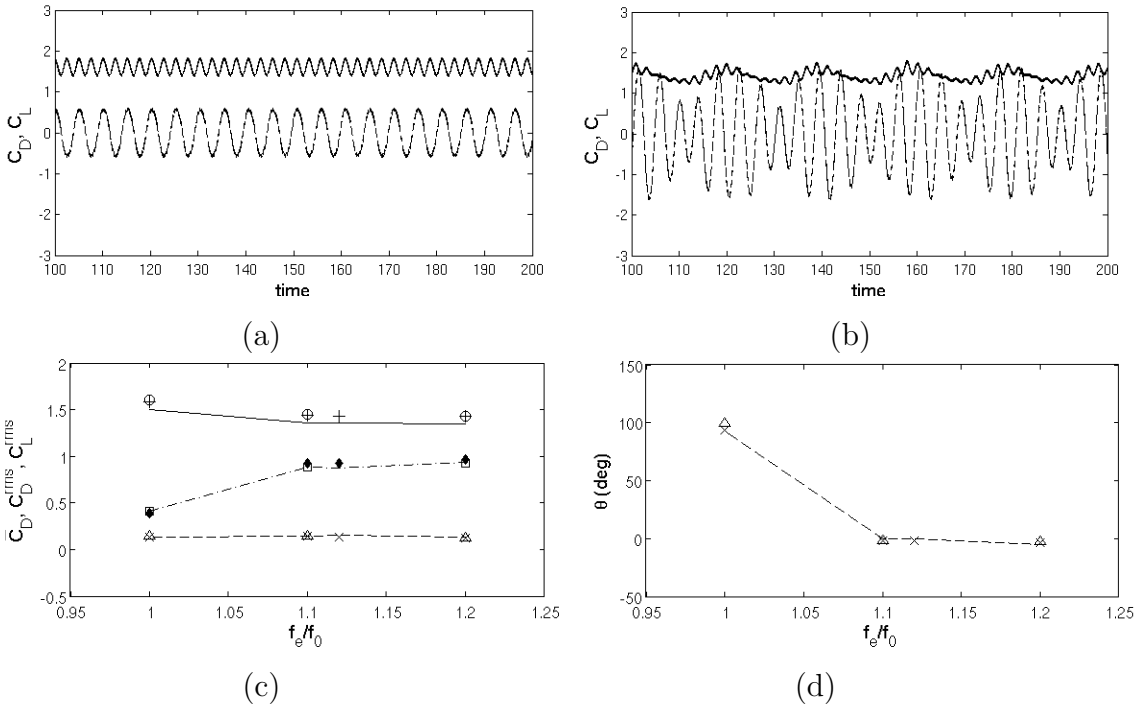


Figure 2.4: Drag and lift coefficients as a function of time for the case of a cylinder oscillating in a cross-flow: (a) $f_e/f_0 = 1.0$, and (b) $f_e/f_0 = 1.2$ (—C_D and - - C_L). (c) Comparison of force coefficients. ∘ $\overline{C}_D$, △ C_D^{rms}, ▫ C_L^{rms} are the present results for the fine grid; —$\overline{C}_D$, - - C_D^{rms}, -·- C_L^{rms} from reference [44], and + $\overline{C}_D$, × C_D^{rms}, ♦ C_L^{rms} from [113]. (d) Phase angle between lift force and vertical displacement. △ are the present results on the fine grid; - - [44] and × [113].

The temporal evolution of the the lift and drag coefficients for the case of the oscillating cylinder are shown in Figure 2.4a, b. It is evident that the proposed scheme results in a smooth variation of the force coefficients without special treatments. In Figure 2.4c a comparison of $\overline{C}_D$, C_D^{rms} and C_L^{rms} for the different excitation frequencies is shown with the corresponding results in the boundary conforming computations in [44] and the computations by using a direct-forcing scheme in [113]. A similar comparison for the phase angle between the lift coefficient and transverse displacement of the cylinder is shown in Figure 2.4d. In general the agreement is excellent. The largest discrepancy appears in $\overline{C}_D$ and is of the order of 3.5%. We should also note that the numerical resolution around the cylinder in our computations is comparable to the one in the reference computations, where $\Delta x \sim 0.005D$.

An important point in the computation of the hydrodynamics forces, especially in fluid-structure interaction problems, is the consistency of the total force and moment exchanged between the fluid and solid systems (action-reaction). Uhlmann [101], for example, who utilizes a similar forcing scheme, proposes a force computation approach that results in equivalence of integral forces. For the case of the oscillating cylinder we compared the force coefficients obtained by direct integration of the local stresses resulting from the normal probe approach, to ones obtained using the approach in [101]. The agreement is very good and the maximum difference is 2.3% for the coarse grid ($\Delta x = 0.008$) and 1.4% for the fine one ($\Delta x = 0.004$) indicating the consistency of our approach.

While mean force predictions is a good indicator of the overall performance

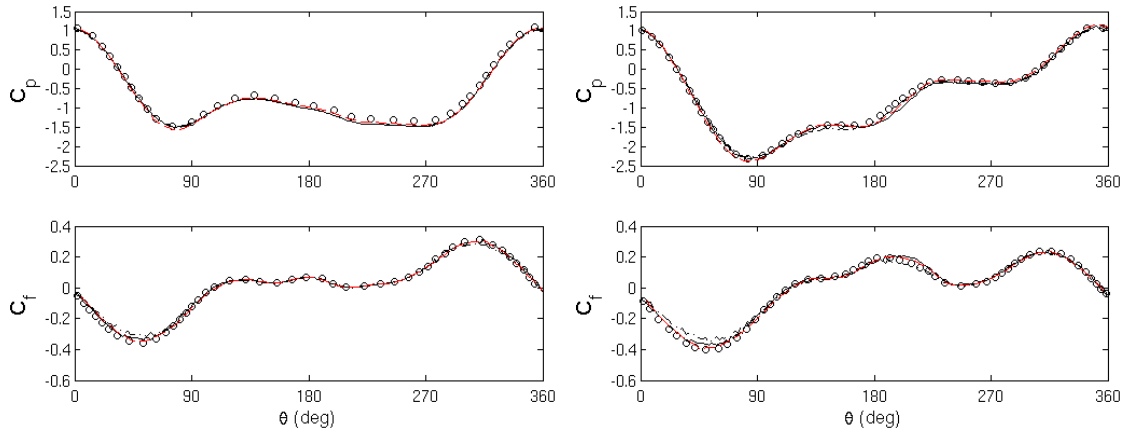


Figure 2.5: Distribution of the pressure and skin friction coefficients C_p and C_f for the case of a cylinder oscillating in a cross-flow. The cylinder is located at the extreme upper position. -.- present results for $\Delta x = 0.008D$, —present results for $\Delta x = 0.004D$, ∘ body-fitted computations in [44], - - non boundary-conforming computations in [113]. (a) $f_e/f_0 = 1.0$, and (b) $f_e/f_0 = 1.2$.

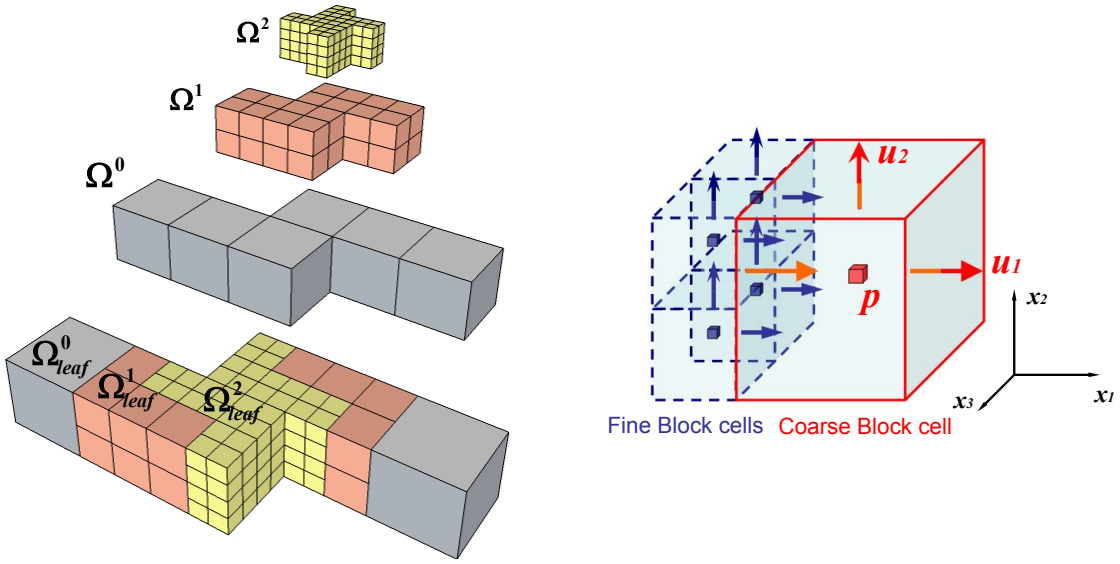
of the method, they do not necessarily translate into an accurate representation of the local forces. In figure 2.5 the distributions of pressure coefficient, C_p, and the skin friction coefficient, C_f, on the cylinder's surface are shown for the time instance corresponding to the extreme upper position. Results for both grids are included from our computations, and are compared with the corresponding results by Guilmineau and Queutey [44] and Yang and Balaras [113]. The higher sensitivity of C_f to the grid resolution results in slightly lower peak values on our coarse grid computations. The results on finer grid agree very well with the reference data. C_p is less sensitive to the grid resolution the results on the different grids are almost indistinguishable.

Chapter 3

Adaptive mesh refinement for fluid-structure interaction problems

In this Chapter, a S-AMR strategy is described for finding solutions of the Navier-Stokes equations in laminar and turbulent incompressible flows. Based on the ideas introduced in [20], the single-block solver presented in the previous chapter is employed on a hierarchy of sub-grids with varying spatial resolution. Each of these sub-grid blocks has a structured Cartesian topology, and it is part of a tree data structure that covers the entire computational domain. One of the main features of the present implementation is the utilization of the Paramesh toolkit [62] to keep track of the grid hierarchy, and perform the required restriction/prolongation and guard-cell filling operations. To compute the flow in complex geometries, the proposed 'direct-forcing' embedded boundary method is used.

In Section 3.1, a description of the AMR grid topology is given and the treatment of the solution at block boundaries is discussed. A strategy to address issues related to mass conservation at interfaces and grid adaptation is also proposed. In Section 3.1.5 an overview of the fluid-structure interaction scheme employed is provided. In Section 3.2, the following examples are discussed to demonstrate the accuracy and robustness of the proposed formulation: i) the Taylor-Green vortex, ii) three-dimensional vortex ring impinging on a wall at Re 570, iii) FSI of two dimensional falling plates, and iv) FSI of a sphere bouncing against a wall at $Re = 830$.



(a) (b)

Figure 3.1: (a) Example grid hierarchy and nested sub-blocks. Three refinement levels, Ω^0, Ω^1, and Ω^2 have been used. (b) Staggered grid arrangement in cells adjacent to a coarse-fine interface: pressure is located at the center and velocities at the face. The velocity component u_3 in the x_3 direction is not shown for clarity.

3.1 Adaptive mesh refinement

3.1.1 Grid topology

The computational grid consists of a number of nested grid blocks with $nx \times ny \times nz$ computational cells at different refinement levels, Ω^l, $l = 0, 1, ..., l_{max}$. The coarsest grid blocks at level Ω^0 always cover the entire computational domain, and local refinement is achieved by the bisection of selected blocks in every coordinate direction. In this process an arbitrary block, b, at level l, for example, will be the origin of eight *children* blocks at level $l + 1$ that occupy the same volume as their

parent block. We call *leaf* blocks the blocks at the highest level of refinement present on a particular region of the domain (blocks that have not been refined, and therefore have no children). Any leaf block at level l shares a common boundary with leaf blocks whose refinement level may differ from l by at most one level. An example is shown in Fig. 3.1a, where an S-shaped domain is discretized with an AMR grid. Level Ω_0 covers the entire domain and consists of 6 blocks (gray). The grid is locally refined near the center of the domain by adding two levels of refinement, Ω^1 (orange) and Ω^2 (yellow). As a result the cell size in the i^{th} direction for a block at the finest level, $l = 2$, is only a quarter of the one at $l = 0$: $\Delta x_i^2 = \Delta x_i^0/4$, where $i = 1, 2, 3$. Note that given the above constraints, Δx_i^l for computational cells on adjacent leaf blocks at different refinement levels will differ by a factor of two. The resulting grid structure is managed using the octree data-structure in the Paramesh toolkit [62], which enables a robust implementation and straightforward parallelization of the proposed algorithm.

A staggered variable arrangement is used in each grid block as shown in Fig. 3.1b, where the velocity component in the x_3 direction has been omitted for clarity. As we did in the previous Chapter we drop the overbar indicating filtered variables, and denote the pressure and velocities for block b at level l as p_{jkm}^{bl} and $u_{i,jkm}^{bl}$, $i = 1, 2, 3$ respectively. jkm are indices that identify a grid cell within block b, and i refers to the orientation of the velocity component. Other variables such as the turbulent viscosity, ν_t, can be similarly defined. We will also denote $P(b)$ as the parent block of block b, and $C(b, i)$ as the i^{th} child block of block b, where $i = 1, ..., 2^d$ and d is the number of dimensions of the problem.

3.1.2 Prolongation and restriction operators

Let us now define the prolongation and restriction operators that are needed to transfer variables between parent and children blocks. The general form of a restriction operation of a variable ϕ from block $C(b,i)$ at level $l+1$ to the parent, $P(b)$, at level l is given by:

$$(R\phi)_{ijk}^{P(b),l} = \sum_{p,q,r=il}^{el} \alpha'_{pqr} \phi_{i'+p,j'+q,k'+r}^{b,l+1}, \tag{3.1}$$

where, ijk are the indexes of a cell on $P(b)$ containing the restricted variable and $i'j'k'$ are the indexes of a cell in b. The limits il and el define the interpolation stencil, where the stencil size and interpolation coefficients, α'_{pqr}, depend on the interpolation scheme used. The prolongation of a variable from block $P(b)$ to b, on the other hand, is given by:

$$(I\phi)_{i'j'k'}^{b,l+1} = \sum_{p,q,r=il}^{el} \alpha_{pqr} \phi_{i+p,j+q,k+r}^{P(b),l}, \tag{3.2}$$

On a staggered grid, prolongation and restriction operators are defined separately for the flow variables collocated at the cell centers (p_{jkm}^{bl}, $\nu_{t\,jkm}^{bl}$, etc.) and the cell faces ($u_{i,jkm}^{bl}$). For the former we use a dimension-by-dimension interpolation strategy that utilizes second-order Lagrange polynomials to perform one-dimensional sweeps over a three-dimensional stencil. In these cases the weight factors, α_{pqr} and α'_{pqr}, in Eqs. (3.1) and (3.2) are products of the corresponding coefficients in the one-dimensional Lagrange polynomials. For the face-centered components we exploit the fact that they are co-planar, and the above strategy is employed in two-dimensions. Example restriction operations in a two-dimensional staggered grid are shown in Fig. 3.2a.

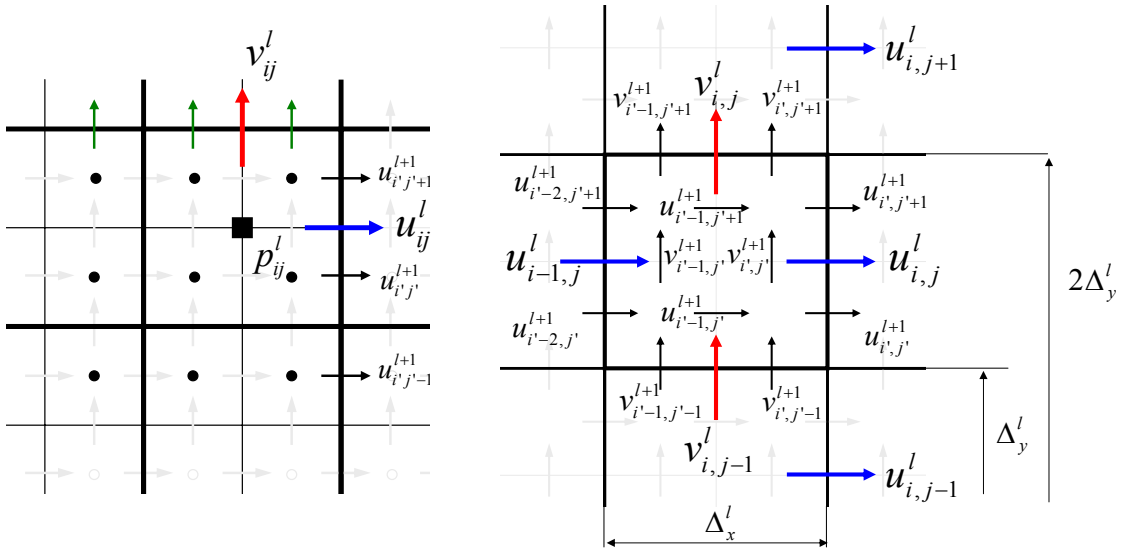


Figure 3.2: (a) Interpolation stencil utilized by two-dimensional prolongation operators. It contains nine points for cell-centered variables such as, p^l_{ij}, and three points for variables located at the cell faces, such as u^l_{ij} and v^l_{ij}.; (b) Variable arrangement for the construction of divergence-preserving prolongation operators in two-dimensions.

The pressure, p^l_{ij}, located at cell-center, ij, of the parent block, is interpolated from pressures on a 3×3 stencil from the corresponding children blocks. The velocities u^l_{ij} and v^l_{ij}, however, are found using the one-dimensional stencils (two-dimensional in three dimensions) shown in the figure. Note that in Fig. 3.2 the superscript indicating the block has been dropped for simplicity and the parent and children blocks are identified by their level superscripts, l and $l+1$ respectively.

The operators given by Eqs. (3.1) and (3.2) satisfy the accuracy requirements

but do not guarantee the divergence-free evolution of the velocity field through the AMR grid. In the present implementation this property is particularly desirable whenever prolongations and restrictions are performed as part of the local grid adaptivity (local refinement and de-refinement) from timestep to timestep. Divergence-free restriction operators can be constructed in a fairly straightforward manner. In our staggered grid arrangement, for example, using face-averaging for all the velocity components preserves the divergence of the velocity vector. Divergence-preserving prolongation of a vector field, on the other hand is not straightforward. Martin et al. [64], in their AMR implementation for the Navier-Stokes equations for incompressible flow, use a divergence cleaning methodology employing an extra Poisson solution. Balsara [12] in the framework of his MHD solver, proposed a divergence-free prolongation reconstruction applicable to AMR meshes with any refinement ratio, utilizing linear and quadratic polynomials. In the present work we have developed divergence-preserving prolongation operators tailored to the specific AMR topology, where the grid size between consecutive refinement levels can only differ by a factor of two. We will describe the proposed scheme in two-dimensions for the configuration shown in Fig. 3.2b. Let us define the discrete divergence, D, at an arbitrary cell ij of the parent block as:

$$D\mathbf{u} = \frac{u^l_{i,j} - u^l_{i-1,j}}{\Delta^l_x} + \frac{v^l_{i,j} - v^l_{i,j-1}}{\Delta^l_y}, \tag{3.3}$$

where Δ^l_x and Δ^l_y is the grid spacing on the parent block in the x and y directions respectively. The above is also the target divergence for the reconstructed velocity field on the corresponding children blocks. We will first determine the eight velocity

components, $u^{l+1}_{i'-2,j'}$, $u^{l+1}_{i'-2,j'+1}$, $u^{l+1}_{i',j'}$, $u^{l+1}_{i',j'+1}$, $v^{l+1}_{i'-1,j'-1}$, $v^{l+1}_{i',j'-1}$, $v^{l+1}_{i'-1,j'+1}$, $v^{l+1}_{i',j'+1}$, which are located at the faces of the parent block as shown in Fig. 3.2b. This is done using one-dimensional, quadratic, mass-flux preserving interpolations on each face. For example, on the right face of the coarse cell one can assume that the velocity can be approximated as, $u(y) = a_0 + a_1 y + a_2 y^2$. Then, utilizing the the known values, $u^l_{i,j-1}$, $u^l_{i,j}$ and $u^l_{i,j+1}$ on that face, together with the discrete mass-flux conservation constraint, $u^l_{i,j} = 0.5(u^{l+1}_{i',j'}+u^{l+1}_{i',j'+1})$, the following algebraic system can be assembled:

$$\begin{aligned} u^l_{i,j-1} &= a_0 + a_1\frac{\Delta^l_y}{2} + a_2\left(\frac{\Delta^l_y}{2}\right)^2 \\ u^l_{i,j} &= \frac{1}{2}(u^{l+1}_{i',j'} + u^{l+1}_{i',j'+1}) = a_0 + a_1\frac{3\Delta^l_y}{2} + a_2\frac{37}{16}\left(\Delta^l_y\right)^2 \\ u^l_{i,j+1} &= a_0 + a_1\frac{5\Delta^l_y}{2} + a_2\left(\frac{5\Delta^l_y}{2}\right)^2 \end{aligned} \tag{3.4}$$

The solution of (3.4) yields the coefficients α_0, α_1, α_2, and the unknown velocities at the midpoints of the face can the be found in a straightforward manner. For example, $u^{l+1}_{i',j'} = a_0 + 5a_1\Delta_y/4 + a_2(5\Delta_y/4)^2$. Note that the system (3.4) can be written in matrix form, and the inverse of the resulting 3×3 Vandermode matrix can be found analytically. The same approach is used on all faces of the parent cell.

Next, the remaining four interior velocities, $u^{l+1}_{i'-1,j'}$, $u^{l+1}_{i'-1,j'+1}$, $v^{l+1}_{i'-1,j'}$ and $v^{l+1}_{i',j'}$, shown in Fig. 3.2b are determined. For each of the children blocks one can write the discrete divergence equations and set it equal to to that from (3.3). For example, for the block $(i'-1, j'+1)$ in Fig. 3.2b:

$$\frac{u^{l+1}_{i'-1,j'} - u^{l+1}_{i'-2,j'}}{\Delta^{l+1}_x} + \frac{v^{l+1}_{i'-1,j'} - v^{l+1}_{i'-1,j'-1}}{\Delta^{l+1}_y} = D\mathbf{u}, \tag{3.5}$$

where Δ^{l+1}_x and Δ^{l+1}_y is the grid spacing on the children block in the x and y

directions respectively. Using similar expressions for all four blocks we can assemble a system of four equations with four unknowns. It is of the form, $\mathbf{A}\mathbf{u_m} = \mathbf{b}$, where $\mathbf{A}$ the 4×4 matrix and $\mathbf{u}_m$, $\mathbf{b}$ the unknowns and source vectors respectively. $\mathbf{A}$, however, is a matrix of rank three and cannot be inverted. To address this issue we will consider an additional independent equation, i.e. express $u^{l+1}_{i'-1,j'}$ (or any of the internal velocities) using quadratic interpolation:

$$u^{l+1}_{i'-1,j'} = -\frac{1}{8}u^{l+1}_{i'-4,j'} + \frac{3}{4}u^{l+1}_{i'-2,j'} + \frac{3}{8}u^{l+1}_{i',j'} \tag{3.6}$$

The resulting system of equations can be written as:

$$\begin{bmatrix} 1 & 0 & \delta & 0 \\ -1 & 0 & 0 & \delta \\ 0 & -1 & 0 & -\delta \\ 0 & 1 & -\delta & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} u^{l+1}_{i'-1,j'} \\ u^{l+1}_{i'-1,j'+1} \\ v^{l+1}_{i'-1,j'} \\ v^{l+1}_{i',j'} \end{Bmatrix} = \begin{Bmatrix} \beta + u^{l+1}_{i'-2,j'} + \delta v^{l+1}_{i'-1,j'-1} \\ \beta + \delta v^{l+1}_{i',j'-1} - u^{l+1}_{i',j'} \\ \beta - u^{l+1}_{i',j'+1} + \delta v^{l+1}_{i',j'+1} \\ \beta + u^{l+1}_{i'-2,j'+1} - \delta v^{l+1}_{i'-1,j'+1} \\ (-u^{l+1}_{i'-4,j'} + 6u^{l+1}_{i'-2,j'} + 3u^{l+1}_{i',j'})/8 \end{Bmatrix} \tag{3.7}$$

where $\delta = \Delta^{l+1}_x/\Delta^{l+1}_y$, $\beta = \Delta^{l+1}_x D\mathbf{u}$. This is now an extended system of five equations and four unknowns which can be written as: $\mathbf{u_m} = (\mathbf{A_e}^T\mathbf{A_e})^{-1}\mathbf{A_e}^T\mathbf{b_e}$. The product $(\mathbf{A_e}^T\mathbf{A_e})^{-1}\mathbf{A_e}^T$ is the 4×5 left-pseudo-inverse of the system, which can be found analytically. The addition of Eq. (3.6) changes the rank of the system and the above *best fit* solution is actually the only solution to the system. There are significant differences between the method above presented and Balsaras scheme [12] when a factor of 2 is used in his reconstruction. Balsaras method starts by defining a piecewise-linear profile for the divergenceless vector field across each face of the rectangular region (the coarse cell in our case). The slopes on this profile are defined

via the minmod slope-limiter. In our case we define a parabolic velocity profile that maintains the mass flux across the face and is defined using neighboring coarse grid velocities. The Balsara method makes use of complete multidimensional quadratic polynomials inside the region, with constraints such that, the continuous divergence at each location in the region is zero. With these, the values of the vector field are interpolated at the internal fine-grid locations. In our case once the coarse faces fine-grid velocities are interpolated, they are used to define one (or more in three dimensions) internal fine grid velocity, by linear or quadratic one dimensional interpolation. The remaining internal fine grid velocities are obtained using the discrete divergence equations for each fine-grid cell. In our scheme the 2^{nd} order accuracy is maintained mainly be the mass-flux conserving quadratic interpolation done on the coarse cell faces. The extension of the above procedure to three-dimensions is straightforward, and the detailed equations are given in Appendix A.

3.1.3 Treatment of the block boundaries

To facilitate the discretization of the equations of motion at block boundaries, overlapping between neighboring blocks is created by means of two layers of ghost cells. A two-dimensional configuration for neighboring blocks at refinement levels l and $l+1$ is shown in Fig. 3.3. To assign values to the ghost cells on the grid-block at level l: i) the solution at level $l+1$ is restricted to its parent grid, which has the same level of refinement as the adjacent coarse block, using Eq. (3.1); ii) *filling* of the ghost cells is done by simple 'injection' of the corresponding variable from the

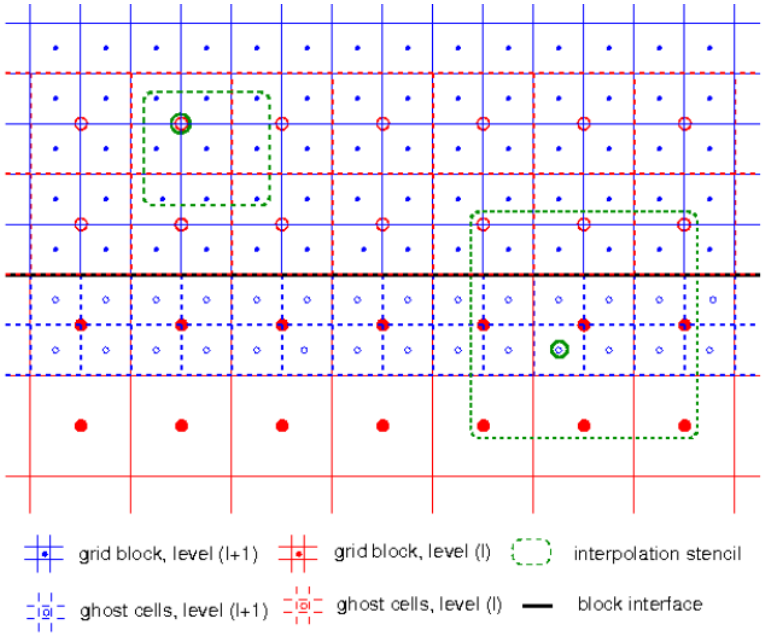


Figure 3.3: Ghost cell configuration at the interface between blocks at levels l and $l+1$. Filled symbols denote the centers of interior cells and open symbols the centers of the ghost cells. The grid nodes surrounded by the green dotted lines are the ones on the interpolation stencil for the ghost cells identified by a green circle.

parent grid. To assign values at ghost cells on the grid-block at level $l+1$ a similar procedure is used, which utilizes the prolongation operator given by Eq. (3.2). In Fig. 3.3 the grid nodes in the interpolation stencil are shown for both cases.

3.1.4 Temporal integration scheme

A standard, second-order, fractional-step method is utilized for the temporal integration of the governing equations [49]. On each leaf block at level $l = 0, ..., l_{max}$

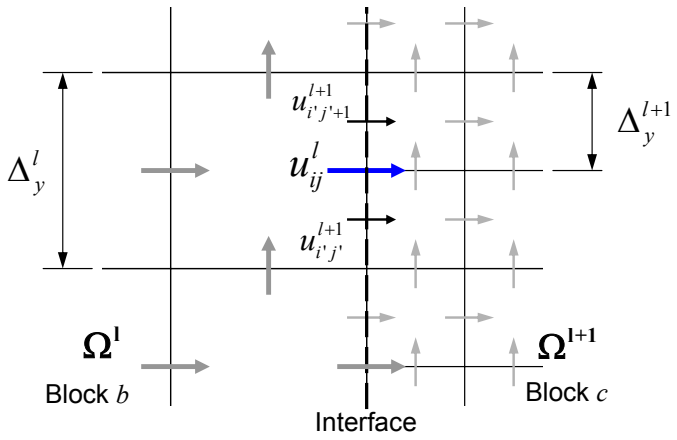


Figure 3.4: Velocity components contributing to the mass flux across a coarse-fine interface.

the predicted velocity, $\widetilde{u}_i^l$, can be written as:

$$\widetilde{u}_i^l = u_i^{l,n} + \frac{\Delta t}{2}\left(3H(u_i^{l,n}) - H(u_i^{l,n-1})\right) - \Delta t \frac{\partial p^{l,n}}{\partial x_i} + \Delta t f_i^{n+1/2}, \tag{3.8}$$

where H is a discrete operator containing the convective, viscous and SGS terms, the superscript, n, refers to the time level, and Δt is the timestep, which is the same on all refinement levels, l. All spatial derivatives are approximated using second-order central differences on a staggered grid. The predicted velocity field, $\widetilde{u}_i^l$, which is not divergence free, can be projected into a divergence-free space by applying a correction of the form:

$$u_i^{l,n+1} = \widetilde{u}_i^l - \Delta t \frac{\partial}{\partial x_i}\left(\delta p^l\right), \tag{3.9}$$

where $\delta p^l = p^{l,n+1} - p^{l,n}$ is the pressure correction, which satisfies the following Poisson equation:

$$\frac{\partial^2\left(\delta p^l\right)}{\partial x_i \partial x_i} = \frac{1}{\Delta t}\frac{\partial \widetilde{u}_i^l}{\partial x_i}. \tag{3.10}$$

The solution of the Poisson equation, (3.10), is done using the multigrid algorithm developed by Martin and Cartwright [63]. This method has been designed for block-structured adaptive grids and maintains second order spatial accuracy, imposing continuity of δp^l and its derivatives across interface jumps. It uses a residual-correction formulation and employs red-black Gauss Seidel (RBGS) point relaxation at each level. In our implementation, in order to increase the performance of the solver, we exploit the uniformity of the mesh at the coarsest level of the V-cycle, l_{solve}, which covers the entire computational domain. In particular, the author utilizes a direct solver, rather than RBGS iterations, to solve the residual-correction equation at that level. The only drawback of this strategy is that the computational domain has to be rectangular since the direct solver utilizes FFT or Cosine transforms, depending on the boundary conditions. This strategy was found to reduce the number of levels on a V-Cycle and relaxation-communication operations, while maintaining the convergence rate of the multigrid scheme.

An important issue with the application of projection schemes to S-AMR grids is related to conservation of mass at coarse-fine block interfaces. In Fig. 3.4 a two-dimensional example of the interface between cells at levels l and $l+1$ is shown. The velocity component, u^l_{ij}, normal to the interface at the coarse level l, as well as the corresponding velocity components, $u^{l+1}_{i'j'}$ and $u^{l+1}_{i'j'+1}$, at fine level $l+1$, are unknowns to be determined during the solution process. They need, however, to satisfy an additional constraint coming form the conservation of the mass flux across the interface:

$$u^l_{ij}\Delta y^l = u^{l+1}_{i'j'}\Delta y^{l+1} + u^{l+1}_{i'j'+1}\Delta y^{l+1}, \tag{3.11}$$

where Δy^l is the grid spacing at level l and Δy^{l+1}=$\Delta y^l/2$, is the grid spacing at level $l+1$. In general, Eq. (3.11) will not hold after the computation of the velocities normal to the refinement jump, resulting to a flux mismatch localized in grid refinement interfaces. To alleviate this problem one can assume that the velocity components on the fine grid at level, $l+1$, are more accurate than the corresponding coarse grid component, u^l_{ij}, which can then be corrected to satisfy Eq. (3.11) (see for example [19], [2]). In the present formulation the flux correction is realized in the following manner: i) the intermediate velocities at the fine level, $\widetilde{u}^{l+1}_{i'j'}$ are computed from Eq. (3.8); ii) the intermediate velocity on the coarse level, $\widetilde{u}^l_{ij}$, is then corrected using Eq. (3.11); and iii) during the iterative solution of the Poisson equation (3.10) the gradient of δp normal to the refinement interface at the u^l_{ij} location is forced to satisfy the following equation:

$$\left.\frac{\Delta(\delta p^l)}{\Delta x}\right|_{ij} \Delta y^l = \left.\frac{\Delta(\delta p^{l+1})}{\Delta x}\right|_{i'j'} \Delta y^{l+1} + \left.\frac{\Delta(\delta p^{l+1})}{\Delta x}\right|_{i'j'+1} \Delta y^{l+1} \tag{3.12}$$

iv) it is now trivial to show that the corrected velocity from Eq. (3.9) will also conserve the mass flux across the interface.

3.1.5 Fluid-structure interaction algorithm

A fundamental complication with two-way, fluid-structure interaction (FSI) problems, is that the prediction of the flow and the corresponding hydrodynamic loads requires knowledge of the motion of the structure and vice-versa. In the present study a strong-coupling scheme is adopted, where the fluid and the structure are treated as elements of a single dynamical system, and all of the governing equations

are integrated simultaneously and iteratively in the time domain. The method is based on Hamming's 4^{th}-order, predictor-corrector formulation, which avoids the evaluation of the hydrodynamic loads at fractional time steps. The details of the overall approach and a demonstration of the accuracy and efficiency for a variety of fluid-structure interaction problems in viscous incompressible flows can be found in [114]. In Fig. 3.5, a flowchart of the overall algorithm as adapted to our AMR implementation is shown:

i) At the beginning of each timestep, and only for the case of LES, the SGS eddy viscosity by means of a LES model. LES will be discussed in next Chapter.

ii) Compute the provisional velocity, $\hat{u}_i^l$, which does not satisfy boundary conditions on the immersed boundary, and is not divergence free. Assign values for $\hat{u}_i^l$ to the ghost cells as described in Section 3.1.3.

iii) Perform AMR if necessary (every n steps). In particular, all leaf-blocks are examined and flagged for refinement or derefinement, according to specified criteria. For example, in all FSI cases reported below, a leaf-block is selected for refinement when it contains an immersed body, or when the vorticity magnitude is larger than a predetermined threshold. A leaf-block is marked for derefinement, on the other hand, when an immersed body is not present within the block, and the vorticity is below a threshold. Other criteria, such as velocity error norms, SGS dissipation etc., could also be used for this purpose. In the case of refinement, the newly created children blocks are added to the octree. All variables other than $\hat{u}_i^l$ and the discrete operator $H(u_i^{l,n-1})$ in (3.8), are interpolated using Eq. (3.2). For $\hat{u}_i^l$ and $H(u_i^{l,n-1})$, the divergence preserving prolongation procedure discussed in Section 3.1.2 is used.

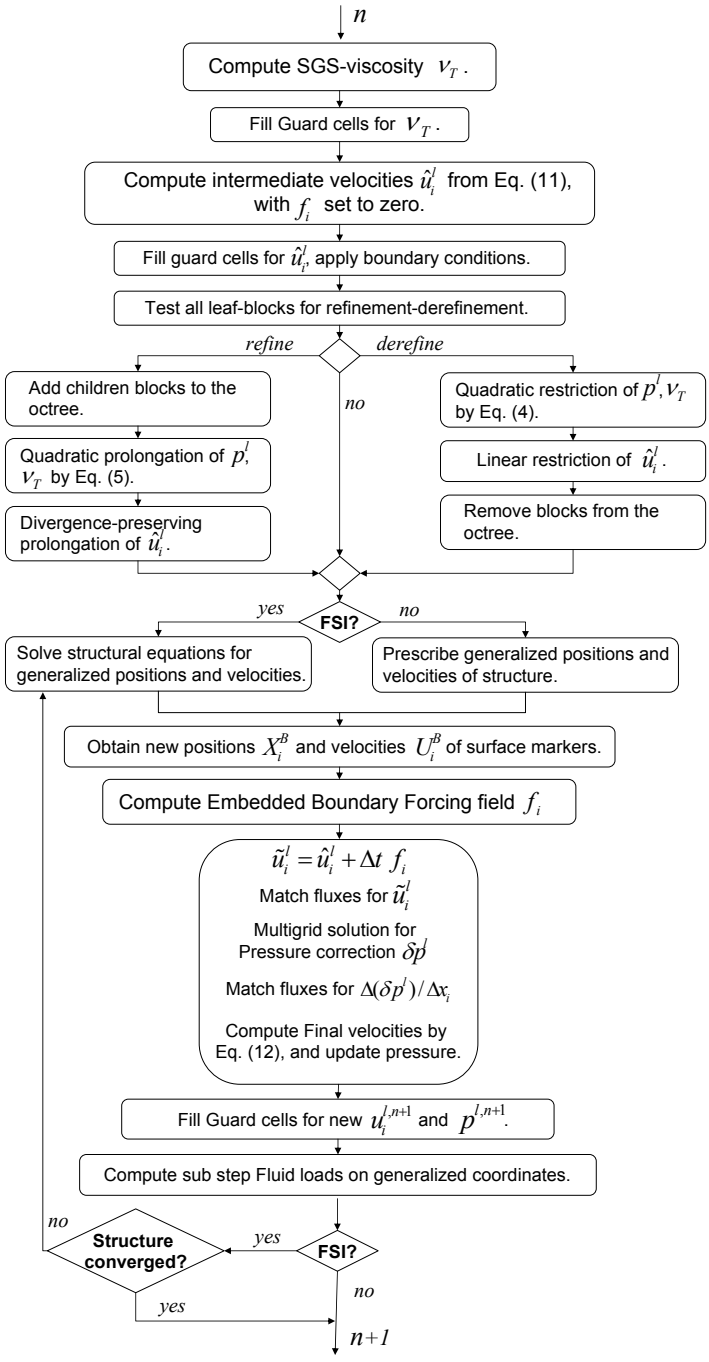


Figure 3.5: Summary of fluid-structure interaction strategy.

In this manner, the intermediate velocities for the time-step following the remeshing step maintain the required divergence level. For each leaf-block flagged for

derefinement, we first check if its parent block contains children that are all flagged for derefinement. In such case a restriction step is performed and the corresponding leaf-blocks are removed from the octree. Quadratic restriction using Eq. (3.1) is utilized for all variables except $\hat{u}_i^l$ and $H(u_i^{l,n-1})$, where linear restriction is used. The latter is a divergence preserving operation.

iv) In the case of FSI a set of predictor-corrector sub-iterations is appied, as shown in the figure. Convergence is assumed when the L_2 error on structures generalized coordinates and velocities is less than a certain tolerance. In all FSI computations of the present study, the tolerance is set to 10^{-8}.

3.2 Numerical studies

In this section, a series of test problems with increasing complexity are studied to evaluate the accuracy and robustness of the proposed AMR formulation. Initially the spatial and temporal accuracy of the method is demonstrated for the Taylor-Green vortex problem. Then, the case of a three-dimensional vortex ring impinging on a wall is considered. Finally two cases, where the accuracy and efficiency of the method in the presence of immersed bodies is examined, are presented: the FSI problem of two falling plates, and sphere bouncing against a wall at $Re = 830$.

3.2.1 Taylor-Green vortex

To investigate the numerical accuracy of the method the Taylor-Green vortex problem is considered. The flow-field is represented by an array of periodic counter-

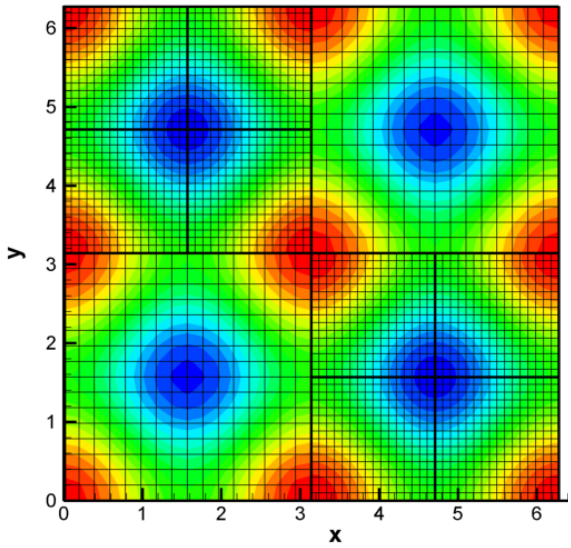


Figure 3.6: AMR grid configuration with two refinement levels for the case of the Taylor-Green vortex. Pressure isolines ranging from $p = -0.4$ to 0.4 at $t = 0.03$ are also shown.

rotating vortices that decay in time, and has an analytical solution of the form:

$$u_a = -e^{-2\nu t} \cos x \sin y, \tag{3.13}$$

$$v_a = e^{-2\nu t} \sin x \cos y, \tag{3.14}$$

$$p_a = -\frac{e^{-4\nu t}}{4} \left(\cos 2x + \cos 2y\right), \tag{3.15}$$

where u_a and v_a are the velocity components in the x and y directions respectively, p_a is the pressure, ν is the kinematic viscosity, and t is the time. The size of our computational domain was $2\pi \times 2\pi$, requiring the use of a mix of homogeneous Dirichlet and Neumann boundary conditions for the velocity components. Grids with two levels of refinement, as well as uniform ones are considered. In the latter case 32^2, 64^2, 128^2 and 256^2 grid points are used and the ghost-cell filling is per-

formed by simple injection of the data between blocks sharing a common boundary. In the former case the base grids utilize 16^2, 32^2, 64^2 and 128^2 computational points on their coarse levels, and the refinement is performed in the second and fourth quadrants of the domain (see Fig. 3.6). To investigate the impact of the ghost-cell filling scheme on the overall accuracy of the method both linear and quadratic interpolation schemes are considered. In all cases the equations of motion are integrated for a total of $T = 0.03$ time units using a timestep of $\Delta t = 5.0 \times 10^{-5}$. The L_2 norm of the residual for the Poisson solution was kept in the order of 10^{-14}.

In Fig. 3.6, pressure isolines at $t = 0.03$ are shown for the AMR grid with two levels of refinement (level 0 being 32^2). It is evident that the main features of the flow are captured. In Figs. 3.7(a) and (c) the L_{inf} error norm for both velocity components at $t = 0.03$, is shown for all cases as a function of the grid size, Δ. Note that in all the two-level calculations, Δ, is computed from the highest refinement level. As expected, in all uniform grid cases second-order accuracy is observed. In the case of AMR the use of linear interpolation for the ghost-cell filling reduces the spatial accuracy, which is now closer to first order. The higher order interpolation scheme described in section 3.1.3, on the other hand, maintains the second order for both velocity components. It is seen that the maximum error incurred by applying quadratic interpolation in the fine grid region of the two-level calculations is of the order of what is obtained for the coarse grid uniform calculation. This is due to contamination of fine grid with coarse-grid error levels and also the inaccuracy derived from the quadratic guard-cell filling itself.

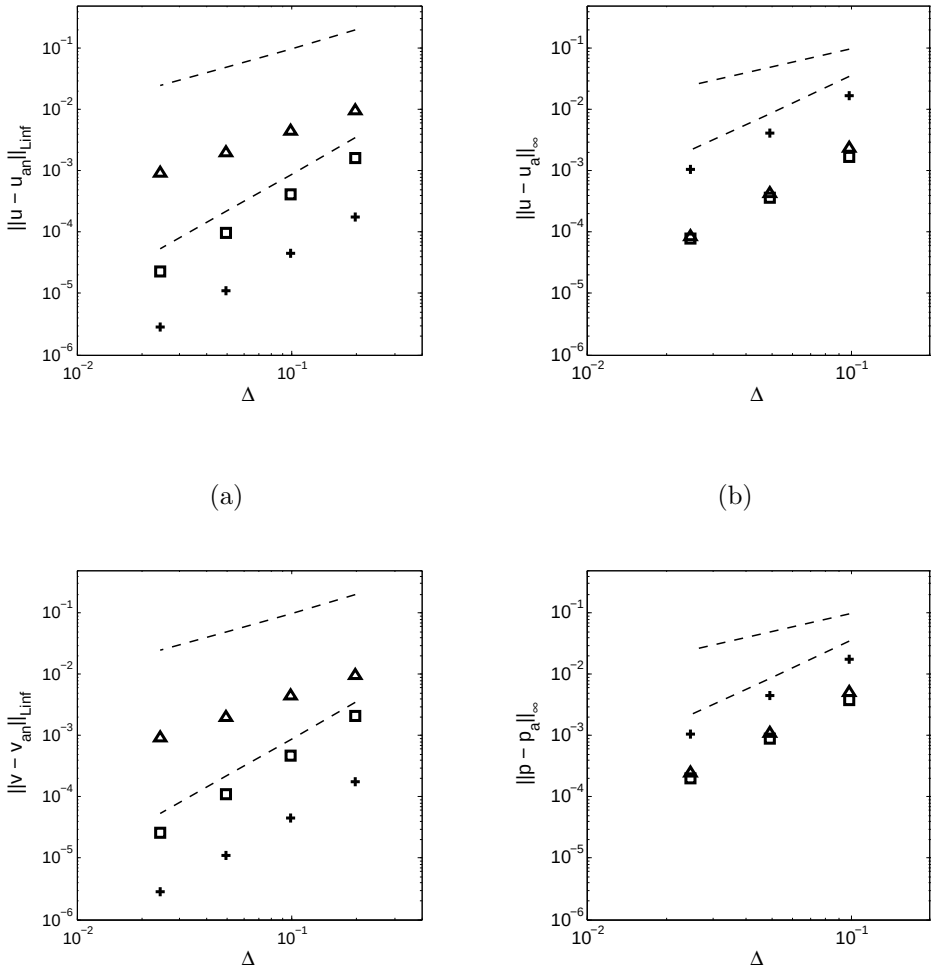


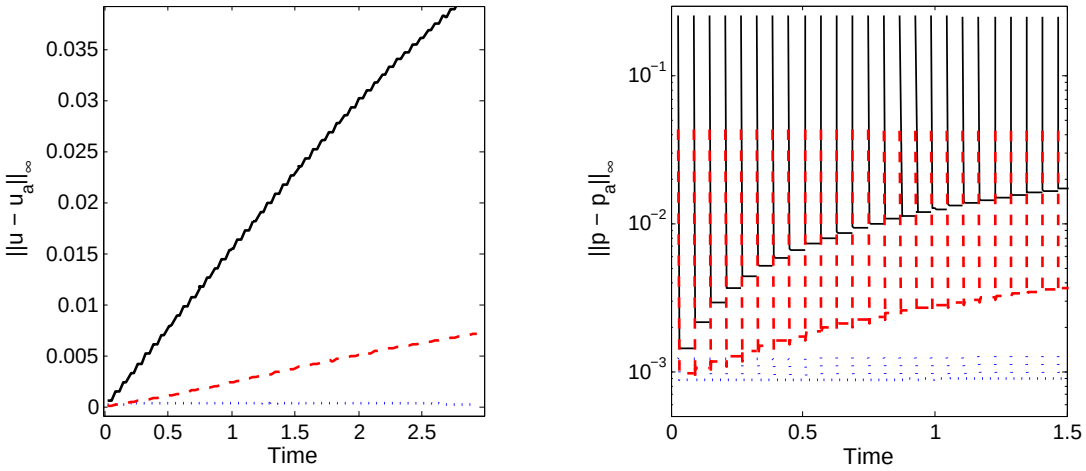
Figure 3.7: Accuracy study on Taylor-Green vortex problem: 1^{st} and 2^{nd} order slopes are added for clarity. (a) & (c) $||u-u_a||_\infty$ and $||v-v_a||_\infty$ respectively as a function of grid spacing at $t = 0.03$. + uniform grid; △ AMR grid, linear interpolation; ▫ AMR grid, quadratic interpolation. (b) & (d) $||u - u_a||_\infty$ and $||p - p_a||_\infty$ as a function of the grid spacing at $t = 0.3$ for the case with dynamic refinement/derefinement performed every 10 timesteps; + linear prolongation; △ quadratic prolongation; ▫ divergence-preserving prolongation.

In all the above computations, the grid is locally refined, but does not evolve with time, as required in most moving boundary problems. To examine the effects of the dynamic adaptation of the grid on the accuracy of the solver, the author also utilizes the Taylor-Green vortex problem with the same setup as above. In this case, however, starting from a single level grid we refine every 10 iterations by adding one refinement level (Fig. 3.6), and then derefine every 10 iterations. Refinement, for example, is performed at iteration count $n = 10, 30, 50, .., 90$, and derefinement at $n = 20, 40, .., 80$. At iteration number 100 the solution is compared to the analytical one, and error norms for the velocity and pressure fields are computed.

An important part of the dynamic grid adaptation is the prolongation of the solution at the newly generated children blocks in refinement areas, and/or the restriction of the solution from eliminated leaf-blocks to its parent, in derefinement areas. As we discussed in section 3.1.5, AMR is performed after the computation of the provisional velocity, $\hat{u}_i^l$. One can show that $\hat{u}_i^l$ is within $O(\Delta t^2)$ of a divergence-free field (see for example [49]) and, therefore, its prolongation/restriction after a grid adaptation step should maintain this property, so the overall temporal accuracy is not affected. As we already discussed in Section 3.1.2, not all prolongation/restriction operators are divergence preserving.

To illuminate their effects on the accuracy of the solution we conducted computations with three different prolongation operators for the velocity field: linear and quadratic, which do not preserve the divergence of $\hat{u}_i^l$, and the quadratic divergence-preserving operator discussed in section 3.1.2. In Figs. 3.7(b) and (d) the L_{inf} error norms for the velocities and pressure are shown as a function of the grid size for all

cases. It is evident that the spatial accuracy is not affected by the choice of prolongation operator, and in all cases second order accuracy maintained. As expected, the error using linear prolongation is about an order of magnitude higher than using the quadratic interpolation variants.



(a) (b)

Figure 3.8: Accuracy study on the Taylor-Green vortex problem: (a) & (b) Variation of $||u - u_a||_\infty$, and $||p - p_a||_\infty$ respectively as a function of time for the case with dynamic refinement/derefinement performed every 10 timesteps;. —— (*black*) linear prolongation; --- (*red*) quadratic prolongation; ····· (*blue*) divergence-preserving prolongation.

The evolution of the error norms with respect to time, on the other hand, reveals some very interesting patterns, as seen in Figs. 3.8(a) and (b). The prolongation operators, which do not preserve the divergence of $\hat{u}_i^l$, result in a monotonically increasing error for both velocity components, while for the divergence-preserving prolongation operator the error is always $O(10^{-4})$. These effects are more profound

in the behavior of the pressure, where the linear and quadratic prolongations result in an error jump of at least three orders of magnitude just after each refinement step, compared to the divergence preserving scheme. The pressure recovers in a few timesteps, but overall the error grows over time. In the case of fluid-structure interaction problems these pressure oscillations can introduce spurious loading on the structure leading to large errors on the solution. For the divergence-preserving scheme the error jump is very small, indicating that the spurious oscillations are practically eliminated. It is important to note that, the choice of other projection scheme variants, where the pressure is not treated incrementally, or making use of high-resolution approximate projections [3] might have a beneficial effect in damping the observed pressure oscillations and numerical errors due to non-divergence preserving interpolations.

3.2.2 Vortex ring impinging on a wall

To demonstrate the ability of the proposed AMR formulation to accurately represent vorticity dynamics, the simple yet very challenging problem of a vortex ring impinging on a wall, is considered. Due to the importance of wall-vortex interactions in many technological applications there is variety of reference data in the literature [104, 28, 73, 96]. In the present work, the author will consider a setup analogous to the one reported in [96], where a vortex ring is generated in the center of the $x - y$ plane, and at a distance $z_o = 1.5D_o$ from the wall as shown in Fig. 3.9. The vortex

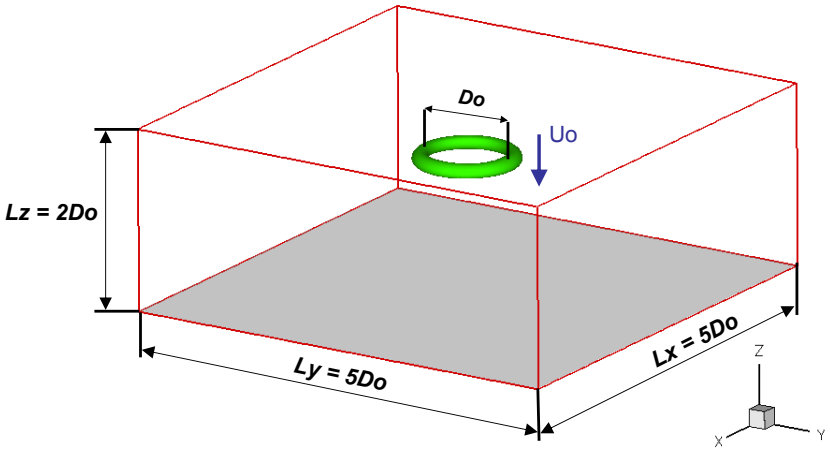


Figure 3.9: Computational setup for the vortex ring impinging on a wall

ring was generated by introducing an impulsive body force of the form:

$$\begin{aligned} f_z(r,z,t) &= -A_o T(t) F(z) H(r), \quad where \\ T(t) &= 0.5[1+\tanh\alpha(\tau - t')] \ , \\ F(z) &= 0.5[1+\tanh\beta(B_z - z')] \ , \\ H(r) &= 0.5[1+\tanh\gamma(C_r - r)] \end{aligned} \tag{3.16}$$

Note that $t' = |t - t_o|$ and $z' = |z - z_o|$. Setting $A_o = 350$, $\alpha = 500$, $t_o = 0.05$, $\tau = 0.04$, $\beta = 100$, $B_z = 0.1$, $\gamma = 100$ and $C_r = 0.5$, the resulting vortex ring had $Re = U_o D_o/\nu = 570$, where U_o and D_o are its initial diameter and self-induced translation velocity and ν is the kinematic viscosity of the fluid. This is lower than the value of $Re_o = 645$, reported in [96]. A closer match of the initial conditions was not possible due to the fact that some of their forcing parameters were not listed.

The AMR grid was adaptively refined in areas of high velocity gradients. Five levels of refinement were used and the total number of points was of the order of 2×10^6. The grid adaptation was done every 10 timesteps using the modulus of

the vorticity field, $|\boldsymbol{\omega}|$, as a test variable. The grid was refined in leaf blocks where $|\boldsymbol{\omega}| > 5.5U_o/L_o$ anywhere within the block, and derefined if $|\boldsymbol{\omega}| < 4.0U_o/L_o$. Periodic boundary conditions are applied in the x and y directions, and non slip at the top and bottom walls in the z direction (see Fig. 3.9). The AMR solution is compared to a computation of exactly the same problem using a single-block, finite-difference solver [9]. The single-block grid in the latter case utilizes $256 \times 256 \times 128$ grid cells, and is uniform in the x and y directions, while it is stretched in the wall-normal direction, z. The resulting cell size in the near wall area was comparable to the one at the highest refinement level in the AMR computation.

In Fig. 3.10, vorticity contours are shown for both the single-block and AMR calculations. The vorticity normal to the $y-z$ plane is shown at four different times during the calculation. In the AMR case the block boundaries are also shown in the figure. The similarity between the two computations is striking. As the primary vortex approaches the wall there is an increase of its radius, which is accompanied by a noticeable decrease in the core size (i.e. Figs. 3.10a-c). The boundary layer generated underneath the primary vortex thickens due to the adverse pressure gradient in the radial direction (Fig. 3.10b-f), and eventually the accumulated vorticity pinches off forming a secondary vortex (Fig. 3.10c-g).

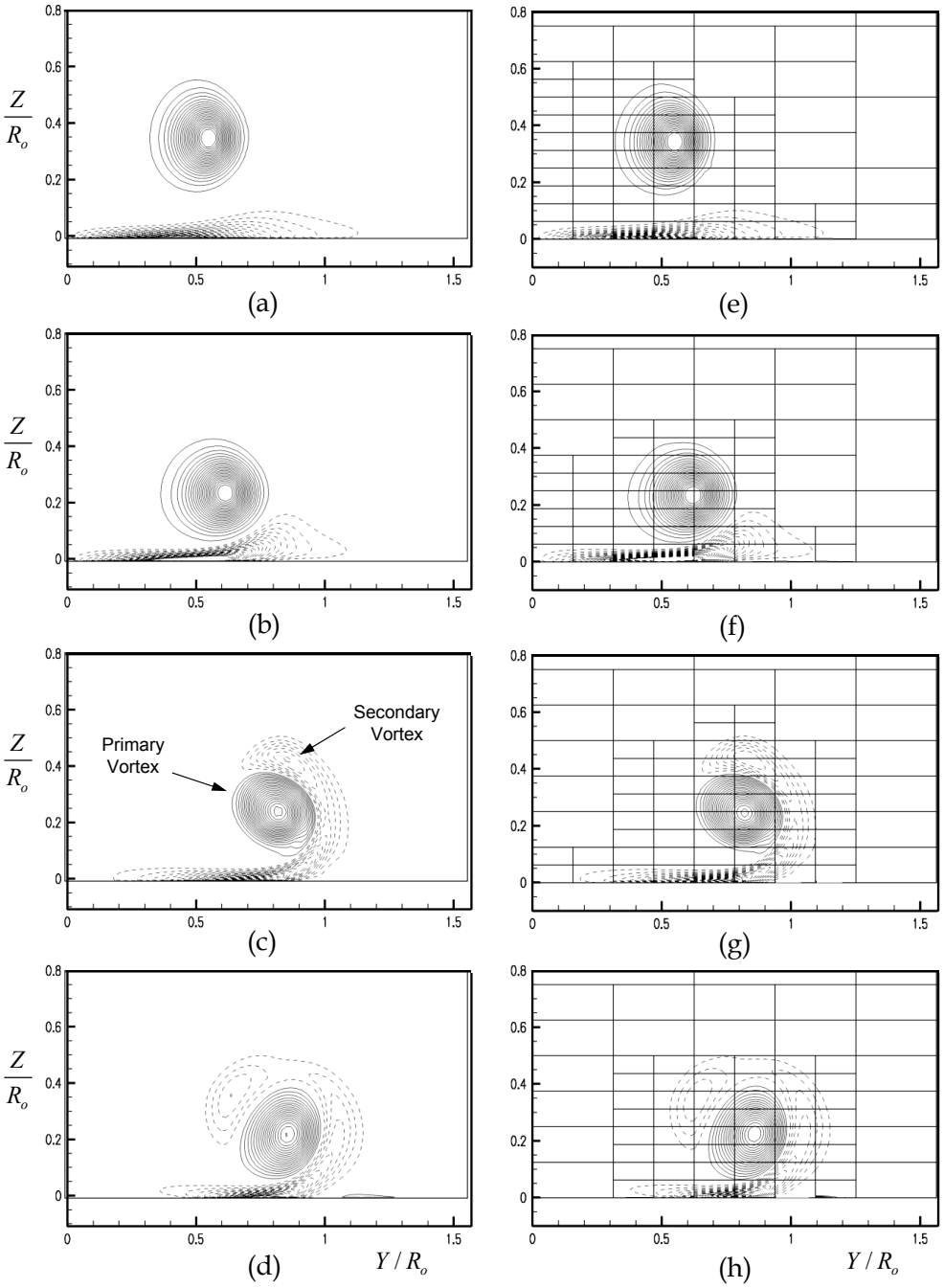
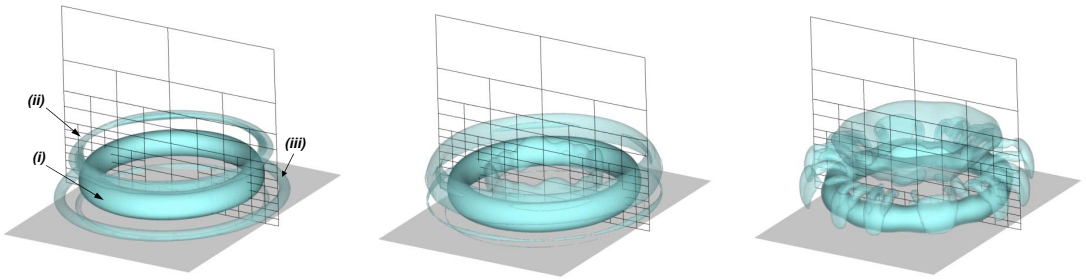


Figure 3.10: Vorticity isolines at an $y - z$ plane for the case of the vortex ring impinging to a wall. Forty five $\omega_x R_o/U_o$ contours from -50 to 50 are used. Left side is from the single-block calculation,and the right side from the AMR. *(a),(e)* $t = 1.3$; *(b),(f)* $t = 1.5$; *(c),(g)* $t = 2.1$; *(d), (h)* $t = 2.5$.



(a) (b) (c)

Figure 3.11: Isosurfaces of Q for the case of a vortex impinging to a wall at three different times. The AMR block distribution is shown at an $y - z$ plane. *(a)* $t = 2.0$; *(b)* $t = 2.7$; *(c)* $t = 4.4$. *(i)*, *(ii)*, and *(iii)* indicate the primary, secondary and tertiary vortices.

A three-dimensional view of primary and secondary vortex structures can be seen in Figure 3.11, where isosurfaces of the second invariant of the velocity gradient tensor, $Q = -1/2(\partial\overline{u}_i/\partial x_j \partial\overline{u}_j/\partial x_i)$, are plotted. The secondary vortex orbits the primary vortex (Fig. 3.11a), and later undergoes a 'buckling' instability as it is compressed by the primary vortex (Fig. 3.11b). The secondary vortex breaks into smaller structures that are advected under the primary vortex and affect the vorticity generation near the wall. This results in small slender structures dominated by radial vorticity (see Fig. 3.11c). At the same time a third vortex pinches-off and also starts orbiting the primary one. This structure at this Re_o is fairly stable, and stays in the proximity of the primary vortex for the rest of the calculation (Fig. 3.11c).

In Figure 3.12 the trajectories of the primary and secondary vortex centers are shown for both computations. The numerical results from Swearingen *et al.* [96]

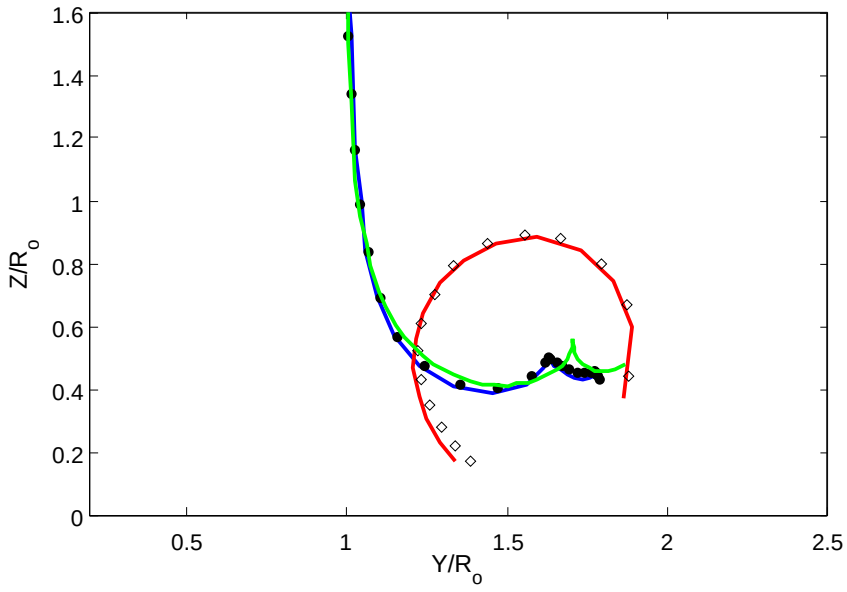


Figure 3.12: Trajectories of the centers of the primary and secondary vortices at $Re_o = 570$. Symbols are from the uniform grid and lines from the AMR grid computation. • primary vortex; ◦ secondary vortex; green line is the trajectory of the primary vortex from [96] at $Re_o = 645$.

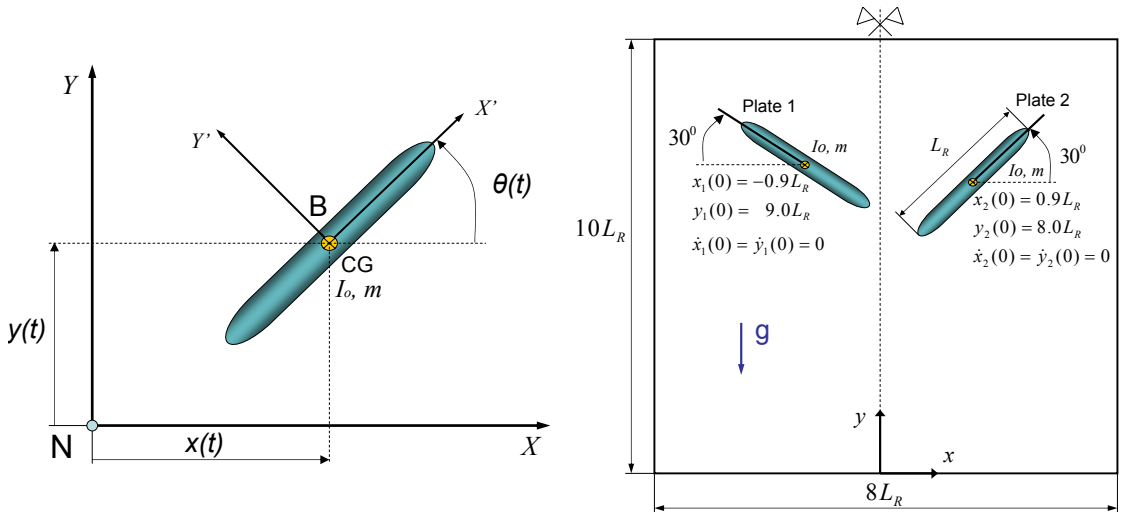
for a similar setup at $Re_o = 645$ are also included for comparison. The agreement between the single-block and AMR calculations is excellent and the trajectories of both vortices are identical indicating that the proposed AMR strategy properly captures the vorticity dynamics. The agreement with the results reported in [96] is also good. Small discrepancies near the wall are primarily due to the ambiguity in the definition of the vortex center as well as the differences in the initial conditions.

3.2.3 Fluid-Structure interaction of two falling plates

To test the accuracy of the approach in fluid-structure interaction problems the case of two falling plates is considered (see Fig. 3.13). In this case, Eq. (2.3) governing the dynamics of each plate can be reduced to the following form:

$$\begin{bmatrix} [\mathbf{I}]_{(3\times3)} & & [\mathbf{0}]_{(3\times3)} & \\ & m & 0 & 0 \\ [\mathbf{0}]_{(3\times3)} & 0 & m & 0 \\ & 0 & 0 & I_o \end{bmatrix} \begin{Bmatrix} \dot{\mathbf{q}}_{1\,(3\times1)} \\ \\ \dot{\mathbf{q}}_{2\,(3\times1)} \\ \\ \end{Bmatrix} = \begin{Bmatrix} \mathbf{q}_{2\,(3\times1)} \\ f_x(\mathbf{q}_1, \mathbf{q}_2) \\ f_y(\mathbf{q}_1, \mathbf{q}_2) - mg \\ M_o(\mathbf{q}_1, \mathbf{q}_2) \end{Bmatrix}, \tag{3.17}$$

with $\mathbf{q}_1 = [x_{(t)}\ y_{(t)}\ \theta_{(t)}]^T$ and $\mathbf{q}_2 = \dot{\mathbf{q}}_1$. $x_{(t)}$, $y_{(t)}$ are the coordinates of the center of mass of the plate in the x and y directions respectively, and $\theta_{(t)}$ its orientation angle (see Fig. 3.13a). f_x, f_y, are the corresponding hydrodynamics forces acting on the plate, and M_o, is their moment with respect to the center of mass. Eq. (3.17) has been made dimensionless using the chord length, c, the mean descent velocity of plate 2, U_R, and the fluid density, ρ_f, as reference variables. The thickness of each plate is 10% of the chord length, and their density is, $\rho_B = 5.1\rho_f$. Also $g = 3.0$, $m = 0.5$, and $I_o = 4.2 \times 10^{-2}$. The resulting Reynolds number is $Re = U_R c/\nu = 200$. The computational domain together with the initial conditions are shown in Fig. 3.13b. Both plates are initially at rest. To evaluate the accuracy of the proposed AMR scheme two separate simulations are considered. First, a single level calculation using a uniform grid of 1024×1280 nodes with spacing $\Delta x = \Delta y = 0.0078 L_R$ is conducted. Then, an AMR computation with four levels of refinement is carried out. Mesh adaptivity in this case was guided by two criteria: i) the presence or not of a

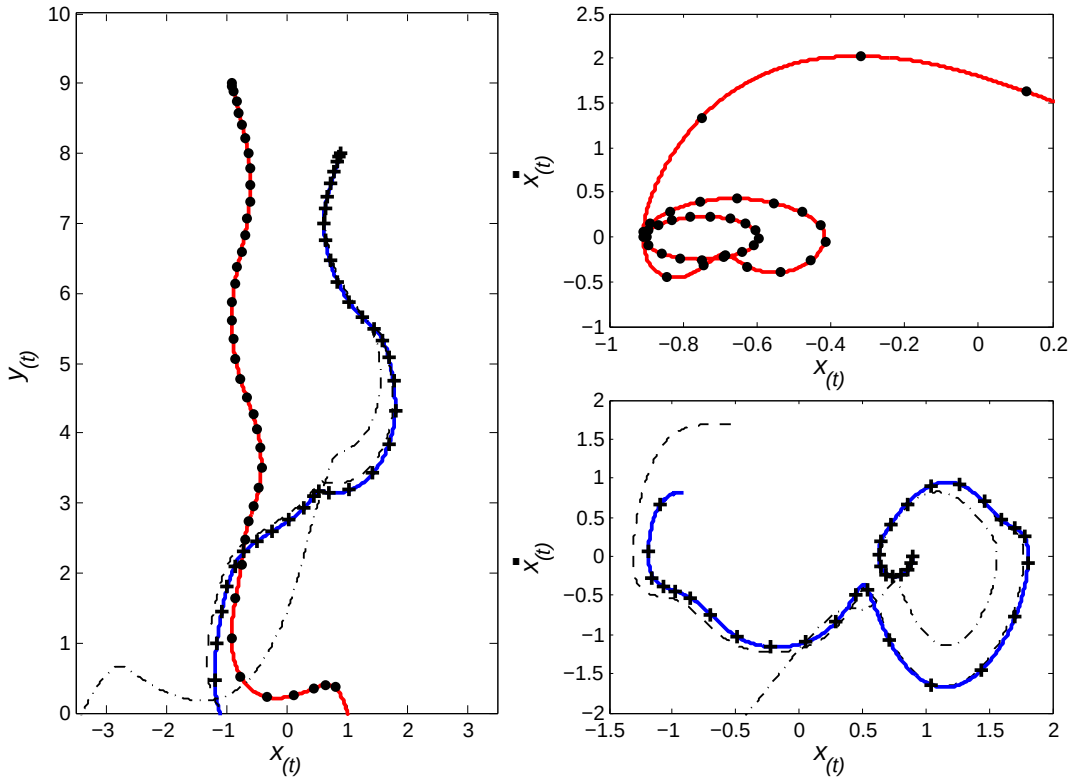


(a) (b)

Figure 3.13: (a) Variables describing two-dimensional rigid body motion for the falling plates. (b) Domain, boundary and initial conditions for the two falling plates problem.

rigid body in a grid block. ii) the vorticity modulus, $|\boldsymbol{\omega}|$, as in the example given in the previous section. As a result the highest refinement level always surrounds the plates, as well as the areas of high velocity gradients in their wake. Note that grid cell size at the highest refinement level is the same as the one used in the uniform grid calculation. Both computations were advanced in time for 22×10^3 steps, with $\Delta t = 2.0 \times 10^{-3}$.

The trajectories of the centers of mass for both bodies as a function of time, and phase diagrams of positions and velocities for the horizontal and vertical degrees-of-freedom are shown in Figs. 3.14a,b respectively. The agreement with the single block computation is excellent demonstrating the fidelity of the proposed AMR



(a) (b)

Figure 3.14: (a) Position of center of mass for each plate as a function of time. (b) Phase diagrams for $x_{(t)}$. Symbols are from the AMR computation and lines from the uniform grid computation. • plate 1; + plate 2. Also, resulting trajectories and phase diagrams for plate 2 with coarser resolution uniform grids are plotted: (- -) 512×640 grid, (-.-) 256×320 grid.

formulation. The trajectories and phase diagram obtained for plate 2 with two uniform grids of coarser resolution than the one obtained with 4 refinement levels are also plotted for comparison. Grid convergence of the solution is observed as the uniform-grid resolution is increased. We see here the effectiveness of the AMR formulation in representing the uniform, fine grid solution, which is not achievable

using coarser uniform meshes.

A frame-by-frame comparison of the vorticity field in the two computations is also shown in Fig. 3.15 with very good agreement as well. It should be mentioned that the proposed FSI treatment has been tested on a single grid-block setting by Yang et al. [114], on different incompressible flow problems, and found second order accurate in space and time. It is interesting to note the transition from steady fluttering fall to tumbling for plate 2 (see Fig. 3.15d,f). Tumbling motion has been observed by Andersen *et al.* [4] for a single falling plate under similar flow conditions, and usually occurs for $Re > 100$ and $I_o > 3 \times 10^{-2}$ [18]. For plate 1 transition from a fluttering fall to tumbling is limited by the wake of plate 2.

3.2.4 Three dimensional example: Sphere-wall collision

To investigate the robustness and accuracy of the proposed method in three dimensional configurations we performed computations of a rigid sphere bouncing off a wall. Problems involving collisions between immersed bodies are particularly challenging for direct-forcing schemes, since the presence of two or more Lagrangian markers from different bodies in the proximity of the same Eulerian grid cell is usually the source of ambiguity. The proposed forcing scheme treats such situations in a robust manner without the need for special treatments.

The particular configuration we selected has applications to particulate flows, and a number of experimental (i.e. [31], [48]) and numerical (i.e. [6]) results are available in the literature for comparison. The dominant parameter in the collision

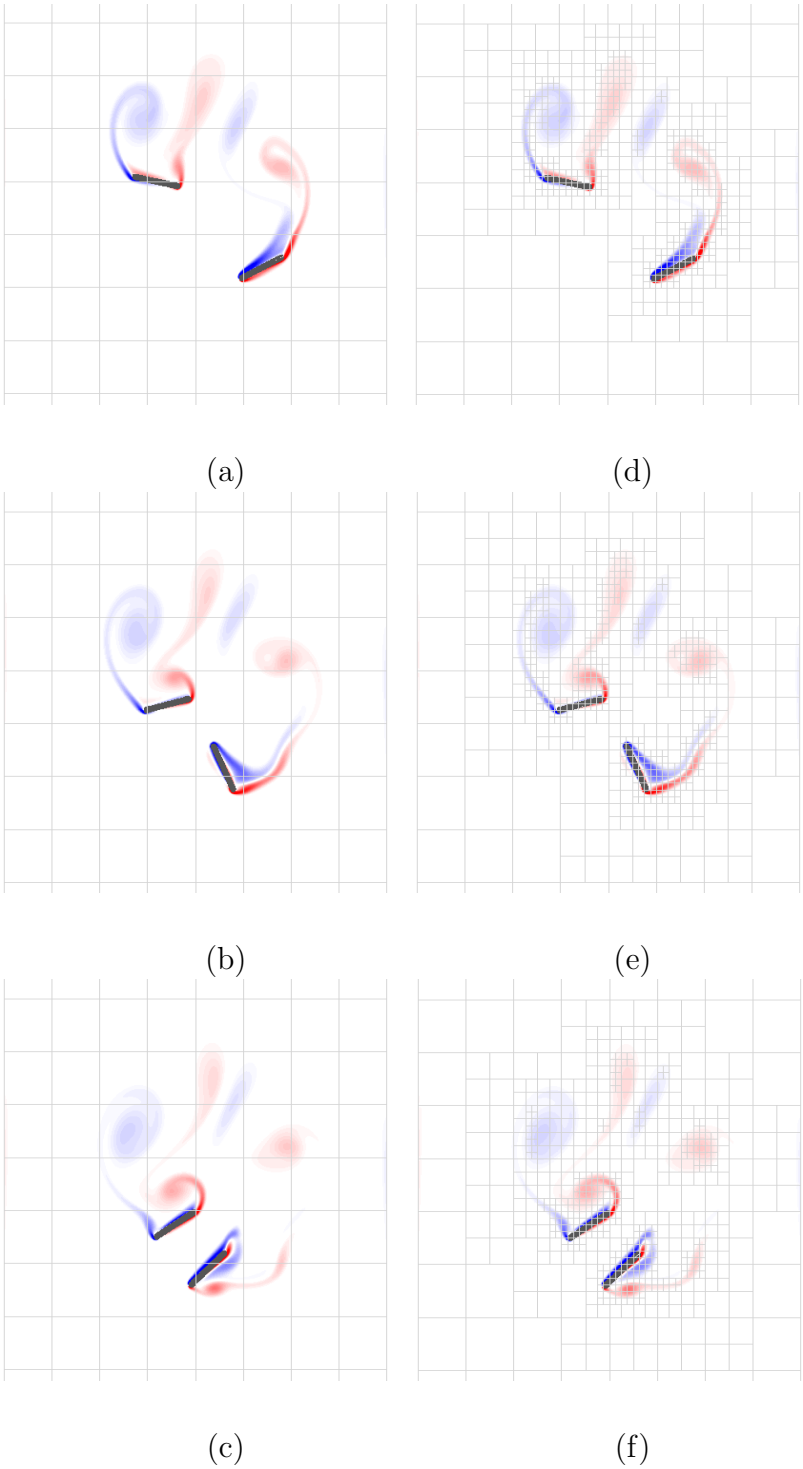


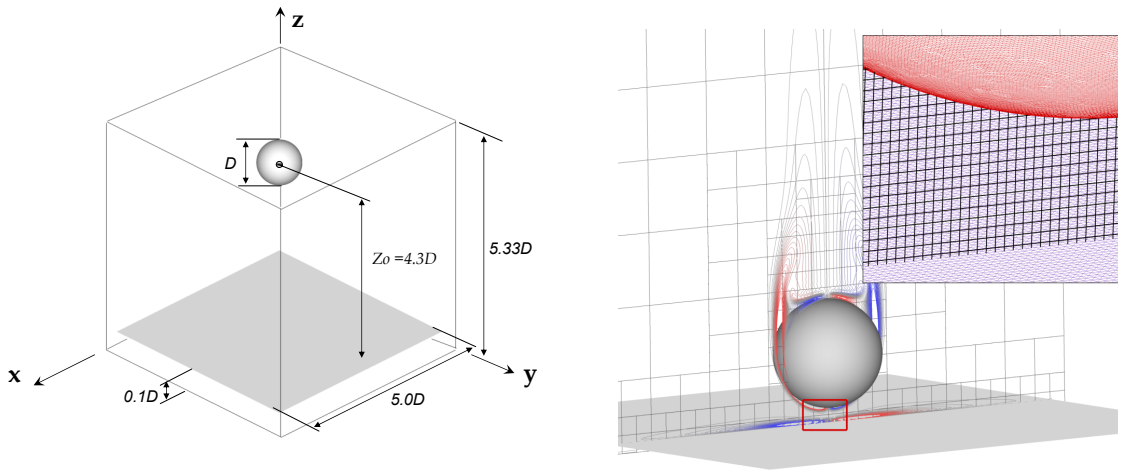
Figure 3.15: Comparison between uniform (left side) and AMR (right side) grid computations for the case of the falling plates. Vorticity isolines are shown at (a) & (d) $t = 4.6$; (b) & (e) $t = 5.2$; (c) & (f) $t = 5.8$.

process is the Stokes number, $St = 1/9(\rho_b/\rho_f)Re$, where ρ_b and ρ_f are the particle and fluid densities respectively, and Re is the Reynolds number based on the particle diameter, D, and the translational velocity, U_f, an instant before impact. For low values of the Stokes number ($St < 10$) no rebound will occur, even if the dry restitution coefficient, e_{dry}, is different from zero. For $St > 10$ rebound occurs, and the total restitution coefficient, e_T, is lower than e_{dry}. For large values of the Stokes number ($St > 500$) the total restitution coefficient, e_T, approaches e_{dry}. If $e_{dry} = 0$, then the Reynolds number is the only dominant parameter.

Below we will present results from two different configurations: i) a case where no rebound occurs, which resembles the conditions in the experiments by Eames and Dalziel [31]; ii) a case where rebound is allowed, which resembles the conditions in the axisymmetric Navier-Stokes computations by Ardekani *et. al* [6]. In figure 3.16a a sketch of the computational domain is shown. Both the sphere and the wall are immersed into a locally refined Cartesian mesh [11], and are represented by an unstructured Lagrangian grid with 2.7×10^5 and 2.9×10^5 markers respectively (see figure 3.16b). The Eulerian grid is arranged in a way that the resolution around the sphere is $\Delta x = 0.01D$. In both cases the horizontal displacements and rotations are constrained and the vertical displacement, $z_s(t)$, is governed by:

$$m_s \ddot{z}_s(t) = -m_s g + f_z(z_s, \dot{z}_s, t), \tag{3.18}$$

where m_s is the mass of the sphere, g is the acceleration of gravity and f_z is the hydrodynamic force on the sphere in the vertical direction. The Navier-Stokes equations governing the dynamics of the fluid, and equation (3.18) governing the dynamics



(a) (b)

Figure 3.16: Sphere-wall interaction problem. (a) computational setup; (b) Lagrangian and Eulerian grids.

of the sphere are solved as a coupled system using the predictor-corrector strategy proposed in [114].

Initially the sphere is located at a distance of $4.3D$ from the horizontal wall, and is impulsively started to reach a velocity corresponding to an initial Reynolds number of $Re_i = 510$. The density ratio is fixed to $\rho_s/\rho_f = 3.2$. The resulting Reynolds number just before impact is $Re = 830$ and the corresponding Stokes number is $St = 295$. Contact is assumed to take place when sphere and floor are within a distance of one cell size. In the experiments by Eames and Dalziel [31] the Reynolds number before impact was $Re = 850$, but the motion of the sphere was prescribed and no bounce was allowed to occur. To simulate these conditions in our computations the dry restitution coefficient, e_{dry}, was set to zero (see equation (3.19) below).

In Figure 3.17 dye flow visualizations form the experiments in [31] are directly compared to our computations, where the flow patterns are visualized using azimuthal vorticity isolines. The computed flow patterns are in very good qualitative agreement with the experiment. Direct quantitative comparisons are not possible due to the fact that vorticity and scalars, such as dye, do not have the same dynamics as a result of their different diffusivities and the absence of vortex stretching in the case of scalars. As the sphere approaches the wall the detached shear layers and the small recirculating areas behind the sphere are evident in both experiments and computations (figure 3.17a, b). Just after impact the vorticity in the shear layers moves towards the wall generating a layer of vorticity with opposite sign on the sphere's surface (see figure 3.17c). As soon as this layer separates (figure 3.17d), a vortex dipole is formed and moves away from the sphere (figures 3.17e, f). By locating the center of these vortices in the computations and the experiments we found that in their trajectories through time are always within 5%.

Next, we considered the bouncing sphere problem. The dry restitution coefficient was the one used in the axisymmetric calculations of Ardekani *et. al* [6], $e_{dry} = 0.97$, which is typical for steel-sphere and glass-wall collisions. The collision process starts when the distance between the particle and the wall is equal to the roughness height, h_r. We assume that rough surface has a negligible effect on the viscous force until the gap between the smooth portions of surfaces becomes equal to the size of largest roughness element, h_r. This is also the moment the impact is assumed to occur. Details can be found in [6]. Just after the collision we define a

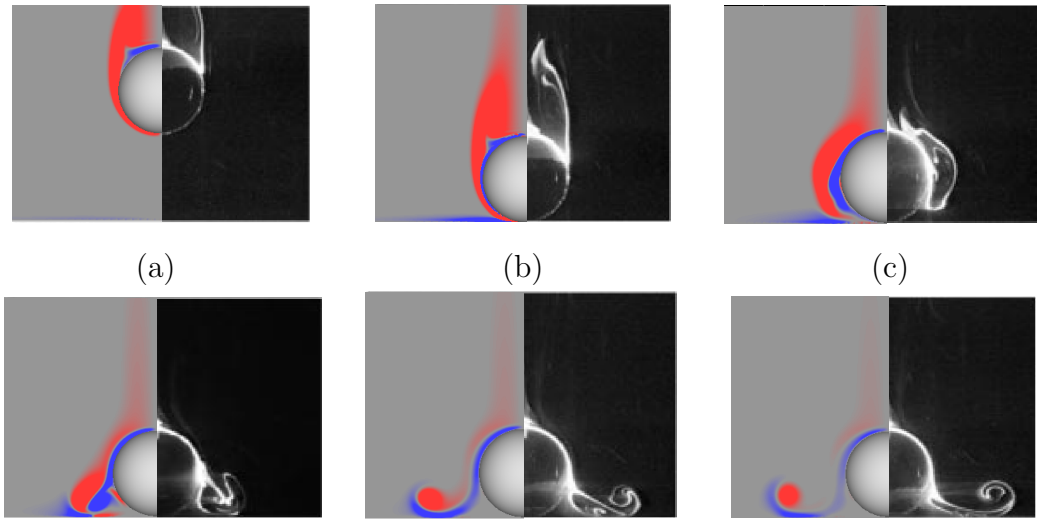


Figure 3.17: Sphere-wall interaction with $e_{dry} = 0.0$. The left half in all figures are azimuthal vorticity isolines from the present computations, and the right half are snapshots from the dye visualizations in [31] at $Re = 850$. (a) $-\tau_f$, (b) 0,(c) τ_f,(d) $2\tau_f$, (e) $3\tau_f$, and (f) $4\tau_f$, where $\tau_f = D/U_f$.

new set of initial conditions for equation (3.18) as follows:

$$z_{s2} = z_{s1}; \qquad \dot{z}_{s2} = -e_{dry}\dot{z}_{s1} \tag{3.19}$$

where z_{s1}, $\dot{z}_{s1}$ and z_{s2}, $\dot{z}_{s2}$ are the sphere's vertical position and velocity before and after the impact respectively.

In numerical simulations of contact problems it is important that the lubrication layer between the bodies is resolved. In all our computations the surface roughness, which practically determines how close the bodies can come, and numerical resolution were selected in a way that a minimum of $5-6$ Eulerian grid points were present between the bodies during impact. We first considered the same case

as in the above example ($Re = 830$ and $St = 295$) with a non-zero restitution coefficient, $e_{dry} = 0.97$, in order to compare the vortex dynamics with the no-bounce case. In figure 3.18 azimuthal vorticity isolines of the no-bounce (left) and bouncing sphere (right) are shown. Just before impact (figure 3.18a) the flow for both cases is identical since we start from the same initial conditions. At a later time and after the first impact (3.18b, c), the layers of vorticity with alternating sign that were observed in the case with $e_{dry} = 0.0$ can also be seen in the bouncing sphere problem. In the latter case, however, the primary vortex originating in the wake is weaker and the upward motion of the sphere causes the shear layer at the surface to roll-up into a strong secondary vortex. As the downward motion of the sphere starts the secondary vortex pinches-off (see figure 3.18d) and by the time the second bounce occurs it is dissipated. As a result the wake and secondary vortices do not form the dipole structure seen in the no bounce case.

Ardekani and Rangel [6] defined a total restitution coefficient, $e_T = U_a/U_f$, where U_a is the velocity of the sphere at $tU_f/D = 0.07$ after the impact time, t_c, which measures the dissipative effect of the fluid, as it is drained and subsequently reenters the layer between sphere and wall. For the present case we found $e_T = 0.63$. A direct comparison with the computations reported in [6], where they reported $e_T = 0.92$, is not possible because of the differences in the Reynolds numbers. Our e_T value, however, is consistent with the trend reported in [6] where a decrease in e_T was observed with increasing Reynolds number and constant Stokes number. In particular they found a decrease of 5.0% for e_T at $St = 301$ when Re increases from 35 to 162. To further verify the accuracy of our formulation we also conducted

a computation that closely matches the low Reynolds number conditions in the simulation by Ardekani and Rangel [6]. The Reynolds Number before impact was $Re = 76.8$ and the Stokes number $St = 299$. In this case we computed an $e_T = 0.91$, which is in very good agreement with the reference results of $e_T = 0.92$.

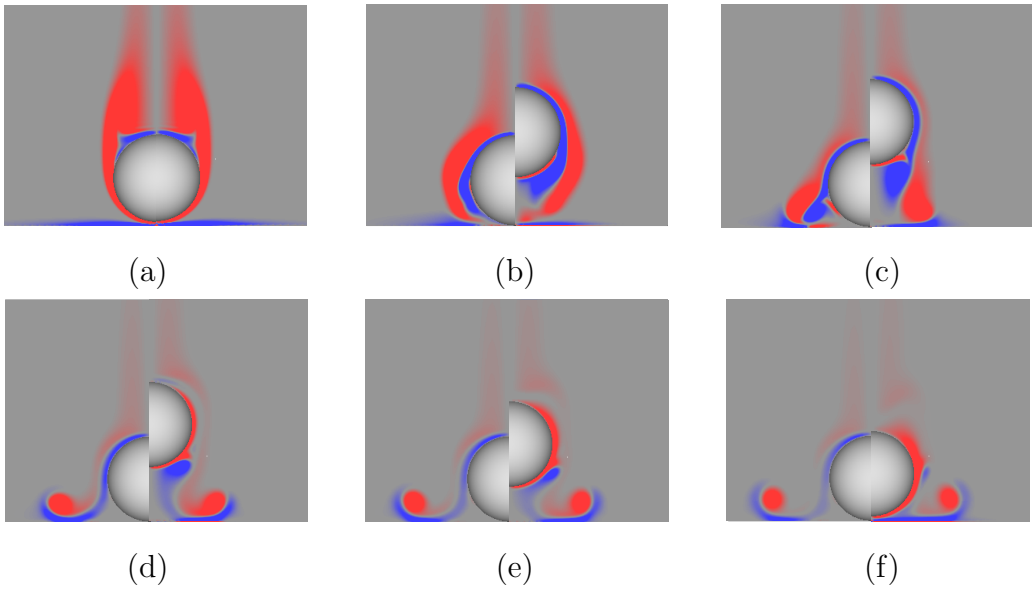


Figure 3.18: A comparison of $e_{dry} = 0.00$ (left) and $e_{dry} = 0.97$ (right) computations for the sphere-wall interaction at $Re = 830$. (a) $-0.1\tau_f$, (b)τ_f, (c)$2\tau_f$, (d) $3\tau_f$, (e) $3.5\tau_f$, and (f) $4.1\tau_f$, where $\tau_f = D/U_f$.

Chapter 4

Large eddy simulation for discontinuous grids

In the introductory Chapter we described a set of features of LES which make its combination with S-AMR methods non trivial. The grid discontinuities introduced in AMR methods may lead to numerical errors, in some cases significant ones. This is especially true if non-dissipative, centered spatial discretization schemes like the one previously proposed are used. In the following section we give a formal introduction to eddy viscosity models in large eddy simulations. We then develop a simple strategy to vary the filter size for filtered variables around grid discontinuities. In order to evaluate the sources of error and develop a strategy to minimize these, we will study in Section 4.2 the flow of spatially decaying isotropic turbulence advected across a refinement interface. In this study we see that using LES together with explicit filtering of the advective terms [60] leads to an effective control of aliasing and interpolation errors when turbulence travels across a derefinement interface. This, strategy is used in Section 4.3 in the simulation of transitional flow around a sphere at $Re = 10000$.

4.1 Mathematical formulation

We employ overbars to denote the resolved quantities on LES computations. We consider the case where the SGS stresses, τ_{ij} in the momentum equations (2.1),

are parameterized using standard eddy-viscosity models:

$$\tau_{ij} - \frac{2}{3}\delta_{ij}\tau_{kk} = -2\nu_T \overline{S}_{ij}, \tag{4.1}$$

where ν_T is the eddy viscosity defined as:

$$\nu_T = C\Delta_f^2 \left|\overline{S}\right|; \tag{4.2}$$

C is a model coefficient, Δ_f the filter width, and $\overline{S}_{ij}$ the resolved strain-rate tensor:

$$\overline{S}_{ij} = \frac{1}{2}\left(\frac{\partial \overline{u}_i}{\partial x_j} + \frac{\partial \overline{u}_j}{\partial x_i}\right). \tag{4.3}$$

In this work we use two different models to compute ν_T: the standard Smagorinsky model [92, 56], and the Lagrangian dynamic eddy-viscosity (LDEV) model proposed by Meneveau *et al.* [66]. In the former the model coefficient is a constant set to $C = 0.65^2$, while in the latter it is dynamically computed during the calculation:

$$C = -\frac{1}{2}\frac{\langle \mathcal{L}_{ij} M_{ij} \rangle}{\langle M_{qr} M_{qr} \rangle}, \tag{4.4}$$

where

$$M_{ij} = \Delta_f^2 \left(\alpha^2 \left|\widehat{\overline{S}}\right| \widehat{\overline{S}}_{ij} - \widehat{\left|\overline{S}\right| \overline{S}_{ij}}\right), \tag{4.5}$$

$$\mathcal{L}_{ij} = \widehat{\overline{u}_i \overline{u}_j} - \widehat{\overline{u}}_i \widehat{\overline{u}}_j, \tag{4.6}$$

Note that $\widehat{.}$ is the test-filter operator, representing a filter with width Δ_t larger than the grid filter Δ_f, and $\alpha = \Delta_t/\Delta_f$ is the ratio between the test and grid filter-widths. The numerator and denominator in equation (4.4), are averaged over particle pathlines. Details on the present implementation of the Lagrangian model can be found in [90].

Any filtered quantity, $\overline{f}$ and $\widehat{f}$, is obtained by recursive application of a one-dimensional filter in the three coordinate directions; the one-dimensional filter is defined as:

$$\overline{f}^{1D} = \sum_{m=j-M}^{j+M} W_m f_m \tag{4.7}$$

(and similarly for $\widehat{f}$). The filter-width corresponding to above operation is given by [59]:

$$\Delta_f = \Delta \left(12 \sum_{j=-M}^{M} j^2 W_j \right)^{1/2}, \tag{4.8}$$

where Δ is the grid size. In our computations we set $\Delta_f/\Delta = \sqrt{4/3}$ and $\alpha = \Delta_t/\Delta_f = \sqrt{6}$.

4.1.1 Explicit filtering of the non-linear term

When explicit filtering of the non-linear term is used in LES, the filtered velocity product in the advective terms of the momentum equations is decomposed as [54]:

$$\overline{u_i u_j} = \overline{\overline{u}_i \overline{u}_j} + \left(\overline{u_i u_j} - \overline{\overline{u}_i \overline{u}_j} \right) \tag{4.9}$$

where $\tau'_{ij} = \overline{u_i u_j} - \overline{\overline{u}_i \overline{u}_j}$ is the new subgrid stress tensor to be modeled. The modeling strategy used here is the same as discussed in the previous Section, and the same filters are applied. The secondary filtering operation has the effect of inhibiting the generation of frequencies higher than the grids characteristic wavenumber [103], thus reducing the numerical error at the higher wave-numbers included by the mesh (this is particularly important in the framework of the LDEV model, as the parameter C and the SGS stresses, depend intrinsically on the accuracy at which the smallest

resolved scales are computed). However, explicit filtering as defined by equation (4.9) is not in general Galilean invariant with respect to translation. Lund [60] showed that the invariance error is reduced if the filter used becomes closer to the sharp Fourier cutoff; adding a scale-similarity type term to the model also leads to a Galilean-invariant formulation [81]. The present study is limited to eddy-viscosity models with trapezoidal filters and Galilean invariance is, therefore, not enforced. The error, however, is proportional to the to velocity of the moving reference frame [60], which in our case is zero.

4.2 Spatially decaying isotropic turbulence past a refinement interface

To quantify the numerical errors arising on LES in the vicinity of grid refinement interfaces, and explore ways of reducing them, we study a simple flow configuration, without some of the complexities present in wall-bounded flows (mean shear, turbulence anisotropy, etc.). We examine homogeneous isotropic turbulence advected by a uniform velocity, U_c, across a fine-coarse grid interface, concentrating on the case in which the interface is normal to the main advection direction. The development of the turbulence in the vicinity and downstream of the interface are examined to evaluate the errors introduced by the grid discontinuity, and the distance required for a return to equilibrium. We also investigate the effect of explicit filtering the advective term in the momentum equations as a means of separating the filter size from the finite-difference grid size, and to control the frequency content

Case	Grid	Filter Variation
C	$48^3 + 48^3$	N/A
F	$96^3 + 96^3$	N/A
FC-S	$96^3 + 48^3$	Sharp
CF-S	$48^3 + 96^3$	Sharp
FC-V	$96^3 + 48^3$	Linear variation over $1.6L_r$

Table 4.1: Summary of Lagrangian Dynamic inflow-outflow computations.

and numerical errors of the solution close to the interface.

4.2.1 Computational setup

We conducted simulations of spatially decaying homogeneous isotropic turbulence. A typical configuration is shown in Figure 4.1 (left). The computational domain consists of two adjoining blocks, each with dimensions $2\pi \times 2\pi \times 2\pi$ in the x, y and z directions respectively. Each block is resolved by 24^3, 48^3 or 96^3 grid points; the mesh is uniform and the grid size is the same in all directions. The two blocks have the same resolution for the single-grid calculations, while in the two-level cases the grid size is discontinuous, with a factor of two ratio in each direction (*i.e.*, we go from a 48^3 grid to a 96^3 one in the coarse-to-fine cases, from a 96^3 grid to a 48^3 one in the fine-to-coarse ones). The refinement jump (the coarse-fine grid interface) is located at the center of the domain ($x = 0$), normal to the x direction. This interface defines two sets of calculations, *coarse-to-fine* (CF) as in Figure 4.1, or *fine-to-coarse* (FC). Tables 4.1-4.2 show the sets of simulations carried out for the

Case	Grid	Filter Variation
C	$48^3 + 48^3$	N/A
F	$96^3 + 96^3$	N/A
FC-S	$96^3 + 48^3$	Sharp
CF-S	$48^3 + 96^3$	Sharp
FC-V	$96^3 + 48^3$	Linear variation over $1.6L_r$
CF-V	$48^3 + 96^3$	Linear variation over $1.6L_r$

Table 4.2: Summary of Smagorinsky inflow-outflow computations.

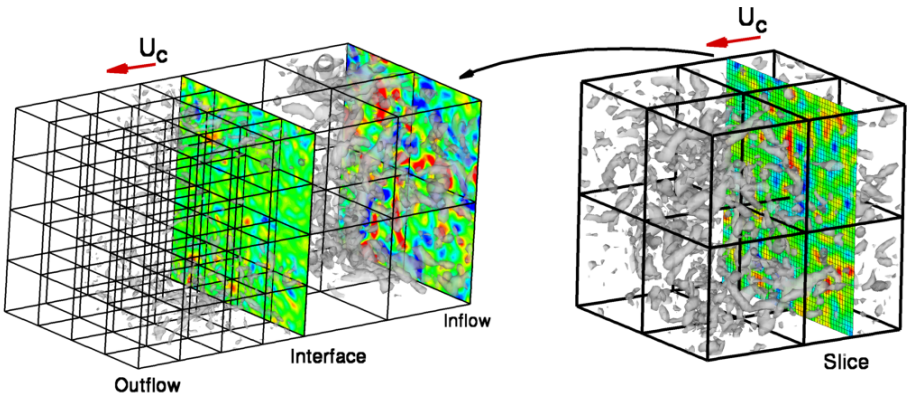


Figure 4.1: Setup of the numerical experiments.

LDEV and Smagorinsky models respectively.

To generate the inflow conditions, computations of forced isotropic turbulence at the same resolution as the block adjacent to the inflow plane were first conducted. The linear turbulence forcing scheme proposed by Lundgren [61] and further studied by Rosales and Meneveau [85] was applied, using a value of mean dissipation per unit mass, $\varepsilon = 0.4U_r^3/L_r$, and a Reynolds number $Re_r = 10,000$. Planes from this simulation were stored on disk and used as inflow conditions in the inflow-outflow

Case	Grid	Filter Variation
C	$24^3 + 2 \times 24^3$	N/A
F	$48^3 + 48^3$	N/A
FC-S	$48^3 + 2 \times 24^3$	Sharp
CF-S	$24^3 + 2 \times 48^3$	Sharp
FC-V	$48^3 + 2 \times 24^3$	Linear variation over $1.6L_r$
CF-V	$24^3 + 2 \times 48^3$	Linear variation over $1.6L_r$

Table 4.3: Summary of Lagrangian Dynamic inflow-outflow computations on lower-resolution grids.

computations (see Figure 4.1). In order to maintain consistency, the same SGS model was applied for the periodic and inflow-outflow simulations. Note, however, that the results of the forced simulations were found to be quite sensitive to the model and grid resolution; thus, at the inflow we had slightly different values of turbulent kinetic energy, depending on the model and resolution used. A specified convective velocity, $U_c = 0.41U_r$, is used to advect turbulence in x direction. This value was chosen based on two considerations: first, we wanted a ratio U_c/u_{rms} representative of the conditions encountered in a wall-bounded flow, especially near the wall where the turbulent eddies are smaller; this would result in $U_c/u_{rms} \simeq 4$. On the other hand, we wanted to use as short a domain as possible, to reduce the computational cost of the simulation. As a result, we chose $U_c/u_{rms} \simeq 2.5$. A convective condition [74] is used at the outflow and periodic conditions in all other directions.

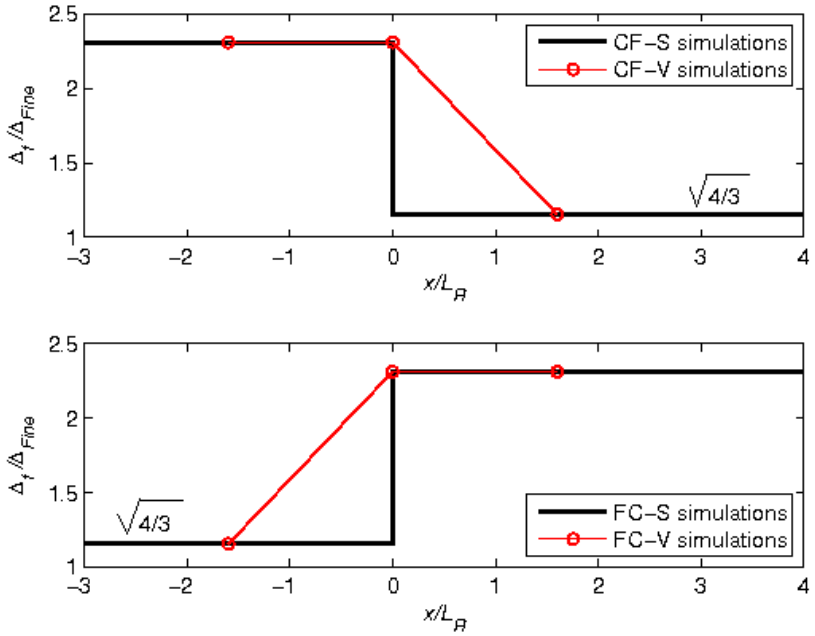


Figure 4.2: Filter width variation.

For each model we performed single-grid calculations on coarse and fine grids, as well as cases in which the grid and filter are changed abruptly by a factor of two in each direction (FC-S and CF-S), and cases in which the filter width was linearly increased or decreased (on the fine-grid side) from the value corresponding to the fine mesh to that corresponding to the coarse one (CF-V and FC-V cases). The variation took place over a distance $1.6L_r$ (comparable to the integral scale of the flow); variations over shorter distances did not prove to be effective. Figure 4.2 illustrates the variation of Δ_f (normalized by the grid size on the fine-resolution side) for the various simulations.

Note that the use of a filter-width different from the grid size is ambiguous, unless the non-linear term is filtered explicitly. When the Smagorinsky model is used, for instance, it is not possible to distinguish a change in Δ_f from a change

in C. With dynamic models, moreover, Δ_f does not enter the equations directly: what matters are the ratio $\alpha = \Delta_t/\Delta_f$, and the weights used to define $\widehat{f}$. We take the view here that the filter-width Δ_f can be assigned *a priori*, independently from the grid size. The coefficient C is determined from the flow physics, either through integration of the spectrum for the Smagorinsky model (see the derivation by Lilly [56]) or dynamically. For the dynamic model, this results in the use of weighting coefficients in (4.7) such that $\alpha = \sqrt{6}$ independent of the value of Δ_f used.

For each calculation, data at several locations along the longitudinal direction was collected in the course of more than 120 computational time units. Then the statistics were computed using an ensemble of no less than 100 time snapshots, and quantities were also plane-averaged for each x location.

4.2.2 Single-grid calculations

First, we performed single-grid calculations with a coarse and a fine grid, using the Smagorinsky and LDEV models. The coarse grid uses 48 points in each direction for each block (for a total of $2 \times 48^3 = 2.2 \times 10^5$ points) while the fine one uses 96^3 points per block, for a total of 1.8 million points.

In these calculations the mesh is cubic, and the filter-width is uniform; commutation errors are, therefore, identically zero. As expected, slight differences between the two models and of course differences between the calculations with the coarse and fine grids were found. The explicit filtering of the advective term results in a larger value of the Taylor microscale λ, reflecting the fact that the filtering reduces

the small-scale energy transfer to small scales. At $x = 0$, where the coarse-fine grid interface is located in the calculations with grid refinement or coarsening, the Reynolds number is $Re_\lambda \simeq 800$. Also, the integral length scale L_{11} (obtained by integrating the longitudinal two-point correlation) is approximately $0.85L_R$, remaining approximately constant along the streamwise direction and between the different eddy-viscosity models.

4.2.3 Two-level computations with the LDEV model

First, we examine the results obtained with the LDEV model. Notice that this model includes memory effects through the integration of the numerator and denominator of (4.4) along Lagrangian particle-paths. Furthermore, the variable filter-width does not affect the eddy viscosity directly (ν_T depends only on α), but only through the explicit filtering performed at the test-filter scale.

Figure 4.3 shows isosurfaces of the second invariant of the filtered velocities gradient tensor:

$$Q = -\frac{1}{2}\left(\frac{\partial \overline{u}_i}{\partial x_j}\frac{\partial \overline{u}_j}{\partial x_i}\right) = -\frac{1}{2}\left(\overline{S}_{ij}\overline{S}_{ij} - \overline{\Omega}_{ij}\overline{\Omega}_{ij}\right) \tag{4.10}$$

colored by the value of the eddy viscosity for three cases. The flow behavior in the Coarse-to-Fine (CF) case, Figure 4.3*(a)*, is relatively benign: shortly after the interface (denoted by the dark plane at $x = 0$) we observe a rapid generation of small scales. When the grid is suddenly coarsened, Figure 4.3*(b)*, on the other hand, we see a sudden depletion of eddies following the interface, and an abrupt increase in their length scale. When the filter is smoothly coarsened on the fine-grid

side, Figure 4.3*(c)*, we notice an increase in the characteristic eddy length-scale and an increase of the eddy viscosity even upstream of the interface, and a smoother transition to the coarse-grid side.

Figure 4.3 shows that, as turbulence is advected through a grid discontinuity, the flow behavior is quite different depending on whether the grid is coarsened or refined. The changes in the turbulence structure are quite dramatic, and one should expect them being reflected in the flow statistics.

Figures 4.4 and 4.5 show the development of statistical results. It should be remarked here that at the interface between fine and coarse grids the interpolation is used to exchange information between fine an coarse grids. Furthermore, because of the staggering, the control points where the velocity components are defined on the coarse grid do not coincide with those on the fine grid. As a consequence, while first-order averaged statistics are continuous across the interface, quantities that are quadratic in the velocity present a discontinuity. Reducing the magnitude of this discontinuity is an important feature of a successful transition methodology. For the CF run the flow statistics switch between the coarse- and fine-grid values reasonably rapidly. In Figure 4.4, the longitudinal spectra at various x locations are shown. These are the transforms of the two-point correlation of the in-plane velocity components with displacement in the direction of the velocity itself. The spectra immediately after the coarse-fine grid interface decay too steeply, as the high wave-numbers initially contain no energy. They are filled rapidly and by $x/L_R = 0.7$, less than one integral scale downstream of the interface, the spectrum returns to the single-grid value.

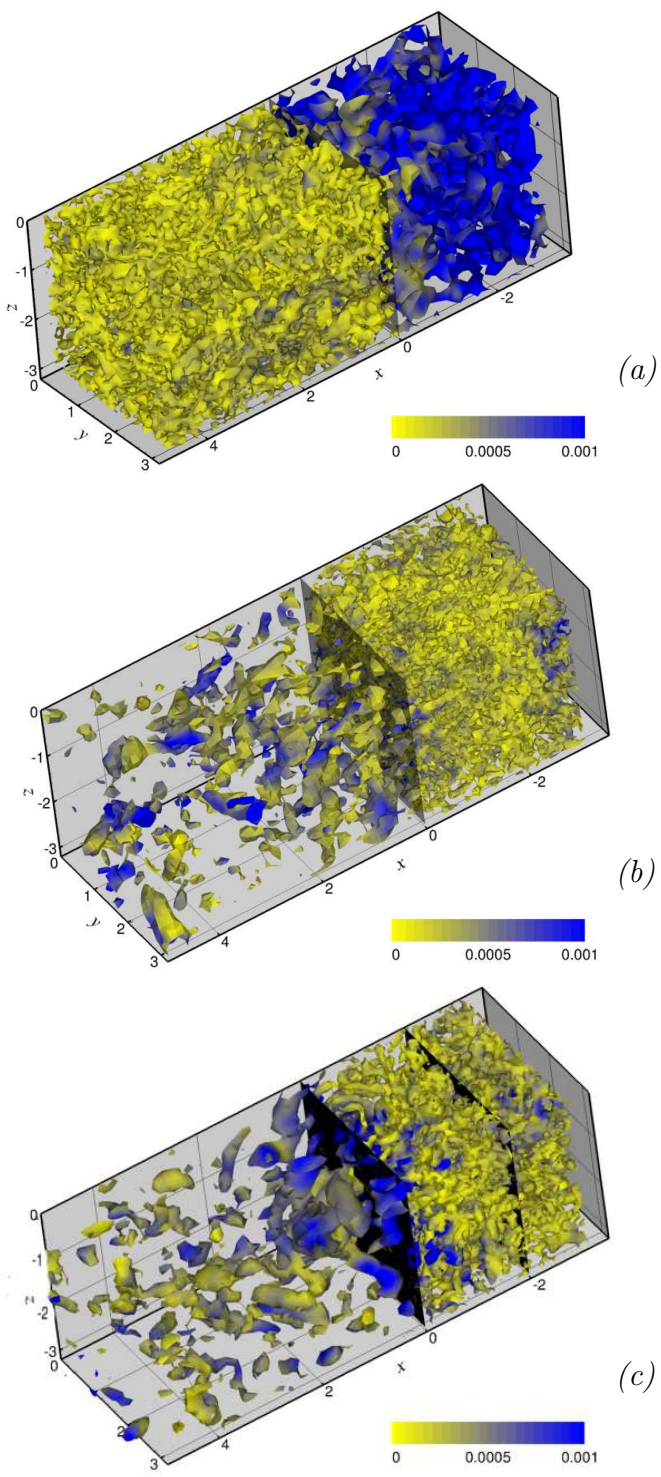


Figure 4.3: Isosurfaces of Q, colored by the value of the eddy viscosity. LDEV model. *(a)* CF-S; *(b)* FC-S; *(c)* FC-V. The flow is from top right to bottom left.

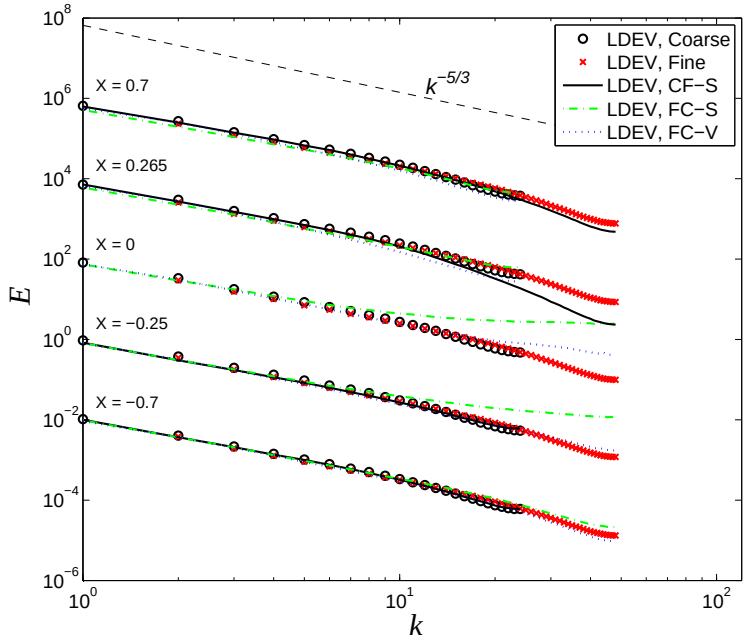


Figure 4.4: Longitudinal spectra, LDEV model. Each spectrum is shifted upward by 100 units for clarity.

The length of the transition downstream of the grid interface depends on the quantity examined, but is generally less than two integral scales L_{11} (recall that $L_{11}/L_r \simeq 0.85$). This distance, of course, is strongly dependent on U_c: As mentioned in before, the value used here, $U_c = 0.41U_r$, corresponds, at the interface, to a mean velocity that is about 2.5 times the rms intensity u_{rms}; in boundary layers U_c/u_{rms} is of the order of 4 near the wall, 20 or more in the outer layer; these values would result in longer transitions.

Coarsening the grid (FC simulations) introduces a much stronger perturbation. When the filter width is discontinuous (FC-S simulation) we observe an accumulation of energy at the small scales, apparent in the spectra at $x = -0.25$ and 0

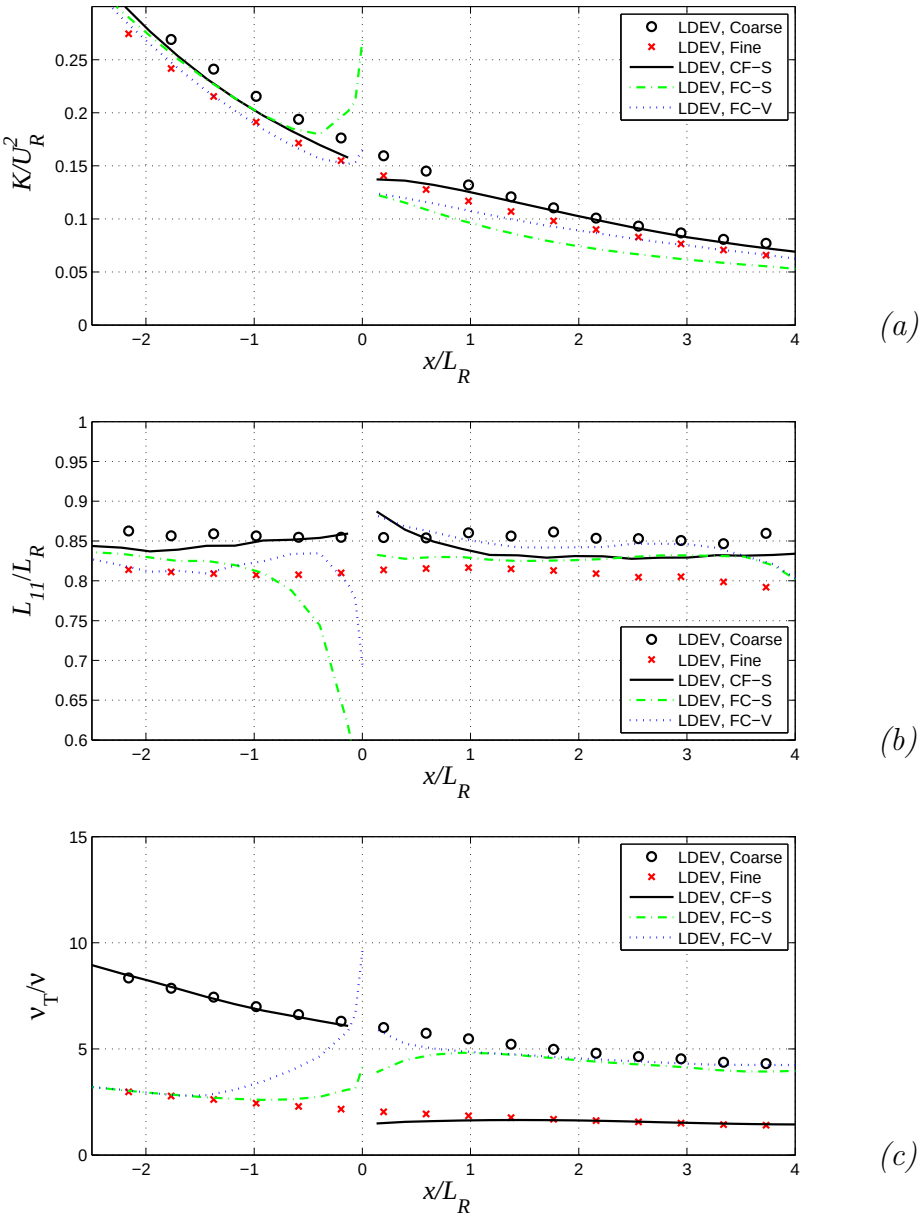


Figure 4.5: Streamwise distribution of *(a)* turbulent kinetic energy K; *(b)* integral length scale L_{11}; and *(c)* normalized eddy viscosity, ν_T/ν. LDEV model.

(see Figure 4.4). The energy pileup results in an overshoot of the turbulent kinetic energy $K = \langle u_i' u_i' \rangle / 2$ immediately before the interface Figure 4.5*(a)*. We also observe discontinuities in the integral scale and the eddy viscosity (normalized by the molecular one).

The smoothly increasing filter-width (FC-V simulation) is superior to the discontinuous one: since the LDEV model is not very dissipative (compared, for instance, to the Smagorinsky model discussed below), the increase of the eddy viscosity upstream of the interface is needed to reduce (but not completely remove) the energy pileup at the grid interface resulting in a much smoother transition. Even in the FC-S case, however, despite the fact that the energy pileup at small scale is very significant, the flow adjusts very quickly to the grid coarsening, and returns to the single-grid, uniform filter-width, data within less than an integral scale of the interface.

The calculations discussed until now were characterized by good resolution of the integral scales, even in the coarse-grid size. In fact, the ratio of integral scale $L = K^{3/2}/\epsilon$ (where is the turbulent kinetic energy and ϵ the dissipation) to the grid size is approximately 15 on the 48^3 grid, 30 for the 96^3 one. In typical wall-bounded flow calculations, much lower resolution is generally used: in the buffer layer and the beginning of the logarithmic layer, typically 5-10 grid points per L are used, and only in the outer layer we reach resolutions of 10-20 grid points per integral scale. To represent more closely the resolution levels used in actual calculations of wall-bounded flows, we performed another set of simulations that used coarser meshes: 24 grid points were used in all directions on the coarse side to resolve a

$2\pi^3$ cube, and 48 grid points on the fine side (see Table 4.3). Except for the fine-level calculation, the domain after the interface in these cases was extended to 4π in the x direction to reduce the effect of the convective outflow boundary on the results. In the 24^3 linearly forced simulations used to obtain inflow data, the value of dissipation was set to $\varepsilon = 0.36U_r^3/L_r$. As before, in all two-level calculations the interface was located at $x = 0$, and the convective velocity $U_c = 0.41U_r$. The Reynolds number, based on λ_f, at the interface location was $Re_{\lambda_f} \simeq 1500$ for the coarse inflow-outflow run.

In Figure 4.6 the author shows the average non dimensional values of turbulent kinetic energy, longitudinal integral length scale and normalized turbulent viscosity for the lower-resolution simulations. In a similar trend as seen in the past section, the Coarse-to-Fine grid and filter-width discontinuity does not affect the statistics as much as the Fine-to-Coarse one. For the FC cases, again, the energy pileup upstream of the interface is corrected by smoothly increasing the filter width more gradually. The sharp FC case exhibits a larger energy pileup zone compared to the higher-resolution simulation of Figure 4.5*(a)*. As expected, the integral length scale is $L_{11}/L_r = 1$ a value higher than that predicted on the finer meshes, $L_{11}/L_r = 0.85$. For the CF case the varying filter in the fine-mesh region delays the return to the fine grid value of all statistics: the increased dissipation associated with the larger filter-width on the fine grid delays the generation of small scale. The sharp discontinuity CF-S results in a recovery of the fine-grid results over $1.6L_r$, a larger distance than observed in the better resolution cases discussed above. Similar observation may be made for the FC cases: the smooth filter-width change gives improved results,

and the length required for recovery from the grid discontinuity is larger than in the previous case. The varying filter CF computation exhibits a bump in the ν_T curve, possibly due to the increase of $|\overline{S}|$ as smaller scales are generated, while the filter width is still interpolated from the coarse side.

4.2.4 Two-level simulations with the Smagorinsky model

Next we discuss simulations in which the SGS stresses were parameterized using the Smagorinsky model. The Lagrangian model includes memory effects; thus, the eddy viscosity immediately after an interface is, to some extent, affected by its previous history and recalls the fact that it is coming from a region of coarser (or finer) filter-width; its changes are, therefore, less dramatic than would be expected when the filter-width is changed by a factor of two. No such memory effects are present in the Smagorinsky model, were the eddy viscosity ν_T is more tightly connected to the filter width Δ_f.

This is clearly observed in Figure 4.7, which shows the streamwise development of turbulent kinetic energy, L_{11}, and the normalized eddy viscosity. We observe sharp discontinuities in the CF-S and FC-S cases for ν_T, which are directly related to the grid size. The variable filter-width is in this case not as beneficial, somewhat smoothing the discontinuities but also resulting in delayed recovery (especially for the CF-V computation).

The difference with the single grid case of the longitudinal spectra (Figure 4.8) is more significant than for the LDEV case as well. We observe a significant delay

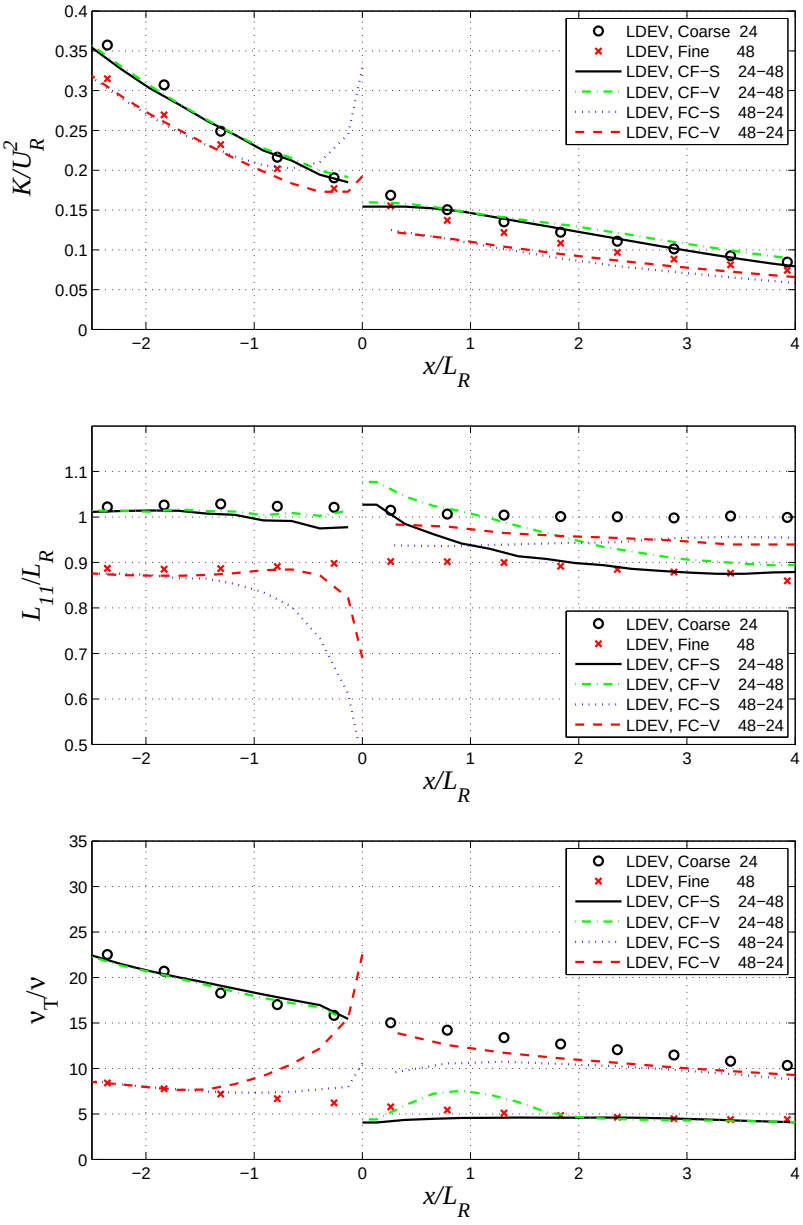


Figure 4.6: Streamwise distribution of *(a)* turbulent kinetic energy K; *(b)* integral length scale L_{11}; and *(c)* normalized eddy viscosity, ν_T/ν. LDEV model with lower resolution, 48^3 points per block on the fine side and 24^3 on the coarse side.

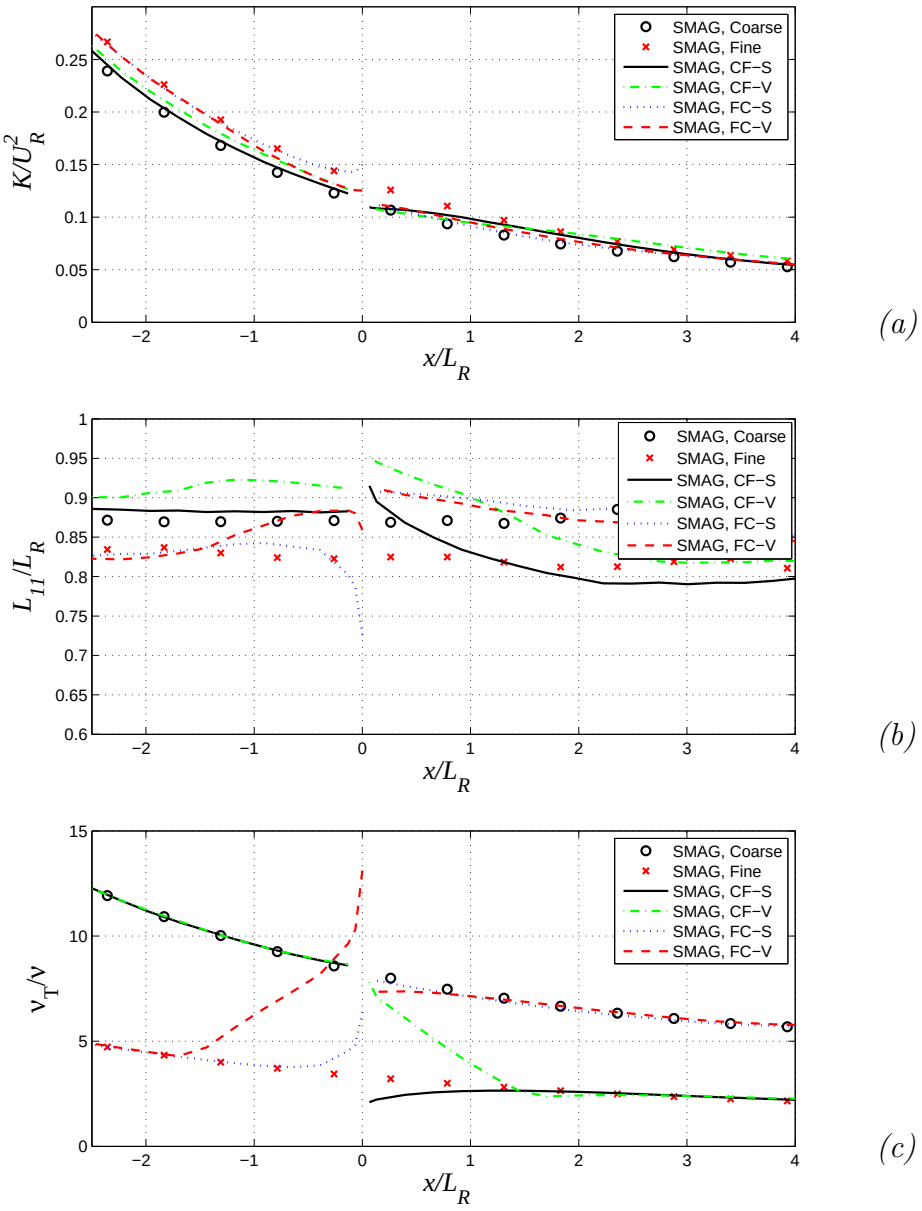


Figure 4.7: Streamwise distribution of *(a)* turbulent kinetic energy K; *(b)* integral length scale L_{11}; and *(c)* normalized eddy viscosity, ν_T/ν. Smagorinsky model.

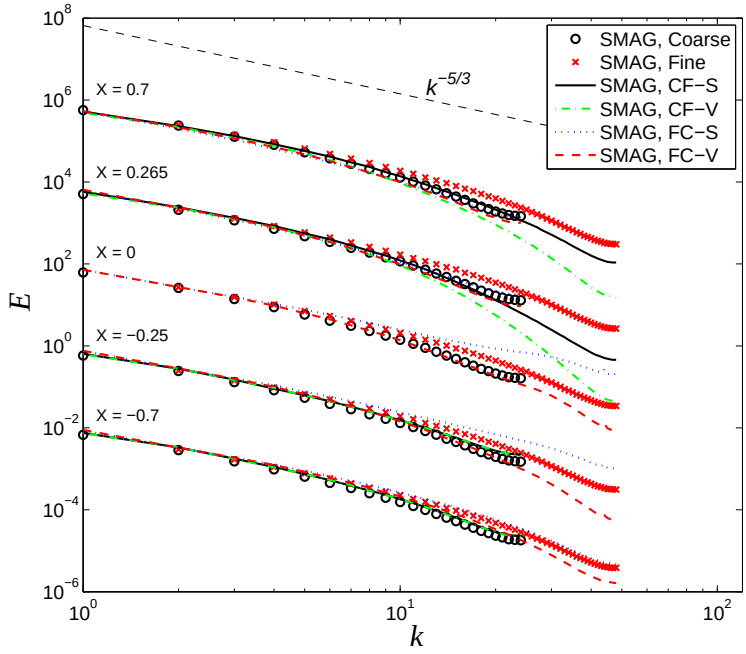


Figure 4.8: Longitudinal spectra. Smagorinsky model. Each spectrum is shifted upward by 100 units for clarity.

in the regeneration of the small-scale energy content for the CF-V case, the energy pileup at small scales for the FC-S simulation and the improved behavior of the flow immediately after the discontinuity in the FC-V calculation.

4.2.5 Two-level calculations with explicit filtering of the advective term

The possible benefit of filtering explicitly the advective term in the momentum equations derives from the fact that, by removing a band of the highest frequencies allowed by the mesh, the truncation and aliasing errors can be reduced [59]. For these reasons we applied explicit filtering together with the LDEV model on the

same cases discussed before. That includes two-level calculations where the grid transitions from 96^2 points to 48^2 points in the Fine-to-Coarse case (FC) or vice-versa in the Coarse-to-Fine (CF) situation. In order to obtain similar levels of inflow turbulent kinetic energy as in Subsection 4.2.3, the dissipation constant for the linearly forced isotropic turbulence runs was set to $\varepsilon = 0.2U_r^3/L_r$. The resulting Reynolds number at the interface location was $Re_\lambda \simeq 800$. The same variable-filter strategy on the fine side discussed before was employed for the filtering of the advective term.

The flow visualizations in Figure 4.9 show the expected trends of eddy depletion on the transition to coarser mesh for the sharp FC, and scales generation as the structures are advected to a better resolution environment, in the CF cases. However, we also notice a more gradual generation of small scales for the CF-S simulation, Figure 4.9*(a)*. More importantly, for the FC-V simulation we observe a more pronounced increase of the eddy size on the fine-grid side, compared with the case with no explicit filtering, Figure 4.3*(c)*. The scale of the eddies crossing the interface is, thus, increased; the better resolution of these eddies results in better accuracy as the flow transitions from the fine to the coarse grid.

Consequently, the spectra evolve much more gradually across the interface (Figure 4.10). In the CF-S case the high wave-number part of the spectrum is filled within a distance of $0.7L_r$ (less than one integral scale of the flow). When the filter-width is decreased gradually (CF-V simulation) a longer transition results, as observed in all previous cases. In the Fine-to-Coarse calculations we observe no energy pileup at small scale, and a very rapid transition for the discontinuous

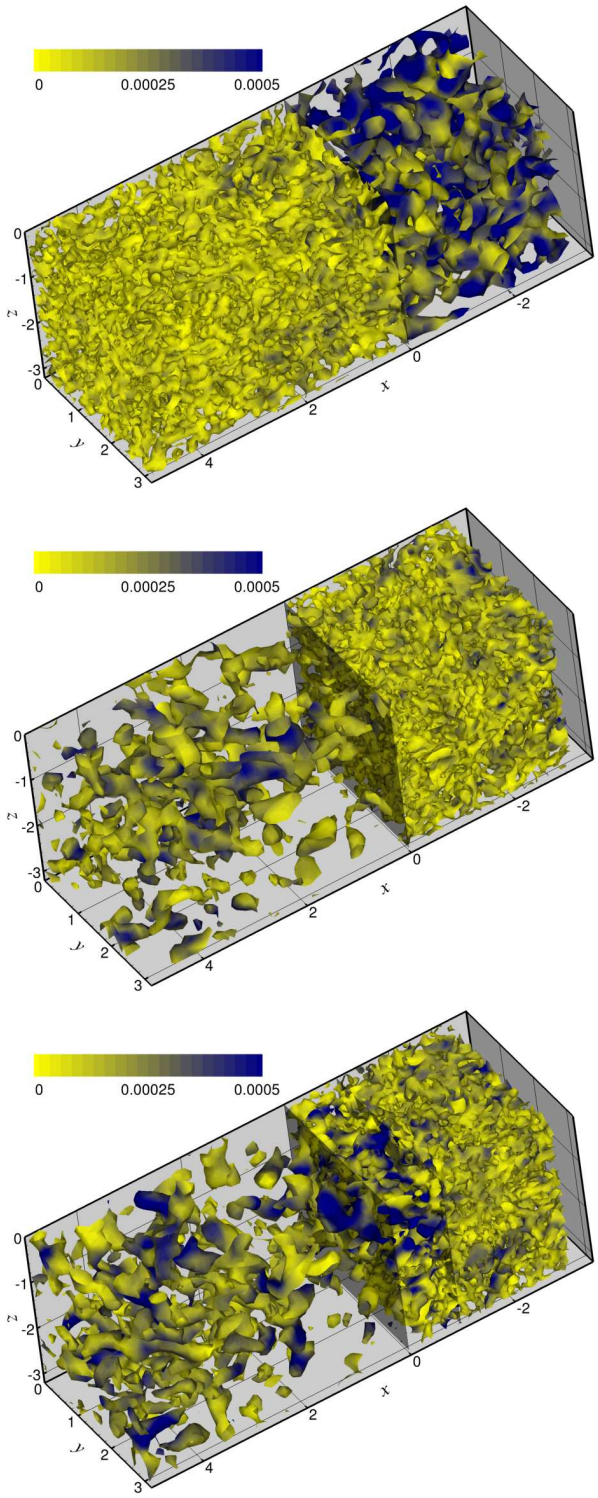


Figure 4.9: Isosurfaces of Q, colored by the value of the eddy viscosity. LDEV model with explicit filtering. *(a)* CF-S; *(b)* FC-S; *(c)* FC-V. The flow is from top right to bottom left.

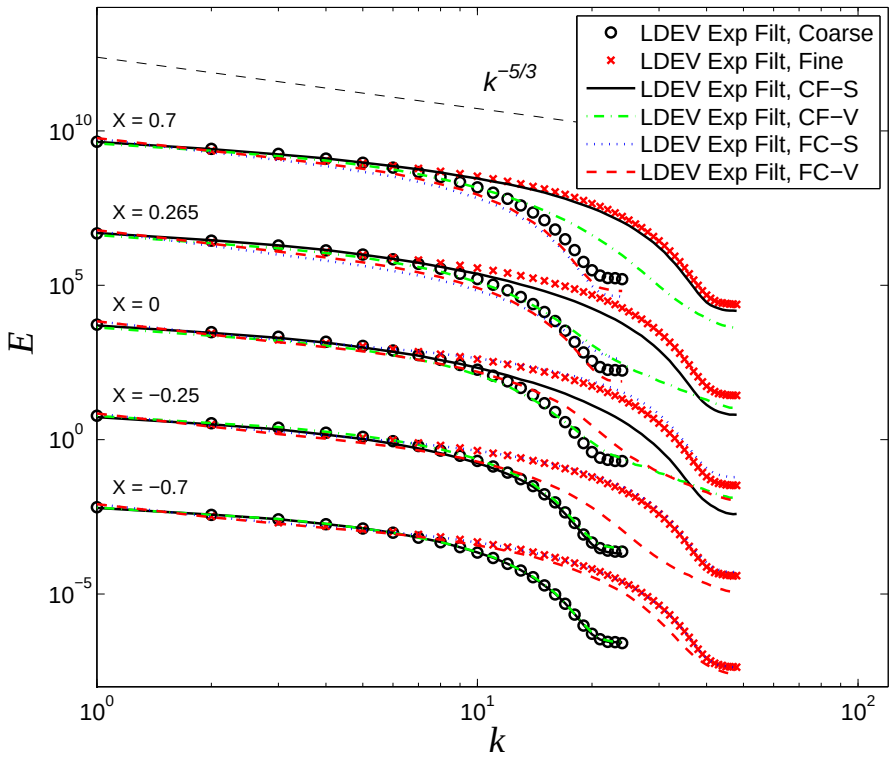


Figure 4.10: Longitudinal spectra for LDEV model with explicit filtering. Each spectrum is shifted upward by 1000 units for clarity.

filter-width variation, a more gradual one for the variable filter-width case, FC-V. Note that the explicit filtering dampens the high-wavenumber content of the non-linear term, which results in more rapid energy decay in all calculations of this type (including the single-grid ones).

In Figure 4.11 the mean turbulent kinetic energy, integral length, and eddy viscosity are shown. The sharp FC transition, as in previous results, exhibits a jump as the grid transitions from fine to coarse; interestingly, however, there is insignificant energy accumulation on the fine side (Figure 4.10). This result can be explained by noting that explicit filtering produces a depletion of energy on the structures at the

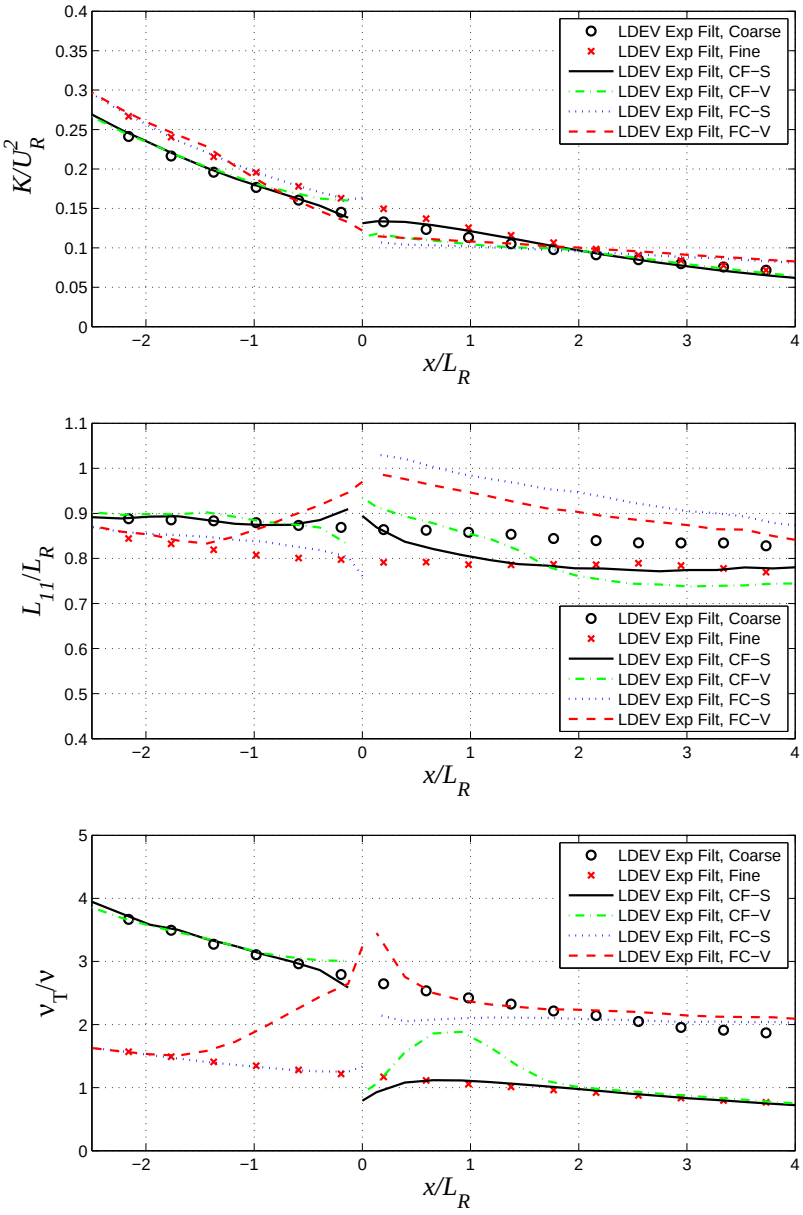


Figure 4.11: Streamwise distribution of *(a)* turbulent kinetic energy K; *(b)* integral length scale L_{11}; *(c)* normalized eddy viscosity, ν_T/ν. LDEV model with Explicit Filtering of advective term.

grid scale, where numerical error is significant (also evident in Figure 4.10 at the first two stations). Since the smallest scales are better resolved by both grid levels, the eddies advected across the interface cause less of a perturbation to the flow. The longitudinal spectra in Figure 4.10 show that, for the varying filter FC-V case, a gradual reduction of energy content on the higher wave-numbers of the fine grid is achieved. This enables a natural transition to the natural energy distribution for the coarse grid, which has a smaller range of resolved wave-numbers. The FC-S case (consistent with the previous sections results) displays a discontinuous transition on the spectra as the resolved wave-number range is reduced. The coarse-to-fine transition behaves in fashion similar to that seen for the previous LDEV calculations, the sharp variation being superior to the variable filter-width case, yielding a faster recovery of fine grid statistical data. The variable filter FC-V calculation, on the other hand exhibits a faster recovery of the coarse grid statistics than the sharp case, for both integral length-scale and turbulent viscosity, Figures 4.11*(b)* and *(c)*.

4.3 Flow around a Sphere at Re=10000

In this section we apply the spatially varying LDEV model with explicit filtering of the advective term to LES of the flow around a sphere at $Re = U_\infty D/\nu = 10^4$ (U_∞ is the freestream velocity and D is the diameter of the sphere) is considered. This Reynolds number falls in the subcritical regime, where laminar boundary layer separation occurs on the surface of the sphere, and transition happens in the detached shear layers in the wake. We have seen in the previous Section that the

varying filter LDEV model with explicit filtering of convective terms is suitable for situations where turbulence is advected across derefinement jumps. That is the case in this problem, as we move away from the sphere along the wake region. A rectangular domain is employed, spanning $38D$ in the streamwise direction and $24D$ in the cross stream directions. The sphere is centered at a distance of $10D$ downstream of the inflow boundary. The inflow velocity, U_∞, is prescribed at this boundary, and a convective condition used at the outflow boundary [74]. Slip wall boundary conditions are used in the other two directions.

A preliminary computation for laminar flow around a sphere at $Re = 300$ was done on a similar setting, using 5 levels of refinement and about 6.5×10^6 points. The resulting grid spacing close to the sphere was $0.0195D$. The predicted mean drag coefficient is $C_D = 0.634$, and its oscillation amplitude $C'_D = 0.0039$. These values are in good agreement with the corresponding results from computations by Johnson & Patel [47] of $C_D = 0.656$ and $C'_D = 0.0035$ respectively. The resulting Strouhal number is $St = 0.132$, which is also in good agreement with the value 0.137 reported in [47].

For the case at $Re = 10^4$ a finer grid with 6 levels of refinement is used with $11 \times 7 \times 7$ blocks at level 0. Each block consists of 16^3 cells. The grid is finest near the surface of the sphere and it is gradually derefined in the wake. The leaf-block arrangement at an $x - z$ plane passing through the center of the sphere is shown in Fig. 4.12. The cell sizes near the surface of the sphere are approximately, $\Delta x_i = 0.003D$, and the resulting distance of the first grid point off the wall is between $r^+ = 1.3 - 2.3$ (following [24], we define $r^+ = ru_\tau/\nu$, where the friction

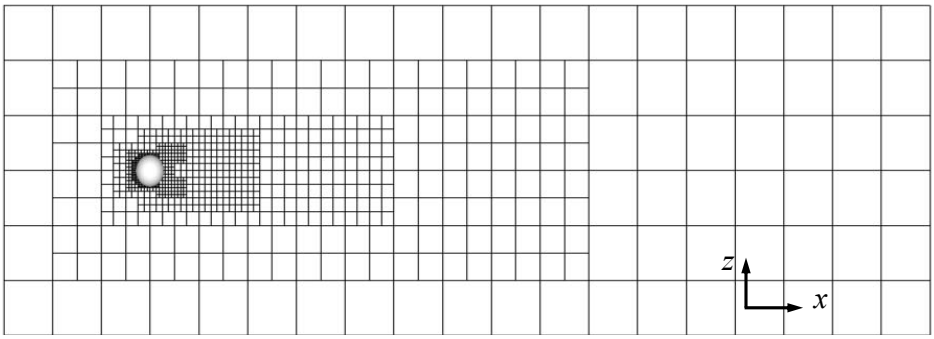


Figure 4.12: AMR grid-blocks for flow around a sphere at $Re = 10^4$. An $x - z$ plane at $y = 0$ is shown. Note that a window spanning from $-5 < x < 28$ in the x-direction and from $-5 < z < 5$ in the z-direction is shown. Six levels of refinement are used with 16^3 points per block.

velocity is assumed to be $u_\tau = 0.04$). The number of points between the wall and $r^+ \sim 10$ is between $5 - 8$. The total number of blocks is 18400, and the number of leaf blocks 16184 corresponding to 66.3 million points.

The equations were integrated in time until the effect of initial conditions was eliminated. Statistics were accumulated over two shedding cycles, which is about $10U_\infty/D$ eddy turnover times or 20000 timesteps. Although the sample is fairly small it is sufficient to provide reasonable estimates of the average force coefficients. LES of the flow around a sphere at the same Reynolds number performed by Constantinescu *et al.* [24], gave a mean drag coefficient $\overline{C}_D = 0.393$. The same value was reported in the LES by Yun *et al.* [117]. Our computation gave $\overline{C}_D = 0.405$, which is in very good agreement with the adobe results. In Fig. 4.13 the distribution of average pressure coefficient, $\overline{C}_p$, and skin friction coefficient, $\overline{C}_f$, are shown as a function of the azimuthal angle, Θ. The corresponding results from the LES by

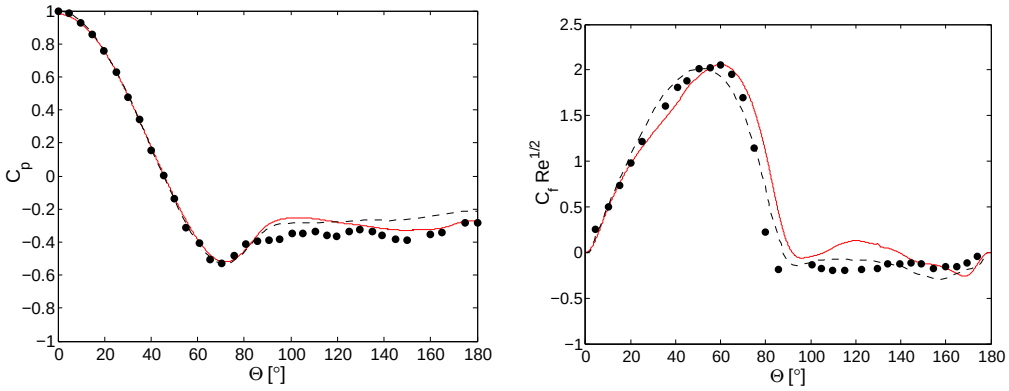


Figure 4.13: Variation of C_p and C_f on the surface of the sphere as a function of angle Θ from front stagnation point. Both are averaged in the azimuthal direction and time. —— current AMR-LES calculation at $Re = 10^4$, --- reference LES at $Re = 10^4$ [24], and • experiment at $Re = 1.6 \times 10^5$ [1].

Constantinescu *et al.*[24] at the same Reynolds number, and the experiments by Achenbach [1] at a higher subcritical Reynolds number, $Re = 1.6 \times 10^5$, are also shown for comparison. In general the agreement is good, and $\overline{C}_p$ is practically the same in all three data sets for $0^o < \Theta < 90^o$. Both simulations also agree very well for $90^o < \Theta < 130^o$, and slightly over-predict $\overline{C}_p$ reported in the experiment. Further behind the sphere at, $130^o < \Theta < 180^o$, our $\overline{C}_p$ is lower by is approximately 6% of the peak value, probably due to our limited statistical sample.

The friction coefficient, $\overline{C}_f$, is also in good agreement with the reference data. In this case discrepancies among the different datasets are larger, due to larger errors in measuring this quantity as well as its sensitivity to numerical resolution. Our mean separation occurs at $\Theta = 91^o$ which is in very good agreement with the value of $\Theta = 90^o$ reported in [117]. Constantinescu *et al.* [24] is reported mean

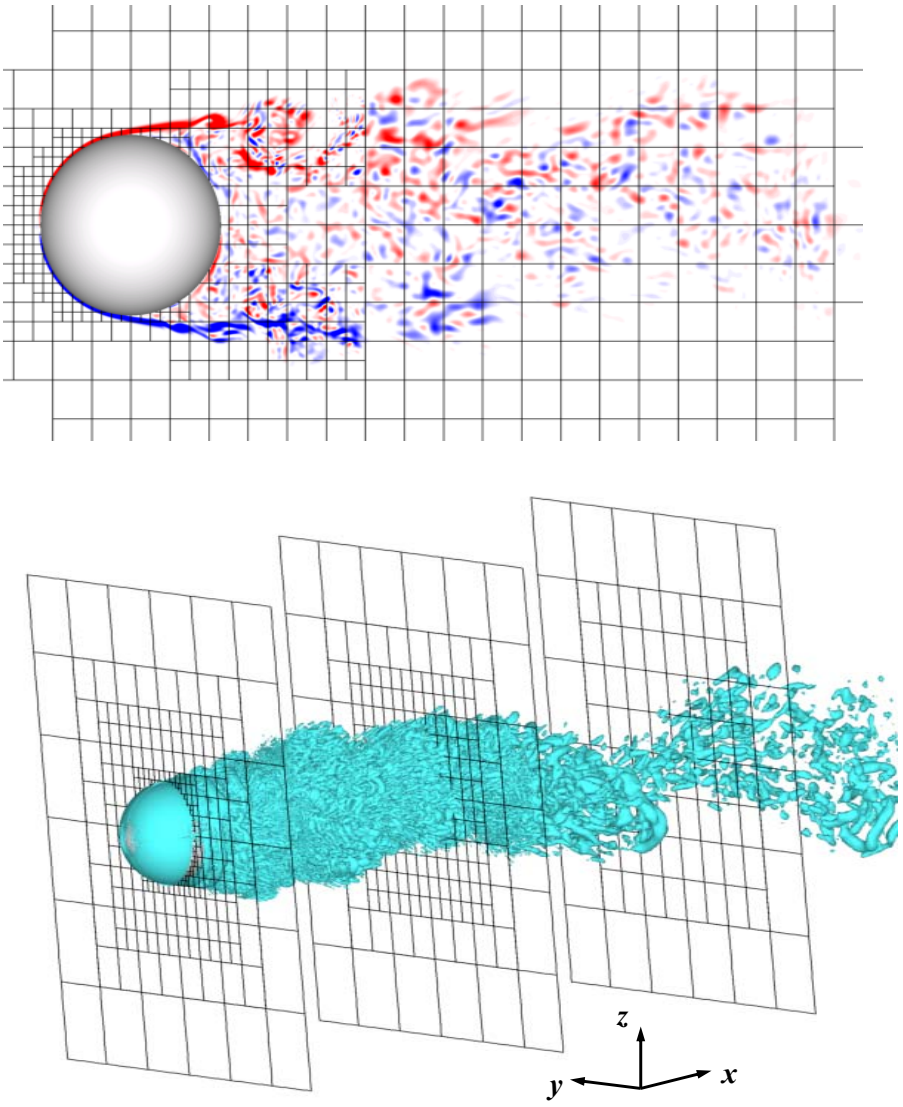


Figure 4.14: A snapshot of the instantaneous flow field around the sphere at $Re = 10^4$. (Top): Vorticity, $\overline{\omega}_y$, contours at an $x - z$ plane through the center of the sphere. Forty contours between $\overline{\omega}_y D/U_\infty = -20$ (*blue*) and 20 (*red*) are shown. (Bottom): Q isosurfaces in the wake. The grid distribution is also shown for 3 slices at positions $x = 0, 3D, 7D$.

separation at $\Theta = 85^o$. They used spherical coordinates and a grid which is much finer near the surface of the sphere, while our resolution is comparable to the one utilized by Yun *et al.* [117].

In Fig. 4.14 an instantaneous snapshot of the spanwise vorticity, $\overline{\omega}_y$, is shown at an $x - z$ plane passing through the center of the sphere. The shear layer instability, roll-up and subsequent breakdown can be observed. The smooth variation of the vorticity between blocks at different refinement levels should also be noted, indicating that turbulent eddies are unaffected by the presence of block boundaries. A three-dimensional view of such eddies is also shown in Fig. 4.14, and is visualized using the second invariant of the velocity gradient tensor, Q. The AMR block structure at three different planes in the wake region ($x = 0$, $x = 3D$ and $x = 7D$) is also shown. As the refinement level is decreased for increasing x, the reduction in the range of scales resolved by the grid is evident.

Chapter 5

Applications to flapping flight

Historically, flapping flight has received great attention from the biology, engineering and science disciplines. For biologists, the interest relies on the mechanisms by which biological motor, sensor and control has evolved and is employed by flying insects in everyday survival [87]. Engineering research on flying has been focused fixed wing aerodynamics, where their detailed understanding and control has driven research over the past century, mainly as a quest to serve the developing aeronautical and aerospace industries. However, advances in material technologies and actuator miniaturization in the past decade has sparked new interest in small scale flapping wing devices [91]. As the size of the vehicle and characteristic Reynolds number (Re) of the flow decrease, friction forces are enhanced, and fixed wing aerodynamics fails to account for the forces required to sustain flapping flight and maneuvering in an efficient manner [7].

Several experimental and computational investigations have been carried out over the past years, to understand the unsteady mechanisms that allow the generation of enough aerodynamic force to hover and maneuver. Experimental studies have given insights on the kinematic and kinetic aspects of flapping flight on thetered and free flying animals ([108], [38]), as well as dynamically scaled robotic devices ([35], [27], [88]). Viscous incompressible flow simulations of flapping wing models has also

been reported ([58], [93], [82], [105], [106], [107], [83]). These studies have been primarily used as an extension of experiments to obtain a more detailed view of the flow fields and the large vortices that dominate the production of aerodynamics forces. Liu and Kawachi [58], for example, conducted one of the first computations of flow around a flapping Manduca Sexta moth wing at $Re = 4000$. Due to the lack of numerical resolution only some of the large scale structures had some similarity to the experimental data in [109]. Ramamurti and Sandberg performed similar study for the smaller Drosophila melanogaster fly at $Re = 136$. Most other computations in the literature utilize two-dimensional (2D) models ([105], [106]). In addition, insect wings are complex flexible structures that add significant complexities to numerical modeling.

Today most of the computational studies on flapping flight focus on the fluid dynamics of rigid flapping wings with prescribed kinematics, or examine the wing as a structure, modeling the fluid loads with added-mass and damping parameters. The coupling between the fluid and the structure is either neglected or taken into account by extremely coarse models. In general, a flexible wing structure deforms under the influence of aerodynamic forces, elastic deformation forces, and also inertia forces due to the accelerations of its own mass ([33], [36], [53], [23]). The deformation is given, to a large extent, by the internal distribution of compliant components and mechanisms ([110]). The relative contributions of aerodynamic and inertial-elastic forces on wing deformation of the *Manduca sexta* hawkmoth, for example, were assessed by Combes and Daniel (2003) [23]. They concluded that, for *Manduca sexta* the equilibrium of forces during the wing's motion is mainly due to inertia and

elastic forces, and the aerodynamic loads had a minor role in the form of damping. For this insect the typical ratio of wing inertial to aerodynamic forces in hovering flight is around 7. In other species this parameter is closer to one [36], and more complex aeroelastic interactions occur. Strongly coupled interactions between fluid and wing dynamics require, accurate and robust accurate FSI solvers [114] such as the one outlined in Chapter 3.

In this chapter we will utilize the AMR/FSI solver to study two important aspects of flapping flight: i) the effects of flexibility on the dynamics of hovering flexible profile, and in particular the nonlinearities of the structure and viscous flow play a part in the evolution of the system. ii) the dynamics of free longitudinal flight in a realistic three-dimensional configuration. Details are given in the following sections.

5.1 Influence of Flexibility on the performance of two-dimensional flapping flexible airfoils

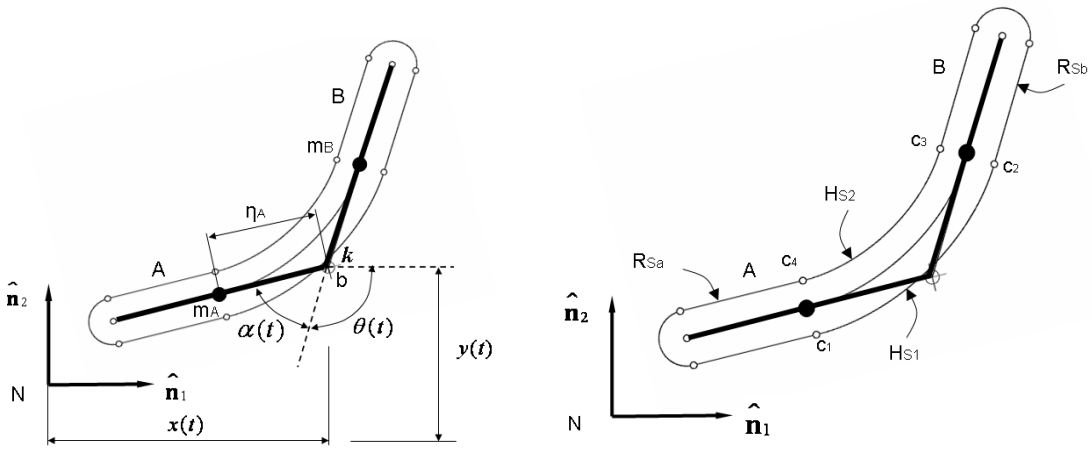
In this section we will study the the effects of structural flexibility on the aerodynamic performance of a flapping wing for a range of Reynolds numbers. We will perform two-dimensional computations in order to explore a wide range of parameters in a cost-efficient manner. In particular, a representative section of a three-dimensional wing (two-dimensional foil) is considered, and spanwise bending and torsion flexibility are neglected. Such a structure can be represented by two plates connected with a torsion spring to account for deformation in the chordwise

direction (see Figure 5.1a). This system has four degrees of freedom, which are effectively reduced to one by prescribing harmonic hovering motions of one of the plates. The plates are rigid and the large angular deformations give rise to cubic and higher order odd non-linearities in the governing equations, similar to those seen in a pendulum and in flexible beams [5]. In a sense, one could consider the two plates as a double pendulum with a torsion spring.

5.1.1 The structural model

A schematic of the structural system is shown in Figure 5.1a. It consists of two rigid links A and B, which are joined at the center, b, by a pin that contains a linear torsion spring. In this model, flexibility is concentrated at one discrete location of the system, and inclusion of elastic links will allow chordwise variations of stiffness and mass to be accounted for. The two links are covered by a set of aerodynamic surfaces that define the boundary between the airfoil and the fluid, and deform as the angle between them changes. The aerodynamic surfaces consist of two rigid segments, R_{Sa} and R_{Sb} (see Figure 5.1b), and two segments that dynamically deform according to the angle between the two links. The deformation is prescribed by fitting the Hermite interpolation polynomials $c1$-$c2$ and $c3$-$c4$. We have found that this modeling is robust and helps maintain the smoothness of the surface even for large values of the angle between the plates.

The dimensionless form of the equations governing the motion of the structural



(a) (b)

Figure 5.1: (a) Two-link profile. The rigid links A and B are connected at hinge b by a torsional spring of stiffness k. The variables $x(t)$, $y(t)$, $\theta(t)$ and $\alpha(t)$ are the generalized coordinates used to describe the wings motion. In the hovering simulations, $x(t)$, $y(t)$ and $\theta(t)$ are prescribed and $\alpha(t)$ is left as only degree of freedom required to define the system. (b) Decomposition of the profiles surfaces into rigid and deformable sections. The two rigid surfaces R_{Sa} and R_{Sb} are connected at points c_1, c_2, c_3 and c_4 by Hermite interpolating polynomials H_{S1} and H_{S2}. Also, m_A, m_B are the total masses of links A and B, and η_A, η_B are the respective distances from their centers of mass to junction b.

system shown in Fig. 5.1a can be written as:

$$\begin{bmatrix} m_A+m_B & 0 & \begin{pmatrix} -m_A\eta_A\sin(\alpha+\theta) \\ +m_B\eta_B\sin(\theta) \end{pmatrix} & -m_A\eta_A\sin(\alpha+\theta) \\ & m_A+m_B & \begin{pmatrix} -m_A\eta_A\cos(\alpha+\theta) \\ +m_B\eta_B\cos(\theta) \end{pmatrix} & m_A\eta_A\cos(\alpha+\theta) \\ & & I_A+I_B & I_A \\ symm. & & & I_A \end{bmatrix} \begin{Bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{\theta} \\ \ddot{\alpha} \end{Bmatrix} = \begin{Bmatrix} Q_x+g_x \\ Q_y+g_y \\ Q_\theta+g_\theta \\ Q_\alpha+g_\alpha \end{Bmatrix} \tag{5.1}$$

where $x(t), y(t), \theta(t)$ are, respectively, the joint horizontal displacement, joint vertical displacement, and orientation angle of link B measured from the inertial reference frame N as shown in Figure 5.1a, and $\alpha(t)$ is the deflection angle between links A and B. Here, m_i is the total mass of plate i $(i = A, B)$; η_i is the distance from the pin joint to the center of mass of bar i; I_i is the moment of inertia of link i respect to hinge point b. Also, Q_x, Q_y are the fluid forces in the x and y directions respectively, and Q_θ, Q_α are the fluid moments acting on the generalized coordinates $\theta(t)$ and $\alpha(t)$. The quantities g_x, g_y, g_θ, g_α are the corresponding contributions of centrifugal, elastic and gravity forces. The equations are made dimensionless by using the chord length of the undeformed airfoil, L_c, as the reference length scale, and the maximum translational velocity of the junction b between the two links, U_c as the reference velocity. The Reynolds number is defined as $Re = L_c U_c/\nu$, where ν is the fluid kinematic viscosity. The fluid forces and moments are determined from equations (2.1).

In the present numerical experiments, the translational motions of the junction as well as the orientation of link B are prescribed. With these prescribed motions, the four degrees of freedom of the system can be effectively reduced to the deflection angle $\alpha(t)$ between plates A and B. Thus, the overall deformation of the wing section is determined by the deflection angle $\alpha(t)$, which is governed by the following reduced form of equations (5.1):

$$I_A\ddot{\alpha} + k\alpha = -I_A\ddot{\theta} + m_A\eta_A sin\left(\theta + \alpha\right)\ddot{x} + Q_\alpha \tag{5.2}$$

Equation (5.2) resembles the equation governing a harmonic oscillator with forcing

due to the prescribed kinematics and the fluid forces (e.g. reference [71]). The non-linearities arise from the $sin(\theta + \alpha)$ term due to the kinematics and the fluid forcing. For this study, we only take into account the fluid damping which arises through the fluid moment Q_α. It should be noted that selecting a proper structural damping model is far from trivial, and this is an active research topic in structural biomechanics. Damping models for insect wings are relatively few [e.g. classical viscous damping model used by Herbert [22] and the viscoelastic model used by Bao and colleagues [13]] and the existing models require a fair amount of empirical information.

5.1.2 Parametric space and computational setup

To prescribe the translational motions of the junction and the orientation of link B, we define the states $x(t)$, $y(t)$ and $\theta(t)$ as:

$$x(t) = \left(1 - e^{-\frac{t}{\tau}}\right) \frac{A_0}{2} cos(\omega_f t); \quad y(t) = 0;$$
$$\theta(t) = \theta_0 + \left(1 - e^{-\frac{t}{\tau}}\right) \gamma sin(\omega_f t + \phi), \tag{5.3}$$

where A_0 is the stroke length of the pin point, θ_0 is the mean orientation angle for link B, γ is the rotation amplitude, ω_f is the frequency of the prescribing or forcing oscillation and ϕ is the phase angle between $x(t)$ and $\theta(t)$. The exponential terms were used in order to reduce transient effects [23]. The time constant was chosen as $\tau = 1.6\pi/\omega_f$ because 99.8% of the prescribed amplitude was reached after a time length of 5 periods. The following parameters corresponding to symmetric hovering

were selected [107]:

$$A_0 = 2.8; \quad \theta_0 = -\frac{\pi}{2}; \quad \gamma = \frac{\pi}{4}; \quad \phi = 0 \tag{5.4}$$

Based on the adopted normalization, the problem is completely defined by the density ratio ρ_b/ρ_f, the frequency ratio ω_f/ω_n and the Reynolds number Re. Here, ρ_b is the density of the wings material and $\omega_n = \sqrt{(k/I_A)}$ is the linear natural frequency of the oscillator (equation (5.2)). The frequency ratio ω_f/ω_n is used to characterize the flexibility of the wing section.

Three Reynolds numbers were considered (Re = 75, 250 and 1000) to investigate the effect of the reduction in viscous dissipation on the system dynamics. The mass ratio was set to $\rho_A/\rho_f = 25$, as this value provided a ratio close to 2 for the maximum translational inertial force over maximum drag force at $Re = 75$ for the chosen geometry and kinematics. The above ratio was determined through numerical experiments with the rigid wing. To compute the maximum horizontal translational inertial force, the total wing mass was multiplied by the maximum acceleration determined from the second derivative of $x(t)$ in equation (5.3). The value of peak drag force, on the other hand, was obtained from the rigid wing simulation at $Re = 75$. The wing has a thickness of 10% of the undeformed chord length and circularly formed edges.

The frequency ratio in the case of $Re = 75$ was set to ω_f/ω_n = 1/2, 1/2.5, 1/3, 1/3.5, 1/4, 1/6, and for Re = 250, 1000 was set to 1/2, 1/3, 1/4, 1/6. The resulting range of maximum deflection angles varied from 10° up to 70°. Also, the rigid profile problem (no angular deformation between the links) was run for each of

the Reynolds numbers mentioned. We note that for frequency ratios below 1/2, the computations would fail since the two plates collide during rotation. This limitation arises from the fact that the flexibility in the present model is concentrated at the hinge point and the distributed chordwise variations of stiffness and mass are not accounted for. A flexible beam model and/or inclusion of structural damping may help to address this issue and enable computations with frequency ratios of about one.

The grids adopted for the $Re = 75$ simulations were tailored to maintain a high refinement zone in the areas where boundary and detached shear layers, as well as the wake vortical structures are present. The rigid wing was set to move in the center of a box with dimensions $30L_c \times 30L_c$, in order to minimize interference effects from the domain boundaries. For all this simulations we chose to use a single block stretched grid, where a fast direct Poisson solver could be used [95]. A near-uniform grid zone was generated near the center, where the motions of the two-link system took place, and this zone was stretched towards the boundaries. The stretching towards the boundaries of the domain was obtained using the hyperbolic tangent function in each direction.

For the $Re = 75$ simulations, the uniform grid zone had a local cell size of $\Delta x = \Delta y = 0.0038L_c$, and the total number of points was 1229×551 along the x and y directions, respectively. Through grid refinement studies, we found that the above resolution was sufficient to capture all flow features. In Figure 5.2, computationally obtained aerodynamic forces are shown for approximately half the resolution throughout the computational domain (total number of points was $664 \times$

400) and the same forcing conditions (i.e. $\tau = 0$ in equations (5.3)) and Reynolds number as that for the baseline rigid wing computation. The corresponding lift and drag coefficients determined in the computations of Wang and colleagues [107], where a hovering ellipse with the same kinematics is considered instead of a plate, are also included in the figure. The agreement between the results obtained with the two different grids is good, with a maximum difference of around 3%. Despite the differences in the wing-section shapes, after the initial transients ($t/T > 2$), good agreement is also seen with the results obtained by Wang and colleagues [107].

For the results of Figure 5.2, within the boundary layers on the link surface, we estimated the number of grid points to be approximately 8 and 16 for the coarse and fine grids, respectively. As the Reynolds number increases, the boundary layer thickness is expected to decrease as $\sqrt{1/Re}$, and in order to keep the resolution within the above range, a grid with 1320×1038 points was found to be sufficient for both $Re = 250$ and $Re = 1000$ cases. All the results presented in this article were obtained with a 1229×551 grid for the $Re = 75$ simulations and a 1320×1038 grid for the $Re = 250$ and $Re = 1000$ simulations. The governing equations were integrated for a time length of 14 periods, 21 periods and 15 periods, for $Re = 75$, 250 and 1000, respectively. The time-averaged quantities were computed over the last 7 periods, 13 periods and 10 periods, for $Re = 75$, 250 and 1000, respectively.

5.1.3 Results

Aerodynamic quantities

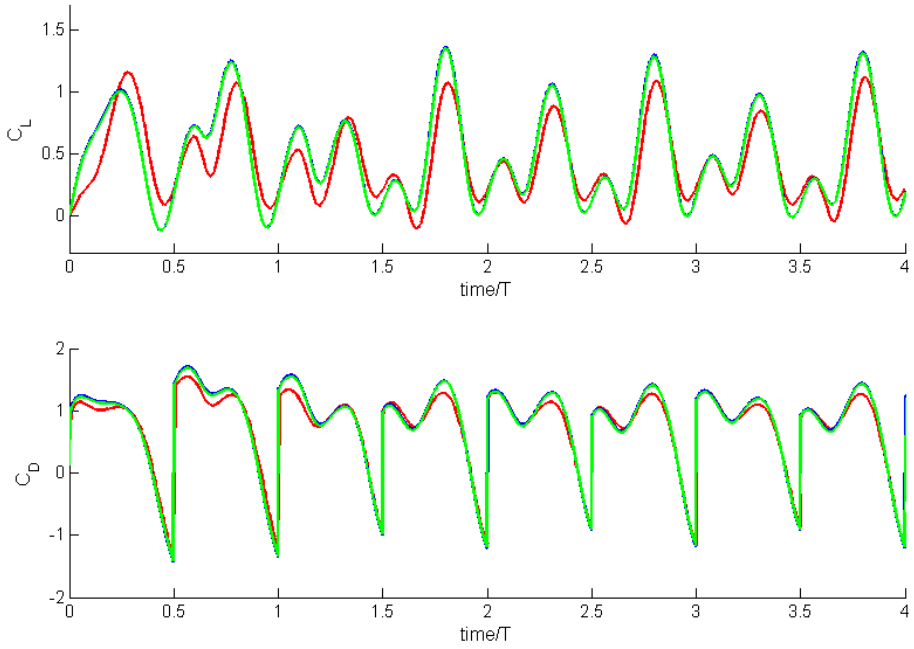


Figure 5.2: Time histories of lift and drag force coefficients (C_L, C_D) for a symmetric harmonic hovering rigid link at $Re = 75$ and two different grid resolutions. Blue line, rigid link, embedded boundary grid 1229×551; green line, embedded boundary grid 666×402; and red line, data from Wang and colleagues [107]. t, time; T is the prescribed motion time period.

Given the complexity of the flexible profile configuration as a function of deformation, a simple, yet consistent, normalization strategy is adopted to obtain force and power nondimensional coefficients. That is, the lift and drag coefficients are defined by:

$$\begin{aligned} C_L(t) &= \frac{Q_y^*(t)/L_c}{\frac{1}{2}\rho_f \cdot U_c^2 L_c} = 2Q_y(t); \\ C_D(t) &= -\frac{sgn(\dot{x}(t))Q_x^*(t)/L_c}{\frac{1}{2}\rho_f \cdot U_c^2 L_c} = -2sgn(\dot{x}(t))Q_x(t) \end{aligned} \tag{5.5}$$

where $Q_x^*(t)$ and $Q_y^*(t)$ are dimensional quantities and $Q_x(t)$ and $Q_y(t)$ are non-dimensional quantities. Once the equation for the deformation $\alpha(t)$ is solved at each integration step, the driving forces in the prescribed generalized coordinates $x(t)$, $y(t)$ and $\theta(t)$ are computed from 5.1.

The total power input, sum of horizontal translational power and rotational power, is computed from

$$\begin{aligned} P_{tr} &= R_x(t) \times \dot{x}(t) \\ P_{rot} &= R_\theta(t) \times \dot{\theta}(t), \end{aligned} \tag{5.6}$$

where P_{tr} and P_{rot} are the translational and rotational power inputs at the hinge b, $R_x(t)$ is the driving force in the x direction, and $R_\theta(t)$ is the driving moment in the θ(t) angular direction. In an ideal case where the driving mechanism on the pin is perfectly elastic, all the negative power exerted on the mechanism can be stored as potential deformation energy for later use, adding to the profiles aerodynamic efficiency. Here, following Berman and Wang [21], a conservative approach is used. We suppose that negative power is not reusable and will not be taken into account in the computation of time averages. Then:

$$P_{tr} = \begin{cases} P_{tr}, & if\ P_{tr} > 0 \\ 0, & if\ P_{tr} < 0 \end{cases} \quad , \quad P_{rot} = \begin{cases} P_{rot}, & if\ P_{rot} > 0 \\ 0, & if\ P_{rot} < 0 \end{cases} \tag{5.7}$$

The power coefficient is defined as:

$$C_{PW}(t) = \frac{(P_{tr}^*(t) + P_{rot}^*(t))/L_c}{\frac{1}{2} \cdot \rho_f \cdot U_c^3 \cdot L_c} = 2(P_{tr}(t) + P_{rot}(t)) \tag{5.8}$$

In Figure 5.3 the dependence of mean values of C_L, C_D, the ratio C_L/C_D and

C_L/C_{PW} with respect to the frequency parameter ω_f/ω_n are shown. The rigid wing has zero torsion stiffness equivalent to $\omega_n = 0$.

For all Reynolds numbers considered, lift and drag coefficients exhibit a peak at the value of $\omega_f/\omega_n = 1/3$. The performance parameter *CL/CD* also exhibits a prominent peak at this frequency ratio. For Re = 75, 250 and 1000 increases of 28%, 23% and 21% respect to the rigid profile performance were observed.

The variations of the aerodynamic quantities with respect to the frequency ratio show similar characteristics for all three Reynolds numbers.The decrease in viscous forces produces a decrease in drag and an increase in lift over the frequency parameter range, therefore, C_L/C_D is generally higher for the higher Re cases. It is interesting to note the striking difference between the graph of C_L/C_D obtained for the $Re = 75$ case and those obtained for the higher Reynolds numbers. For the lowest Reynolds number and $\omega_f/\omega_n = 1/4$, the above ratio is over 13% higher than that obtained for the rigid wing, while for $Re = 250$ it is only increased by 0.5%. In Figure 5.3c, it can be seen that for $\omega_f/\omega_n = 1/3$, the performance ratio C_L/C_{PW} is 39% and 28% higher than that obtained for the rigid wing for $theRe = 75$ and $Re = 250$ cases, respectively. Interestingly, this measure is only about 13% higher than that obtained for the rigid case at $Re = 1000$.

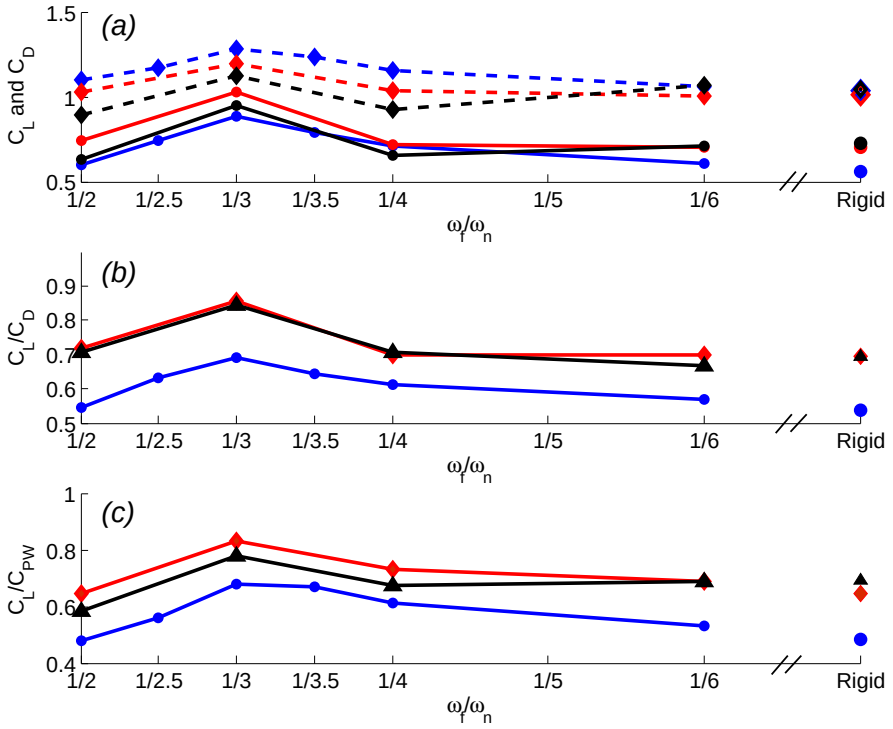


Figure 5.3: *(a)* Mean values of C_L and C_D as a function of the frequency ratio ω_f/ω_n: blue circle, C_L at $Re = 75$; red circle, C_L at $Re = 250$; black circle, C_L at $Re = 1000$; blue diamond, C_D at $Re = 75$; red diamond, C_D at $Re = 250$; and black diamond, C_D at $Re = 1000$. *(b)* Ratio of mean C_L/C_D versus ω_f/ω_n: blue circle, $Re = 75$; red diamond, $Re = 250$; and black triangle, $Re = 1000$. *(c)* Mean lift coefficient per unit of driving power coefficient (C_{PW}) versus ω_f/ω_n: same definitions as in (b).

In Figure 5.4, the time histories of the lift and drag coefficients are shown for all Reynolds numbers at three selected frequency ratios. The effects of flexibility are noticeable on the lift-force peaks at the initiation of the stroke (indicated with a black arrow in Fig. 5.4a). For $Re = 75$ and $\omega_f/\omega_n = 1/2$, corresponding to the

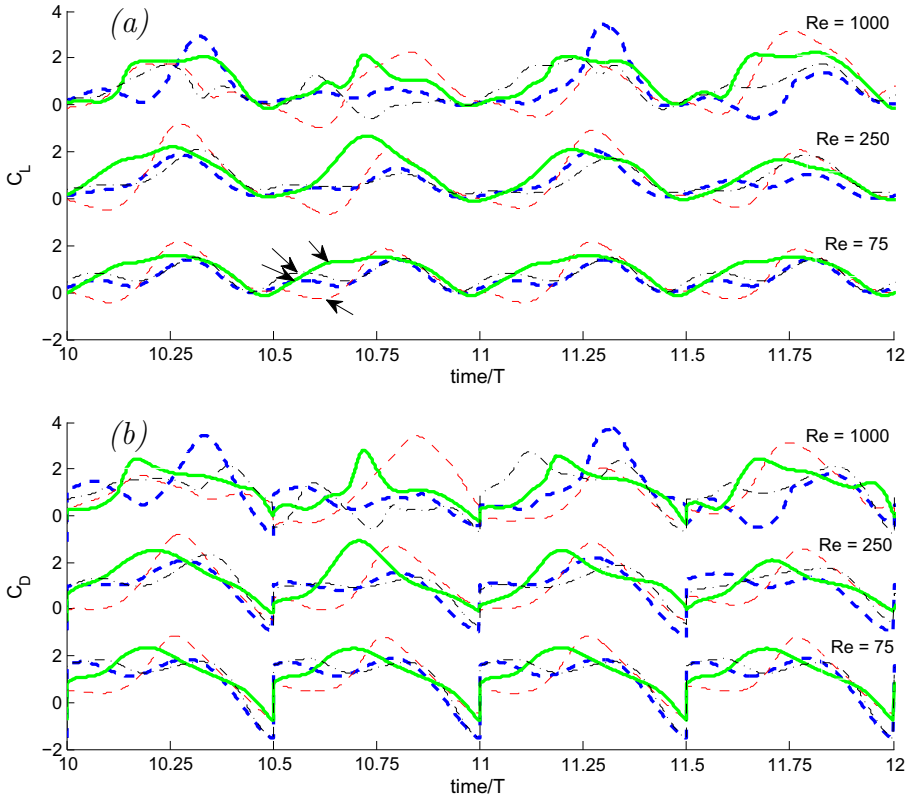


Figure 5.4: Time histories of lift and drag coefficients for $Re = 75$, $Re = 250$ and $Re = 1000$. *(a)* lift coefficient and *(b)* drag coefficient; blue dashed line, rigid wing; red dashed line, flexible wing with $\omega_f/\omega_n = 1/2$; green line, flexible wing with $\omega_f/\omega_n = 1/3$; black dash-dot line, flexible wing with $\omega_f/\omega_n = 1/4$.

most flexible airfoil, this peak is negative, while for the rigid case the coefficient of lift peaks at 0.5. For all cases in between, the enhancement of the mean lift force seen in Figure 5.3a comes from the gradual increase of this peak, which is at 0.83 and 1.28 for $\omega_f/\omega_n = 1/4$ and $\omega_f/\omega_n = 1/3$, respectively. For the latter frequency ratio, where a structural non-linear resonance occurs, the lift peak is also delayed and nearly coincides with the translational lift peak. This is translated into

a larger area under the lift curve per period and a larger time-averaged C_L value. The temporal variations of the lift and drag coefficients for $Re = 250$ and 1000 are more complex than the variations seen at $Re = 75$, and the periodicity is practically lost. Still, in an average sense, negative lift peaks after stroke reversal and larger translational lift peaks are seen when $\omega_f/\omega_n = 1/2$. Also, a widened two-peak lift curve is observed when $\omega_f/\omega_n = 1/3$.

Flow structures

In order to relate the temporal variations of the lift and drag forces to specific flow structures, we examine several realizations of the instantaneous flow fields. In Figure 5.5, vorticity isolines are shown for the rigid and $\omega_f/\omega_n = 1/3$ cases at $Re = 250$. For clarity, the lift coefficient variation has been added (Figure 5.5k), together with the temporal variation of the phase-averaged circulation of the most important vortical structures generated during a flapping cycle. These are the leading edge vortex (LEV) shown in Figure 5.5a, the end of stroke vortex (ESV) shown in Figure 5.5c and trailing edge vortex (TEV) shown in Figure 5.5e. The circulation of each of these vortices was computed as a function of time by direct integration of the vorticity within a given threshold contour around each vortex. The selection of the threshold contour, although arbitrary, was consistently taken to be the lowest closed vorticity isoline in the vicinity of the given vortex. The range of contours was given by 48 isolines within an interval of 10 vorticity units placed around a hand picked value at each time. This value was obtained on a position of the shear layer associated with the given vortex and close to it. Then, the circulation of each vortex

was first computed in time for both forward and reverse strokes over nine periods, and then, both of these quantities were averaged. The number of frames available per period for the simulations was 40.

As the flexible wing approaches the end of the stroke in Figure 5.5a, it exhibits different rotation velocities on the two components A and B. The driven link B rotates with the prescribed angular velocity $\dot{\theta}(t)$, and the lower link A rotates with an angular speed $[\dot{\theta}(t)+ \dot{\alpha}(t)]$. The added angular speed $\dot{\alpha}(t)$ affects the overall dynamics at stroke reversal. First, the camber generated by the angular deformation $\alpha(t)$ at the end of the stroke (see Figure 5.5b) reorients the zero lift direction on the wing and enhances wake capture effects. This enhancement mechanism is analogous to the one produced by orientation advancement in rigid wings [107] . It is important to note that an excessive degree of flexibility (low frequency ratios) produces a large camber at stroke reversal, which has a negative effect on the lift production (e.g. at $\omega_f/\omega_n = 1/3$ in Figure 5.4). The evolution and strength of the LEV on the other hand (see Figure 5.5a) is only a weak function of the wings flexibility. The formation time as well as the maximum circulation shown in Figure 5.5l are approximately the same for the rigid and the flexible wings.

Another effect of the higher rotation speeds at the trailing edge for the flexible wings is the formation of a stronger shear layer, where higher vorticity values are found. This shear layer rolls up into a stronger ESV as can be seen in Figure 5.5c.

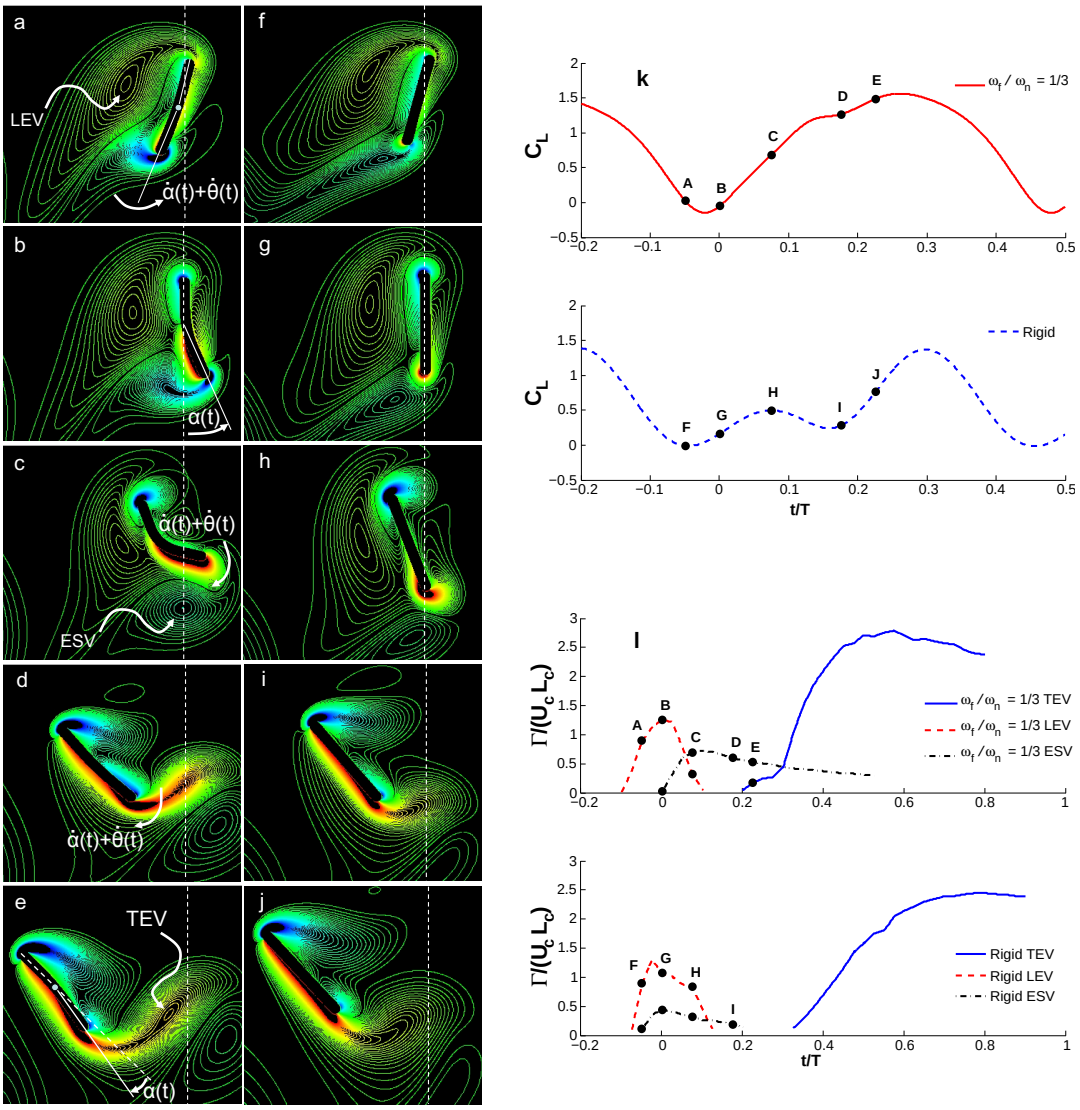


Figure 5.5: Behavior of rigid and flexible profile with $\omega_f/\omega_n = 1/3$ at $Re = 75$ during stroke reversal. (a-e) Vorticity contours for flexible profile at $t/T = -0.0491$, 0.0009, 0.0759, 0.1760, and 0.2260; 80 contours, $\omega_{\min} = -10, \omega_{\max} = 10$. (f-j) Vorticity contours for rigid profile at same time locations. White dashed lines denote end of stroke position. (k) lift coefficient history. (l) Circulation histories of leading edge vortex (LEV), end of stroke vortex (ESV), and trailing edge vortex (TEV). Γ, circulation.

On examining the ESV circulation plots (Figure 5.5k), one finds that the strength and life span are significantly enhanced when compared with those of a rigid wing. The ESV pinches off later, forming a pair of counter-rotating vortices together with the LEV. This vortex pair generates flow directed towards the wing, enhancing the wake-capturing effects. This is more clearly reflected in the lift coefficient evolution shown in Figure 5.5k.

In contrast to the rigid wing, where the lift curve reaches a maximum (point H) and starts to decrease, for the flexible wing the production of increased lift continues for longer (point C). Once a flexible wings deflection has reached its maximum, the elastic energy stored in the torsion spring is released to generate a restoring motion, whose timing again depends on the degree of flexibility of the structure (see Figures 5.5d and 5.5e). This restoring motion produces a dynamical change in the wings camber with a resulting increase in the fluid forces. This also affects the formation and growth of the TEV. It is well established that this flow structure generates a low-pressure zone, which translates into increasing forces up to the pinch-off time (see, for example, reference [106]). It can be seen in Figures 5.5k-l, that the time at which the TEV pinches off is correlated with the translational force peak; this peak is advanced in the $\omega_f/\omega_n = 1/3$ case when compared with that for the rigid wing.

In Figure 5.6, a quantitative comparison of the LEV, ESV and TEV dynamics is shown for different flexibilities at Re=75 by determining their average circulations as a function of time.

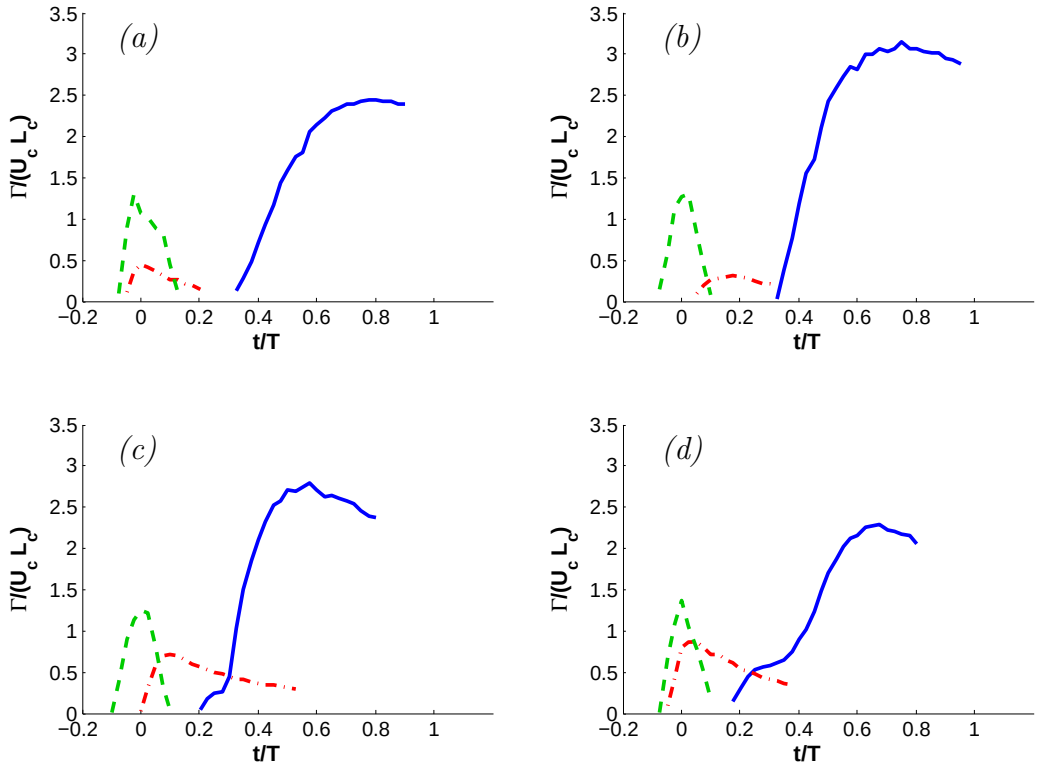


Figure 5.6: Averaged circulations as a function of time with respect to stroke reversal at $Re = 75$; blue line, TEV; green dashed line, LEV; and red dashed-dot line, ESV. (a) Rigid airfoil, (b) Flexible wing with $\omega_f/\omega_n = 1/2$, (c) Flexible airfoil with $\omega_f/\omega_n = 1/3$, and (d) Flexible profile with $\omega_f/\omega_n = 1/4$.

The maximum averaged circulation for each vortex and the time at which it occurs with respect to the stroke reversal are provided in Table 5.1. As expected, from what was observed in Figure 5.5, the LEV dynamics is similar for all frequency ratios in terms of both strength and timing. The TEV on the other hand, attains a higher maximum circulation as the wing becomes more flexible. However, the time it takes to reach this maximum circulation is shortest for $\omega_f/\omega_n = 1/3$, where the best aerodynamic performance is seen. For the ESV vortex, the maximum circulation

ω_f/ω_n	**Γ TEV**	**Time TEV**	**Γ LEV**	**Time LEV**	**Γ ESV**	**Time ESV**
Rigid	2.44	0.78	1.30	-0.02	0.44	0.001
1/2	3.14	0.75	1.30	0.03	0.31	0.18
1/3	2.78	0.58	1.25	0.001	0.72	0.10
1/4	2.29	0.68	1.36	0.001	0.87	0.03
1/6	2.36	0.80	1.43	0.001	0.59	0.02

Table 5.1: Values of maximum mean circulation Γ obtained for TEV, LEV, and ESV at $Re = 75$. The circulations were obtained by integrating the vorticity function for each vortex for two strokes and taking the average value for each time frame. Time is defined as in Figure 5.6 and corresponds to the maximum mean circulation.

increases for the frequency ratios $\omega_f/\omega_n = 1/3$ and 1/4. The peak circulation for f/n=1/3 is 20% lower than that obtained for f/n=1/4, but the deflection at stroke reversal in terms of the maximum deformation angle is 90% larger when $\omega_f/\omega_n = 1/3$ (56 deg. for $\omega_f/\omega_n = 1/3$ and 29 deg. for $\omega_f/\omega_n = 1/4$). This is translated into a larger projected area contributing to the lift force. Also, as seen from Figure 5.6, the time delay of the peak circulation of the ESV vortex is increased as the wing becomes more flexible.

A more direct illustration of the above-mentioned vortex evolutions is given in Figure 5.7, where instantaneous vorticity isolines are shown for eight characteristic instances during a flapping cycle. For $\omega_f/\omega_n = 1/3$ and $\omega_f/\omega_n = 1/4$, it is clear that the enhanced ESV vortices produce an oblique-shaped TEV vortex. For $\omega_f/\omega_n = 1/4$, an excessive negative camber is produced at stroke reversal, which

then generates a high suction zone on the lower side of the wing leading to the negative peak in the CL curve seen in Figure 5.4. The Reynolds number effects on the temporal evolution of the lift and drag forces seen in Figure 5.4 can also be observed in Figure 5.8, where the instantaneous vorticity isolines are shown for $Re = 250$ for all frequency ratios. Clearly, as the viscous damping is decreased, the system dynamics system ceases to be periodic and the important vortices are stronger and are not dissipated as quickly as seen in the $Re = 75$ case (see Figure 5.7). For the case of a rigid wing, for example, the LEV from a given stroke interacts with the shear layer being generated in the next stroke, and this induces a premature formation of the new LEV. This process is not periodic, which is also reflected in the evolution of lift and drag forces. A similar interaction is observed at $\omega_f/\omega_n = 1/2$ (see last three frames in Figure 5.8b). Also, in all flexibility the cases, especially for $\omega_f/\omega_n = 1/3$, strong interaction between the profile and previous ESV is seen during the translation phase. Regardless, the usual components of the flow field, namely TEV, LEV and ESV vortices are noticed for the different flexible profiles studied.

5.1.4 Remarks on flexibility effects

Insect wings are flexible structures that undergo large displacements and deformations during flapping, as the wing structures interact with the surrounding flow. There has been speculation that many insects flap their wings at frequencies close to the natural frequency of the structure. For example, analysis of Manduca sexta wings (see for example references [23], [111]) has shown that the wings first

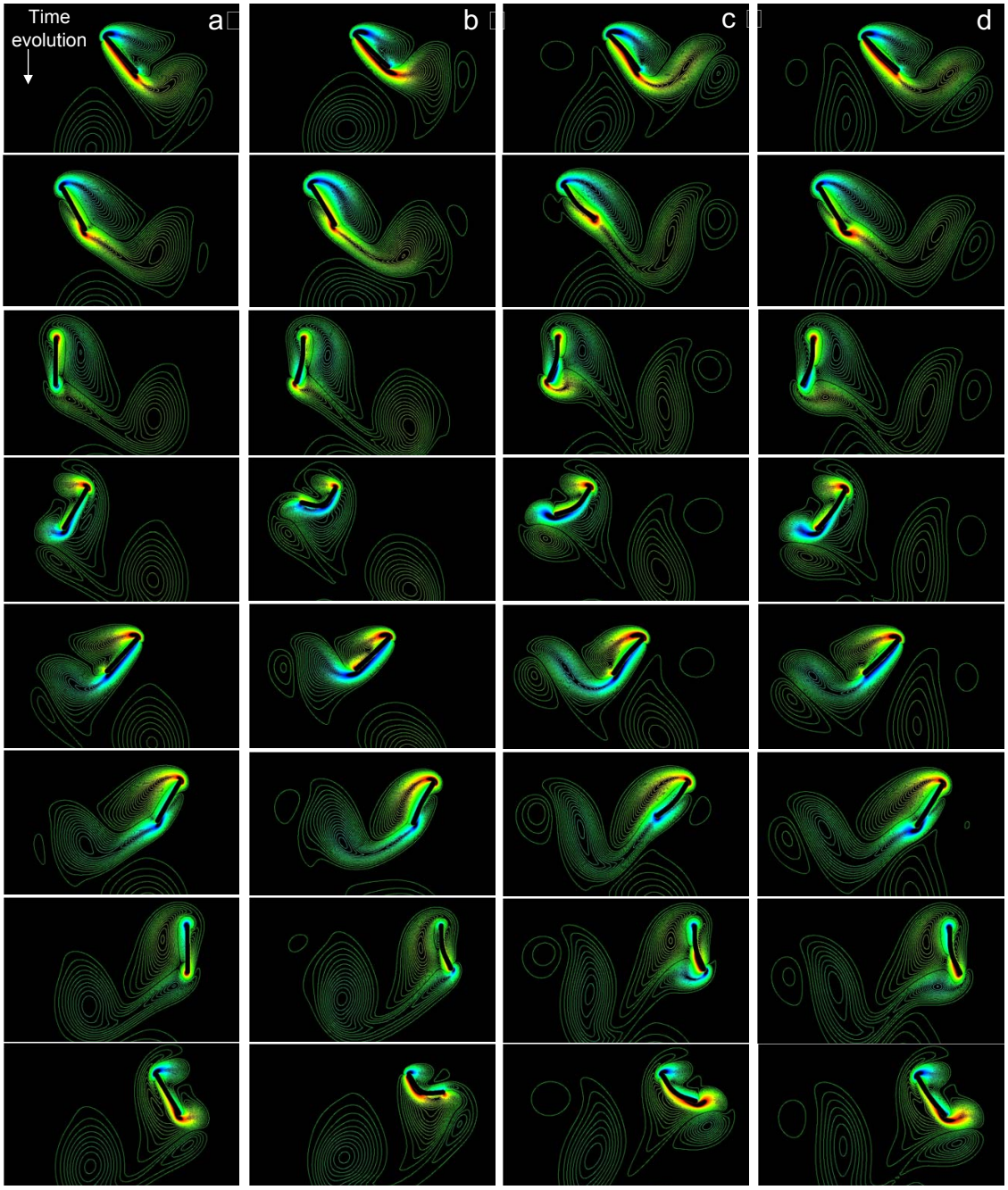


Figure 5.7: Instantaneous vorticity contours over one period at $Re = 75$. Contours range from -10 (blue) to 10 (red) with 80 isolines. Column (a), rigid wing; columns (b-d), flexible profile with $\omega_f/\omega_n = 1/2, 1/3$, and $1/4$, respectively.

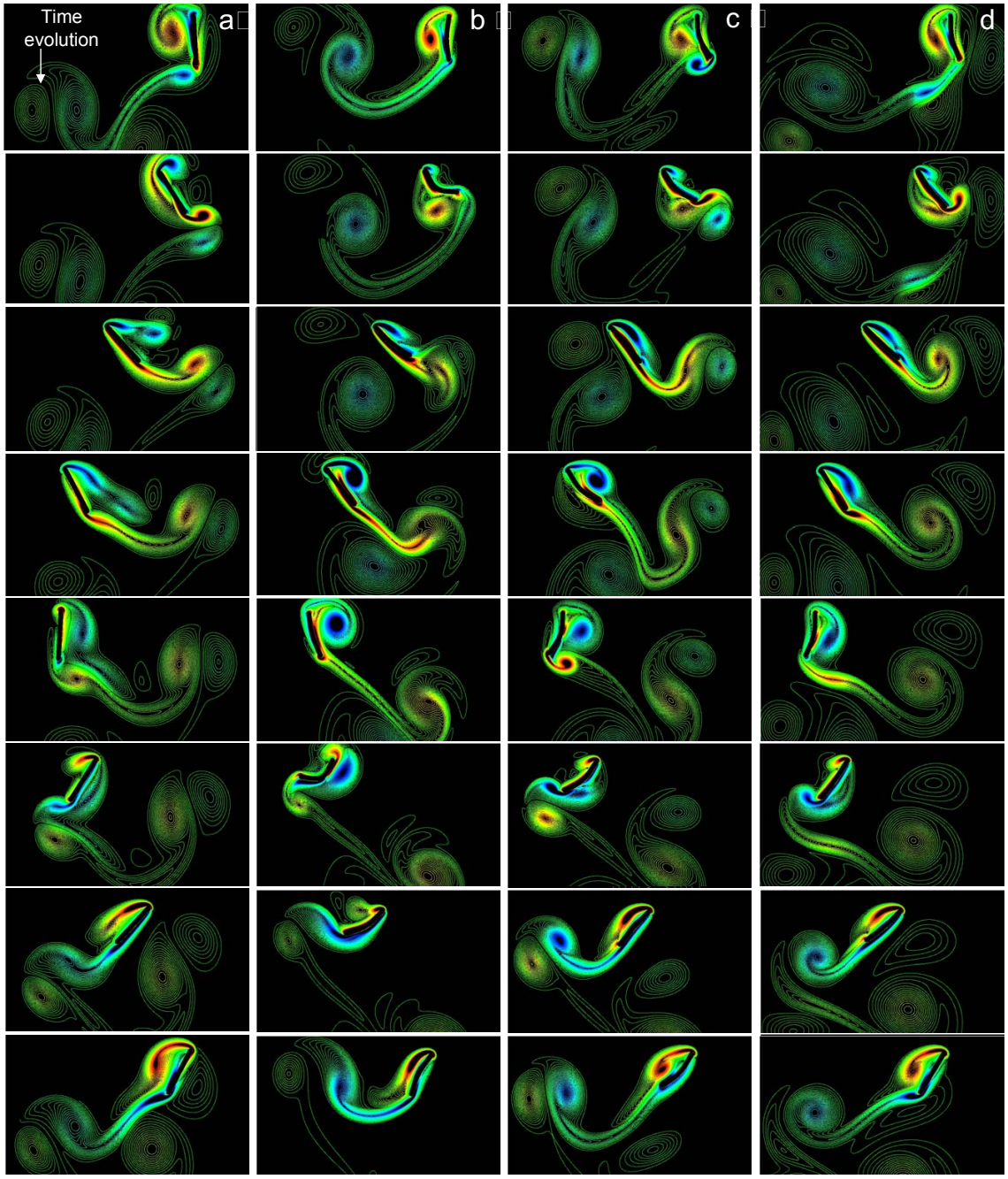


Figure 5.8: Instantaneous vorticity contours over one period at $Re = 75$. Contours range from -10 (blue) to 10 (red) with 80 isolines. Column (a), rigid wing; columns (b-d), flexible profile with $\omega_f/\omega_n = 1/2, 1/3$, and $1/4$, respectively.

natural frequency is close to the driving frequency in normal flapping motion. This suggests that insects may be taking advantage of a structural resonance to reduce energy consumption and enhance aerodynamic performance. Despite the significance of such a hypothesis, only a limited number of studies have addressed the problem due to the exceedingly complex fluid-structure interactions that are encountered in experimental or numerical work. Although three-dimensional computations of the NavierStokes system coupled with a wing structural system are within the reach of todays computers, one still needs to develop appropriate mathematical models and tools to capture all important phenomena in this complex system. In this regard, the present study extends computational work that has been conducted before with simplified two-dimensional rigid wings to include the effects of flexibility. The wing is represented by two rigid links, which are joined at the center by a pin that contains a torsion spring. The kinematics of one of the links is prescribed, while the motion of the other link is determined by the fluidstructure interactions. Although a fairly wide range of Reynolds numbers and frequency ratios was examined, we found that the computations would fail at forcing frequencies close to the linear resonance frequency due to an excessive degree of flexibility. This limitation is due to the concentration of flexibility at a discrete point, and replacement of rigid links with elastic links modeled as elastic beams is expected to help in overcoming this limitation.

As mentioned earlier, the two-link structural system can be perceived as a double pendulum with a common hinge. In particular, when one of the link motions is prescribed, the other link behaves as a pendulum subjected to a constraint arising

from the prescribed motion and complex fluidstructure interactions. Equation (5.2) does resemble the equation of a pendulum driven in a fluid. A straightforward perturbation analysis [71] shows that the structural system can exhibit non-linear resonances at $\omega_f = 1/3\omega_n$ and $\omega_f = 3\omega_n$. These resonances are expected in systems with cubic non-linearities; for example, in the equations governing local oscillations of a pendulum about an equilibrium position and elastic systems such as beams [5]. Operation of the flexible wing at the non-linear superharmonic resonance $\omega_f = 1/3\omega_n$ was found to be beneficial for aerodynamic performance. Inclusion of fluid effects will give rise to quadratic non-linearities and additional non-linear resonances.

For the specific set of kinematics that we considered, most of the benefits of having a flexible wing are associated with the stroke reversal phase of the cycle. Especially for the optimal flexibility cases ($\omega_f = 1/3\omega_n$), the strength and timing of the ESV, as well as the dynamical changes of the wings camber due to structural deformations, are responsible for the performance enhancement. The overall enhancement mechanism is analogous to that produced by orientation advancement in rigid wings (see, for example, reference [107]). It is noted that the present computations cover a wide range of frequency ratios and, consequently, wing deflections range from a few degrees to very large values. For example, in the case of highly stiff wings (e.g. $\omega_f/\omega_n = 1/6$), the maximum deflection angle between the links was about 11, 13 and 16 deg. for Re=75, 250 and 1000, respectively, while for highly flexible wings (e.g. $\omega_f/\omega_n = 1/2$) the corresponding numbers were 67, 68 and 91 deg., respectively. In insects, the wing deformation magnitudes increase as the body size and mass increase, and it is conceivable that deformations seen in this study at

the aerodynamically preferred frequency ratio of $\omega_f/\omega_n = 1/3$ could be possible in some species. On the other hand, for small insects such as Drosophila only small magnitude wing deformations have been observed. The computations of this study show that as the wing is made stiffer, the performance enhancements are marginal when compared with a rigid foil. For example, at $Re = 75$ and the highly stiff case $\omega_f/\omega_n = 1/6$, C_L/C_D is approximately 6% higher than that of a rigid foil. For the higher Reynolds numbers $Re = 250$ and 1000 there is actually no enhancement, and the performance is worse than that obtained with a rigid foil. The above results indicate that low Reynolds number regimes might benefit performance even at small chordwise distortions. The force histories, in particular for the low Reynolds numbers, appear to reach a periodic steady state after the initial transients for all of the frequency ratios that were considered, suggesting that quasi-steady models might be able to reproduce this behavior. Such models have been reported in the literature, and have been adapted for flapping flight based on models developed for high Reynolds number fixed wing aeroelasticity studies by including wing rotation along with translation ([32];[34]), and forces due to added mass ([89]). In a more recent study by Wang and colleagues [107] the unsteady forces from experiments and with two-dimensional computations were compared with the quasi-steady model predictions. They pointed out that the force predictions, which were made by using models based on potential flow theory [70] for a constant pitching amplitude and constant translating speed wing, deviated substantially from the experimentally determined unsteady forces. The force predictions from a semi-empirical model based on numerical results from steady translating wings at a fixed angle of attack were in

broad qualitative agreement with the unsteady forces. However, detailed comparisons revealed that, depending on the kinematics, the unsteady effects can reduce the total lift by a factor of 2 to 3. In the present case, due to the wings flexibility, the identification of the quasi-steady contributions is more complex as additional new states have been included.

5.2 An Insect at free longitudinal flight

While aerodynamics of insects and other flying animals has received considerable attention over the past decade, the same is not true of response stability of flying animals. Currently there is a lack of quantitative understanding of the stability of flying insects. In this section we will present preliminary results from our effort towards the development of a comprehensive framework that will enable us to conduct sophisticated numerical experiments to understand and assess the response stability of insect-like systems in different disturbance environments.

5.2.1 Multi-body insect model kinematics

The insect model is composed of one pair of rigid wings, RW and LW, hinged to a rigid body, $RB1$, which respresents the insect's head and thorax, and another body, $RB2$, representing the insect's abdomen. The latter is also articulated to $RB1$ (See Figure 5.9). The coordinate transformations among the different reference frames and with respect to the inertial frame N are defined in terms of Euler angle sequences. The kinematic analysis for the given system starts by selecting the

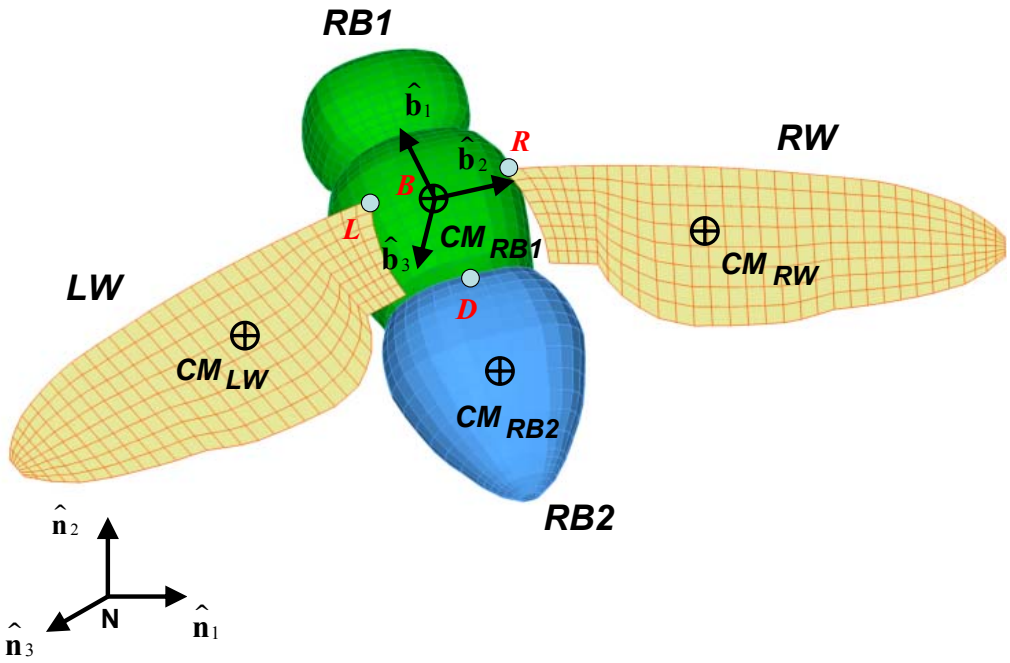


Figure 5.9: Four rigid body model for the full insect. Rigid Body $RB1$, head-thorax, is represented by the tracking frame B attached to its center of mass. Rigid body $RB2$, the abdomen, is hinged to $RB1$ at point D where its respective body frame is defined. The wing bodies RW and LW are articulated to $RB1$ at points R and L respectively, where their tracking frames are defined.

sequences that will define the transformations among the reference systems N, B, D, R and L. The transformation between the inertial system N and the $RB1$ tracking system, B, defined by the unitary vectors $\hat{\mathbf{b}}_1$, $\hat{\mathbf{b}}_2$, $\hat{\mathbf{b}}_3$, is given by a 180° rotation with respect to the axis given by $\hat{\mathbf{n}}_1$, followed by a 3-2-1 Euler angle sequence with angular coordinates ψ (yaw respect to $\hat{\mathbf{e}}_3$), θ (pitch respect to $\hat{\mathbf{e}}'_2$) and ϕ (roll respect to $\hat{\mathbf{e}}''_1$), as shown in Figure 5.10. This is the typical flight mechanics coordinate transformation and yields a singularity for pitch angles multiple of $\pi/2$.

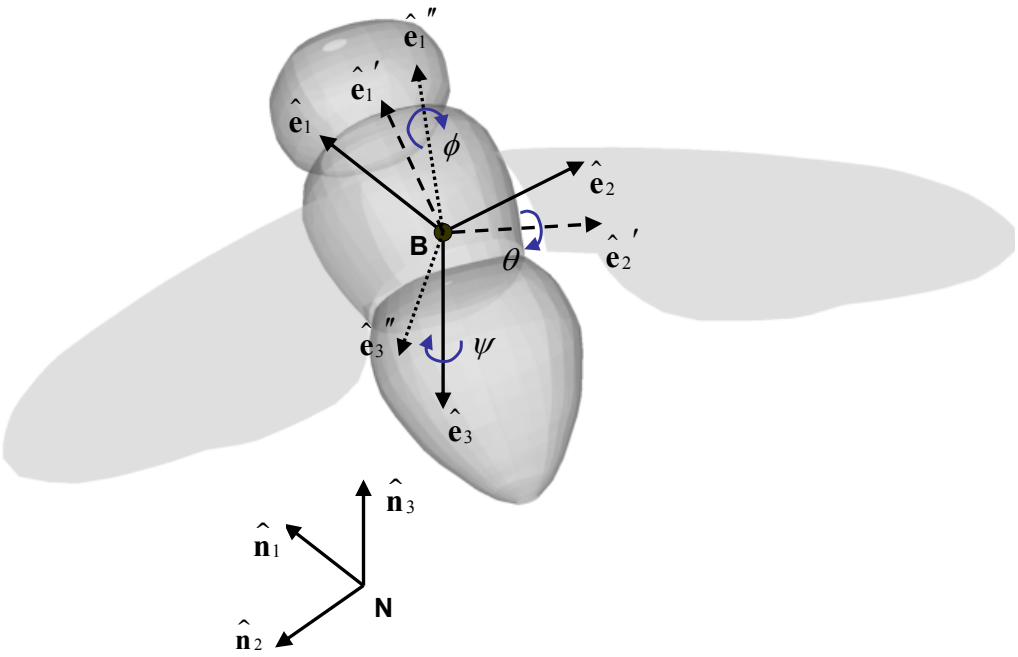


Figure 5.10: Transformation from inertial reference frame N and $RB1$ body frame B. A 180° rotation respect to $\hat{\mathbf{n}}_1$ is given, followed by a 3-2-1 Euler angle sequence with angular coordinates ψ (yaw respect to $\hat{\mathbf{e}}_3$), θ (pitch respect to $\hat{\mathbf{e}}_2'$) and ϕ (roll respect to $\hat{\mathbf{e}}_1''$)

The transformation matrix from N to B is

$$[\mathbf{T}_{BN}] = \begin{bmatrix} c\theta c\psi & -c\theta s\psi & s\theta \\ s\phi s\theta c\psi - c\phi s\psi & -s\phi s\theta s\psi - c\phi c\psi & -s\phi c\theta \\ c\phi s\theta c\psi + s\phi s\psi & -c\phi s\theta s\psi + s\phi c\psi & -c\phi c\theta \end{bmatrix} \tag{5.9}$$

where $s()$ and $c()$ are the sin() and cos() functions respectively. The transformation from system B to the system D local to the abdomen is given by two rotation angles: angle ψ_2 with respect to $\hat{\mathbf{b}}_3$, and θ_2 with respect to the intermediate axis $\hat{\mathbf{b}}_2'$. The

corresponding transformation matrix is:

$$[\mathbf{T}_{DB}] = \begin{bmatrix} c\theta_2 c\psi_2 & c\theta_2 s\psi_2 & -s\theta_2 \\ -s\psi_2 & c\psi_2 & 0 \\ s\theta_2 c\psi_2 & s\theta_2 s\psi_2 & c\theta_2 \end{bmatrix} \tag{5.10}$$

Next the transformation between the system B and the system R local to the right wing is done in the following steps:

1. 90° rotation with respect to $\hat{\mathbf{b}}_3$, followed by a 90° rotation with respect to $\hat{\mathbf{b}}_1'$
2. β_3' degrees rotation with respect to $\hat{\mathbf{b}}_1''$
3. An 3-2-1 Euler angle sequence of angles ψ_3 with respect to $\hat{\mathbf{b}}_3'''$ (beating angle in the stroke plane, defined here relative to the body coordinate system), θ_3 with respect to $\hat{\mathbf{b}}_2^{IV}$ (out of stroke plane angle), and ϕ_3 with respect to $\hat{\mathbf{b}}_1^{V}$ (angle that modifies the angle of attack of the wing).

The resulting transformation matrix, $[\mathbf{T}_{RB}]$, is defined as follows:

$$\begin{bmatrix} c\theta_3 s\psi_3 s\beta_3' - s\theta_3 c\beta_3' & c\theta_3 c\psi_3 & c\theta_3 s\psi_3 c\beta_3' + s\theta_3 s\beta_3' \\ (\alpha_4 s\theta_3 + \alpha_1)s\beta_3' + s\phi_3 c\theta_3 c\beta_3' & \alpha_2 s\theta_3 - \alpha_3 & (\alpha_4 s\theta_3 + \alpha_1)c\beta_3' - s\phi_3 c\theta_3 s\beta_3' \\ (\alpha_3 s\theta_3 - \alpha_2)s\beta_3' + c\phi_3 c\theta_3 c\beta_3' & \alpha_1 s\theta_3 + \alpha_4 & (\alpha_3 s\theta_3 - \alpha_2)c\beta_3' - c\phi_3 c\theta_3 s\beta_3' \end{bmatrix} \tag{5.11}$$

where the R system is defined by the versor triad $\hat{\mathbf{r}}_1, \hat{\mathbf{r}}_2, \hat{\mathbf{r}}_3$ (see Figure 5.11 for reference on the different coordinate systems). Here, $\alpha_1 = c\phi_3 c\psi_3$, $\alpha_2 = s\phi_3 c\psi_3$, $\alpha_3 = c\phi_3 s\psi_3$ and $\alpha_4 = s\phi_3 s\psi_3$. The transformation from frame B to frame L is very similar to the one described above:

1. $-90°$ rotation with respect to $\hat{\mathbf{b}}_3$, followed by a $-90°$ rotation to respect to $\hat{\mathbf{b}}_1'$

2. β_4' degrees of rotation with respect to $\hat{\mathbf{b}}_1''$. Assuming that the pitch angle of body B is zero, the angle β_4' is related to the stroke plane angle, β, reported in many studies ([108], [7]) by the equation $\beta_4' = 90° - \beta$.

3. Euler angle sequence 3-2-1 of angles ψ_4 with respect to $\hat{\mathbf{b}}_3'''$, θ_4 with respect to $\hat{\mathbf{b}}_2^{IV}$, and ϕ_4 with respect to $\hat{\mathbf{b}}_1^{V}$.

In the same manner the transformation matrix $[\mathbf{T}_{LB}(\phi_4, \theta_4, \psi_4, \beta_4')]$ can be written as:

$$\begin{bmatrix} c\theta_4 s\psi_4 s\beta_4' - s\theta_4 c\beta_4' & -c\theta_4 c\psi_4 & -c\theta_4 s\psi_4 c\beta_4' - s\theta_4 s\beta_4' \\ (\gamma_4 s\theta_4 + \gamma_1)s\beta_4' + s\phi_4 c\theta_4 c\beta_4' & -\gamma_2 s\theta_4 + \gamma_3 & -(\gamma_4 s\theta_4 + \gamma_1)c\beta_4' + s\phi_4 c\theta_4 s\beta_4' \\ (\gamma_3 s\theta_4 - \gamma_2)s\beta_4' + c\phi_4 c\theta_4 c\beta_4' & -\gamma_1 s\theta_4 - \gamma_4 & -(\gamma_3 s\theta_4 - \gamma_2)c\beta_4' + c\phi_4 c\theta_4 s\beta_4' \end{bmatrix} \tag{5.12}$$

The L system is defined by the versor triad $\hat{\mathbf{l}}_1, \hat{\mathbf{l}}_2, \hat{\mathbf{l}}_3$, and $\gamma_1 = c\phi_4 c\psi_4$, $\gamma_2 = s\phi_4 c\psi_4$, $\gamma_3 = c\phi_4 s\psi_4$, $\gamma_4 = s\phi_4 s\psi_4$. The value that must be given to the angular variable β_3' on the right wing, in order to conserve the same stroke plane as the one present for the left wing is $\beta_3' = -\beta_4'$.

In order to produce symmetric flapping, the relation among the right wing Euler angle sequence and left wing angular variables must be:

$$\phi_3(t) = -\phi_4(t)\ ; \quad \theta_3(t) = \theta_4(t)\ ; \quad \psi_3(t) = -\psi_4(t) \tag{5.13}$$

Once the transformations among coordinate systems and the positions of the origins of systems D, R and L are given with respect to the origin of the thorax, B, the positions of all points in the different component bodies can be computed in terms of the inertial frame N. Since the rotation matrices form an orthonormal set

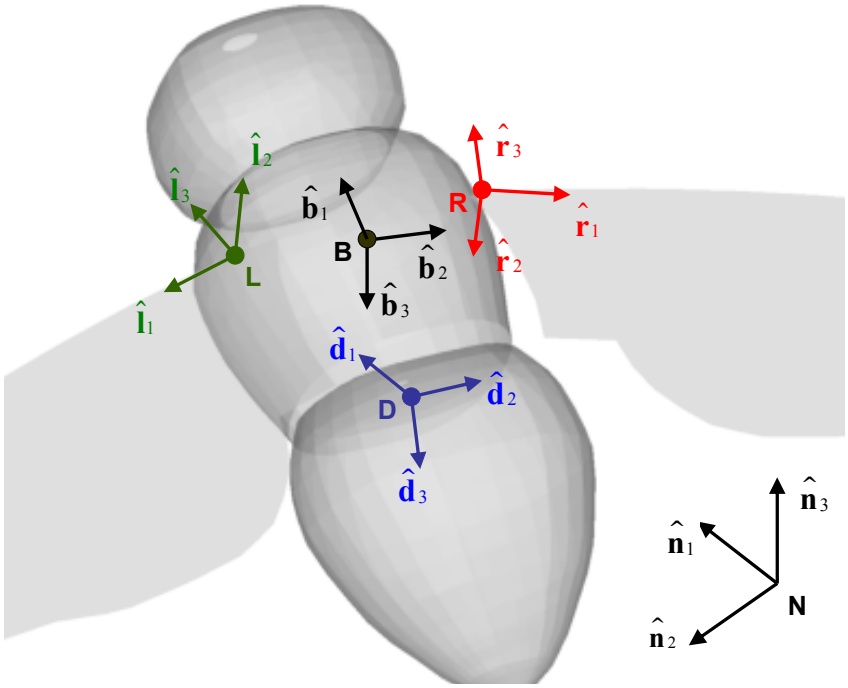


Figure 5.11: Reference frames for rigid body system.

for the points of body B we have:

$$\mathbf{r}_p^N = \mathbf{R}_B^N + [\mathbf{T}_{NB}]\,\mathbf{r}_{p/B}^B \tag{5.14}$$

where $\mathbf{R}_B^N = [x(t)\ y(t)\ z(t)]$, $[\mathbf{T}_{NB}] = [\mathbf{T}_{BN}]^T$, $\mathbf{r}_p^N$ represents the position of a generic point in body B with respect to the origin of the inertial frame in terms of the $\hat{\mathbf{n}}_1$, $\hat{\mathbf{n}}_2$, $\hat{\mathbf{n}}_3$ triad, and $\mathbf{r}_{p/B}^B$ is the position vector from the origin of frame B to point p in local $\hat{\mathbf{b}}_1$, $\hat{\mathbf{b}}_2$, $\hat{\mathbf{b}}_3$ coordinates. A similar expression can be written for the remaining bodies:

$$\mathbf{r}_{p_D}^N = \mathbf{R}_B^N + [\mathbf{T}_{NB}]\,\mathbf{r}_{D/B}^B + [\mathbf{T}_{ND}]\,\mathbf{r}_{p_D/D}^D \quad (\textit{point in the abdomen}) \tag{5.15}$$

$$\mathbf{r}_{p_R}^N = \mathbf{R}_B^N + [\mathbf{T}_{NB}]\,\mathbf{r}_{R/B}^B + [\mathbf{T}_{NR}]\,\mathbf{r}_{p_R/R}^R \quad (\textit{point in right wing}) \tag{5.16}$$

$$\mathbf{r}_{p_L}^N = \mathbf{R}_B^N + [\mathbf{T}_{NB}]\,\mathbf{r}_{L/B}^B + [\mathbf{T}_{NL}]\,\mathbf{r}_{p_L/L}^L \quad (\textit{point in left wing}) \tag{5.17}$$

To complete the kinematic analysis of this mechanical system the angular velocities and accelerations of the different bodies in terms of the time derivatives of the Euler angles must be obtained. In figure 5.10, the angular velocity of frame B with respect to the inertial reference frame N can be expressed in terms of the time derivatives of the Euler angles:

$$ {}^N\boldsymbol{\omega}^B = \dot{\psi}\hat{\mathbf{e}}_3 + \dot{\theta}\hat{\mathbf{e}}'_2 + \dot{\phi}\hat{\mathbf{e}}''_1 \tag{5.18} $$

where the $\hat{\mathbf{e}}_3$, $\hat{\mathbf{e}}'_2$, $\hat{\mathbf{e}}''_1$ unit vectors in terms of the versors of the B frame are:

$$ \hat{\mathbf{e}}_3 = -s\theta\hat{\mathbf{b}}_1 + c\theta s\phi\hat{\mathbf{b}}_2 + c\theta c\phi\hat{\mathbf{b}}_3 \quad ; \quad \hat{\mathbf{e}}'_2 = c\phi\hat{\mathbf{b}}_2 - s\phi\hat{\mathbf{b}}_3 \quad ; \quad \hat{\mathbf{e}}''_1 = \hat{\mathbf{b}}_1 \tag{5.19} $$

Substituting ((5.19)) into ((5.18)) and rearranging into matrix-vector form we get:

$$ {}^N\boldsymbol{\omega}^B_B = \begin{bmatrix} 1 & 0 & -s\theta \\ 0 & c\theta & c\theta s\phi \\ 0 & -s\phi & c\theta c\phi \end{bmatrix} \begin{Bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{Bmatrix} \tag{5.20} $$

which is the angular velocity of frame B respect to N in terms of the unit vectors $\hat{\mathbf{b}}_1$, $\hat{\mathbf{b}}_2$, $\hat{\mathbf{b}}_3$. We denote the matrix in the above equation as $[{}^N\mathbf{B}^B_{(\phi,\theta)}]$, and $\dot{\boldsymbol{\phi}}_B = [\dot{\phi}\ \dot{\theta}\ \dot{\psi}]^T$. Considering the vector identity ${}^N\boldsymbol{\omega}^B_B \times {}^N\boldsymbol{\omega}^B_B = \mathbf{0}$, the corresponding angular acceleration is given by:

$$ {}^N\boldsymbol{\alpha}^B_B = \frac{{}^N d}{dt}\left({}^N\boldsymbol{\omega}^B_B\right) = \frac{{}^B d}{dt}\left({}^N\boldsymbol{\omega}^B_B\right) = [{}^N\mathbf{B}^B]\ddot{\boldsymbol{\phi}}_B + [{}^N\dot{\mathbf{B}}^B]\dot{\boldsymbol{\phi}}_B \tag{5.21} $$

The velocity and acceleration of a point p on $RB1$, in terms of the unit vectors of frame B are:

$$ \mathbf{v}^B_p = \frac{{}^N d}{dt}\left(\mathbf{r}_p\right)_B = \frac{{}^N d}{dt}\left(\mathbf{R}_B\right)_B + {}^N\boldsymbol{\omega}^B_B \times \mathbf{r}^B_{p/B} \tag{5.22} $$

$$ \mathbf{a}^B_p = \frac{{}^N d^2}{dt^2}\left(\mathbf{r}_p\right)_B = \frac{{}^N d^2}{dt^2}\left(\mathbf{R}_B\right)_B + {}^N\boldsymbol{\alpha}^B_B \times \mathbf{r}^B_{p/B} + {}^N\boldsymbol{\omega}^B_B \times {}^N\boldsymbol{\omega}^B_B \times \mathbf{r}^B_{p/B} \tag{5.23} $$

The angular velocity of frame D with respect to frame B is ${}^B\boldsymbol{\omega}^D = \dot{\psi}_2\hat{\mathbf{b}}_3 + \dot{\theta}_2\hat{\mathbf{b}}_2'$, and in terms of the B frame decomposition:

$$ {}^B\boldsymbol{\omega}_B^D = \begin{bmatrix} -s\psi_2 & 0 \\ c\psi_2 & 0 \\ 0 & 1 \end{bmatrix} \begin{Bmatrix} \dot{\theta}_2 \\ \dot{\psi}_2 \end{Bmatrix} = [{}^B\mathbf{B}_{(\psi_2)}^D]\dot{\boldsymbol{\phi}}_D \tag{5.24} $$

Then, the angular velocity and angular acceleration of frame D with respect to the inertial frame N is:

$$ {}^N\boldsymbol{\omega}_B^D = {}^N\boldsymbol{\omega}_B^B + {}^B\boldsymbol{\omega}_B^D \tag{5.25} $$

$$ {}^N\boldsymbol{\alpha}^D = \frac{{}^N d}{dt}\left({}^N\boldsymbol{\omega}^B + {}^B\boldsymbol{\omega}^D\right) = \frac{{}^B d}{dt}\left({}^N\boldsymbol{\omega}^B\right) + \frac{{}^B d}{dt}\left({}^B\boldsymbol{\omega}^D\right) + {}^N\boldsymbol{\omega}^B \times {}^B\boldsymbol{\omega}^D \tag{5.26} $$

Then, using equation (5.24), ${}^B\boldsymbol{\alpha}_B^D = \frac{{}^B d}{dt}\left({}^B\boldsymbol{\omega}^D\right) = [{}^B\dot{\mathbf{B}}^D]\dot{\boldsymbol{\phi}}_D + [{}^B\mathbf{B}^D]\ddot{\boldsymbol{\phi}}_D$, the absolute angular acceleration of frame D in terms of the B frame unit vectors is:

$$ {}^N\boldsymbol{\alpha}_B^D = {}^N\boldsymbol{\alpha}_B^B + {}^B\boldsymbol{\alpha}_B^D + {}^N\boldsymbol{\omega}_B^B \times {}^B\boldsymbol{\omega}_B^D \tag{5.27} $$

The linear velocity and acceleration of a point p_D on $RB2$ can be found as follows:

$$ \mathbf{v}_{p_D}^B = \frac{{}^N d}{dt}\left(\mathbf{R}_B\right)_B + {}^N\boldsymbol{\omega}_B^B \times \mathbf{r}_{D/B}^B + {}^N\boldsymbol{\omega}_B^D \times \mathbf{r}_{p_D/D}^B \tag{5.28} $$

$$ \mathbf{a}_{p_D}^B = \mathbf{a}_D^B + {}^N\boldsymbol{\alpha}_B^D \times \mathbf{r}_{p_D/D}^B + {}^N\boldsymbol{\omega}_B^D \times {}^N\boldsymbol{\omega}_B^D \times \mathbf{r}_{p_D/D}^B \tag{5.29} $$

where $\mathbf{a}_D^B$ is the acceleration of the D frame origin, and is computed from equation (5.23) for this point in $RB1$.

For the right wing, the angular velocity definition, and its expression in terms

of frame B are:

$$
\begin{aligned}
{}^{B}\boldsymbol{\omega}^{R} &= \dot{\psi}_3\hat{\mathbf{b}}_3^{'''} + \dot{\theta}_3\hat{\mathbf{b}}_2^{IV} + \dot{\phi}_3\hat{\mathbf{b}}_1^{V}, \quad \text{and} && (5.30)\\
{}^{B}\boldsymbol{\omega}_B^R &= \begin{bmatrix} c\theta_3 s\psi_3 s\beta_3 - s\theta_3 c\beta_3 & c\psi_3 s\beta_3 & c\beta_3 \\ c\theta_3 c\psi_3 & -s\psi_3 & 0 \\ c\theta_3 s\psi_3 c\beta_3 + s\theta_3 s\beta_3 & c\psi_3 c\beta_3 & -s\beta_3 \end{bmatrix} \begin{Bmatrix} \dot{\phi}_3 \\ \dot{\theta}_3 \\ \dot{\psi}_3 \end{Bmatrix}, && (5.31)
\end{aligned}
$$

respectively. Then the corresponding absolute angular velocity ${}^{N}\boldsymbol{\omega}_B^R$ and acceleration ${}^{N}\boldsymbol{\alpha}_B^R$ can be readily obtained from expressions analogous to equations (5.25)-(5.27). Then the linear velocity and acceleration for a point p_R laying on the right wing, RW, are:

$$\mathbf{v}_{p_R}^B = \frac{{}^{N}d}{dt}\left(\mathbf{R}_B\right)_B + {}^{N}\boldsymbol{\omega}_B^B \times \mathbf{r}_{R/B}^B + {}^{N}\boldsymbol{\omega}_B^R \times \mathbf{r}_{p_R/R}^B \tag{5.32}$$

$$\mathbf{a}_{p_R}^B = \mathbf{a}_R^B + {}^{N}\boldsymbol{\alpha}_B^R \times \mathbf{r}_{p_R/R}^B + {}^{N}\boldsymbol{\omega}_B^R \times {}^{N}\boldsymbol{\omega}_B^R \times \mathbf{r}_{p_R/R}^B \tag{5.33}$$

Equations similar to (5.25)-(5.27) and (5.32)-(5.33) can de derived for the kinematics of the left wing in terms of frame B. The angular velocity, for example, relative to B is:

$$
\begin{aligned}
{}^{B}\boldsymbol{\omega}^{L} &= \dot{\psi}_4\hat{\mathbf{b}}_3^{'''} + \dot{\theta}_4\hat{\mathbf{b}}_2^{IV} + \dot{\phi}_4\hat{\mathbf{b}}_1^{V} && (5.34)\\
{}^{B}\boldsymbol{\omega}_B^L &= \begin{bmatrix} c\theta_4 s\psi_4 s\beta_4 - s\theta_4 c\beta_4 & c\psi_4 s\beta_4 & c\beta_4 \\ -c\theta_4 c\psi_4 & s\psi_4 & 0 \\ -\left(c\theta_4 s\psi_4 c\beta_4 + s\theta_4 s\beta_4\right) & -c\psi_4 c\beta_4 & s\beta_4 \end{bmatrix} \begin{Bmatrix} \dot{\phi}_4 \\ \dot{\theta}_4 \\ \dot{\psi}_4 \end{Bmatrix} && (5.35)
\end{aligned}
$$

5.2.2 Kinematics of longitudinal flight

In the present study we will limit ourselves to longitudinal flight, where the insect's motion is simplified to longitudinal mechanics. Such a flight mode is possible

when: i) a symmetric wing motion is used and the lateral force and moments are very small compared to their longitudinal counterparts; ii) the mass distribution is symmetric and has the same symmetry plane as the model geometry. In such case the the coordinates for $RB1$ reduce to the displacements $x(t)$ and $z(t)$ along the symmetry plane, $\hat{\mathbf{n}}_\mathbf{1} - \hat{\mathbf{n}}_\mathbf{3}$, and the $\theta(t)$ angle with respect to the body axis, $\hat{\mathbf{b}}_\mathbf{2}$, which is aligned at all times with the $-\hat{\mathbf{n}}_\mathbf{2}$ direction. The transformation matrix from frame N to B and the angular velocity vector reduce to:

$$[\mathbf{T}_{BN}] = \begin{bmatrix} cos(\theta) & 0 & sin(\theta) \\ 0 & -1 & 0 \\ sin(\theta) & 0 & -cos(\theta) \end{bmatrix}, \quad {}^N\boldsymbol{\omega}_B^B = \dot{\theta}\hat{\mathbf{b}}_\mathbf{2} \tag{5.36}$$

The coordinates that describe the relative orientation of $RB2$ with respect to $RB1$ reduce to the variable θ_2. The corresponding tansformation matrix from frame B to D, and angular velocity are:

$$[\mathbf{T}_{DB}] = \begin{bmatrix} cos(\theta_2) & 0 & -sin(\theta_2) \\ 0 & 1 & 0 \\ sin(\theta_2) & 0 & cos(\theta_2) \end{bmatrix}, \quad {}^B\boldsymbol{\omega}_B^D = \dot{\theta}_2\hat{\mathbf{b}}_\mathbf{2} \tag{5.37}$$

The rest of the kinematic analysis follows the same path as in the previous section.

5.2.3 Dynamics of longitudinal flight

Here we derive the equations of motion for the longitudinal dynamics using the classical Lagrange's equations approach [14]. We define the generalized coordinates $x(t)$, $z(t)$, $\theta(t)$ and $\theta_2(t)$ to be force driven, and all angular variables that describe the relative motion of the wing with respect to $RB1$ to be prescribed in time. We

first need to define the mass inertia properties associated with each of the four rigid bodies. For $RB1$ they are: m_B and I_{Byy}, the total mass and mass moment of inertia with respect to the body axis given by $\hat{\mathbf{b}}_\mathbf{2}$ (see figure 5.12a). We make the additional assumption that the origin of the B frame is at the same location as the center of mass, CM_{RB1}, of body $RB1$. All mass moments of inertia are defined with respect to the corresponding body center of mass in local coordinates. For $RB2$: m_D and I_{Dyy}, are the total mass and mass moment of inertia with respect to the body axis passing through CM_{RB2} and is parallel to $\hat{\mathbf{d}}_\mathbf{2}$, which is in longitudinal motion the same as $\hat{\mathbf{b}}_\mathbf{2}$. The center of mass of $RB2$ is located on the symmetry plane, $\hat{\mathbf{d}}_\mathbf{1} - \hat{\mathbf{d}}_\mathbf{3}$, at positions x_{CD} and z_{CD} from the origin of frame D (see figure 5.12b).

The right wing, RW, is on $\hat{\mathbf{r}}_\mathbf{1} - \hat{\mathbf{r}}_\mathbf{3}$ plane and in finite thickness cases the wing is assumed to be symmetric with respect to this plane. The required inertia properties in this case are: m_R, I_{Rxx}, I_{Ryy}, I_{Rzz}, I_{Rxz}, the wing mass and moments of inertia with respect to axes passing through CM_{RW} and parallel to $\hat{\mathbf{r}}_\mathbf{1}$, $\hat{\mathbf{r}}_\mathbf{2}$ and $\hat{\mathbf{r}}_\mathbf{3}$, and the product of inertia on the $\hat{\mathbf{r}}_\mathbf{1} - \hat{\mathbf{r}}_\mathbf{3}$ plane respectively (see figure 5.12c). The center of mass of RW is located at the local coordinates x_{CR} and z_{CR} from the origin of the R system. The left wing, LW, has similar properties as the right wing: m_L, I_{Lxx}, I_{Lyy}, I_{Lzz}, I_{Lxz}. Due to the symmetry assumption the inertia properties of the left wing are equal to the ones on the right wing. The position of the center of mass of LW is given by the x_{CL}, z_{CL} local coordinates.

Next we will define the kinetic and potential energies for each rigid body in

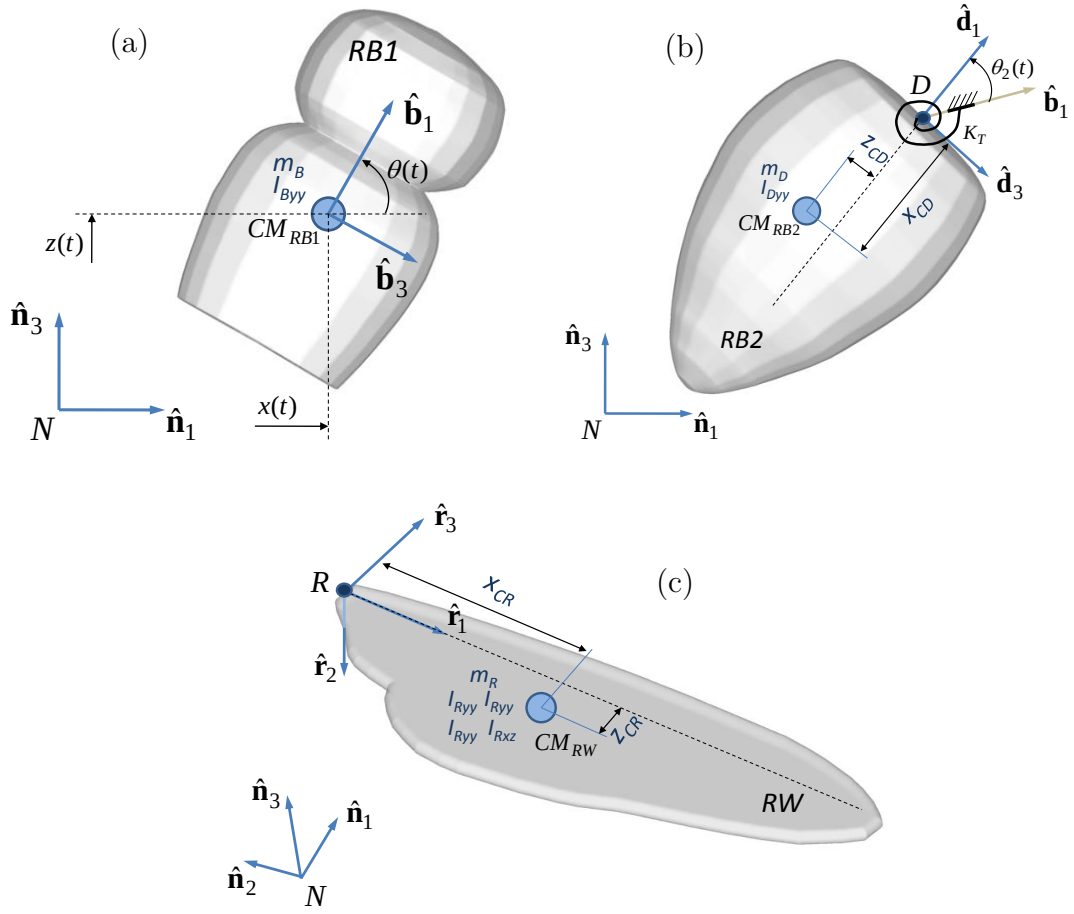


Figure 5.12: Inertia properties and reference frames for bodies: (a) $RB1$, (b) $RB2$, and (c) RW. Properties of the left wing LW are analogous to RW. The degrees of freedom of the system are the variables $x(t)$, $z(t)$, $\theta(t)$ and $\theta_2(t)$.

the model. In particular the kinetic energy can be defined as:

$$T_{B_i} = \frac{1}{2}\left(\mathbf{v}_{CM_{B_i}}\right)^T m_{B_i}\mathbf{v}_{CM_{B_i}} + \frac{1}{2}\left({}^N\boldsymbol{\omega}^{B_i}\right)^T [I_{B_i}]\ {}^N\boldsymbol{\omega}^{B_i} \tag{5.38}$$

where B_i is any of the $RB1$, $RB2$, RW or LW bodies; $\mathbf{v}_{CM_{B_i}}$ is the absolute velocity vector of the center of mass of B_i; ${}^N\boldsymbol{\omega}^{B_i}$ is the absolute angular velocity of B_i; and m_{B_i}, $[I_{B_i}]$ are the mass and mass inertia matrix of B_i. These vectors, as

we discussed in the previous section, can be expressed in terms of the B frame unit vectors $\hat{\mathbf{b}}_1\hat{\mathbf{b}}_2\hat{\mathbf{b}}_3$. The inertia matrix for each body also needs to be expressed in terms of these versors. The potential energy for each body is given by the gravitational field:

$$V_{B_i} = m_{B_i}\, g\, z_{CM_{B_i}} \tag{5.39}$$

where g is the gravitational acceleration and $z_{CM_{B_i}}$ is the vertical position of the center of mass coordinate along the Newtonian unit vector $\hat{\mathbf{n}}_3$ for body B_i.

In the simplest case of $RB1$ the resulting kinetic and potential energy functions are:

$$T_{RB1} = \frac{1}{2}m_B\left(\dot{x}(t)^2 + \dot{z}(t)^2\right) + \frac{1}{2}I_{Byy}\dot{\theta}(t)^2, \quad V_{RB1} = m_B\, g\, z(t) \tag{5.40}$$

Then applying Lagrange's equations on the Lagrangian function $L_{RB1} = T_{RB1} - V_{RB1}$, using $x(t)$, $z(t)$, $\theta(t)$ and $\theta_2(t)$, as generalized coordinates we get the following system:

$$\begin{bmatrix} m_B & 0 & 0 & 0 \\ 0 & m_B & 0 & 0 \\ 0 & 0 & I_{Byy} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} \ddot{x} \\ \ddot{z} \\ \ddot{\theta} \\ \ddot{\theta}_2 \end{Bmatrix} = \begin{Bmatrix} Q_{Bx} \\ Q_{Bz} \\ Q_{B\theta} \\ Q_{B\theta_2} \end{Bmatrix} + \begin{Bmatrix} 0 \\ -m_B\, g \\ 0 \\ 0 \end{Bmatrix} \tag{5.41}$$

where Q_{Bx}, Q_{Bz}, $Q_{B\theta}$ and $Q_{B\theta_2}$ are the generalized fluid forces acting on the generalized coordinates. We see that the θ_2 equation has no contributions from $RB1$ ($Q_{B\theta_2}$ is zero). The contributions to the θ_2 equation come from $RB2$. Note that for $RB2$ there is a component of the potential energy arising from a linear torsional spring K_T located at point D, which exerts a restituting moment proportional to θ_2

(figure 5.12b). A torsional damper could also be added. The computation of fluid forces and moments for each of the immersed bodies in the model is done using the approach in Chapter 2, which enables us to compute the force $\mathbf{F}_B^N = [F_{Bx}, F_{By}, F_{Bz}]^T$ and moment with respect to the center of mass, $\mathbf{M}_B^N = [M_{Bx}, M_{By}, M_{Bz}]^T$, of $RB1$ in the N frame. In terms of the B frame these forces are simply $\mathbf{F}_B^B = [T_{BN}]\,\mathbf{F}_B^N$ and $\mathbf{M}_B^B = [T_{BN}]\,\mathbf{M}_B^N$. Then, the generalized fluid forces are computed from [14]

$$Q_{Bx_j} = \mathbf{F}_B^B \cdot \frac{\partial \mathbf{v}_{C_{M_{RB1}}}^B}{\partial \dot{x_j}} + \mathbf{M}_B^B \cdot \frac{\partial^N \boldsymbol{\omega}_B^B}{\partial \dot{x_j}}, \ , \quad j = 1, 2, .., 4 \tag{5.42}$$

where x_j represents any of x, z, θ, θ_2. The resulting generalized fluid forces on $RB1$ are: $Q_{Bx} = F_{Bx}$, $Q_{Bz} = F_{Bz}$, $Q_{B\theta} = -M_{By}$, $Q_{B\theta_2} = 0$. The procedure for the equations for $RB2$, RW and LW is similar, where the corresponding velocities of the centers of mass and absolute angular velocities for each of these bodies is utilized. The problem becomes intractable to derive analytically, specially for the wings which undergo three-dimensional motion given by the beating and out of stroke plane angles ψ_3, ψ_4 and θ_3, θ_4 as well as the angles of attack ϕ_3, ϕ_4. For this reason the derivation of the equations for each body was done using the symbolic manipulation software Mathematica©.

Testing the consistency of the model

To verify the above model two test cases were setup, where the effects of the fluid flow are neglected. In the first problem the system is composed by $RB1$ and $RB2$ only. The degrees of freedom in this case are the four: x, z, θ and θ_2. This system undergoes two-dimensional motion and is equivalent to the two-link model presented in Section 5.1.1. If the inertial properties and initial conditions are selected

to be the same, then both sets of equations should provide the same response. A comparison between the two models is shown in Figure 5.13 for the following initial conditions: for the RB1-RB2 computation, $\theta_o = -0.5$, $\theta_{2o} = 1$, $x_o = x_{DB}cos(\theta_o)$ and $z_o = 0$; for two-link model, $\theta_o = -0.5 + \pi$, $\theta_{2o} = 1$, $x_o = 0$ and $z_o = \eta_b sin(\theta_o)$. As it can be seen the evolution of the resulting energies in time for the both models

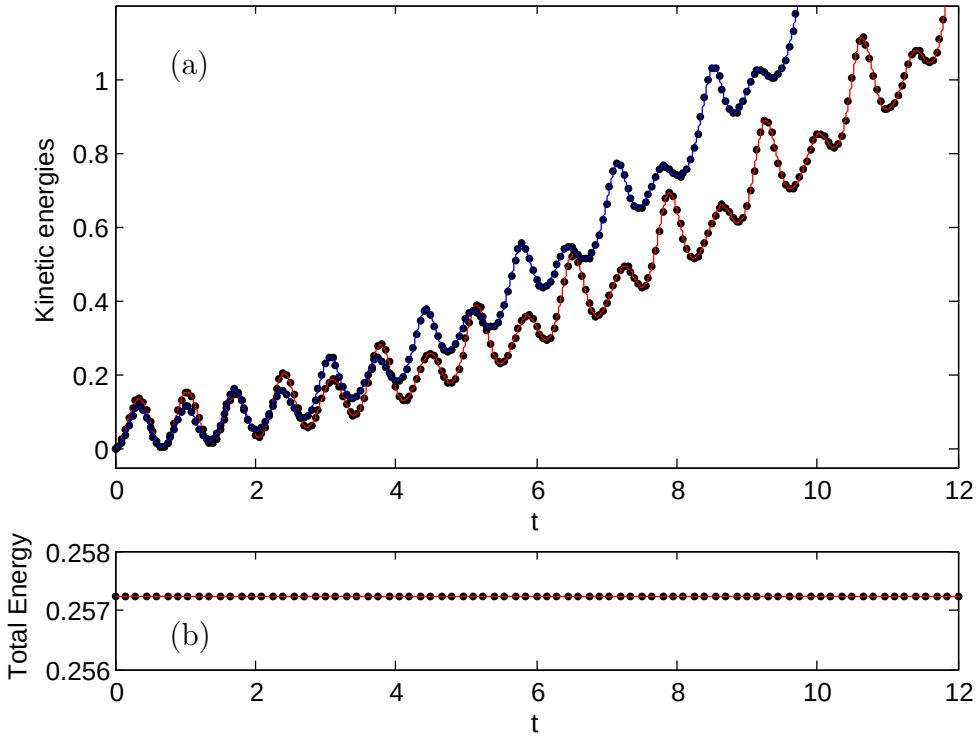


Figure 5.13: Energy plots as a function of integration time for comparison among $RB1$-$RB2$ and two-link model: (a) (blue) kinetic energy of $RB1$, (red) kinetic energy of $RB2$, (•) corresponding kinetic energies for plates B and A of the two-link model; (b) (red) total energy of the system $RB1$-$RB2$, (•) total energy of the two-link model.

match perfectly. The maximum difference for an integration of 15 computational

units (letot) was of the order of the tolerance imposed on the numerical integration (about 10^{-10}). Excellent agreement was also found on the horizontal and vertical positions of the centers of mass of each body (figure 5.14). These results confirm that the contributions of $RB1$ and $RB2$ to the four rigid body system in longitudinal dynamics are correctly computed.

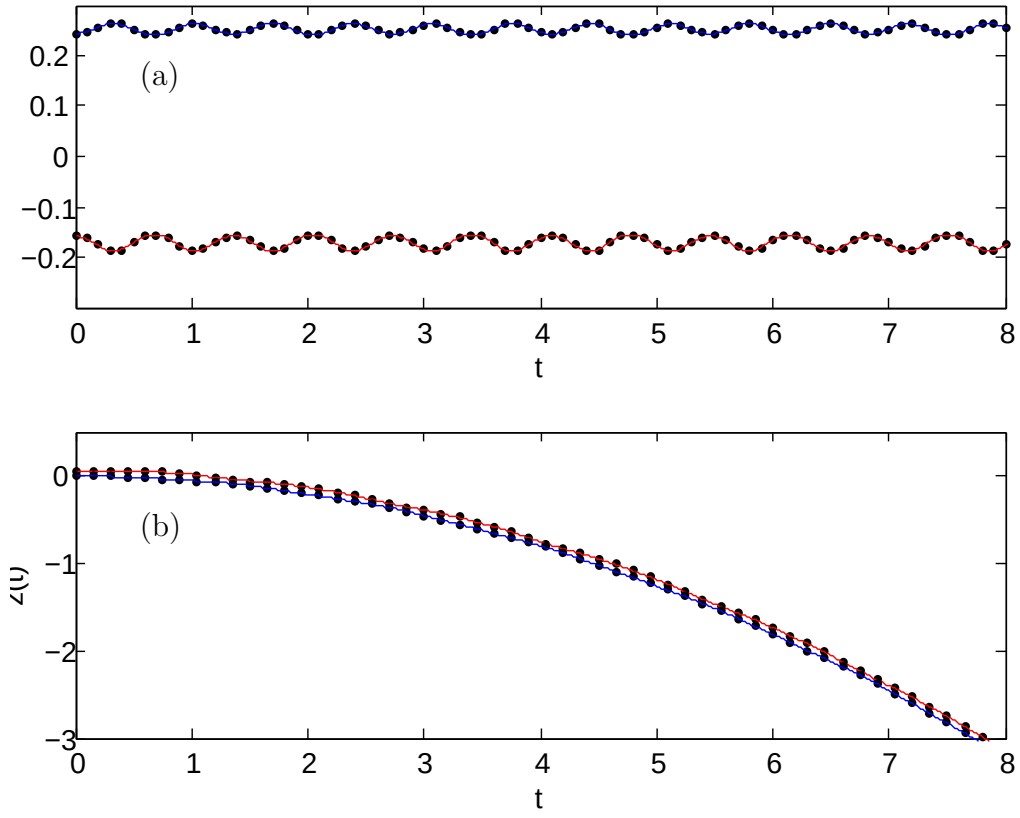


Figure 5.14: Time variation of positions of centers of mass: (a) Horizontal positions, (blue) $RB1$, (red) $RB2$, (•) corresponding positions for plates B and A of the two-link model; (b) Vertical positions, bodies are identified as in (a).

The second test was performed on the four rigid body model, using harmonic

kinematics on the wing's stroke angle and angle of attack. The oscillation amplitude, forcing frequency, and also the inertia and stiffness properties, were the ones outlined in the following sections. The equations of motion were integrated using a fourth order Runge-Kutta method in Matlab© (function ode45). A zero gravity force was prescribed. As there are no external forces applied to the system throughout the calculation the position, $x_{CM_T}, y_{CM_T}, z_{CM_T}$, of the center of mass of the four rigid body system:

$$\begin{aligned} x_{CM_T} &= \frac{m_B x_{CM_{RB1}} + m_D x_{CM_{RB2}} + m_R x_{CM_{RW}} + m_L x_{CM_{LW}}}{m_B + m_D + m_R + m_L} \\ y_{CM_T} &= \frac{m_R y_{CM_{RW}} + m_L y_{CM_{LW}}}{m_R + m_L} \\ z_{CM_T} &= \frac{m_B z_{CM_{RB1}} + m_D z_{CM_{RB2}} + m_R z_{CM_{RW}} + m_L z_{CM_{LW}}}{m_B + m_D + m_R + m_L} \end{aligned} \tag{5.43}$$

is expected to be stationary in the Newtonian frame axes, $\hat{\mathbf{n}}_1, \hat{\mathbf{n}}_2, \hat{\mathbf{n}}_3$. The time integration was performed over 20 flapping cycles and the integrator absolute error tolerance was set to 10^{-13}. The maximum variations of the $x_{CM_T}, y_{CM_T}, z_{CM_T}$ variables in time were found to be less than $2.5e^{-12}$, which is of the order of the tolerance level imposed on the numerical integration. Different values of the integration tolerance provided corresponding variation values for $x_{CM_T}, y_{CM_T}, z_{CM_T}$. This test showed that, although the generalized coordinates $x(t), z(t), \theta(t), \theta_2(t)$ of the model did undergo important variation, there is no spurious momentum introduced to the system.

5.2.4 Hovering of a *Musca Domestica* model at $Re = 500$

In this section we will present simulations for a fly model hovering at Reynolds number of 500. Both prescribed kinematics and FSI of free flight are considered. The model was created using CAD software from *Musca Domestica* digital images. The different bodies can be seen in figure 5.12. The reference length is the wingspan, b (one wing hinge to tip distance). The head-abdomen length of the model is $L_{TA} = 1.03b$. The wings have rounded edges and a thickness of $0.025b$. A simple set of wing kinematics, representative of *Diptera* wing motion is prescribed in all simulations. For the right wing, for example, we set:

$$\begin{aligned} \phi_3(t) &= A_\phi sin\left(\omega_f t + \alpha\right) \\ \theta_3(t) &= 0 \\ \psi_3(t) &= \psi_m + A_\psi cos\left(\omega_f t\right), \end{aligned} \tag{5.44}$$

where $A_\phi = \pi/4$ is the amplitude of the angle of attack, and $\alpha = \pi/6$ (advanced rotation) is the phase . The mean stroke angle is $\psi_m = -\pi/36$, and the stroke amplitude $A_\psi = 55\pi/180$. The reference velocity U_R used is the mean wing tip velocity given by $U_R = \dot{\psi}_{3mean} b$, where $\dot{\psi}_{3mean} = 2A_\psi \omega_f/\pi$. The Reynolds number is $Re = U_R b/\nu = 500$. The $RB1$ orientation angle on the prescribed kinematics simulation was set to $\theta = \pi/3$. Using a stroke angle, $\beta_3' = pi/6$, the stroke plane was aligned with the horizontal plane. The values given to the angles of the left wing were such that symmetric flapping occurs.

Prescribed kinematics simulation

First, a calculation of a tethered fly with with prescribed kinematics was conducted. The relative angle, θ_2, between $RB1$ and $RB2$ was set to zero. The model was placed in the center of a cubic domain of dimensions $7b \times 7b \times 7b$, with the origin of frame B at a distance $z = 4b$ from the bottom wall. No-slip boundary conditions were set on the x and z directions, and no penetration on the y direction, where the Poisson equation was assumed to have a periodic solution. Level 0 on the AMR grid consists of $8 \times 8 \times 8$ blocks with 16^3 cells. Three levels of refinement were used, and together with dynamic adaptivity, the cell size around all solid boundaries was $\Delta x = \Delta y = \Delta z = 0.0068b$. With this resolution approximately 150 points were clustered along each wing span. Mesh adaptation was performed every 10 timesteps. The grid was refined/derefined based on two criteria: the presence of a solid boundary and/or the magnitude of the vorticity vector on a block. A block was refined when the maximum vorticity magnitude within it was larger than $5.5U_R/b$, and a set of children where derefined when it was less than $4.8U_R/b$. The problem was integrated at constant CFL number, for a time span of four flapping cycles.

In order to examine the symmetry of the flow and corresponding loads we evaluate the maximum total force in the Eulerian reference frame. The corresponding force coefficients are defined as:

$$C_{x_i}^{max} = \frac{max\left(F_{x_iRB1}(t) + F_{x_iRB2}(t) + F_{x_iRW}(t) + F_{x_iLW}(t)\right)}{\frac{1}{2}\rho_f U_R^2 S_w} \tag{5.45}$$

where x_i, $i = 1, 2, 3$ represents the Eulerian x, y and z coordinates, and $S_w = 0.274b^2$ is the one-wing planform area. The forces $F_{x_i}(t)$ are the hydrodynamic forces for

each of the bodies in the model.

The maximum fluid force coefficients in the Eulerian frame are $C_x^{max} = 3.35$, $C_y^{max} = 5.3e^{-3}$, and $C_z^{max} = 2.13$. The fluid force on the y direction normal to the model symmetry plane is 0.25% the value of the vertical force, which demonstrates the validity of the longitudinal flight assumption for this particular set of wing kinematics. In figure 5.15 the time variation of C_x for each of the four rigid bodies is shown. Both RW and LW results are identical, and bodies $RB1$ and $RB2$ contribute a maximum C_x that is 6% of the amount produced by the wings. This

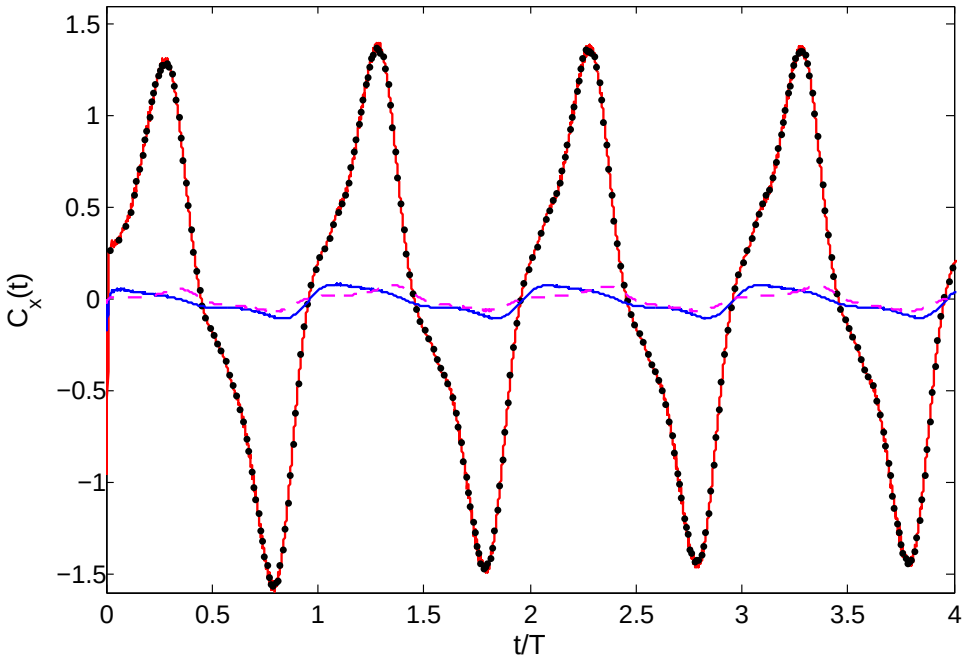


Figure 5.15: Time variation of C_x coefficient for different bodies on prescribed simulation: (blue) $RB1$, (dashed magenta) $RB2$, (red) RW, and (•) LW.

result is important since it points to the need of modeling the thorax-head and abdomen bodies in some high resolution simulations of hovering. The lift and drag

coefficients for this normal hovering simulation is computed for RW by

$$C_{L_RW} = \frac{F_{zRW}(t)}{\frac{1}{2}\rho_f U_R^2 S_w} \tag{5.46}$$

$$C_{D_RW} = -sgn\left(\dot{\psi}_3(t)\right) \frac{F_{xRW}(t)\, cos\left(\psi_3(t)\right) + F_{yRW}(t)\, sin\left(\psi_3(t)\right)}{\frac{1}{2}\rho_f U_R^2 S_w} \tag{5.47}$$

and similar expressions can be used for RW. Also the lift coefficient can be computed directly for $RB1$ and $RB2$. These results can be seen in figure 5.16. The mean lift and drag coefficients computed in periods 3 and 4 for the right wing are $\overline{C}_{L_{RW}} = 0.42$ and $\overline{C}_{D_{RW}} = 0.77$. Same coefficients were found for LW. The maximum

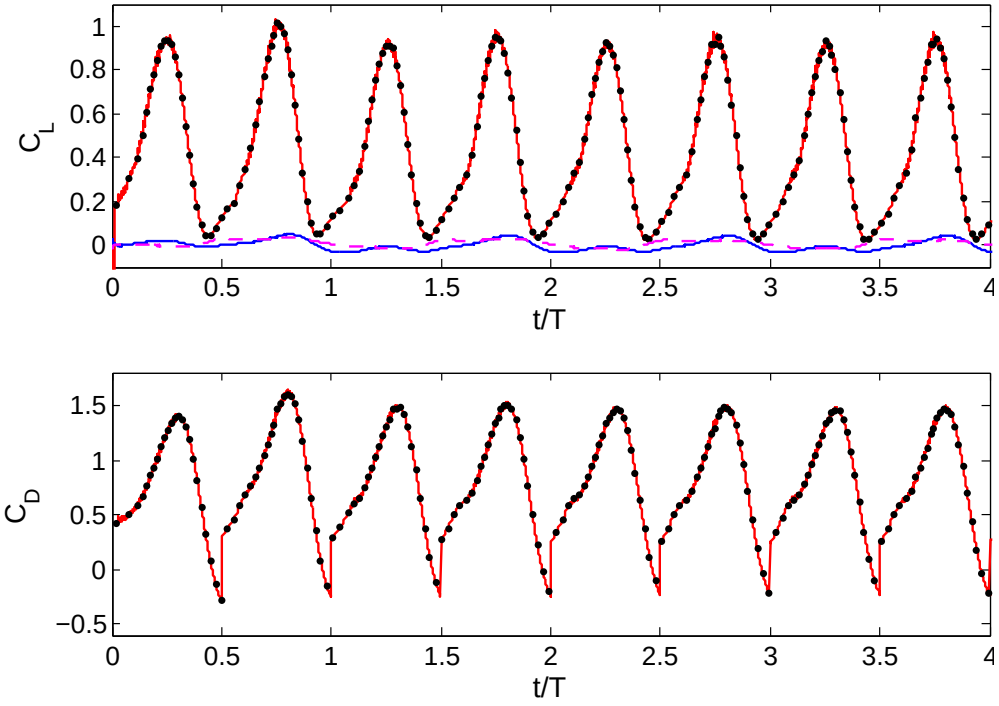


Figure 5.16: Time variation of C_L and C_D coefficients for different bodies on prescribed simulation: (blue) $RB1$, (dashed magenta) $RB2$, (red) RW, and (•) LW.

contribution to lift provided by $RB1$ and $RB2$ is about 4% of the lift given by both wings.

Free longitudinal flight simulation

To be able to conduct simulations of free flapping flight we first need to get estimates for the inertia properties for each of the rigid bodies in our model. Since there are no readily available data in the literature, we present here an approach to obtain realistic estimates for the model we use. We start by setting the ratio of inertia to aerodynamic force we expect to obtain on the course of the simulation. An estimate of this ratio is given by comparing the mean inertia force due to tangential acceleration of the wing center of mass $\overline{F}_{in} = 2m_w x_{CR} A_\psi \omega^2/\pi$ to the mean drag force $\overline{D}_w$ from the prescribed kinematics calculation above. In a manner similar to the 2D airfoil calculations presented earlier, we assume the regime is one of high fluid-structure coupling and $\overline{F}_{in}/\overline{D}_w \simeq 1.3$. The inertia effects are expected to be higher due to the contribution of the mass moments of inertia in three dimensional wing motion. From this ratio, the mean drag force coefficient previously obtained, and the wing geometry we can derive the value of $m_w = 0.2$. All inertia properties are made dimensionless using the fluid reference variables ρ_f and b. On the other hand, from the wings CAD model the volume contained by the wing can be computed. Assuming a constant density of the wing $\rho_w/\rho_f \simeq 30$ the mass estimated before can be obtained. Then, all mass inertia properties for the wings can be computed performing numerical integration along their volume

$$\begin{aligned} m_w &= \tfrac{\rho_w}{\rho_f b^3} \textstyle\int_{\Omega_w} d\Omega \\ I_{wxx} = \tfrac{\rho_w}{\rho_f b^5} \textstyle\int_{\Omega_w} (y^2+z^2)\, d\Omega \quad & I_{wyy} = \tfrac{\rho_w}{\rho_f b^5} \textstyle\int_{\Omega_w} (x^2+z^2)\, d\Omega \\ I_{wzz} = \tfrac{\rho_w}{\rho_f b^5} \textstyle\int_{\Omega_w} (x^2+y^2)\, d\Omega \qquad & I_{wxz} = \tfrac{\rho_w}{\rho_f b^5} \textstyle\int_{\Omega_w} x\ z d\Omega \end{aligned} \tag{5.48}$$

here w refers to any of RW and LW. All integrations are performed in the local

Body i	m_{B_i}	I_{B_ixx}	I_{B_iyy}	I_{B_izz}	I_{B_ixz}	x_{CB_i}	z_{CB_i}
$RB1$	2.323	–	$6.646e^{-2}$	–	–	0.0	0.0
$RB2$	1.547	–	$2.448e^{-2}$	–	–	-0.180	$-1.984e^{-2}$
RW	0.2	$1.741e^{-3}$	$1.326e^{-2}$	$1.154e^{-2}$	$3.962e^{-4}$	0.500	$-9.923e^{-2}$
LW	0.2	$1.741e^{-3}$	$1.326e^{-2}$	$1.154e^{-2}$	$3.962e^{-4}$	0.500	$-9.923e^{-2}$

Table 5.2: Dimensionless inertia properties for bodies of four rigid body model, computed in local coordinate systems. x_{CB_i} and z_{CB_i} are the local coordinates of the center of mass of body B_i respect to the origin of the local frame of reference.

coordinate system centered on the wings center of mass location. The numerical integration was performed by meshing the wings volume with linear tetrahedra elements, and performing a finite element sum of 4 point quadratures on the elements. This quadrature and the linear interpolation functions used on the spatial variables ensures that the integrations are exact on the discretized volumes. Grid refinement was performed to ensure that the inertia properties were converged up to 10^{-6}.

To define the mass of the thorax-head and abdomen we assume that one wing mass in our model is 5% of the mass of $RB1$ and $RB2$ together. Wu et. al [112] report this percentage to be 1% for Hoverflyes and 6% for the Manduca Sexta hawkmoth. Then, the density of $RB1$ and $RB2$ was found to be $\rho_B/\rho_f \simeq 50$. The computation of inertia properties for $RB1$ and $RB2$ followed the same procedure as the wings. Table 5.2 shows the final dimensionless inertia properties for the model components.

In order to complete the set of parameters required for the FSI simulation of

longitudinal flight the dimensionless field acceleration (gravity) was set to $g = 0.01$, a value that gives a weight force lower than the required to balance the mean lift force of the prescribed simulation. Thus, it is expected that the model would have an upward net motion. Also the initial conditions on the state variables were set to $x_o = 0$, $z_o = -0.5b$, $\theta_o = 90^0$, $\theta_{2o} = 0$, and zero velocities. A unit dimensionless torsion stiffness $K_T = 1$ was employed at the abdomen hinge. This was seen to maintain the relative angle between $RB1$ and $RB2$ within a 5^o amplitude. An initial attempt using the kinematics of the previous prescribed motion simulation, where the mean stroke angle is 5^o behind the $\hat{\mathbf{b}}_\mathbf{2}$ axis, gave a large pitch down acceleration, due to the moment imbalance with respect to y. In the simulation we present below, we changed the mean stroke angle to $\psi_m = 5.5\pi/180$, advanced respect to the $\hat{\mathbf{b}}_\mathbf{2}$ axis. In this case a slight pitch up motion was found along the simulation. It is important to note that, in several flapping wing systems, both hovering and forward flight have been found to be dynamically unstable through linear stability analysis (i.e. [94], [97]). Given the system of aerodynamic forces produced through flapping, and regardless how balanced the mean force and moment equations are initially, it is expected that the solution will eventually diverge as the integration progresses.

In figure 5.17 the values of the state variables $x(t)$, $y(t)$, $\theta(t)$ and $\theta_2(t)$ as a function of integration time are shown. The vertical position $z(t)$ is seen to increase throughout the calculation consistent with the fact that the mean resulting vertical force is directed upwards. The horizontal coordinate of the center of mass of $RB1$ takes oscillating positive values, which diminish as the calculation progresses. This is due to the fact that the orientation angle $\theta(t)$ of $RB1$ increases with time and so

does the stroke plane angle $\theta(t) + \beta_i'$, introducing a component of the force normal to the stroke plane angle (lift in the prescribed simulation) along the negative x direction. The oscillatory component of $x(t)$ is about $0.1b$, which is consistent to the data reported in [112] for a *Manduca Sexta* hawkmoth model with similar wing to body mass ratio. The orientation angle $\theta(t)$ increases steadily, due to moment imbalance. A lower value of ψ_m is required to reduce this effect. Also, $\theta(t)$ oscillates with a peak to peak amplitude of 4^o similarly to what is reported in reference [112].

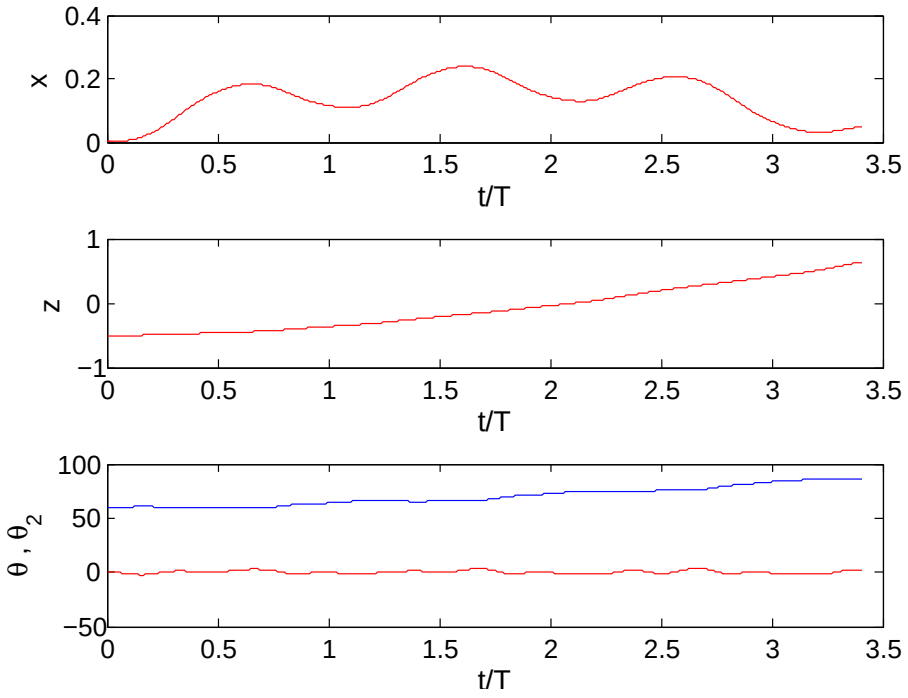


Figure 5.17: Variation of positions $x(t)$, $z(t)$ and angular coordinates $\theta(t)$ (blue), $\theta_2(t)$ (red) with time for FSI simulation of free longitudinal flight.

In figures 5.18 the positions of the center of mass of $RB1$ are shown, together with the location of the thorax, abdomen and the orientation of the stroke plane at

different time points. The increase of the angle $\theta(t)$ as the simulation progresses is given by a pitch up momentum of the force normal to the stroke plane. Aerodynamic forces are follower forces in the sense that they are always aligned (i.e. are normal or tangent) to the aerodynamic surfaces, and their dynamic effect is in general destabilizing. This fact points to the need of developing control strategies using some of the wing kinematics parameters described (i.e. ψ_m, α) in order to minimize pitch oscillations or divergence.

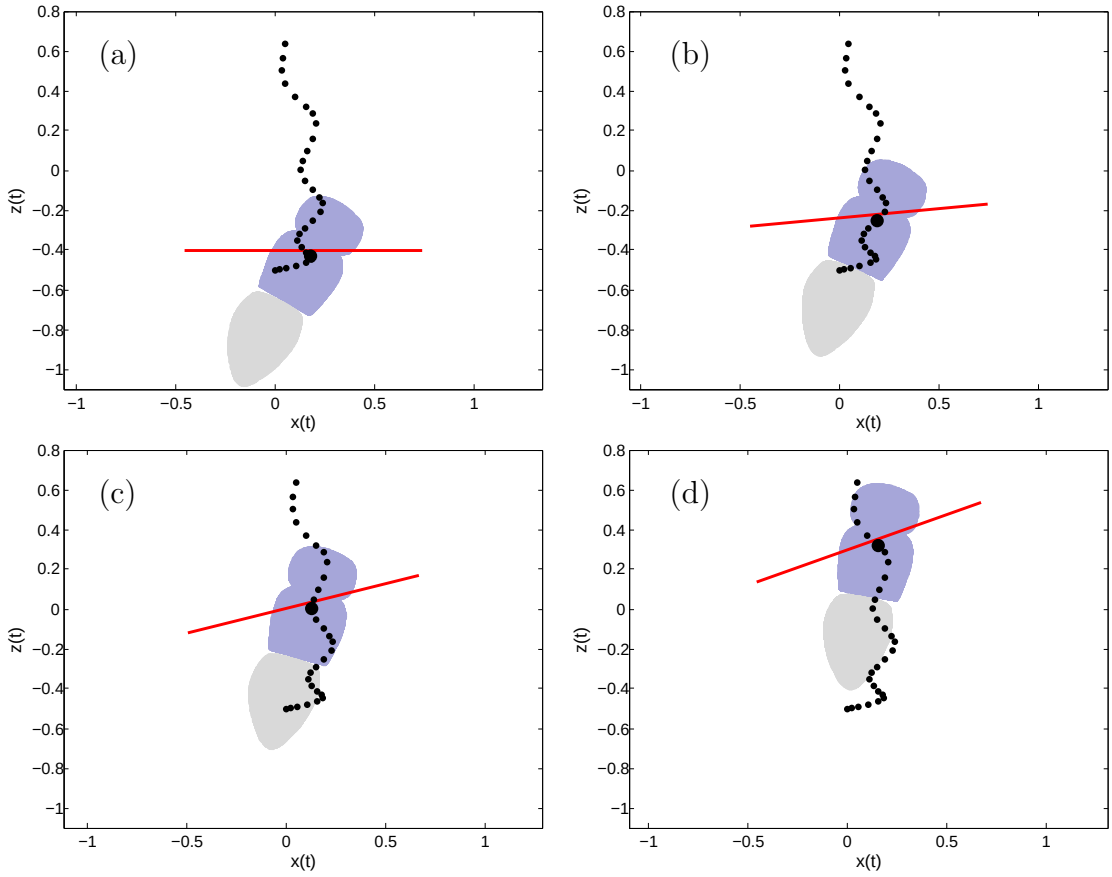


Figure 5.18: Variation of position of the center of mass of $RB1$ with time. The model section on the plane $\hat{\mathbf{n}}_\mathbf{1} - \hat{\mathbf{n}}_\mathbf{3}$ is shown along with the orientation of the stroke plane (in red) for: (a) $t/T = 0.73$, (b) $t/T = 1.38$, (c) $t/T = 2.13$, and (d) $t/T = 2.80$.

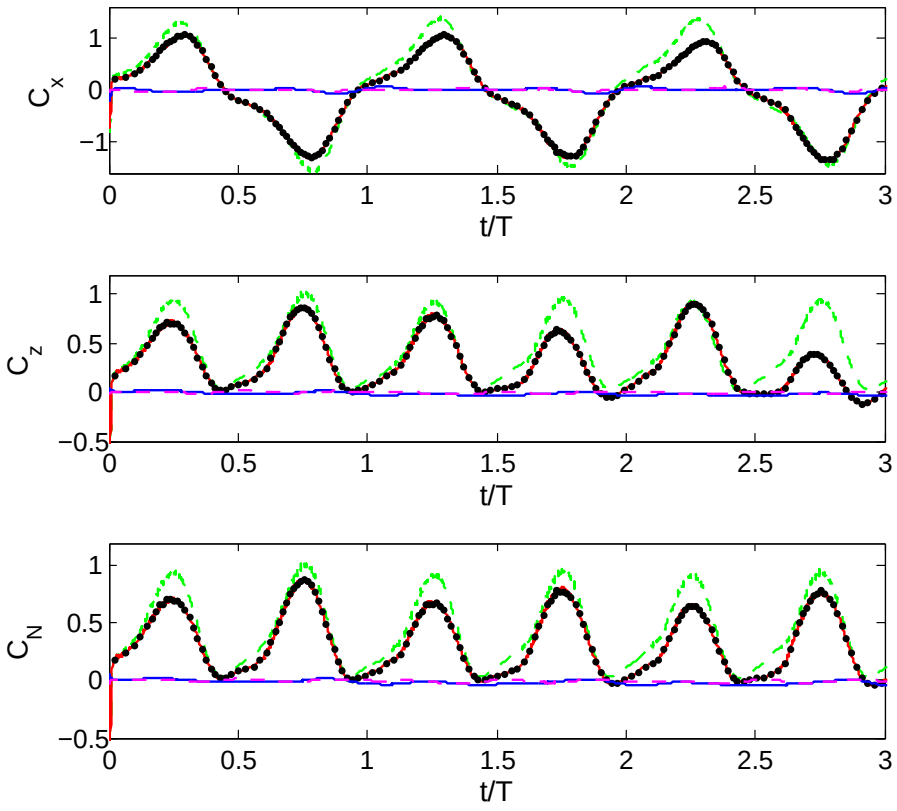


Figure 5.19: Variation of force coefficients $C_x(t)$, $C_z(t)$, and $C_N(t)$ (force normal to stroke plane) with time for FSI calculation of flapping wing model in free longitudinal flight: (blue) $RB1$, (dashed magenta) $RB2$, (red) RW, and (•) LW. The green curves correspond to $C_x(t)$, $C_z(t)$, and $C_L(t)$ from the prescribed kinematics simulation.

In figures 5.19 the time variation of force coefficients in the horizontal x, and vertical z directions, and also in a direction normal to the stroke plane (lift direction for the prescribed kinematics simulation) are shown. We see that the fluid forces computed for RW and LW are again equal. Also, as in the prescribed kinematics simulation the force in the y direction was found to be negligible with respect to

the other components. The maximum force on the x direction provided by $RB1$ and $RB2$ is 8% the force on the wings. It is seen that, due to the stroke plane rotation, C_x and C_z vary significantly as with time. The coefficients resulting from the prescribed kinematics calculation are plotted on the same figures for comparison. All force coefficients take lower values than the fixed body calculation. This is due to the fact that body motion lowers the ability of the wings to transfer momentum to the fluid (see also [112]).

The instantaneous flow-fields at different simulation times are seen in figures 5.20. Here, an isocontour of Q colored by the vorticity ω_y shows the evolution of the fundamental flow structures, namely leading edge and tip vortices. In figures 5.20a-b we see the leading edge vortices attached to each wing, which detach in the vicinity of the wing tips. Here vorticity is reoriented forming the wing tip vortices. Vorticity is also shed from the regions of the wing proximal to the bodies, and therefore, two vortical structures are being generated on each wing (figures 5.20c-d). This secondary vortices are dependent on the planform geometry of the wings used. The vertical displacement of the model due to the imbalance in the lift force and its own weight can be clearly seen in the sequence of instantaneous snapshots in figures 5.20.

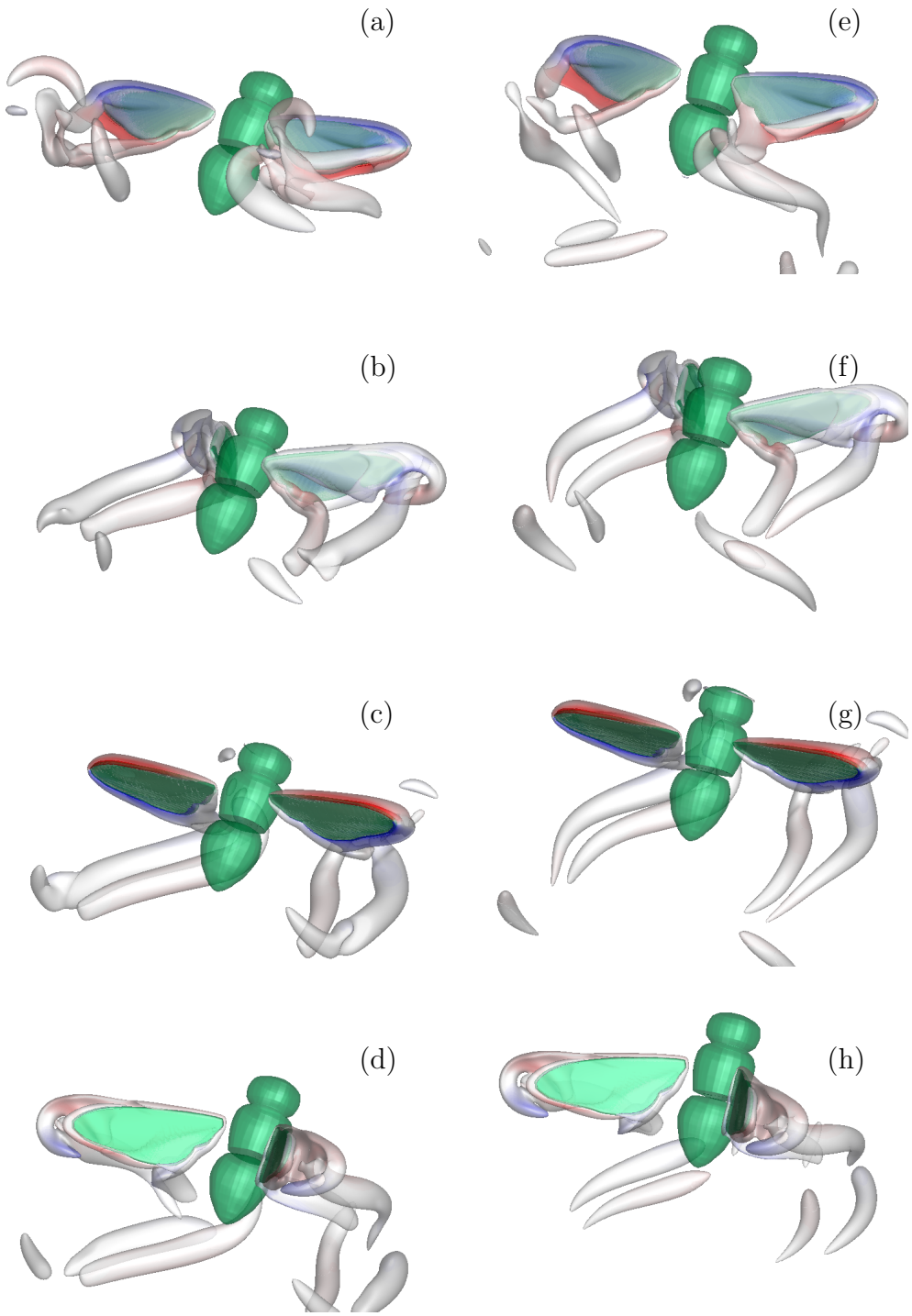


Figure 5.20: Q isocontour colored by vorticity on the y direction. 40 contours of ω_y from -20 to 20 are used. (a)-(h): $t/T = 0.75$, 1, 1.25, 1.5, 1.75, 2, 2.25 and 2.5.

Chapter 6

Summary, Contributions and Directions for Future Work

In this work, an adaptive mesh refinement, large eddy simulation strategy has been developed for fluid-structure interaction problems in transitional and turbulent flow regimes. An immersed boundary reconstruction procedure to represent the moving and/or deforming bodies immersed in the flow has been proposed. The overall method is a generalization of the formulation initially proposed by Uhlmann [101], where the main difference with existing direct-forcing schemes (i.e. [37], [50], [9]) is that the evaluation of the forcing function is done on the Lagrangian markers instead of the Eulerian points. The main advantages of this strategy compared to existing direct-forcing schemes can be summarized as follows:

i) It is more versatile since it decouples the local discretization from the computation of the forcing function and, therefore, can be implemented into structured or unstructured codes in a straightforward manner. Most of the available direct-forcing schemes have been developed in the framework of finite-difference or finite-volume formulations on Cartesian grids. The proposed scheme can be used with other spatial discretization approaches (i.e., finite elements).

ii) It is very robust in dealing with contact and collisions of multiple bodies. The forcing function is built and appropriately scaled based on the contributions of all bodies in the vicinity of an Eulerian point without any special treatment. In

most direct-forcing methods, the presence of two or more Lagrangian markers from different bodies in the proximity of the same Eulerian grid point is usually the source of ambiguity.

The immersed boundaqry method was also found to maintain the second-order spatial accuracy of the underlying finite-difference solver. Most importantly, it has been demonstrated that when combined with the scheme that has been proposed to compute the surface forces, it has a sharp-like behavior similar to sharp Eulerian direct-forcing schemes and boundary conforming methods. The overall computational cost of the proposed formulation is comparable to other direct forcing approaches available in the literature (i.e. [9], [50]). In the case of stationary bodies, the interface tracking step, as well as the computations of the shape functions can be carried out in a pre-processing module and stored in memory. Then, the computational effort associated with the forcing step is reduced to a small fraction of the overall cost. In the case of moving bodies the operations described in Section 2.3.1 need to be performed at each timestep. The overall cost depends on the total number of Lagrangian markers and the size of the Eulerian grid. The tracking step in the current formulation, is probably less expensive than other direct-forcing schemes since only the closest point to each marker needs to be identified. The forcing step on the other hand, is more expensive than typical direct-forcing schemes since the shape function computation involves the solution of a 4×4 linear system (equation 2.11). However, like all other direct-forcing schemes, the cost per Lagrangian marker is constant. Therefore, the overall cost is proportional to the number of Lagrangian

markers.

To extend the range of applicability of immersed boundary methods such as the one proposed above, to high Reynolds number complex flows, the author has also developed a structured AMR method. In this approach, a single-block solver is employed on a hierarchy of sub-grids with varying spatial resolutions. Each of these sub-grid blocks has a structured Cartesian topology, and each block is part of a tree data structure that covers the entire computational domain. Time advancement is done by using a fractional step method. All spatial derivatives are approximated with second order finite-differences on a staggered grid. The Paramesh toolkit [62] is utilized to keep track of the grid hierarchy, and perform the required restriction/prolongation and guard-cell filling operations. The author has demonstrated that the accuracy of dynamic AMR is greatly affected by the conservation properties of the prolongation and restriction operators. The author has developed a divergence-preserving prolongation operator tailored to the specific AMR topology, where the grid size between consecutive refinement levels can only differ by a factor of two. Overall, the second-order spatial and temporal accuracy of the basic solver are maintained. The computational efficiency of the proposed formulation is a balancing act between the lower CPU and memory cost incurred bu using AMR, and the additional work derived from the augmented computational complexity. In general, the proposed solver has a significant advantage over single block solvers, in high Reynolds number FSI problems, where the fine grid patches needed to resolve the boundary layers on the body surfaces have to be constantly rearranged following the motion, For the falling plates simulation of Section 3.2.3, for example, the

number of grid points utilized in the AMR computation is 1/10 of the ones used in the uniform grid solution. In terms of wall time, the AMR calculation took about 3 seconds per timestep on a single processor, while the equivalent uniform grid run required roughly twice that amount. It is also noted that the uniform grid solver utilizes a direct solver for the Poisson equation (i.e. [9]), which is much more efficient compared to the multigrid solver. If a multigrid solver is to be adopted for the case of the uniform grid too, then, the computational savings by using AMR would have been an order of magnitude higher.

In general, it is found for projection schemes for the Navier-Stokes equations that the Poisson equation solution takes a substantial part of the computing cost. The problem is particularly acute for parallel multigrid solution schemes on *octree* meshes, where sequential relaxation across refinement levels requires extensive data communication among processors. Also, work load balancing among processors becomes a very difficult task. The use of a hybrid multigrid and a highly parallel direct solver for the coarse grid solution, together with sizable meshes on the coarsest level, is found to improve the computation times on some problems, for example, the computations of flow around a sphere. It is believed that highly scalable Poisson solvers for AMR grid structures like the one employed in this work, need to be developed to perform the next generation of numerical simulations.

The author has also shown that the proposed AMR scheme is well suited for carrying out LES studies. In Chapter 4, we performed numerical simulations of spatially developing homogeneous isotropic turbulence, convected through an interface where the grid is suddenly coarsened or refined by a factor of two in each

direction. The author has compared calculations where the filter width is proportional to the grid size and is discontinuous at the coarse-fine grid interface, with cases in which the filter-width varies gradually between the values corresponding to the coarse and fine grids. Both the Smagorinsky and the Lagrangian-Dynamic Eddy viscosity (LDEV) SGS models have been used. In addition, the explicit filtering of the advective term in conjunction with the application of the LDEV model has been evaluated. A sudden refinement of the grid does not result in a significant flow perturbation: small scales are gradually generated downstream of the interface, resulting in a smooth flow across the interface. When the grid is suddenly coarsened, on the other hand, a considerable energy pileup at small scales is observed near the interface. For coarse-fine interfaces a discontinuous filter width gives more accurate results than a smoothly decreasing one: decreasing the eddy viscosity allows small eddies to be generated more rapidly, while the smoothly varying filter gives increased eddy viscosity on the fine-grid side, which delays the generation of small scales. When the grid is suddenly coarsened, on the other hand, an increase of the eddy viscosity upstream of the interface through a smooth increase of the filter-width is found to be beneficial. For fine-to-coarse interfaces, the LDEV model with a smoothly varying interface is found to give more accurate results than the Smagorinsky model, and the flow transitions to the single-grid results within one integral scale L_{11}. The interface effect is inherently related to the amount of eddy viscosity modeling employed in the simulations. As the Reynolds number Re_{λ_f} is increased, for the fine-coarse transition, the intensity and extension of the energy concentration zone on the fine side is magnified. Correspondingly, on the coarse-fine

situation, the distance required for the restoration of equilibrium is enhanced as the resolution becomes coarser.

Explicit filtering has the the ability to decrease the high wavenumber energy content of the flow prior to the interface, and this improves the transition across the grid discontinuities. This strategy works very well with the present AMR scheme, and was found to give excellent results for the case of the flow around a sphere at $Re = 10^4$. The fact that the LDEV model, which has some memory effects, gives better results than the Smagorinsky one indicates that, perhaps models that include a transport equation (which also would include memory effects) may be beneficial. Further investigation of other LES modeling strategies for AMR is left as a potential future work direction. Compared to available structured [64] or unstructured AMR [45] formulations, the constraint that neighboring blocks can only differ by one level of refinement may result in larger overall grids to achieve the same local resolution, especially in internal flows. This is not a major concern in LES studies of turbulent and transitional flows, where grid discontinuities generated by neighboring blocks that differ by more than a factor of two can contaminate the high frequency content and unphysically enhance turbulent fluctuations on the fine grid side; therefore, they are not desirable.

In Chapter 5, the numerical tools were applied to study and simulate problems relevant to flapping wing systems at low Reynolds numbers regimes. First, the influence of flexibility on the aerodynamic performance of a two-dimensional hovering wing section was numerically studied. Here, the wing model consists of two rigid links that are joined at the center with a linear torsion spring. By prescribing

the kinematics of the top link, the structural system is effectively reduced to a single degree-of-freedom non-linear oscillator. The results obtained demonstrate that flexibility can be beneficial in terms of enhancing the aerodynamic performance. Furthermore, it has been found that in the frequency range below the first natural frequency, the best performance is achieved when the wing is driven at a frequency close to one of the non-linear resonances (a superharmonic resonance of order three) of the system. This behavior is common to all of the Reynolds numbers studied. In terms of the flow physics, the wake capture mechanism is enhanced partially due to a stronger flow around the wing at stroke reversal. However, it needs to be noted that the cases where the wing is driven at or close to the first natural frequency of the system were not considered in this study, and it is possible that a better aerodynamic performance may be achieved at the fundamental resonance and this remains to be explored. The study also leads to the following open questions:

i) Why is there a performance enhancement when the system is excited at a flapping wings non-linear resonance and would one achieve a better performance with a non-linear resonance compared with a linear resonance?

ii) Which kinematics is preferable from an aerodynamic efficiency standpoint?

The interplay between wing flexibility and kinematics together with qualitative changes in the system dynamics as a function of the Re number requires further investigation, and this is left as a direction for future research.

A four rigid body, flapping wing model has also been developed. The model is composed of the thorax-head, abdomen, and a pair of wings. The kinematics for the

general motion in three dimensions has been studied, and the equations of motion have been derived for the simple case of free longitudinal flight. A simple strategy to estimate the inertia properties of each of the components has been developed. Both all prescribed kinematics and FSI simulations of longitudinal flight were performed, showing that the proposed numerical methodology is well suited to approach these complex problems. It is believed that this work will serve as a basis for more in-depth studies into dynamics, stability and control in free flight regimes.

Appendix A

Divergence preserving prolongation in three dimensions

The development of divergence preserving operators in three-dimensions is similar to the one for two-dimensions outlined in section 3.1.2. In particular, the discrete divergence of the coarse-grid cell shown in Fig. A.1 is:

$$\frac{u^l_{i,j,k} - u^l_{i-1,j,k}}{\Delta x^l} + \frac{v^l_{i,j,k} - v^l_{i,j-1,k}}{\Delta y^l} + \frac{w^l_{i,j,k} - w^l_{i,j,k-1}}{\Delta z^l} = D\mathbf{u} \tag{A.1}$$

To interpolate the velocities on the cell faces, the one-dimensional interpolations in two-dimensions, are now replaced with two-dimensional quadratic interpolations assuming a polynomial variation of the form:

$$u(\xi,\eta) = a_0 + a_1\xi + a_2\eta + a_3\xi\eta + a_4\xi^2 + a_5\eta^2 + a_6\xi^2\eta + a_7\xi\eta^2 + a_8\xi^2\eta^2 \tag{A.2}$$

An example stencil is shown in figure A.1a, and a system analogous to (3.4) can be constructed by applying Eq. (A.2) in the $(\xi_{i+\alpha}, \eta_{j+\beta})$, $\alpha, \beta = -1, 0, 1$ positions at level l, except for (ξ_i, η_j). Then the known value of the coarse grid variable at location (ξ_i, η_j) is used in

$$u^l_{i,j,k} = \frac{1}{4}\left(u^{l+1}_{i',j',k'} + u^{l+1}_{i',j'+1,k'} + u^{l+1}_{i',j'+1,k'+1} + u^{l+1}_{i',j',k'+1}\right), \tag{A.3}$$

and the four fine grid values are computed by interpolations from Eq. (A.2). The matrix associated to the resulting system can be inverted using symbolic manipulation software, such that the computation of the interpolation coefficients a_i, $i = 0, 1, \ldots, 8$

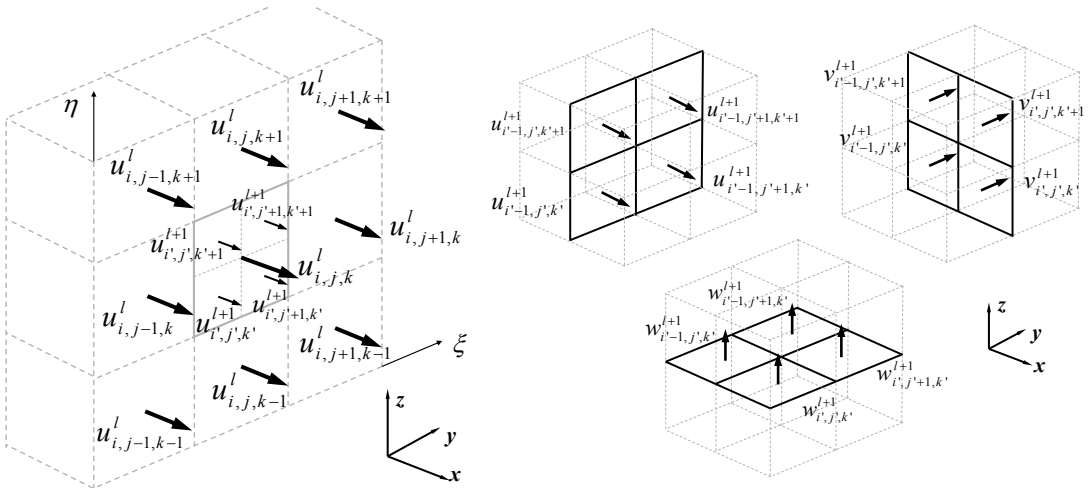


Figure A.1: (a) Interpolation stencil for 2D face interpolations used in three dimensional divergence preserving prolongation. (b) Internal variables to a given coarse-grid cell indexed by i, j, k.

will require a 9×9 matrix-vector multiplication. The computation of the four fine grid variables requires four additional 9×1 vector-vector multiplications.

Once the velocity field has been found on the cell faces, the velocities internal to the coarse cell need to be obtained. There are twelve internal fine-grid variables, $u^{l+1}_{i'-1,j',k'}$, $u^{l+1}_{i'-1,j'+1,k'}$, $u^{l+1}_{i'-1,j',k'+1}$, $u^{l+1}_{i'-1,j'+1,k'+1}$, $v^{l+1}_{i',j',k'}$, $v^{l+1}_{i'-1,j',k'}$, $v^{l+1}_{i',j',k'+1}$, $v^{l+1}_{i'-1,j',k'+1}$, $w^{l+1}_{i',j',k'}$, $w^{l+1}_{i'-1,j',k'}$, $w^{l+1}_{i'-1,j'+1,k'}$ and $w^{l+1}_{i',j'+1,k'}$, that need to be determined (see Fig. A.1b). Four of the twelve ($u^{l+1}_{i'-1,j',k'}$, $u^{l+1}_{i'-1,j'+1,k'+1}$, $v^{l+1}_{i'-1,j',k'}$, $v^{l+1}_{i',j',k'+1}$), can be found using one dimensional quadratic interpolation from the corresponding coarse-face variables, and for the remaining ones, eight discrete divergence equations are used as in the two-dimensional example. The resulting system can be written

as:

$$\begin{bmatrix} 0 & 0 & 0 & 0 & \gamma & 0 & 0 & 0 \\ 0 & 0 & \beta & 0 & 0 & \gamma & 0 & 0 \\ 0 & \alpha & 0 & 0 & 0 & 0 & \gamma & 0 \\ 0 & -\alpha & -\beta & 0 & 0 & 0 & 0 & \gamma \\ \alpha & 0 & 0 & \beta & -\gamma & 0 & 0 & 0 \\ -\alpha & 0 & 0 & 0 & 0 & -\gamma & 0 & 0 \\ 0 & \alpha & 0 & -\beta & 0 & 0 & -\gamma & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\gamma \end{bmatrix} \begin{Bmatrix} u^{l+1}_{i'-1,j',k'+1} \\ u^{l+1}_{i'-1,j'+1,k'} \\ v^{l+1}_{i',j',k'} \\ v^{l+1}_{i'-1,j',k'+1} \\ w^{l+1}_{i'-1,j',k'} \\ w^{l+1}_{i',j',k'} \\ w^{l+1}_{i'-1,j'+1,k'} \\ w^{l+1}_{i',j'+1,k'} \end{Bmatrix} = \begin{Bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \\ b_8 \end{Bmatrix} \tag{A.4}$$

where $\alpha = 1/\Delta x^{l+1}$, $\beta = 1/\Delta y^{l+1}$ and $\gamma = 1/\Delta z^{l+1}$. The coefficients in the right hand side are given by

$$\begin{aligned}
b_1 &= D\mathbf{u} - \frac{u^{l+1}_{i'-1,j',k'} - u^{l+1}_{i'-2,j',k'}}{\Delta x^{l+1}} - \frac{v^{l+1}_{i'-1,j',k'} - v^{l+1}_{i'-1,j'-1,k'}}{\Delta y^{l+1}} + \frac{w^{l+1}_{i'-1,j',k'-1}}{\Delta z^{l+1}} \\
b_2 &= D\mathbf{u} - \frac{u^{l+1}_{i',j',k'} - u^{l+1}_{i'-1,j',k'}}{\Delta x^{l+1}} + \frac{v^{l+1}_{i',j'-1,k'}}{\Delta y^{l+1}} + \frac{w^{l+1}_{i',j',k'-1}}{\Delta z^{l+1}} \\
b_3 &= D\mathbf{u} + \frac{u^{l+1}_{i'-2,j'+1,k'}}{\Delta x^{l+1}} - \frac{v^{l+1}_{i'-1,j'+1,k'} - v^{l+1}_{i'-1,j',k'}}{\Delta y^{l+1}} + \frac{w^{l+1}_{i'-1,j'+1,k'-1}}{\Delta z^{l+1}} \\
b_4 &= D\mathbf{u} - \frac{u^{l+1}_{i',j'+1,k'}}{\Delta x^{l+1}} - \frac{v^{l+1}_{i',j'+1,k'}}{\Delta y^{l+1}} + \frac{w^{l+1}_{i',j'+1,k'-1}}{\Delta z^{l+1}} \\
b_5 &= D\mathbf{u} + \frac{u^{l+1}_{i'-2,j',k'+1}}{\Delta x^{l+1}} + \frac{v^{l+1}_{i'-1,j'-1,k'+1}}{\Delta y^{l+1}} - \frac{w^{l+1}_{i'-1,j',k'+1}}{\Delta z^{l+1}} \\
b_6 &= D\mathbf{u} - \frac{u^{l+1}_{i',j',k'+1}}{\Delta x^{l+1}} - \frac{v^{l+1}_{i',j',k'+1} - v^{l+1}_{i',j'-1,k'+1}}{\Delta y^{l+1}} - \frac{w^{l+1}_{i',j',k'+1}}{\Delta z^{l+1}} \\
b_7 &= D\mathbf{u} + \frac{u^{l+1}_{i'-2,j'+1,k'}}{\Delta x^{l+1}} - \frac{v^{l+1}_{i'-1,j'+1,k'+1}}{\Delta y^{l+1}} - \frac{w^{l+1}_{i'-1,j'+1,k'+1}}{\Delta z^{l+1}}
\end{aligned} \tag{A.5}$$

$$\begin{aligned}
b_8 &= D\mathbf{u} - \frac{u^{l+1}_{i',j'+1,k'+1} - u^{l+1}_{i'-1,j'+1,k'+1}}{\Delta x^{l+1}} - \frac{v^{l+1}_{i',j'+1,k'+1} - v^{l+1}_{i',j',k'+1}}{\Delta y^{l+1}} \\
&- \frac{w^{l+1}_{i',j'+1,k'+1}}{\Delta z^{l+1}}
\end{aligned}$$

As this system is of rank 7, one more equation similar to (3.6) is added. As in the two-dimensional case, the left-pseudo-inverse of the resulting 9×8 coefficient matrix A_e can be computed symbolically. The implementation requires additional 8×9 matrix-vector operations to obtain the unknown variables. It is possible to reduce the size of system (A.4), by replacing the values for $u^{l+1}_{i'-1,j',k'+1}$ and $w^{l+1}_{i',j'+1,k'}$ given by the first and last equations, on the remainder ones. Then, the condensed system is of size 6×6 and still requires an extra equation to be full rank. In this case the computer implementation would require a smaller 6×7 matrix-vector multiplication to obtain the rest of the fine-grid variables.

Bibliography

[1] Achenbach, E. Experiments on flow past spheres at very high reynolds-numbers. *J. Fluid. Mech. 54* (1972), 565–575.

[2] Almgren, A., Bell, J., Colella, P., Howell, L., and Welcome, M. A conservative adaptive projection method for the variable density incompressible navier-stokes equations. *Journal of Computational Physics 142*, 1 (Jan 1998), 1–46.

[3] Almgren, A. S., Bell, J. B., and Crutchfield, W. Y. Approximate projection methods: Part I. Inviscid analysis. *SIAM Journal on Sci. Comput. 22* (2000), 1139–1159.

[4] Andersen, A., Pesavento, U., and J., W. Z. Unsteady aerodynamics of fluttering and tumbling plates. *J. Fluid Mech. 541* (2005), 65–90.

[5] Anderson, T. J., Balachandran, B., , and Nayfeh, A. H. Nonlinear resonances in a flexible cantilever beam. *J. Vibration and Acoustics 116* (1995), 480–484.

[6] Ardekani, A., and Rangel, R. Numerical investigation of particle–particle and particle–wall collisions in a viscous fluid. *J Fluid Mech* (Jan 2008).

[7] Azuma, A. *The biokinetics of flying and swimming. Second edition.* AIAA Education Series, 2006.

[8] Baaijens, F. P. T. A fictitious domain/mortar element method for fluid structure interaction. *Int. J. Num. Meth. Fluids 35* (2001), 743–761.

[9] Balaras, E. Modeling complex boundaries using an external force field on fixed cartesian grids in large-eddy simulations. *Comput Fluids 33* (Jan 2004), 375–404.

[10] Balaras, E., Piomelli, U., and Wallace, J. Self-similar states in turbulent mixing layers. *J Fluid Mech 446* (Jan 2001), 1–24.

[11] Balaras, E., and Vanella, M. Adaptive mesh refinement strategies for immersed boundary methods. *AIAA-2009-162* (Jan 2009), 9.

[12] Balsara, D. Divergence-free adaptive mesh refinement for magnetohydrodynamics. *Journal of Computational Physics 174*, 2 (Jan 2001), 614–648.

[13] Bao, L., Hu, J. S., Yu, Y. L., Cheng, P., Xu, B. Q., and Tong, B. G. Viscoelastic constitutive model related to deformation of insect wing under loading in flapping motion. *Appl. Math. Mech. 27* (2006), 741–748.

[14] Baruh, H. *Analytical Dynamics*. WCB/McGraw-Hill, 1999.

[15] Bayyuk, S., Powellt, K., and van Leer, B. A simulation technique for 2-d unsteady inviscid flows around arbitrarily moving and deforming bodies of arbitrary geometry. *AIAA-1993-3391* (1993), 1013–1024.

[16] Bell, J., Berger, M. J., Saltzman, J., and Welcome, M. Three dimensional adaptive mesh refinement for hyperbolic conservation laws. *SIAM J. Sci. Comput. 15* (1994), 1:127–138.

[17] Bell, J. B., Colella, P., and M., G. H. A second-order projection method for the incompressible Navier-Stokes equations. *J. Comput. Phys. 85* (1989), 257–283.

[18] Belmonte, A., Eisenberg, H., and Moses, E. From flutter to tumble: Inertial drag and froude similarity in falling paper. *Phys Rev Lett 81*, 2 (Jan 1998), 345–348.

[19] Berger, M. J., and Colella, P. Local Adaptive mesh refinement for shock hydrodynamics. *J. Comput. Phys. 82* (1989), 64–84.

[20] Berger, M. J., and Oliger, J. Adaptive mesh refinement for hyperbolic partial differential equations. *J. Comput. Phys. 53* (1984), 484–512.

[21] Berman, G. J., and Wang, Z. J. Energy-minimizing kinematics in hovering insect flight. *J. Fluid Mech. 582* (2007), 153–168.

[22] C., H. R. *Modelling insect wings using the finite element method*. PhD thesis, University of Exeter, 2002.

[23] Combes, S. A., and Daniel, T. L. Into thin air: contributions of aerodynamic and inertial-elastic forces to wing bending in the hawkmoth Manduca sexta. *J. Exp. Biology 206* (2003), 2999–3006.

[24] Constantinescu, G. S., and Squires, K. D. LES and DES investigations of turbulent flow over a sphere at Re=10,000. *Flow Turb. and Combustion 70* (2003), 267–298.

[25] Cubero, A., and Piomelli, U. Large-eddy simulations of channel flows with variable filter-width-to-grid-size ratios. In *Direct and Large-Eddy Simulation VI* (2006), E. Lamballais, R. Friedrich, B. J. Geurts, and O. Métais, Eds., Springer Netherlands, pp. 151–158.

[26] Del Alamo, J. C., Jimenez, J., Zandonade, P., and D., M. R. Scaling of the energy spectra of turbulent channels. *J. Fluid Mech. 500* (2004), 135–144.

[27] Dickinson, M. H., Lehmann, F. O., and Sane, S. P. Wing rotation and the aerodynamic basis of insect flight. *Science 284* (1999), 1954–1960.

[28] Doligalski, T. L., Smith, C. R., and Walker, J. D. A. Vortex interactions with walls. *Annu. Rev. Fluid Mech. 26* (1994), 573–616.

[29] Donea, J., Giuliani, S., and Halleux, J. P. An arbitrary Lagrangian-Eulerian finite element method for transient dynamic fluid-structure interactions. *Computer Meth. Appl. Mech. Engr. 33* (1982), 689–723.

[30] Dong, S., Karniadakis, G. E., and Karonis, N. T. Cross-site computations on the teragrid. *Computing in Science and Engr. 7* (2005), 14–23.

[31] Eames, I., and Dalziel, S. Dust resuspension by the flow around an impacting sphere. *J Fluid Mech 403* (Jan 2000), 305–328.

[32] Ellington, C. P. The aerodynamics of hovering insect flight. Part I: The quasi-steady analysis. *Phil. Trans. R. Soc. Lond. 305* (1984), 145–181.

[33] Ellington, C. P. The aerodynamics of hovering insect flight. Part VI: Lift and power requirements. *Phil. Trans. R. Soc. Lond. 305* (1984), 145–181.

[34] Ellington, C. P. The novel aerodynamics of insect flight: applications to micro-air vehicles. *J. Exp. Biology 202* (1999), 3439–3448.

[35] Ellington, C. P., Van den Berg, C., Willmott, A. P., and Thomas, A. L. Leading edge vortices in insect flight. *Nature 384* (1996), 626–630.

[36] Ennos, A. R. Inertial and Aerodynamic torques on the wings of Diptera in flight. *J. Exp. Biology 142* (1989), 87–95.

[37] Fadlun, E., Verzicco, R., Orlandi, P., and Mohd-Yusof, J. Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations. *Journal of Computational Physics 161*, 1 (Jan 2000), 35–60.

[38] Fry, S. N., Sayaman, R., and Dickinson, M. H. The aerodynamics of hovering flight in Drosophila. *J. Exp. Biology 208* (2005), 2303–2318.

[39] Glowinski, R., Pan, T. W., Hesla, T. I., and Joseph, D. D. A distributed Lagrange multiplier/fictitious domain method for particulate flows. *Int. J. Multiphase Flow 25* (1999), 755–794.

[40] Glowinski, R., Pan, T. W., Hesla, T. I., Joseph, D. D., and Periaux, J. A fictitious domain approach to the direct numerical simulation of incompressible viscous flow past moving rigid bodies: application to particulate flow. *J. Comput. Phys. 169* (2000), 363–426.

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

[42] Griffith, B. E., Hornung, R. D., McQueen, D. M., and Peskin, C. S. An adaptive, formally second order accurate version of the immersed boundary method. *Journal of Computational Physics 223*, 1 (Jan 2007), 10–49.

[43] Gu, W., Chyu, C., and Rockwell, D. Timing of vortex formation from an oscillating cylinder. *Phys Fluids 6*, 11 (Jan 1994), 3677–3682.

[44] Guilmineau, E., and Queutey, P. A numerical simulation of vortex shedding from an oscillating circular cylinder. *J Fluid Struct 16*, 6 (Jan 2002), 773–794.

[45] Ham, F., Lien, F., and Strong, A. A cartesian grid method with transient anisotropic adaptation. *Journal of Computational Physics 179* (Jan 2002), 469–494.

[46] Hu, H. H., Patankar, N. A., and Zhu, M. Y. Direct numerical simulations of fluid-solid systems using the arbitrary lagrangian-eulerian technique. *J Comp. Phys. 169* (2001), 427–462.

[47] Johnson, T., and Patel, V. Flow past a sphere up to a reynolds number of 300. *J Fluid Mech 378* (Jan 1999), 19–70.

[48] Joseph, G., Zenit, R., Hunt, M., and Rosenwinkel, A. Particle–wall collisions in a viscous fluid. *J Fluid Mech* (Jan 2001).

[49] Kan, J. V. A second-order accurate pressure-correction scheme for viscous incompressible flow. *SIAM Journal on Scientific and Statistical Computing* (Jan 1986).

[50] Kim, J., Kim, D., and Choi, H. An immersed-boundary finite-volume method for simulations of flow in complex geometries. *Journal of Computational Physics 171*, 1 (Jan 2001), 132–150.

[51] Kravchenko, A. G., Moin, P., and Moser, R. D. Zonal embedded grids for numerical simulations of wall-bounded turbulent flows. *J. Comput. Phys. 127* (1996), 412–423.

[52] Lancaster, P., and Salkauskas, K. Surfaces generated by moving least squares methods. *Math. Comp. 37*, 155 (1981), 141–158.

[53] Lehmann, F. O., and Dickinson, M. H. The changes in power requirements and muscle efficiency during elevated force production in the fruit fly Drosophila melanogaster. *J. Exp. Biology 200* (1997), 1133–1143.

[54] Leonard, A. Energy cascade in large-eddy simulations of turbulent fluid flows. *Adv. Geophys. 18A*, 237–248.

[55] Li, L., Sherwin, S. J., and Bearman, P. W. A moving frame of reference algorithm for fluid/structure interaction of rotating and translating bodies. *Int. J Num. Meth. in Engr. 38* (2002), 187–206.

[56] Lilly, D. K. The representation of small scale turbulence in numerical simulation experiments. In *Proceedings of the IBM Scientific Computing Symposium on Environmental Sciences* (1967), pp. 195–210.

[57] Liu, G., and Gu, Y. T. *An Introduction to Meshfree Methods and Their Programming*. Springer, Jan 2005.

[58] Liu, H., and Kawachi, K. A numerical study of insect flight. *J. Comput. Phys 146* (1998), 124–156.

[59] Lund, T. S. On the use of discrete filters for large-eddy simulation. In *Center for Turbulence Research Annual Research Briefs 1997* (1997), Stanford University, pp. 83–95.

[60] Lund, T. S. The use of explicit filters in large-eddy simulations. *Comput. Math. Appl. 46* (2003), 603–616.

[61] Lundgren, T. S. Linearly forced isotropic turbulence. In *Center for Turbulence Research Annual Research Briefs 2003*. Stanford University, Stanford CA, 2003, pp. 461–473.

[62] MacNeice, P., Olson, K. M., Mobarry, C., deFainchtein, R., and Packer, C. Paramesh: a parallel adaptive mesh refinement community toolkit. *Comput. Phys. Commun. 126* (2000), 330–354.

[63] Martin, D. F., and Cartwright, K. Solving Poisson's equation using Adaptive Mesh Refinement. *Tech. Rep. UCB/ERL M96/66* (1996).

[64] Martin, D. F., Colella, P., and Graves, D. A cell-centered adaptive projection method for the incompressible navier-stokes equations in three dimensions. *Journal of Computational Physics 227*, 3 (Jan 2008), 1863–1886.

[65] Mavriplis, D. J. *Unstructured mesh generation and adaptivity*. AGARD Publications, von Karman Institute for Fluid Dynamics Lecture Series., 1995-02.

[66] Meneveau, C., Lund, T., and Cabot, W. A lagrangian dynamic subgrid-scale model of turbulence. *J Fluid Mech 319* (Jan 1996), 353–385.

[67] Mittal, R., Dong, H., Bozkurttas, M., Najjar, F. M., Vargas, A., and von Loebbecke, A. A versatile sharp interface immersed boundary method for incompressible flows with complex boundaries. *Journal of Computational Physics 227*, 10 (Jan 2008), 4825–4852.

[68] Mittal, R., and Iaccarino, G. Immersed boundary methods. *Annu Rev Fluid Mech 37* (Jan 2005), 239–261.

[69] Mohd-Yusof, J. Combined immersed boundaries/b-splines methods for simulations of flows in complex geometries. Ctr annual research briefs, NASA Ames/Stanford University, Stanford, CA, 1997.

[70] Munk, M. M. Note on the air forces on a wing caused by pitching. NACA Technical Notes 217, 1925.

[71] Nayfeh, A. H., and Balachandran, B. *Applied Nonlinear Dynamics: Analytical, Computational and Experimental Methods*. Wiley Series in Nonlinear Science, 1995.

[72] Newman, D. J., and Karniadakis, G. E. A direct numerical simulation study of flow past a freely vibrating cable. *J Fluid Mech. 344* (1997), 95–136.

[73] Orlandi, P., and Verzicco, R. Vortex rings impinging on walls: axisymmetric and three-dimensional simulations. *J. Fluid Mech. 256* (1993), 615–646.

[74] Orlanski, I. A simple boundary condition for unbounded hyperbolic flows. *Journal of Computational Physics* (Jan 1976).

[75] Pantano, C., Deiterding, R., Hill, D. J., and Pullin, D. I. A low numerical dissipation patch-based adaptive mesh refinement for large-eddy simulation of compressible flows. *J. Comput. Phys. 221* (2007), 63–87.

[76] Peskin, C. Flow patterns around heart valves - numerical method. *Journal of Computational Physics 10*, 2 (Jan 1972), 252–&.

[77] Peskin, C. S., and McQueen, D. M. Modeling prosthetic heart-values for numerical analysis of blood-flow in the heart. *J. Comput. Phys. 37* (1980), 113–132.

[78] Peskin, C. S., and McQueen, D. M. Cardiac fluid dynamics. *Crit. Rev. Biomed. Eng. 20* (1992), 451–459.

[79] Piomelli, U., Balaras, E., and Pascarelli, A. Turbulent structures in accelerating boundary layers. *J Turbul 1* (Jan 2000), 001.

[80] Piomelli, U., Kang, S., Ham, F., and Iaccarino, G. Effect of discontinuous filter width in large-eddy simulations of plane channel flow. In *Studying turbulence using numerical databases XI* (2006), Stanford University, pp. 151–162.

[81] Piomelli, U., Moin, P., and Ferziger, J. H. Model consistency in large eddy simulation of turbulent channel flows. *Phys. Fluids 31*, 7 (1988), 1884–1891.

[82] Ramamurti, R., and Sandberg, W. A three-dimensional computational study of the aerodynamic mechanisms of insect flight. *J. Exp. Biology 205* (2002), 1507–1518.

[83] Ramamurti, R., and Sandberg, W. A computational investigation of the three-dimensional unsteady aerodynamics of Drosophila hovering and maneuvering. *J. Exp. Biology 210* (2006), 881–896.

[84] Roma, A. M., Peskin, C. S., and Berger, M. J. An adaptative version of the immersed boundary method. *J. Comput. Phys. 153* (1999), 509–534.

[85] Rosales, C., and Meneveau, C. Linear forcing in numerical simulations of isotropic turbulence: Physical space implementations and convergence properties. *Phys. Fluids 17* (2005), 5106–1–8.

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

[87] Sane, S. P. The Aerodynamics of insect flight. *J. Exp. Biology 206* (2003), 4191–4208.

[88] Sane, S. P., and Dickinson, M. H. The control of lift force by a flapping wing: Lift and drag production. *J. Exp. Biology 204* (2001), 2607–2626.

[89] Sane, S. P., and Dickinson, M. H. The aerodynamic effects of wing rotation and a revised quasi-steady model for flapping flight. *J. Exp. Biology 205* (2002), 1087–1096.

[90] Sarghini, F., Piomelli, U., and Balaras, E. Scale-similar models for large-eddy simulations. *Phys Fluids 11* (Jan 1999), 1596–1607.

[91] Shyy, W., Berg, M., and D., L. Flapping and flexible wings for biological and micro air vehicles. *Prog. Aer. Sc. 35* (1999), 455–505.

[92] Smagorinsky, J. General circulation experiments with the primitive equations. i. the basic experiment. *Mon. Weather Rev. 91* (1963), 99–164.

[93] Sun, M., and Tang, J. Unsteady aerodynamic force generation by a model fruit fly wing in flapping motion. *J. Exp. Biology 205* (2002), 55–70.

[94] Sun, M., and Xiong, Y. Dynamic flight stability of a hovering bumblebee. *J Exp. Biol. 208* (2005), 447–459.

[95] Swartzrauber, P. N. A direct method for the discrete solution of separable elliptic equations. *S.I.A.M. J. Nume. Anal. 11* (1974), 1136–1150.

[96] Swearingen, J. D., Crouch, J. D., and Handler, R. A. Dynamics and stability of a vortex ring impacting a solid boundary. *J. Fluid Mech. 297* (1995), 1–28.

[97] Taylor, G. K., and Thomas, A. L. R. Dynamic flight stability in the desert locust schistocerca gregaria. *J Exp. Biol. 206* (2003), 2803–2829.

[98] Tezduyar, T. E. Finite Element methods for flow problems with moving boundaries and interfaces. *Arch. Comput. Methods Engr. 8:2* (2001), 83–130.

[99] Tullio, M. D. D., Cristallo, A., Balaras, E., and Verzicco, R. Direct numerical simulation of the pulsatile flow through an aortic bileaflet mechanical heart valve. *J Fluid Mech 622* (Jan 2009), 259–290.

[100] Udaykumar, H., Shyy, W., and Rao, M. Elafint: A mixed eulerian-lagrangian method for fluid flows with complex and moving boundaries. *Int J Numer Meth Fl 22*, 8 (Jan 1996), 691–712.

[101] Uhlmann, M. An immersed boundary method with direct forcing for the simulation of particulate flows. *Journal of Computational Physics 209*, 2 (Jan 2005), 448–476.

[102] van der Bos, F., and Geurts, B. J. Commutator errors in the filtering approach to large-eddy simulation. *Phys. Fluids 17*, 035108 (2005), 1–20.

[103] Vasilyev, O., Lund, T., and Moin, P. A general class of commutative filters for les in complex geometries. *Journal of Computational Physics 146*, 1 (Jan 1998), 82–104.

[104] Walker, J. D. A., Smith, C. R., Cerra, A. W., and Doligalski, T. L. The impact of a vortex ring on a wall. *J. Fluid Mech. 181* (1987), 99–140.

[105] Wang, Z. J. Two dimensional mechanism for insect hovering. *Phys. Rev. Lett. 85* (2000), 2216–2219.

[106] Wang, Z. J. Vortex shedding and frequency selection in flapping flight. *J. Fluid Mech. 410* (2000), 323–341.

[107] Wang, Z. J., Birch, J. M., and Dickinson, M. H. Unsteady forces and flows in low Reynolds number hovering flight: two-dimensional computations vs. robotic wing experiments. *J. Exp. Biology 207* (2004), 449–460.

[108] Willmott, A. P., and Ellington, C. P. The Mechanics of flight in the hawkmoth *Manduca Sexta*. I Kinematics of hovering and forward flight. *J. Exp. Biology 200* (1997), 2705–2722.

[109] Willmott, A. P., and Ellington, C. P. The Mechanics of flight in the hawkmoth *Manduca Sexta*. II Aerodynamic consequences of kinematic and morphological variation. *J. Exp. Biology 200* (1997), 2723–2745.

[110] Wootton, R. J. Invertebrate paraxial locomotory appendages: design, deformation and control. *J. Exp. Biology 202* (1999), 3333–3345.

[111] Wootton, R. J., Herbert, R. C., Young, P. G., and Evans, K. E. Approaches to structural modeling of insect wings. *Phil. Trans. R. Soc. Lond. 358* (2003), 1577–1587.

[112] Wu, J. H., Zhang, Y. L., and Sun, M. Hovering of model insects: simulation by coupling equations of motion with navier-stokes equations. *J Exp. Biol. 212* (2009), 3313–3329.

[113] Yang, J., and Balaras, E. An embedded-boundary formulation for large-eddy simulation of turbulent flows interacting with moving boundaries. *Journal of Computational Physics 215* (Jan 2006), 12–40.

[114] Yang, J., Preidikman, S., and Balaras, E. A strongly coupled, embedded-boundary method for fluid-structure interactions of elastically mounted rigid bodies. *J Fluid Struct 24*, 2 (Jan 2008), 167–182.

[115] Yeung, P. K., Pope, S. B., and Sawford, B. L. Reynolds number dependence of lagrangian statistics in large numerical simulations of isotropic turbulence. *J. Turbulence 7*, 58 (2006).

[116] Yu, Z. A DLM/FD method for fluid/flexible-body interactions. *J. Comput. Phys. 207* (2005), 1–27.

[117] Yun, G., Kim, D., and Choi, H. Vortical structures behind a sphere at subcritical reynolds numbers. *Physics of Fluids 18* (2006), p. 015102.