University of Southern Queensland
Faculty of Health, Engineering & Sciences

COMPACT INTEGRATED RADIAL BASIS FUNCTION MODELLING OF PARTICULATE SUSPENSIONS

A dissertation submitted by

Nha Thai-Quang

B.Eng., M.Sci.

For the award of the degree of

Doctor of Philosophy

2014

Dedication

To my parents, my brother and my wife.

Certification of Dissertation

I certify that the ideas, experimental work, results and analyses, software and conclusions reported in this dissertation are entirely my own effort, except where otherwise acknowledged. I also certify that the work is original and has not been previously submitted for any other award.

Nha Thai-Quang, Candidate	Date

ENDORSEMENT

Prof. Nam Mai-Duy, Principal supervisor	Date
Prof. Thanh Tran-Cong, Co-supervisor	Date
Dr. Canh-Dung Tran, Co-supervisor	Date

Acknowledgments

I would like to express my profound gratitude to my supervisors, Prof. Nam Mai-Duy, Prof. Thanh Tran-Cong and Dr. Canh-Dung Tran for their invaluable support and effectively helpful guidance throughout this research work. Without their continuous support and guidance, my knowledge would not have been improved as today and this thesis would not have been completed.

I would like to acknowledge Dr. Khoa Le-Cao, my senior, for the helpful discussion during the period when we worked together. I am also thankful to all my seniors, labmates, colleagues and friends, for their assistance and encouragement over the course of my study.

I gratefully acknowledge the financial supports for my Ph.D. study including a Postgraduate Research Scholarship of The University of Southern Queensland, a Top Up of Faculty of Health, Engineering and Sciences, and a Supplement of Computational Engineering and Science Research Centre. In addition, I would like to thank Dr. Andrew Wandel and Dr. Canh-Dung Tran for offering me a position of teaching assistant. I also would like to thank staffs at the Faculty (A/Prof. Armando Apan, Ms. Juanita Ryan, Ms. Katrina Hall, Ms. Marie Morris, Ms. Melanie Loach, and Mr. Martin Geach) for their kind assistance in the matter of paper works.

Last but not least, I am eternally indebted to my family and my wife Thao Duong for their unconditional love, support, and encouragement to pursue this research endeavour. I would like to dedicate this work to them with my sincerest thanks.

Notes to Readers

All content of the present thesis is recorded on the attached CD-ROM, including the following files:

1. Thesis.pdf: An electronic version of the present thesis.
2. Chapter6-Oscillating-Circular-Cylinder-Re100.wmv: An animation showing the evolution of the horizontal velocity field of the flow induced by an oscillating circular cylinder for $Re = 100$ (Chapter 6, Section 6.4.2).
   Online at http://www.youtube.com/watch?v=kOuQsA51YY0.
3. Chapter6-Oscillating-Circular-Cylinder-Re800.wmv: An animation showing the evolution of the horizontal velocity field of the flow induced by an oscillating circular cylinder for $Re = 800$ (Chapter 6, Section 6.4.2).
   Online at http://www.youtube.com/watch?v=YJlFojDuPs4.
4. Chapter6-Single-Particle-Vertical-Velocity.wmv: An animation showing the sedimentation of a particle and the evolution of the vertical velocity field of the flow in a closed box (Chapter 6, Section 6.4.3).
   Online at http://www.youtube.com/watch?v=OItYnm8rbvQ.
5. Chapter6-Single-Particle-Vorticity.wmv: An animation showing the sedimentation of a particle and the evolution of the vorticity of the flow in a closed box (Chapter 6, Section 6.4.3).
   Online at http://www.youtube.com/watch?v=X2miOpFafIU.
6. Chapter6-Two-Particles-Drafting-Kissing-Tumbling-Velocity.wmv: An animation showing the drafting-kissing-tumbling phenomenon of two settling particles and the evolution of the velocity magnitude field of the flow in a closed box (Chapter 6, Section 6.4.4).
   Online at http://www.youtube.com/watch?v=7WZZCsG6bm8.
7. Chapter6-Two-Particles-Drafting-Kissing-Tumbling-Vorticity.wmv: An animation showing the drafting-kissing-tumbling phenomenon of two settling particles and the evolution of the vorticity of the flow in a closed box (Chapter 6, Section 6.4.4).
   Online at http://www.youtube.com/watch?v=Qc1FqNJsLbI.

Abstract

The present Ph.D. thesis is concerned with the development of computational procedures based on Cartesian grids, point collocation, immersed boundary method, and compact integrated radial basis functions (CIRBF), for the simulation of heat transfer and steady/unsteady viscous flows in complex geometries, and their applications for the prediction of macroscopic rheological properties of particulate suspensions.

The thesis consists of three main parts. In the first part, integrated radial basis function approximations are developed into compact local form to achieve sparse system matrices and high levels of accuracy together. These stencils are employed for the discretisation of the Navier-Stokes equation in the pressure-velocity formulation. The use of alternating direction implicit (ADI) algorithms to enhance the computational efficiency is also explored. In the second part, compact local IRBF stencils are extended for the simulation of flows in multiply-connected domains, where the direct forcing-immersed boundary (DFIB) method is adopted to handle such complex geometries efficiently. In the third part, the DFIB-CIRBF method is applied for the investigation of suspensions of rigid particles in a Newtonian liquid, and the prediction of their bulk viscosity and stresses.

The proposed computational procedures are verified successfully with several test problems in Computational Fluid Dynamics and Computational Rheology. Accurate results are achieved using relatively coarse grids.

Papers Resulting from the Research

Journal Articles

1. **N. Thai-Quang**, K. Le-Cao, N. Mai-Duy, T. Tran-Cong (2011): Discretisation of the Velocity-Pressure Formulation with Integrated Radial-Basis-Function Networks. *Structural Longevity*, vol. 6, no. 2, pp. 77-92.

2. **N. Thai-Quang**, K. Le-Cao, N. Mai-Duy, T. Tran-Cong (2012): A High-order Compact Local Integrated- RBF Scheme for Steady-state Incompressible Viscous Flows in the Primitive Variables. *Computer Modeling in Engineering and Sciences (CMES)*, vol. 84, no. 6, pp. 528-557.

3. **N. Thai-Quang**, N. Mai-Duy, C.-D. Tran, T. Tran-Cong (2012): High-order Alternating Direction Implicit Method Based on Compact Integrated-RBF Approximations for Unsteady/Steady Convection-Diffusion Equations. *Computer Modeling in Engineering and Sciences (CMES)*, vol. 89, no. 3, pp. 189-220.

4. **N. Thai-Quang**, K. Le-Cao, N. Mai-Duy, C.-D. Tran, T. Tran-Cong (2013): A Numerical Scheme Based on Compact Integrated-RBFs and Adams-Bashforth/Crank-Nicolson Algorithms for Diffusion and Unsteady Fluid Flow Problems. *Engineering Analysis with Boundary Elements*, vol. 37, no. 12, pp. 1653-1667.

5. N. Mai-Duy, **N. Thai-Quang**, T.-T. Hoang-Trieu, T. Tran-Cong (2013): A Compact 9 point Stencil Based on Integrated RBFs for the Convection-Diffusion Equation. *Applied Mathematical Modelling*. (Accepted, In Press).

6. **N. Thai-Quang**, N. Mai-Duy, C.-D. Tran, T. Tran-Cong (2013): A Direct Forcing Immersed Boundary Method Employed with Compact Integrated RBF Approximations for Heat Transfer and Fluid Flow Problems. *Computer Modeling in Engineering and Sciences (CMES)*, vol. 96, no. 1, pp. 49-90.

7. **N. Thai-Quang**, N. Mai-Duy, C.-D. Tran, T. Tran-Cong (2013): Direct Numerical Simulation of Particulate Flows with Direct Forcing Immersed Boundary-Compact Integrated RBF Method. (Submitted).

8. **N. Thai-Quang**, N. Mai-Duy, C.-D. Tran, T. Tran-Cong (2013): Direct Simulation of Particulate Suspension and Numerical Prediction of Its Rheological Properties by Compact Integrated RBF Approximations and Direct Forcing Immersed Boundary Method. (Submitted).

Conference Papers

1. **N. Thai-Quang**, N. Mai-Duy, T. Tran-Cong (2011, September 6-10): A Numerical Study of Integrated Radial-Basis-Functions for the Pressure-Velocity Formulation. *7th International Conference on Computational & Experimental Engineering and Sciences (ICCES), Special Symposium on Meshless & Other Novel Computational Methods*, Bülent Ecevit University, Zonguldak, Turkey.

2. **N. Thai-Quang**, K. Le-Cao, N. Mai-Duy, C.-D. Tran, T. Tran-Cong (2012, November 25-28): A High-order Compact Integrated-RBF Scheme for Time-Dependent Problems. *4th International Conference on Computational Methods (ICCM)*, Crowne Plaza, Gold Coast, Australia.

3. C.M.T. Tien, **N. Thai-Quang**, N. Mai-Duy, C.-D. Tran, T. Tran-Cong (2013, October 3-4): A fully coupled scheme for viscous flows in regular and irregular domains using compact integrated RBF approximation. *1st Australasian Conference on Computational Mechanics (ACCM)*, Sydney, Australia.

Contents

Acronyms & Abbreviations

1D	One Dimensional
1D-IRBF	One-dimensional Integrated Radial Basis Function
2D	Two Dimensional
ALE	Arbitrary Lagrangian-Eulerian
ALE-FEM	Arbitrary Lagrangian-Eulerian Finite Element Method
ADI	Alternating Direction Implicit
ADI-CIRBF-1	Alternating Direction Implicit-Compact Integrated Radial Basis Function-Scheme 1
ADI-CIRBF-2	Alternating Direction Implicit-Compact Integrated Radial Basis Function-Scheme 2
BEM	Boundary Element Method
BICGSTAB	Biconjugate Gradient Stabilised Method
BSQI	B-spline Quasi-Interpolation
CBGM	Cubic B-spline Galerkin Methods
CDD	Convection-Dominated Diffusion
CFD	Computational Fluid Dynamics
CIRBF	Compact Integrated Radial Basis Function
CIRBF-1	Compact Integrated Radial Basis Function-Scheme 1
CIRBF-2	Compact Integrated Radial Basis Function-Scheme 2
CIRBF-3	Compact Integrated Radial Basis Function-Scheme 3
CLIRBF	Compact Local Integrated Radial Basis Function
CPU	Central Processing Unit
DF	Direct Forcing
DFD	Domain-Free Discretisation Method
DFIB	Direct Forcing Immersed Boundary
DFIB-CIRBF	Direct Forcing Immersed Boundary-Compact Integrated Radial Basis Function
DKT	Drafting Kissing Tumbling
DLM/FDM	Distributed Lagrange Multiplier/Fictitious Domain Method
DNS	Direct Numerical Simulation
DQM	Differential Quadrature Method
DRBFN	Direct Radial Basis Function Network

EHOC-ADI	Exponential High-order Compact Alternating Direction Implicit
FBM	Fictitious Boundary Method
FD	Finite Difference
FDM	Finite Difference Method or Fictitious Domain Method
FE	Finite Element
FEM	Finite Element Method
FV	Finite Volume
FVM	Finite Volume Method
GMRES	Generalized Minimal Residual
HOC-ADI	High-order Compact Alternating Direction Implicit
HPD-ADI	High-order Hybrid Pade Alternating Direction Implicit
IB	Immersed Boundary
IBM	Immersed Boundary Method
IIM	Immersed Interface Method
IRBF	Integrated Radial Basis Function
IRBFN	Integrated Radial Basis Function Network
ISPM	Immersed Structural Potential Method
LBM	Lattice Boltzmann Method
MQ	Multiquadric
MQQI	Multiquadric Quasi-Interpolation
Ne	Error Norm
ODE	Ordinary Differential Equation
P_e	Peclet number
PDE	Partial Differential Equation
PDE-ADI	Pade Scheme-Based Alternating Direction Implicit
PR-ADI	Peaceman Rachford Alternating Direction Implicit
QBCM	Quartic B-spline Collocation Methods
QBGM	Quadratic Galerkin Methods
RBF	Radial Basis Function
RBFN	Radial Basis Function Network
Re	Reynolds number
RMS	Root Mean Square
SIM	Sharp Interface Method
SM	Spectral Method
SVD	Singular Value Decomposition
VBM	Virtual Boundary Method

List of Tables

List of Figures

Chapter 1

Introduction

The chapter starts with an overview of particulate suspensions. Conventional numerical methods used for solving fluid flows and fluid-solid systems are then reviewed. Next, we describe radial basis functions that will be utilised to develop approximation tools in the proposed computational procedures, which is followed by the motivation and objectives of the present research project. Finally, an outline of the thesis is presented.

1.1 Suspensions

Particulate suspensions widely exist in nature and are commonly produced in industry (Phan-Thien, 2013). Typical examples include paint, inks, biofluids (e.g. blood, mucus and cartilage), cosmetics, food stuff, etc. Particulate suspensions are systems formed by particles suspended in liquids. If the suspending liquid is Newtonian, one speaks of Newtonian suspensions; otherwise, non-Newtonian suspensions. Suspended particles can be rigid bodies, droplets or bubbles. If the suspended particles are identical, one has monodispersed suspensions - otherwise, polydispersed suspensions. The particle size also matters too. If the size of suspended particles is sufficiently small, they will undergo their Brownian motion and one speaks of colloidal suspensions. If this is not met, one has non-colloidal suspensions. Further details can be found in (Phan-Thien, 2013). In this research project, we are mainly concerned with suspensions of rigid spheres (3D) and circular discs (2D) in a Newtonian fluid (droplets and bubbles will not be considered here).

The behaviour of suspensions is non-Newtonian in nature, due to the evolution of microstructure (the spatial arrangement of rigid bodies). There are three regimes of suspension, namely dilute, semi-dilute and concentrated. For a concentrated regime (i.e. volume fractions of the solid phase larger than 0.25),

lubrication forces become dominant and special care is needed, particularly for numerical modelling (one needs to maintain meshes/grids between the particle surfaces when the volume fraction is increased). Suspensions are typically characterised by (i) the dependence of viscosity on concentration (volume fraction of the solid phase); (ii) non-zero first and second normal stress differences; and (iii) migration of rigid particles from high to low shear regions in an inhomogeneous shear flow such as Poiseuille flow.

1.2 Numerical methods

1.2.1 Simulating fluid flows

From a mathematical point of view, physical processes in nature such as the motion of fluids and heat transfer can be described by a set of partial differential equations (PDEs). These equations are based on the fundamental conservative laws in physics: conservation of mass, conservation of momentum and conservation of energy. The governing equations for Newtonian fluid flows are known as the Navier-Stokes equation. Due to the complex nature of the governing equations, analytical solutions cannot be obtained in most cases. Numerical methods have been developed to find an approximate solution. In numerical methods, one needs to discretise the PDEs in both space and time. As a result, a linear/non-linear PDE will be converted into a system of linear/non-linear algebraic equations. Such algebraic systems can then be solved for values of the field variables (e.g. the velocity, pressure and temperature) at discrete points within the computational domains and at all points in the domain by virtue of the assumed interpolation.

Computational fluid dynamics (CFD) is concerned with the numerical simulation of fluid flows. Common numerical methods used include finite difference methods (FDMs), finite volume methods (FVMs), finite element methods (FEMs), boundary element methods (BEMs) and spectral methods (SMs). Fundamental backgrounds and applications of conventional methods have been well documented, e.g. in (Roache, 1998; Tannehill et al., 1997; Peyret and Taylor, 1983) for FDM; (Leveque, 2002; Tannehill et al., 1997; Patankar, 1980) for FVM; (Zienkiewicz and Taylor, 2000; Girault and Raviart, 1986) for FEM; (Pozrikidis, 2002; Tran-Cong, 1989; Phan-Thien and Kim, 1994; Tanner, 1985) for BEM and (Peyret, 2002; Karniadakis and Sherwin, 1999; Canuto et al., 1988) for SM. They have been used with varying degree of success in various practical problems. It should be noted that (i) generating a mesh is not a straightforward task; and (ii) the distortion of the mesh can result in a severe reduction in accuracy. The latter can be alleviated by means of remeshing.

Recently, the concept of solving PDEs without using a mesh/grid has been introduced. The problem domain is simply discretised by a set of nodes that can be randomly distributed. Like conventional methods, meshless methods have also been developed to deal with the governing equations in their strong form (reproducing kernel particle method (RKPM) (Liu et al., 1995)), weak form (element-free Galerkin (EFG) method (Belytschko et al., 1994), meshless local Petrov-Galerkin (MLPG) (Atluri and Zhu, 1998; Atluri and Shen, 2002; Atluri et al., 2004)) and inverse statement (boundary node method (BNM) (Mukherjee and Mukherjee, 1997; Yang et al., 2011), local boundary integral equation (LBIE) method (Zhu et al., 1998)). Applications of meshless methods for fluid flows have received a great deal of attention because of their economic benefits in dealing with complex geometries such as free surfaces and channels of varying cross-sections.

1.2.2 Modelling fluid-solid systems

If the Navier-Stokes equation governing the motion of a fluid and the Newton-Euler equation governing the motion of solid bodies are solved directly, one speaks of direct numerical simulations (DNSs). Since the solvent is modelled explicitly, DNSs have the ability to deal with any kinds of the suspending liquid and also any shapes and sizes of suspended particles. Based on the fluid-phase solver employed, DNSs can be classified into two categories. In the first category, a mesh following the movement of the particles, i.e. moving mesh, is used. Methods in this category are usually based on the Arbitrary Lagrangian-Eulerian (ALE) approach, e.g. (Yu et al., 2007, 2004, 2002), (Hu et al., 2001; Hu, 1995; Hu et al., 1992). Finite element methods employed with unstructured meshes have been developed to simulate the motion of a large number of rigid objects in Newtonian and viscoelastic fluids. Nodes on the particle surfaces are allowed to move with the particles, while nodes in the interior of the fluid are smoothly updated by solving a Laplace's equation. One needs to generate a new mesh when the old one becomes too distorted; the flow field is then projected onto the new mesh. In the second category, a mesh covering the whole domain and independent of the particle movement, i.e. a fixed mesh, is used. Methods in this category include immersed structural potential methods (ISPMs) (Gil et al., 2010), fictitious boundary methods (FBMs) (Turek et al., 2003), immersed boundary methods (IBMs) (Peskin, 2002), sharp interface methods (SIMs) (also called ghost-fluid method) (Liu et al., 2000), virtual boundary methods (VBMs) (Saiki and Biringen, 1996), immersed interface methods (IIMs) (Leveque and Li, 1994), fictitious domain methods (FDMs) (Glowinski et al., 1994), etc. The advantage of the second category over the first one is that it allows a fixed grid to be used and thus eliminating the remeshing procedure.

Much research effort has also been put in the development of numerical methods in the Lagrangian framework (i.e. particle-based methods). Unlike the Eulerian

approach using a fixed frame of reference, the Lagrangian one follows the trajectory of individual parts of the material. An advantage in using this approach is that convective terms vanish in Lagrangian formulations and therefore the numerical method employed is simpler. A popular particle-based Lagrangian method is smoothed particle hydrodynamics (SPH) method pioneered by Lucy (1977) and Gingold and Monaghan (1977). Up to date, SPH methods have been improved in terms of both accuracy and stability and they have been applied to many engineering problems successfully (Monaghan, 2005; Randles and Libersky, 1996; Morris et al., 1997; Hashemi et al., 2012). Dilts (1999) proposed a novel method, namely Moving-Least-Squares-Particle Hydrodynamics (MLSPH), in which the conventional SPH method was newly derived by means of a Galerkin approximation. In this derivation, the SPH interpolant was replaced by the Moving-Least-Squares (MLS) interpolant.

1.3 Radial basis functions

Radial basis functions (RBFs) have emerged as a powerful numerical tool in scattered data approximations and have found applications in various research fields. Kansa (1990a,b) applied RBFs to solve parabolic, hyperbolic and elliptic PDEs, where the multiquadrics (MQ) function is adopted and used in a global fashion, and the collocation technique is employed to discretise PDEs. Kansa's method and its variants are hereby referred to as differential/direct RBF (DRBF) methods. The accuracy of MQ is strongly influenced by the so-called shape/width parameter (Carlson and Foley, 1991; Rippa, 1999). In approximating the function and its derivatives, MQ and some other RBFs are known to possess spectral accuracy with an error estimate as $O\left(\lambda^{\sqrt{a}/h}\right)$ (Cheng et al., 2003), where $0 < \lambda < 1$, a is the RBF width and h is the characteristic distance between RBFs' centres. If one approximates the kth derivative, the convergence rate is reduced to $O\left(\lambda^{\sqrt{a}/h-k}\right)$ as shown in Madych (1992). Noting that any errors associated with a function approximation will be badly magnified in the differentiation process. An indirect/integral (IRBF) approach was proposed in Mai-Duy and Tran-Cong (2001a,b, 2003) in order to avoid such a reduction in convergence rate. In this approach, derivatives of a function are first approximated by RBFs. Lower-order derivatives and the function itself are then obtained through the integration process. Since integration is a smooth process, it is expected that the indirect approach yields a more stable solution than the direct approach. The indirect/integral approach was developed into global forms (Mai-Duy and Tran-Cong, 2001a), one-dimensional forms (1D-IRBF) (Mai-Duy and Tanner, 2007; Mai-Duy and Tran-Cong, 2005), local forms (Mai-Duy and Tran-Cong, 2009) and compact local forms (Thai-Quang, Le-Cao, Mai-Duy, Tran and Tran-Cong, 2013; Mai-Duy and Tran-Cong, 2013; Thai-Quang, Mai-Duy, Tran and Tran-Cong, 2012; Thai-Quang, Le-Cao, Mai-Duy and Tran-Cong, 2012; Mai-Duy and Tran-Cong, 2011). Each form has its

own strengths and weaknesses. For compact local forms, the information about the governing equation or derivatives of the field variable is also included in local approximations, which allows the achievement of sparse system matrices and high levels of accuracy together. This research is concerned with the development of compact local IRBF approximations for the discretisation of the pressure-velocity formulation in solving viscous flows, fluid-solid interactions and particulate suspensions.

1.4 Motivation and objectives

An understanding of the rheological properties and dynamic behaviours of particulate suspensions is clearly vital in industrial particulate-flow processes, e.g. slurries, colloids, fluidised beds, spray drying/cooling in chemical engineering, river sediment in environmental engineering, and rock cuttings in drilling operation in mining and petroleum engineering. Direct numerical simulations (DNSs) for the description of microstructures in fluid media, from which bulk properties are derived, have made significant advances over the past twenty years. However, to date, existing DNSs still face major deficiencies with respect to efficiency and accuracy. Given the vastness of industrial processes involving particulate suspensions, any achievement in solving these problems will bring enormous benefits to the industry and consumers. The main objectives of this research are

- To develop computational procedures, which will be both accurate and efficient, for the simulation of heat transfer and steady/unsteady viscous flows.
- To develop accurate and efficient computational procedures for the simulation of steady/unsteady viscous flows in multiply-connected domains that are stationary.
- To develop accurate and efficient computational procedures for the simulation of steady/unsteady viscous flows in multiply-connected domains that vary with time.
- To predict rheological properties of particulate suspensions.

The enhancement of accuracy is achieved through

- Using RBFs to represent the field variables in the governing equations. RBFs converge much faster with respect to spacing refinement than conventional low-order polynomials.

- Using the integral formulation rather than the conventional differential one to construct the RBF approximations (IRBF). A reduction in convergence rate in the approximation of derivatives caused by differentiation (Madych, 1992) is avoided in the present integral formulation.
- Constructing the IRBF approximations, where values of the governing equation or derivatives at selected points are also included (compact forms).

The enhancement of efficiency is achieved through

- Using point collocation to discretise the governing equations. No integrations are involved in this process.
- Using Cartesian grids rather than finite-element meshes to represent the spatial problem domain. Generating a Cartesian grid is much simpler and easier than generating a finite element mesh.
- Constructing IRBF approximations in a local form, leading to a sparse system matrix.
- Using ADI algorithms of You (2006) to save memory storages and reduce computational costs.
- Converting a multiply-connected domain into a simply-connected domain by means of the direct forcing immersed boundary method.

1.5 Outline

The present thesis has a total of eight chapters including this chapter (Introduction). Below are brief descriptions of the remaining chapters.

- In chapter 2, IRBFs are developed for the discretisation of the velocity-pressure formulation on Cartesian grids to simulate 2D steady-state incompressible viscous flows. A high-order compact local integrated RBF (CLIRBF) approximation scheme and an effective boundary treatment for the pressure variable are proposed. A number of linear and non-linear problems are considered to investigate the performance of the proposed scheme/treatment numerically.
- In chapter 3, CLIRBF approximations are further developed for solving 1D and 2D time-dependent problems, where second-order Adams-Bashforth/Crank-Nicolson algorithms are employed to discretise the temporal derivatives. Four types of time-dependent equations, namely the diffusion, Burgers, Stokes and Navier-Stokes equations, are considered.

- In chapter 4, CLIRBF approximations are employed with the ADI algorithm of You (2006) to solve 2D convection-diffusion equation efficiently. Several steady and unsteady problems are considered to verify the present schemes. Results obtained are compared with those by some other ADI schemes.

- In Chapter 5, CLIRBF approximations are employed with the direct forcing immersed boundary (DFIB) method to simulate heat transfers and viscous flows in multiply-connected domains. The proposed method, namely DFIB-CIRBF, is verified through the solution of several test problems including Taylor-Green vortices, rotational flow, lid-driven cavity flow with multiple solid bodies, flow between rotating circular and fixed square cylinders, and natural convection in an eccentric annulus between two circular cylinders.

- In chapter 6, the present DFIB-CIRBF method is applied for the simulation of fluid-solid systems (the interactions between fluid and rigid particles). Problems considered include Taylor-Green vortices, induced flow by an oscillating circular cylinder, single particle sedimentation and drafting-kissing-tumbling behaviour of two settling particles.

- In chapter 7, the present DFIB-CIRBF method is applied for the simulation of particulate suspensions in a sliding bi-periodic frame and the prediction of their rheological properties numerically. The motion of a liquid and particles are solved in a decoupled manner, where methods for computing the rigid body motion are derived. Results concerning viscometric behaviour (e.g. viscosity and flow index) are presented.

- Chapter 8 concludes the thesis and suggests some possible research developments/extensions in the future.

Chapter 2

A compact IRBF scheme for steady-state fluid flows

This chapter is concerned with the development of IRBF method for the simulation of 2D steady-state incompressible viscous flows governed by the velocity-pressure formulation on Cartesian grids. Instead of using low-order polynomial interpolants, a high-order compact local IRBF scheme is employed to represent the convection and diffusion terms. Furthermore, an effective boundary treatment for the pressure variable, where Neumann boundary conditions are transformed into Dirichlet ones, is proposed. This transformation is based on global 1D-IRBF approximators using values of the pressure at interior nodes along a grid line and first-order derivative values of the pressure at the two extreme nodes of that grid line. The performance of the proposed scheme is investigated numerically through the solution of several linear (analytic tests including Stokes flows) and non-linear (recirculating cavity flow driven by combined shear & body forces and lid-driven cavity flow) problems. Unlike the global 1D-IRBF scheme, the proposed scheme leads to a sparse system matrix. Numerical results indicate that (i) the present solutions are more accurate and converge faster with grid refinement in comparison with standard finite-difference results; and (ii) the proposed boundary treatment for the pressure is more effective than conventional direct application of the Neumann boundary condition.

2.1 Introduction

It is known that the equations of motion of a Newtonian fluid can be obtained via several formulations, including those based on the velocity-pressure ($\mathbf{u}-p$), the stream function-vorticity ($\psi-\omega$) and the stream function (ψ) variables.

The last two involve less dependent variables than the first one. However, they require some special treatments for the handling of the vorticity boundary condition (the $\psi - \omega$ formulation) and the calculation of high-order derivatives including the cross-ones (the ψ formulation). Furthermore, the pressure field needs be resolved, which is generally recognised as a complicated process. For the $\mathbf{u} - p$ formulation, the pressure and velocity fields are obtained directly from the discretised equations and it is straightforward to extend the formulation to 3D problems.

It was reported, e.g. (Roache, 1998; Cheng, 1968; Cyrus and Fulton, 1967), that the use of a conservative form of the governing equation has the ability to give more accurate results than the use of a non-conservative form. In Torrance et al. (1972), through the simulation of a flow in a cavity, it was shown that results by using the conservative equations with first-order accurate interpolants are better than those by using the non-conservative equations with second-order accurate interpolants.

To facilitate a numerical calculation, the spatial domain needs be discretised. Generating a Cartesian grid, which is associated with finite-difference (FD) methods, can be seen to be much more straightforward than generating a finite-element (FE) mesh, which is associated with FE methods and finite-volume (FV) methods.

A fractional-step/projection approach, which is originally suggested by Chorin (1968), is widely applied for the simulation of incompressible viscous flows modelled with the $\mathbf{u} - p$ formulation. Variations of this approach have been published in, for example, (Almgren, 1996; Perot, 1993; Bell et al., 1989; Van Kan, 1986; Kim and Moin, 1985). In this study, we will propose a numerical projection method, based on Cartesian grids and a compact local IRBF scheme, for the discretisation of the $\mathbf{u} - p$ formulation in two dimensions. Boundary conditions for the pressure are taken in the form of Dirichlet type, and to do so, we propose a treatment based on global 1D IRBF approximations using values of the pressure at interior nodes along a grid line and first-order derivative values of the pressure at the two extreme nodes of that grid line. The performance of the present method is investigated numerically through the solution of linear and non-linear problems.

The remainder of the chapter is organised as follows. Sections 2.2 and 2.3 briefly outline the mathematical model of incompressible viscous flows and the global 1D-IRBF approximation scheme, respectively. The proposed compact local IRBF scheme and the proposed boundary treatment for the pressure are described in Section 2.4. In Section 2.5, numerical results are presented and compared with some benchmark solutions, where appropriate. Finally, some concluding remarks are given in Section 2.6.

2.2 Mathematical model

The transient Navier-Stokes equations for an incompressible Newtonian fluid in a domain of interest Ω at the time t can be written in the non-dimensionalised conservative form and in the primitive variables as

$$\nabla.\mathbf{u} = 0 \qquad \text{in} \quad \Omega,\ t \geq 0, \tag{2.1}$$

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla.(\mathbf{u}\,\mathbf{u}) = -\nabla p + \frac{1}{Re}\nabla^2\mathbf{u} + \mathbf{f}_b \qquad \text{in} \quad \Omega,\ t \geq 0, \tag{2.2}$$

where p, $\mathbf{u} = (u, v)^T$, $\mathbf{f}_b = \left(f_{bx}, f_{by}\right)^T$ are the static pressure, the fluid velocity vector, and the fluid body force density vector, respectively, defined in the Cartesian x and y coordinate system, the superscript T denotes the transpose; $Re = UL/\nu$ the Reynolds number, in which ν is the kinematic viscosity, L is the characteristic length and U is the characteristic speed of the flow.

For the projection method (Chorin, 1968), the velocity and the pressure variables in the above set of PDEs are solved separately in each iteration. The temporal discretisation of (2.2) with an explicit Euler scheme gives

$$\frac{\mathbf{u}^n - \mathbf{u}^{n-1}}{\Delta t} = -\boldsymbol{\nabla} p^n + \frac{1}{Re}\boldsymbol{\nabla}^2\mathbf{u}^{n-1} - \nabla.(\mathbf{u}^{n-1}\mathbf{u}^{n-1}) + \mathbf{f}_b^{n-1}, \tag{2.3}$$

where the superscript n denotes the current time level.

An intermediate velocity vector, denoted by $\mathbf{u}^{*,n} = (u^{*,n}, v^{*,n})^T$, is defined as

$$\frac{\mathbf{u}^{*,n} - \mathbf{u}^{n-1}}{\Delta t} = \frac{1}{Re}\boldsymbol{\nabla}^2\mathbf{u}^{n-1} - \nabla.(\mathbf{u}^{n-1}\mathbf{u}^{n-1}) + \mathbf{f}_b^{n-1}. \tag{2.4}$$

This equation, which does not involve the pressure gradient term, can be rewritten as

$$\mathbf{u}^{*,n} = \mathbf{u}^{n-1} + \Delta t\left[\frac{1}{Re}\boldsymbol{\nabla}^2\mathbf{u}^{n-1} - \nabla.(\mathbf{u}^{n-1}\mathbf{u}^{n-1}) + \mathbf{f}_b^{n-1}\right]. \tag{2.5}$$

It is seen that $\mathbf{u}^{*,n}$ does not satisfy the continuity equation (2.1). From (2.3) and (2.4), one can derive the following equation

$$\frac{\mathbf{u}^n - \mathbf{u}^{*,n}}{\Delta t} = -\boldsymbol{\nabla} p^n. \tag{2.6}$$

The Poisson equation for the pressure is then obtained by applying the gradient operator to both sides of (2.6) and forcing $\mathbf{u}^n$ to satisfy (2.1)

$$\boldsymbol{\nabla}^2 p^n = \frac{1}{\Delta t}\boldsymbol{\nabla}\cdot\mathbf{u}^{*,n}. \tag{2.7}$$

After solving (2.7), the velocity field at the next time level is calculated through (2.6) as

$$\mathbf{u}^n = \mathbf{u}^{*,n} - \Delta t \boldsymbol{\nabla} p^n. \tag{2.8}$$

2.3 A brief review of the global 1D-IRBF scheme

Consider the approximation of a univariate function $u(\eta)$ and its derivatives up to second order. The second-order derivative of u is decomposed into RBFs

$$\frac{d^2u(\eta)}{d\eta^2} = \sum_{i=1}^{m} w_i G_i(\eta), \tag{2.9}$$

where m is the number of RBFs; $\{G_i(\eta)\}_{i=1}^m$ the set of RBFs; and $\{w_i\}_{i=1}^m$ the set of weights/coefficients to be found. Approximate representations for the first-order derivative and the function itself are then obtained through integration

$$\frac{du(\eta)}{d\eta} = \sum_{i=1}^{m} w_i H_i(\eta) + c_1, \tag{2.10}$$

$$u(\eta) = \sum_{i=1}^{m} w_i \overline{H}_i(\eta) + c_1\eta + c_2, \tag{2.11}$$

where $H_i(\eta) = \int G_i(\eta)d\eta$; $\overline{H}_i(\eta) = \int H_i(\eta)d\eta$; and c_1 and c_2 are the constants of integration. In the IRBF methods, basis functions are obtained through

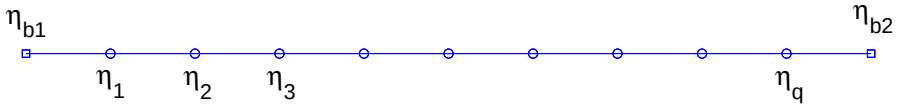


Figure 2.1: 1D-IRBF centres on a Cartesian grid line.

integration. For several RBFs including the multiquadric function used in the present project, integrated basis functions can be obtained in analytic form and thus equations (2.10) and (2.11) do not require any additional costs when compared to their counterparts of the conventional (differential) RBF methods. Let $\{\eta_i\}_{i=1}^q$ ($q = m-2$) and $\{\eta_{b1}, \eta_{b2}\}$ be a set of interior nodal points and a set of boundary nodal points, respectively, as shown in Figure 2.1. We choose the set of RBF centres as the set of nodes. Evaluation of (2.11) at the interior and boundary nodes results in

$$\begin{pmatrix} \widehat{u} \\ \widehat{u}_b \end{pmatrix} = \overline{\mathcal{H}} \begin{pmatrix} \widehat{w} \\ c_1 \\ c_2 \end{pmatrix}, \tag{2.12}$$

where

$$\begin{aligned}
\widehat{u} &= (u_1, u_2, \cdots, u_q)^T, \\
\widehat{u}_b &= (u_{b1}, u_{b2})^T, \\
\widehat{w} &= (w_1, w_2, \cdots, w_m)^T, \\
\overline{\mathcal{H}} &= \begin{bmatrix}
\overline{H}_1(\eta_1) & \cdots & \overline{H}_m(\eta_1) & \eta_1 & 1 \\
\overline{H}_1(\eta_2) & \cdots & \overline{H}_m(\eta_2) & \eta_2 & 1 \\
\vdots & \ddots & \vdots & \vdots & \vdots \\
\overline{H}_1(\eta_q) & \cdots & \overline{H}_m(\eta_q) & \eta_q & 1 \\
\overline{H}_1(\eta_{b1}) & \cdots & \overline{H}_m(\eta_{b1}) & \eta_{b1} & 1 \\
\overline{H}_1(\eta_{b2}) & \cdots & \overline{H}_m(\eta_{b2}) & \eta_{b2} & 1
\end{bmatrix}.
\end{aligned} \tag{2.13}$$

The system (2.12), which represents the relation between the RBF space and the physical space and hereafter is called a conversion system, can be solved for the unknown vector of weights $(\widehat{w}, c_1, c_2)^T$ by means of the singular value decomposition (SVD) technique as

$$\begin{pmatrix} \widehat{w} \\ c_1 \\ c_2 \end{pmatrix} = \overline{\mathcal{H}}^{-1} \begin{pmatrix} \widehat{u} \\ \widehat{u}_b \end{pmatrix}, \tag{2.14}$$

where $\overline{\mathcal{H}}^{-1}$ is the pseudo-inverse of $\overline{\mathcal{H}}$.

Making use of (2.14), (2.10) and (2.9), values of the first and second derivatives of u at the interior and boundary nodes are, respectively, computed as

$$\begin{pmatrix} \frac{du_1}{d\eta} \\ \frac{du_2}{d\eta} \\ \vdots \\ \frac{du_q}{d\eta} \\ \frac{du_{b1}}{d\eta} \\ \frac{du_{b2}}{d\eta} \end{pmatrix} = \begin{bmatrix}
H_1(\eta_1) & \cdots & H_m(\eta_1) & 1 & 0 \\
H_1(\eta_2) & \cdots & H_m(\eta_2) & 1 & 0 \\
\vdots & \ddots & \vdots & \vdots & \vdots \\
H_1(\eta_q) & \cdots & H_m(\eta_q) & 1 & 0 \\
H_1(\eta_{b1}) & \cdots & H_m(\eta_{b1}) & 1 & 0 \\
H_1(\eta_{b2}) & \cdots & H_m(\eta_{b2}) & 1 & 0
\end{bmatrix} \overline{\mathcal{H}}^{-1} \begin{pmatrix} \widehat{u} \\ \widehat{u}_b \end{pmatrix}, \tag{2.15}$$

$$\begin{pmatrix} \frac{d^2u_1}{d\eta^2} \\ \frac{d^2u_2}{d\eta^2} \\ \vdots \\ \frac{d^2u_q}{d\eta^2} \\ \frac{d^2u_{b1}}{d\eta^2} \\ \frac{d^2u_{b2}}{d\eta^2} \end{pmatrix} = \begin{bmatrix}
G_1(\eta_1) & \cdots & G_m(\eta_1) & 0 & 0 \\
G_1(\eta_2) & \cdots & G_m(\eta_2) & 0 & 0 \\
\vdots & \ddots & \vdots & \vdots & \vdots \\
G_1(\eta_q) & \cdots & G_m(\eta_q) & 0 & 0 \\
G_1(\eta_{b1}) & \cdots & G_m(\eta_{b1}) & 0 & 0 \\
G_1(\eta_{b2}) & \cdots & G_m(\eta_{b2}) & 0 & 0
\end{bmatrix} \overline{\mathcal{H}}^{-1} \begin{pmatrix} \widehat{u} \\ \widehat{u}_b \end{pmatrix}. \tag{2.16}$$

These expressions can be rewritten in the following compact form

$$\widehat{\frac{du}{d\eta}} = \widehat{\mathcal{D}}_{1\eta} \widehat{u} + \widehat{k}_{1\eta}, \tag{2.17}$$

and

$$\widehat{\frac{d^2u}{d\eta^2}} = \widehat{\mathcal{D}}_{2\eta}\widehat{u} + \widehat{k}_{2\eta}, \tag{2.18}$$

where the matrices $\widehat{\mathcal{D}}_{1\eta}$ and $\widehat{\mathcal{D}}_{2\eta}$ consist of all but the last two columns of the product of two matrices on the right-hand side of (2.15) and (2.16), respectively; and $\widehat{k}_{1\eta}$ and $\widehat{k}_{2\eta}$ are obtained by multiplying the vector $\widehat{u}_b$ with the last two columns of (2.15) and (2.16) respectively. It is noted that entries of $\widehat{k}_{1\eta}$ and $\widehat{k}_{2\eta}$ are functions of the two boundary values.

It can be seen that derivatives of the function u at nodes are expressed in terms of nodal values of u.

2.4 Proposed method

Consider an interior grid point $\mathbf{x}_0 = (x_0, y_0)^T$ and its associated local 3-point stencil $[\eta_1, \eta_2, \eta_3]$ $(\eta_1 < \eta_2 < \eta_3,\ \eta_0 \equiv \eta_2)$ as shown in Figure 2.2, in which η represents x and y.

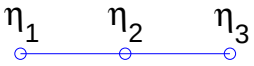


Figure 2.2: Local 3-point 1D-IRBF stencil.

2.4.1 A high-order compact local IRBF scheme

Over a local 3-point stencil, we can represent the conversion system as a matrix-vector multiplication

$$\begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ \frac{d^2u_1}{d\eta^2} \\ \frac{d^2u_3}{d\eta^2} \end{pmatrix} = \underbrace{\begin{pmatrix} \overline{\mathcal{H}} \\ \mathcal{G} \end{pmatrix}}_{\mathcal{C}} \begin{pmatrix} w_1 \\ w_2 \\ w_3 \\ c_1 \\ c_2 \end{pmatrix}, \tag{2.19}$$

where $u_i = u(\eta_i)$ $(i \in \{1,\ 2,\ 3\})$; $\frac{d^2u_i}{d\eta^2} = \frac{d^2u}{d\eta^2}(\eta_i)$ $(i \in \{1,\ 3\})$; $\mathcal{C}$ is the conversion matrix and $\overline{\mathcal{H}}$, $\mathcal{G}$ are submatrices defined as

$$\overline{\mathcal{H}} = \begin{bmatrix} \overline{H}_1(\eta_1) & \overline{H}_2(\eta_1) & \overline{H}_3(\eta_1) & \eta_1 & 1 \\ \overline{H}_1(\eta_2) & \overline{H}_2(\eta_2) & \overline{H}_3(\eta_2) & \eta_2 & 1 \\ \overline{H}_1(\eta_3) & \overline{H}_2(\eta_3) & \overline{H}_3(\eta_3) & \eta_3 & 1 \end{bmatrix}, \tag{2.20}$$

$$\mathcal{G} = \begin{bmatrix} G_1(\eta_1) & G_2(\eta_1) & G_3(\eta_1) & 0 & 0 \\ G_1(\eta_3) & G_2(\eta_3) & G_3(\eta_3) & 0 & 0 \end{bmatrix}. \tag{2.21}$$

Solving (2.19) yields

$$\begin{pmatrix} w_1 \\ w_2 \\ w_3 \\ c_1 \\ c_2 \end{pmatrix} = \mathcal{C}^{-1} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ \frac{d^2u_1}{d\eta^2} \\ \frac{d^2u_3}{d\eta^2} \end{pmatrix}, \tag{2.22}$$

which maps the vector of nodal values of the function and of its second derivative to the vector of RBF coefficients including two integration constants. Approximate expressions for u and its derivatives in the physical space are obtained by substituting (2.22) into (2.11), (2.10) and (2.9), respectively.

$$u(\eta) = \begin{bmatrix} \overline{H}_1(\eta) & \overline{H}_2(\eta) & \overline{H}_3(\eta) & \eta & 1 \end{bmatrix} \mathcal{C}^{-1} \begin{pmatrix} \widehat{u} \\ \widehat{\frac{d^2u}{d\eta^2}} \end{pmatrix}, \tag{2.23}$$

$$\frac{du(\eta)}{d\eta} = \begin{bmatrix} H_1(\eta) & H_2(\eta) & H_3(\eta) & 1 & 0 \end{bmatrix} \mathcal{C}^{-1} \begin{pmatrix} \widehat{u} \\ \widehat{\frac{d^2u}{d\eta^2}} \end{pmatrix}, \tag{2.24}$$

$$\frac{d^2u(\eta)}{d\eta^2} = \begin{bmatrix} G_1(\eta) & G_2(\eta) & G_3(\eta) & 0 & 0 \end{bmatrix} \mathcal{C}^{-1} \begin{pmatrix} \widehat{u} \\ \widehat{\frac{d^2u}{d\eta^2}} \end{pmatrix}, \tag{2.25}$$

where $\eta_1 \le \eta \le \eta_3$; $\widehat{u} = (u_1, u_2, u_3)^T$ and $\widehat{\frac{d^2u}{d\eta^2}} = (\frac{d^2u_1}{d\eta^2}, \frac{d^2u_3}{d\eta^2})^T$. The above three equations can be rewritten in the form

$$u(\eta) = \sum_{i=1}^{3} \varphi_i(\eta) u_i + \varphi_4(\eta) \frac{d^2u_1}{d\eta^2} + \varphi_5(\eta) \frac{d^2u_3}{d\eta^2}, \tag{2.26}$$

$$\frac{du(\eta)}{d\eta} = \sum_{i=1}^{3} \frac{d\varphi_i(\eta)}{d\eta} u_i + \frac{d\varphi_4(\eta)}{d\eta} \frac{d^2u_1}{d\eta^2} + \frac{d\varphi_5(\eta)}{d\eta} \frac{d^2u_3}{d\eta^2}, \tag{2.27}$$

$$\frac{d^2u(\eta)}{d\eta^2} = \sum_{i=1}^{3} \frac{d^2\varphi_i(\eta)}{d\eta^2} u_i + \frac{d^2\varphi_4(\eta)}{d\eta^2} \frac{d^2u_1}{d\eta^2} + \frac{d^2\varphi_5(\eta)}{d\eta^2} \frac{d^2u_3}{d\eta^2}, \tag{2.28}$$

where $\{\varphi_i(\eta)\}_{i=1}^5$ is the set of IRBFs in the physical space. It can be seen from (2.26)-(2.28) that the present IRBF approximations are expressed in terms of

not only nodal function values but also nodal second-derivative values.

The present compact local 3-point IRBF scheme is utilised to represent the variations of the velocity components, the intermediate velocity components and the pressure in (2.3)-(2.8).

2.4.2 Two boundary treatments for the pressure

In order to solve the pressure Poisson equation (2.7), a boundary condition for the pressure is required. On the non-slip boundaries, from the momentum equation (2.2), one can derive the Neumann boundary condition for the pressure as

$$\frac{\partial p_b^n}{\partial x} = \frac{1}{Re}\left(\frac{\partial^2 u_b^{n-1}}{\partial x^2} + \frac{\partial^2 u_b^{n-1}}{\partial y^2}\right) - \left(\frac{\partial (u_b^{n-1} u_b^{n-1})}{\partial x} + \frac{\partial (v_b^{n-1} u_b^{n-1})}{\partial y}\right) + {f_{b_x}}^{n-1} = \frac{u_b^{*,n} - u_b^n}{\Delta t}, \quad (2.29)$$

$$\frac{\partial p_b^n}{\partial y} = \frac{1}{Re}\left(\frac{\partial^2 v_b^{n-1}}{\partial x^2} + \frac{\partial^2 v_b^{n-1}}{\partial y^2}\right) - \left(\frac{\partial (u_b^{n-1} v_b^{n-1})}{\partial x} + \frac{\partial (v_b^{n-1} v_b^{n-1})}{\partial y}\right) + {f_{b_y}}^{n-1} = \frac{v_b^{*,n} - v_b^n}{\Delta t}. \quad (2.30)$$

In what follows, we will describe an implementation of the Neumann boundary condition in the context of IRBFs (Treatment 1), and present a new treatment, which transforms the Neumann boundary condition into the Dirichlet one, and its detailed implementation (Treatment 2).

Treatment 1

The boundary condition for the pressure is imposed in the Neumann form. Assume that η_1 is a boundary node (i.e. $\eta_{b1} \equiv \eta_1$). At the current time level n, one can calculate the value of $\partial p/\partial \eta$ at η_{b1} through (2.29) and (2.30). We modify the conversion system (2.19) as

$$\begin{pmatrix} p_1^n \\ p_2^n \\ p_3^n \\ \frac{\partial p_{b1}^n}{\partial \eta} \\ \frac{\partial^2 p_3^{n-1}}{\partial \eta^2} \end{pmatrix} = \begin{pmatrix} \overline{\mathcal{H}} \\ \mathcal{H} \\ \mathcal{G} \end{pmatrix} \begin{pmatrix} w_1^n \\ w_2^n \\ w_3^n \\ c_1^n \\ c_2^n \end{pmatrix}, \quad (2.31)$$

where $\partial p_{b1}^n/\partial\eta$ and $\partial^2 p_3^{n-1}/\partial\eta^2$ are known values; $\overline{\mathcal{H}}$ is defined as in (2.20); and

$$\mathcal{H} = \left[\begin{array}{ccccc} H_1(\eta_{b1}) & H_2(\eta_{b1}) & H_3(\eta_{b1}) & 1 & 0 \end{array}\right], \tag{2.32}$$

$$\mathcal{G} = \left[\begin{array}{ccccc} G_1(\eta_3) & G_2(\eta_3) & G_3(\eta_3) & 0 & 0 \end{array}\right]. \tag{2.33}$$

Equation (2.31) leads to

$$\begin{pmatrix} w_1^n \\ w_2^n \\ w_3^n \\ c_1^n \\ c_2^n \end{pmatrix} = \begin{pmatrix} \overline{\mathcal{H}} \\ \mathcal{H} \\ \mathcal{G} \end{pmatrix}^{-1} \begin{pmatrix} p_1^n \\ p_2^n \\ p_3^n \\ \frac{\partial p_{b1}^n}{\partial\eta} \\ \frac{\partial^2 p_3^{n-1}}{\partial\eta^2} \end{pmatrix}. \tag{2.34}$$

It can be seen that there are two unknowns over the stencil associated with $\eta_0 \equiv \eta_2$, namely p_{b1}^n and p_2^n. As a result, apart from collocating (2.7) at η_2 for the unknown p_2^n, one also needs to collocate (2.7) at η_{b1} for the unknown p_{b1}^n. Values of the second derivative of p at η_{b1} and η_2 at the current time level are thus computed as

$$\begin{pmatrix} \frac{\partial^2 p_{b1}^n}{\partial\eta^2} \\ \frac{\partial^2 p_2^n}{\partial\eta^2} \end{pmatrix} = \left[\begin{array}{ccccccc} G_1(\eta_{b1}) & \cdots & G_m(\eta_{b1}) & \eta_{b1} & 1 \\ G_1(\eta_2) & \cdots & G_m(\eta_2) & \eta_2 & 1 \end{array}\right] \begin{pmatrix} \overline{\mathcal{H}} \\ \mathcal{H} \\ \mathcal{G} \end{pmatrix}^{-1} \begin{pmatrix} p_1^n \\ p_2^n \\ p_3^n \\ \frac{\partial p_{b1}^n}{\partial\eta} \\ \frac{\partial^2 p_3^{n-1}}{\partial\eta^2} \end{pmatrix}. \tag{2.35}$$

Treatment 2

The boundary condition for the pressure is imposed in the Dirichlet form. The process of deriving Dirichlet boundary conditions for the pressure is based on the global 1D-IRBF approximation scheme, i.e. (2.9)-(2.11), using the previous values of the pressure at interior nodes along a grid line and the current first-order derivative values of the pressure at the two extreme nodes of that grid line (Thai-Quang et al., 2011).

Consider a grid line η and let m be the number of nodes on the grid line. From (2.29)-(2.30), one can obtain derivative values of the pressure at the two extreme nodes, i.e. $\partial p_{b1}^n/\partial\eta$ and $\partial p_{b2}^n/\partial\eta$. We modify the conversion system (2.12) as

$$\begin{pmatrix} \widehat{p}^{n-1} \\ \frac{\partial p_{b1}^n}{\partial\eta} \\ \frac{\partial p_{b2}^n}{\partial\eta} \end{pmatrix} = \begin{pmatrix} \overline{\mathcal{H}} \\ \mathcal{H} \end{pmatrix} \begin{pmatrix} \widehat{w}^n \\ c_1^n \\ c_2^n \end{pmatrix}, \tag{2.36}$$

where the left-hand side is a known vector

$$\widehat{p}^{n-1} = \left(p_1^{n-1}, p_2^{n-1}, \cdots, p_q^{n-1}\right)^T, \quad (q = m-2),$$
$$\widehat{w}^{n} = \left(w_1^{n}, w_2^{n}, \cdots, w_m^{n}\right)^T,$$

$$\overline{\mathcal{H}} = \begin{bmatrix} \overline{H}_1(\eta_1) & \cdots & \overline{H}_m(\eta_1) & \eta_1 & 1 \\ \overline{H}_1(\eta_2) & \cdots & \overline{H}_m(\eta_2) & \eta_2 & 1 \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ \overline{H}_1(\eta_q) & \cdots & \overline{H}_m(\eta_q) & \eta_q & 1 \end{bmatrix}, \tag{2.37}$$

$$\mathcal{H} = \begin{bmatrix} H_1(\eta_{b1}) & \cdots & H_m(\eta_{b1}) & 1 & 0 \\ H_1(\eta_{b2}) & \cdots & H_m(\eta_{b2}) & 1 & 0 \end{bmatrix}. \tag{2.38}$$

Values of the pressure at the two extreme nodes at the current time level are then estimated by collocating (2.11) at η_{b1} and η_{b2} and making use of (2.36)

$$\begin{pmatrix} p_{b1}^n \\ p_{b2}^n \end{pmatrix} = \begin{bmatrix} \overline{H}_1(\eta_{b1}) & \cdots & \overline{H}_m(\eta_{b1}) & \eta_{b1} & 1 \\ \overline{H}_1(\eta_{b2}) & \cdots & \overline{H}_m(\eta_{b2}) & \eta_{b2} & 1 \end{bmatrix} \begin{pmatrix} \overline{\mathcal{H}} \\ \mathcal{H} \end{pmatrix}^{-1} \begin{pmatrix} \widehat{p}^{n-1} \\ \frac{\partial p_{b1}^n}{\partial \eta} \\ \frac{\partial p_{b2}^n}{\partial \eta} \end{pmatrix}. \tag{2.39}$$

We use these known values as Dirichlet boundary conditions in solving the pressure Poisson equation (2.7).

2.4.3 Solution procedure

The proposed solution procedure is outlined as follows.

- Step 1: Guess initial values for the pressure and velocity fields. For the $Re = 0$ case, we use the rest state as the initial guess. For a $Re > 0$ case, we use the solution corresponding to a smaller Re as the initial guess.
- Step 2: Compute the intermediate velocity field

$$\mathbf{u}^{*,n} = \mathbf{u}^{n-1} + \Delta t \left[\frac{1}{Re} \boldsymbol{\nabla}^2 \mathbf{u}^{n-1} - \nabla . \left(\mathbf{u}^{n-1} \mathbf{u}^{n-1} \right) + \mathbf{f}_b^{n-1} \right], \tag{2.40}$$

  using the proposed compact local IRBF scheme, i.e (2.27)-(2.28), in which, for $n > 2$, nodal values of the field variable and its second derivatives are taken from the time level $(n-1)$ and $(n-2)$, respectively. It is noted that, on the boundary, some nodal second-derivative values are estimated through the governing equations, e.g.
  $\frac{\partial^2 u_b^{n+1}}{\partial x^2}$, $\frac{\partial^2 v_b^{n+1}}{\partial x^2}$, $\frac{\partial^2 u_b^{n+1}}{\partial y^2}$ and $\frac{\partial^2 v_b^{n+1}}{\partial y^2}$,
  and some through the global 1D-IRBF scheme, e.g.

$\frac{\partial^2 (uu)_b^{n+1}}{\partial x^2}$, $\frac{\partial^2 (uv)_b^{n+1}}{\partial x^2}$, $\frac{\partial^2 u_b^{*(n+1)}}{\partial x^2}$, $\frac{\partial^2 p_b^{n+1}}{\partial x^2}$, $\frac{\partial^2 (vu)_b^{n+1}}{\partial y^2}$, $\frac{\partial^2 (vv)_b^{n+1}}{\partial y^2}$, $\frac{\partial^2 v_b^{*(n+1)}}{\partial y^2}$ and $\frac{\partial^2 p_b^{n+1}}{\partial y^2}$.

- Step 3: Compute $\nabla.\mathbf{u}^{*,n}$ according to the formula (2.27), in which, for $n > 1$, nodal values of the function and its second derivatives are taken from the time level n and $(n-1)$, respectively.

- Step 4: Compute the pressure gradients on the boundary from the momentum equations

$$\nabla p^n = \frac{1}{\Delta t}\left(\mathbf{u}^{*,n} - \mathbf{u}^{n-1}\right). \tag{2.41}$$

- Step 5: Derive a Dirichlet boundary condition for the pressure, i.e. p_b^n, in the case of using Treatment 2. Otherwise, skip this step.

- Step 6: Solve the pressure Poisson equation

$$\nabla^2 p^n = \frac{1}{\Delta t}\nabla.\mathbf{u}^{*,n}, \tag{2.42}$$

  subject to the corresponding boundary conditions.

- Step 7: Compute ∇p^n using (2.27) and estimate the velocity field at the current time level

$$\mathbf{u}^n = \mathbf{u}^{*,n} - \Delta t \nabla p^n. \tag{2.43}$$

- Step 8: Check to see whether the flow reaches a steady state. If not, repeat from Step 2. Otherwise, stop and output the results.

2.5 Numerical examples

The proposed method can be applied to any type of RBF (e.g. Gaussian, multiquadric (MQ) and inverse multiquadric). However, it has generally been accepted that, among RBFs, the multiquadric (MQ) function tends to result in the most accurate approximation (Franke, 1982). We choose the following MQ as the basis function here as well as in the remaining chapters.

$$G_i(\mathbf{x}) = \sqrt{(\mathbf{x} - \mathbf{c}_i)^T(\mathbf{x} - \mathbf{c}_i) + a_i^2}, \tag{2.44}$$

where $\mathbf{x} = (x, y)^T$ is the position vector of the point of interest; $\mathbf{c}_i = (x_{c_i}, y_{c_i})^T$ and a_i the position vector of the centre and the width of the ith MQ, respectively. In 1D form, MQ function in (2.44) becomes

$$G_i(\eta) = \sqrt{(\eta - c_i)^2 + a_i^2}, \tag{2.45}$$

where η represents x or y.

Analytic forms of the integrated MQ basis functions are given below (Mai-Duy and Tran-Cong, 2011).

$$H_i(\eta) = \int G_i(\eta)d\eta = \frac{r}{2}Q + \frac{a_i^2}{2}R, \tag{2.46}$$

$$\overline{H}_i(\eta) = \int H_i(\eta)d\eta = \left(\frac{r^2}{6} - \frac{a_i^2}{3}\right)Q + \frac{a_i^2 r}{2}R, \tag{2.47}$$

where $r = \|\eta - c_i\|$; $Q = \sqrt{r^2 + a_i^2}$; and $R = \ln(r + Q)$. The plots of 3 functions $G(\eta)$, $H(\eta)$ and $\overline{H}(\eta)$ are shown in Figure 2.3. For each stencil, the set of nodal

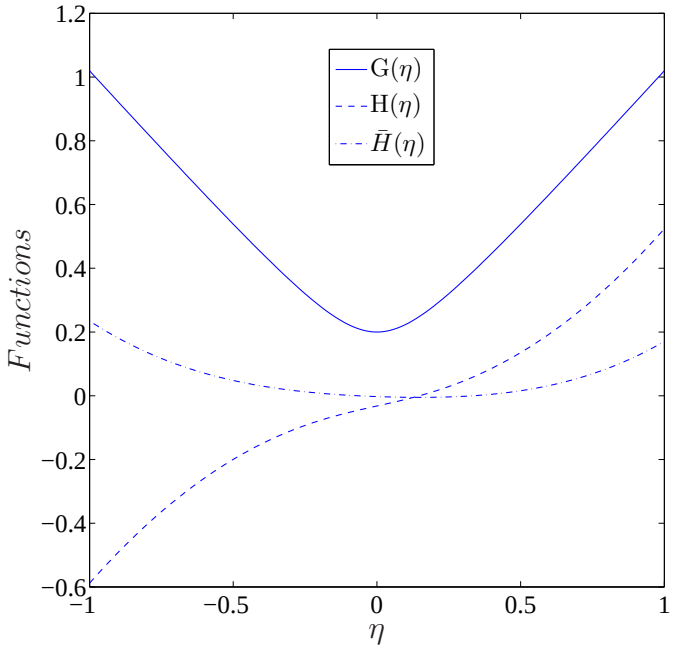


Figure 2.3: The plots of basis functions employed in the present studies with $c = 0$ and $a = 0.2$.

points is taken to be the set of MQ centres. We simply choose the MQ width as $a_i = \beta h_i$ in which β is a given positive number and h_i the distance between the ith node and its nearest neighbouring node. We assess the performance of the proposed method through two measures:

- the root mean square (RMS) error defined as

$$RMS = \frac{\sqrt{\sum_{i=1}^{N}(u_i - \bar{u}_i)^2}}{\sqrt{N}}, \tag{2.48}$$

where N is the number of nodes over the whole domain and $\bar{u}$ is the exact solution, and

- let Ne denote the error norm (i.e. RMS in this chapter). The average rate of convergence with respect to the grid refinement is determined by considering the error norm behaviour as $Ne \approx O(h^{\alpha})$, where the value of the convergence rate α is determined by the method of least squares with the data taken in the logarithmic scale, and h is the average grid spacing. For the local rates of convergence, they are given by

$$\alpha = \frac{\log(Ne^{(r)}/Ne^{(s)})}{\log(h^{(r)}/h^{(s)})}, \tag{2.49}$$

where the superscripts (r) and (s) indicate the data obtained from the rth and sth calculations $(r < s)$, respectively.

The proposed method is verified through the solution of a linear second-order ODE and the simulation of viscous flows to obtain their structures at the steady state. For the latter, the steady state is considered to have been reached when

$$\sqrt{\frac{\sum_{i=1}^{N}(u_i^n - u_i^{n-1})^2}{N}} < 10^{-9}, \tag{2.50}$$

where u^n and u^{n-1} are the approximate solutions at the current and previous time levels, respectively.

2.5.1 Ordinary differential equation (ODE)

As a first test, we consider the following boundary-value second-order problem

$$\frac{d^2u}{dx^2} = -(2\pi)^2 \sin(2\pi x), \quad 0 \le x \le 1, \quad u(0) = u(1) = 0. \tag{2.51}$$

The exact solution to this problem can be verified to be $\bar{u}(x) = \sin(2\pi x)$. We add a pseudo time-derivative term to equation (2.51) to facilitate an iterative calculation

$$\frac{d^2u}{dx^2} = -(2\pi)^2 \sin(2\pi x) + \frac{du}{dt}. \tag{2.52}$$

When the difference of u between two successive time levels is small, i.e. $\frac{du}{dt} \simeq 0$ (the iterative process is said to converge), the obtained solution is also a solution to (2.51).

In the present calculation, a time step of 0.5 is used. It is noted that the higher the value of a permissible time step, the faster the convergence of the solution

will be. One can reduce the grid size h and/or vary the MQ width β to enhance the solution accuracy.

For $\beta-$adaptivity study, the value of β is chosen in a wide range of 2-100. Results obtained at a grid of 51 are shown in Figure 2.4. As β increases, the error Ne reduces significantly. However, at very large values of β, the behaviour of Ne becomes unstable. It appears that the optimal value of β is 8 and the corresponding condition number of the system matrix is 5.84×10^8. It is noted that, from a theoretical point of view, it is still not clear how to choose the optimal value of the MQ width. Unlike global IRBF versions (where β=1 is a preferred value), the present compact IRBF scheme can work well with a wide range of β ($20 \leq \beta \leq 60$).

For $h-$adaptivity study, the present IRBF and standard second-order FD calculations are conducted on various sets of uniformly distributed points, from 5 to 51 with an increment of 2. Results obtained by the two methods are given in Figure 2.5. It can be seen that the present scheme outperforms the FDM in terms of the solution accuracy and convergence rate, whereas the two schemes have similar values of the matrix condition number.

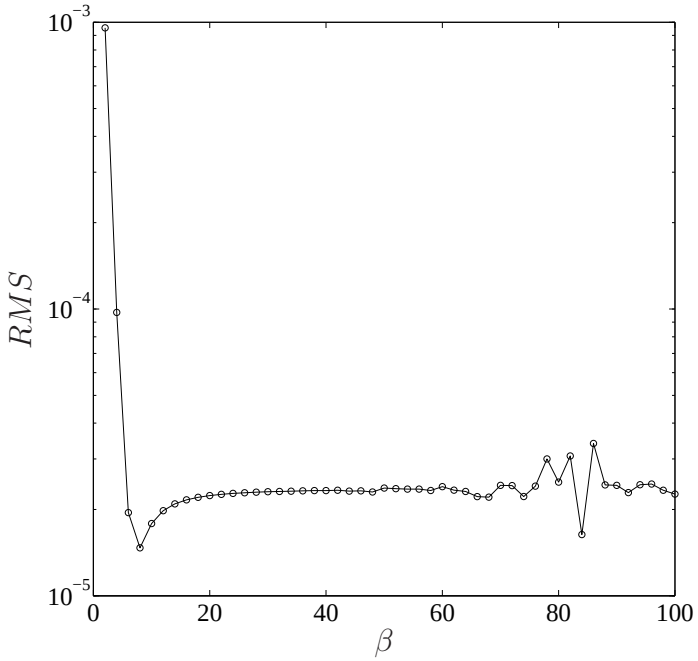
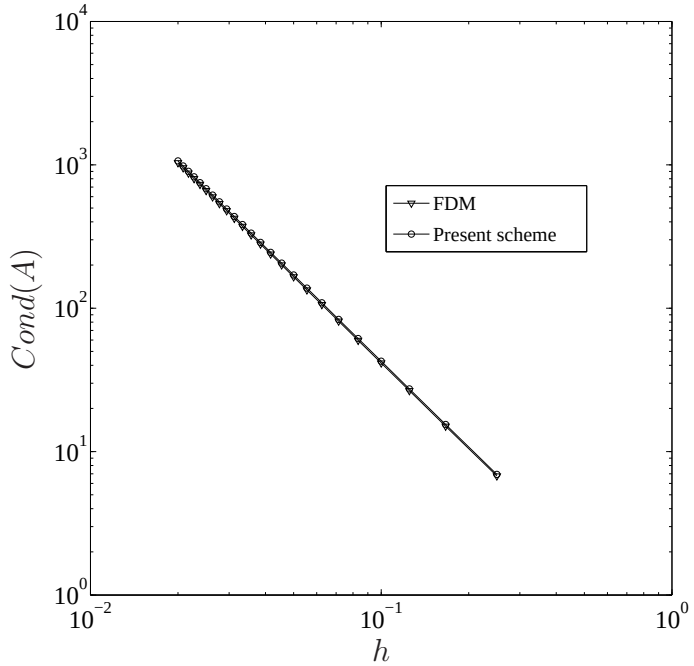
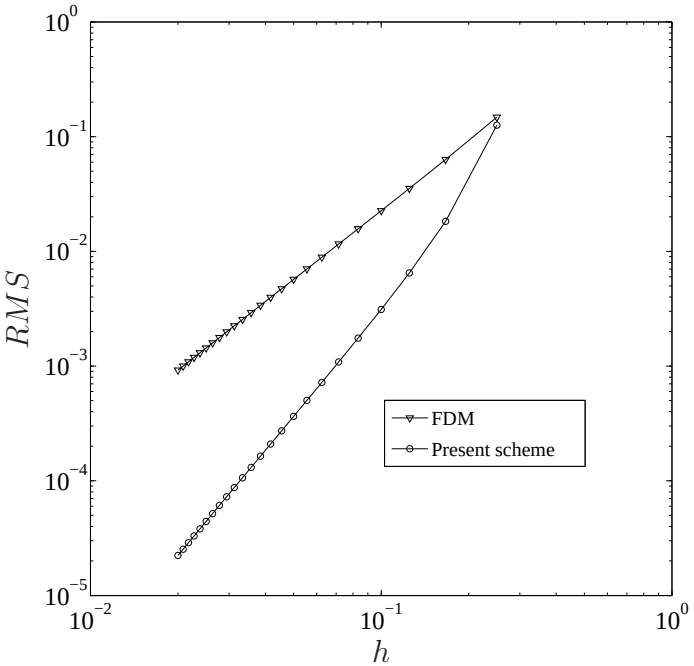


Figure 2.4: ODE, $N = 51$: the effects of the MQ width β on the solution accuracy.



(a)



(b)

Figure 2.5: ODE, $\beta = 20$, $N \in \{5,\ 7,\ 9, ...,\ 51\}$: the effects of the grid size h on the system matrix condition (a) and the solution accuracy (b) for the FDM and the present scheme. The matrix condition number grows as $O(h^{-2})$ for the two methods while the solution converges as $O(h^2)$ for FDM and $O(h^{3.23})$ for the IRBF method.

2.5.2 Analytic Stokes flow

Consider a Stokes flow (Fadel and Agouzoul, 2011). Equations (2.1)-(2.2) reduce to

$$\nabla.\mathbf{u} = 0, \tag{2.53}$$

$$\frac{\partial \mathbf{u}}{\partial t} = -\nabla p + \nabla^2 \mathbf{u} + \mathbf{f}_b. \tag{2.54}$$

The exact solution is chosen as

$$\overline{u}(x,y) = 2\pi x^2(1-x)^2 \sin(\pi y)\cos(\pi y), \tag{2.55}$$

$$\overline{v}(x,y) = -2x(x-1)(2x-1)\sin^2(\pi y), \tag{2.56}$$

$$\overline{p}(x,y) = \sin(x)\cos(y), \tag{2.57}$$

from which, one can derive the corresponding body force

$$f_{b_x} = 2\pi(-1+6x+2(\pi^2-3)x^2-4\pi^2x^3+2\pi^2x^4)\sin(2\pi y)+\cos(x)\cos(y), \tag{2.58}$$

$$f_{b_y} = 4\pi^2 x(1-3x+2x^2)\cos(2\pi y)-12(1-2x)\sin^2(\pi y)-\sin(x)\sin(y). \tag{2.59}$$

Let $\Omega = [0,1]\times[0,1]$ be the flow domain. Values of u and v are prescribed on the boundaries according to (2.55) and (2.56), respectively, whereas the hydrostatic pressure mode is eliminated by fixing the pressure value at a single node. In the present study, we take the centre of the cavity as a reference point.

The simulations are performed for a set of uniform grids, namely $\{11\times 11,\ 21\times 21, \cdots,\ 51\times 51\}$. In Table 2.1, we present an accuracy analysis with respect to grid refinement for results obtained by the proposed scheme and the standard second-order central FD scheme. It can be seen that the former outperforms the latter regarding both the solution accuracy and the rate of convergence. The overall convergence rates are of $O(h^{3.01})$ for u, $O(h^{3.11})$ for v and $O(h^{2.88})$ for p by the proposed scheme, while the corresponding values are $O(h^{2.09})$, $O(h^{2.18})$ and $O(h^{2.78})$ by the FDM.

2.5.3 Recirculating cavity flow driven by combined shear and body forces

This problem is taken from Shih and Tan (1989) which has an exact solution. The recirculating flow of a Newtonian fluid in a square cavity (Figure 2.6) is induced by the combined shear and body forces. The governing equations are of the form (2.1)-(2.2). It is assumed that the velocity profile along the top boundary is $v = 0$ and $u(x,1) = 16x^2(x-1)^2$ while the other walls are non-slip and stationary. The body force, which is present in the y-direction only, is given

Table 2.1: Stokes flow: RMS errors, local and overall convergence rates for u, v and p by the proposed method and FDM. The overall convergence rate α is presented in the form of $O(h^{\alpha})$.

Grid	$Ne(u)$	Rate	$Ne(v)$	Rate	$Ne(p)$	Rate
			Present method			
11×11	6.5648E-04	—-	5.3296E-04	—-	1.7813E-02	—-
21×21	8.3206E-05	2.98	6.0128E-05	3.15	2.3210E-03	2.94
31×31	2.4489E-05	3.02	1.6978E-05	3.12	7.2212E-04	2.88
41×41	1.0289E-05	3.01	7.0329E-06	3.06	3.2314E-04	2.80
51×51	5.1893E-06	3.07	3.6338E-06	2.96	1.7496E-04	2.75
		$O(h^{3.01})$		$O(h^{3.11})$		$O(h^{2.88})$
			FDM			
11×11	3.9284E-03	—-	1.3077E-03	—-	5.8633E-02	—-
21×21	8.7393E-04	2.17	2.0142E-04	2.70	7.0630E-03	3.05
31×31	3.8109E-04	2.05	9.6704E-05	1.81	2.3485E-03	2.72
41×41	2.1146E-04	2.05	5.8460E-05	1.75	1.1005E-03	2.63
51×51	1.3579E-04	1.98	3.7685E-05	1.97	6.8175E-04	2.15
		$O(h^{2.09})$		$O(h^{2.18})$		$O(h^{2.78})$

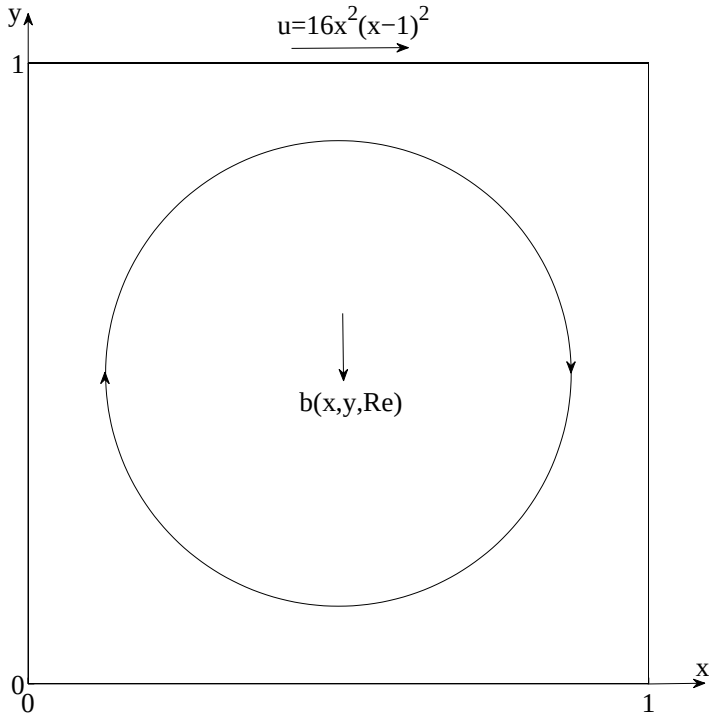


Figure 2.6: Recirculating cavity flow: A schematic diagram of the physical domain (non-dimensionalised).

by

$$f_{b_x} = 0, \tag{2.60}$$

$$f_{b_y} = -\frac{8}{Re}\left[24F(x) + 2f'(x)g''(y) + f'''(x)g(y)\right] - 64\left[F_2(x)G_1(y) - g(y)g'(y)F_1(x)\right], \tag{2.61}$$

where

$$f(x) = x^4 - 2x^3 + x^2, \tag{2.62}$$

$$g(y) = y^4 - y^2, \tag{2.63}$$

$$F(x) = \int f(x)dx = 0.2x^5 - 0.5x^4 + x^3/3, \tag{2.64}$$

$$F_1(x) = f(x)f''(x) - [f'(x)]^2 = -4x^6 + 12x^5 - 14x^4 + 8x^3 - 2x^2, \tag{2.65}$$

$$F_2(x) = \int f(x)f'(x)dx = 0.5[f(x)]^2, \tag{2.66}$$

$$G_1(y) = g(y)g'''(y) - g'(y)g''(y) = -24y^5 + 8y^3 - 4y. \tag{2.67}$$

The exact solution to this problem is known to be

$$\overline{u}(x,y) = 8f(x)g'(y) = 8(x^4 - 2x^3 + x^2)(4y^3 - 2y), \tag{2.68}$$

$$\overline{v}(x,y) = -8f'(x)g(y) = -8(4x^3 - 6x^2 + 2x)(y^4 - y^2), \tag{2.69}$$

$$\overline{p}(x,y,Re) = \frac{8}{Re}\left[F(x)g'''(y) + f'(x)g'(y)\right] + 64F_2(x)\left[g(y)g''(y) - [g'(y)]^2\right]. \tag{2.70}$$

We employ several uniform grids, namely $\{21\times 21,\ 31\times 31, \ldots,\ 71\times 71\}$, and the two previously discussed boundary treatments for the pressure to simulate the flow. Table 2.2 compares the present results with those obtained by FD approximation schemes. In the case of IRBFs, the imposition of the pressure boundary condition in the Dirichlet form (Treatment 2) yields more accurate results than those in the Neumann form (Treatment 1). In the case of FDs, the two treatments have similar performances. The IRBF solutions are seen to be more accurate and to converge faster than the FD ones. To achieve a similar level of accuracy, the FDM requires a denser grid than the proposed scheme. For example, with Treatment 1, RMS errors of the p solution are 2.9×10^{-4} using a grid of 61×61 for the former and 2.8×10^{-4} using a grid of 21×21 for the latter. Figure 2.7 shows profiles of the velocity on the horizontal and vertical centrelines of the cavity, which are in very good agreement with the exact solution.

2.5.4 Lid-driven cavity flow

It differs from the previous problem in that the velocity of the lid is now prescribed as $\mathbf{u} = (1,0)^T$ and the body force components are set to zeros. There are thus two values of u at the two top corners, making the stress solution there singular. The singular lid-driven cavity flow is widely used as a test problem for the assessment of accuracy of numerical solvers in CFD. From the literature, FD results using very dense grids by Ghia et al. (1982) and pseudo-spectral results by Botella and Peyret (1998) have been often cited for comparison purposes. It is noted that for the latter, the field variables were decomposed into the regular part that is approximated with Chebyshev polynomials and the singular part that is treated analytically; and a benchmark spectral solution for $Re = 100$ and $Re = 1000$ were provided.

We use Treatment 2 only in the imposition of the pressure boundary condition. A wide range of Re, namely $\{100,\ 400,\ 1000,\ 3200\}$ and uniform grids, namely $\{11\times 11,\ 31\times 31,\ 51\times 51,\ 71\times 71,\ 91\times 91,\ 111\times 111,\ 129\times 129\}$ are considered in the simulation. The time step is chosen in the range of 0.1 to 0.00025. Smaller time steps are employed for higher Re and higher grid densities. Tables 2.3, 2.4

Table 2.2: Recirculating cavity flow, $Re = 100$: RMS errors and local convergence rates for u, v and p.

	Present (Dirichlet)		Present (Neumann)		FDM (Dirichlet)		FDM (Neumann)	
Grid	$Ne(u)$	Rate	$Ne(u)$	Rate	$Ne(u)$	Rate	$Ne(u)$	Rate
21×21	3.7323E-04	—-	6.8994E-04	—-	2.7515E-03	—-	2.8806E-03	—-
31×31	8.8758E-05	3.54	3.0028E-04	2.05	1.2587E-03	1.93	1.2598E-03	2.04
41×41	3.3153E-05	3.42	1.8214E-04	1.74	7.2116E-04	1.94	7.1861E-04	1.95
51×51	1.6052E-05	3.25	1.0607E-04	2.42	4.7417E-04	1.88	4.7156E-04	1.89
61×61	8.8940E-06	3.24	6.6161E-05	2.59	3.3465E-04	1.91	3.3254E-04	1.92
71×71	5.2855E-06	3.38	3.8683E-05	3.48	2.5021E-04	1.89	2.4875E-04	1.88
Grid	$Ne(v)$	Rate	$Ne(v)$	Rate	$Ne(v)$	Rate	$Ne(v)$	Rate
21×21	3.0814E-04	—-	1.1666E-03	—-	3.3290E-03	—-	3.0724E-03	—-
31×31	6.9064E-05	3.69	4.3065E-04	2.46	1.5300E-03	1.92	1.5246E-03	1.73
41×41	2.7564E-05	3.19	2.3730E-04	2.07	8.7191E-04	1.95	8.9158E-04	1.86
51×51	1.3899E-05	3.07	1.3227E-04	2.62	5.5126E-04	2.05	5.6429E-04	2.05
61×61	8.2254E-06	2.88	7.9870E-05	2.77	3.8059E-04	2.03	3.8919E-04	2.04
71×71	5.2222E-06	2.95	4.6700E-05	3.48	2.7666E-04	2.07	2.8119E-04	2.11
Grid	$Ne(p)$	Rate	$Ne(p)$	Rate	$Ne(p)$	Rate	$Ne(p)$	Rate
21×21	2.8508E-04	—-	7.3830E-04	—-	2.5036E-03	—-	5.8569E-03	—-
31×31	6.2890E-05	3.73	2.9702E-04	2.25	1.1474E-03	1.92	2.7822E-03	1.84
41×41	2.3775E-05	3.38	1.7771E-04	1.79	6.5178E-04	1.97	1.5035E-03	2.14
51×51	1.2035E-05	3.05	1.0679E-04	2.28	4.1769E-04	1.99	9.5988E-04	2.01
61×61	7.0999E-06	2.89	6.7371E-05	2.53	2.9038E-04	1.99	6.6316E-04	2.03
71×71	4.5087E-06	2.95	4.0092E-05	3.37	2.1349E-04	2.00	4.9218E-04	1.93

Table 2.3: Lid-driven cavity flow, $Re = 100$: Extrema of the vertical and horizontal velocity profiles along the horizontal and vertical centrelines, respectively, of the cavity. "Errors" are relative to the "Benchmark" solution.

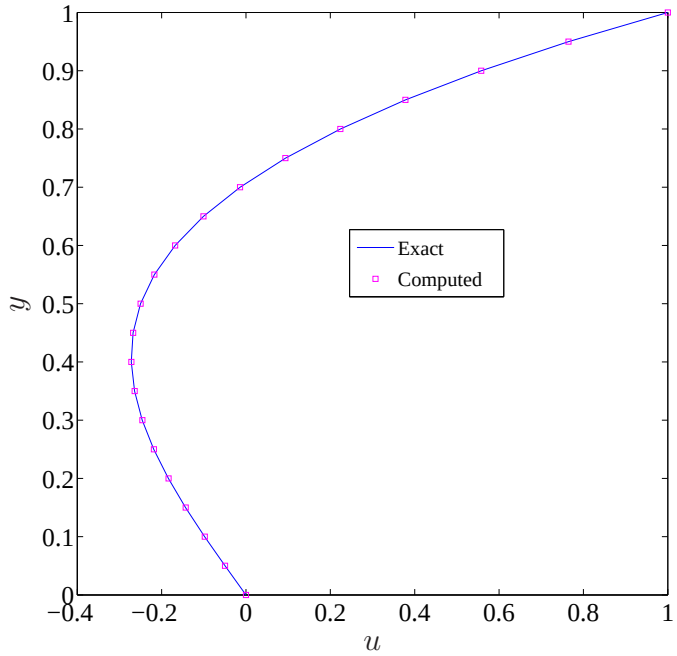
Method	Grid	u_{min}	Error (%)	y_{min}	v_{max}	Error (%)	x_{max}	v_{min}	Error (%)	x_{min}
Present	11×11	-0.1912173	10.66	0.4807	0.1595908	11.13	0.2307	-0.2236027	11.90	0.8136
Present	31×31	-0.2102259	1.78	0.4578	0.1768808	1.50	0.2370	-0.2501843	1.43	0.8107
Present	51×51	-0.2121503	0.88	0.4579	0.1781849	0.77	0.2372	-0.2520400	0.69	0.8107
FDM ($\psi - \omega$) (Ghia et al., 1982)	129×129	-0.2109	1.47	0.4531	0.17527	2.40	0.2344	-0.24533	3.34	0.8047
FDM ($\mathbf{u} - p$) (Bruneau and Saad, 2006)	129×129	-0.2106	1.61	0.4531	0.1786	0.54	0.2344	-0.2521	0.67	0.8125
FVM ($\mathbf{u} - p$) (Sahin and Owens, 2003)	257×257	-0.213924	0.06	0.4598	0.180888	0.73	0.2354	-0.256603	1.10	0.8127
FVM ($\mathbf{u} - p$), cpi. (Deng et al., 1994)	128×128	-0.21315	0.42	—	0.17896	0.34	—	-0.25339	0.16	—
Benchmark (Botella and Peyret, 1998)		-0.2140424		0.4581	0.1795728		0.2370	-0.2538030		0.8104

Table 2.4: Lid-driven cavity flow, $Re = 1000$: Extrema of the vertical and horizontal velocity profiles along the horizontal and vertical centrelines, respectively, of the cavity. "Errors" are relative to the "Benchmark" solution.

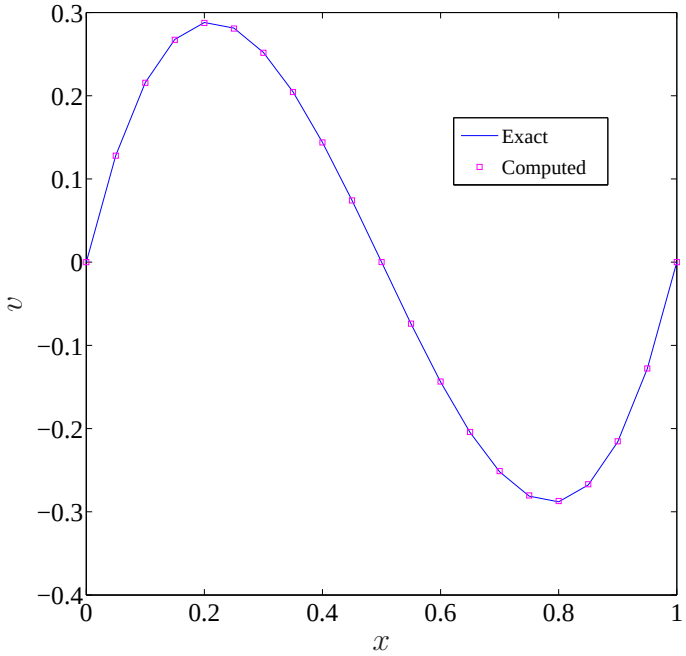
Method	Grid	u_{min}	Error (%)	y_{min}	v_{max}	Error (%)	x_{max}	v_{min}	Error (%)	x_{min}
Present	51×51	-0.3629562	6.59	0.1787	0.3515585	6.73	0.1637	-0.4898251	7.07	0.9052
Present	71×71	-0.3755225	3.36	0.1753	0.3637009	3.51	0.1608	-0.5086961	3.49	0.9078
Present	91×91	-0.3815923	1.80	0.1735	0.3698053	1.89	0.1594	-0.5174658	1.82	0.9085
Present	111×111	-0.3840354	1.17	0.1728	0.3722634	1.24	0.1588	-0.5209683	1.16	0.9088
Present	129×129	-0.3848064	0.97	0.1724	0.3729119	1.07	0.1586	-0.5223350	0.90	0.9089
FDM ($\psi - \omega$) (Ghia et al., 1982)	129×129	-0.38289	1.46	0.1719	0.37095	1.59	0.1563	-0.5155	2.20	0.9063
FDM ($\mathbf{u} - p$) (Bruneau and Saad, 2006)	256×256	-0.3764	3.13	0.1602	0.3665	2.77	0.1523	-0.5208	1.19	0.9102
FVM ($\mathbf{u} - p$), cpi. (Deng et al., 1994)	128×128	-0.38511	0.89	—	0.37369	0.86	—	-0.5228	0.81	—
Benchmark (Botella and Peyret, 1998)		-0.3885698		0.1717	0.3769447		0.1578	-0.5270771		0.0908

Table 2.5: Lid-driven cavity flow: Extrema of the vertical and horizontal velocity profiles along the horizontal and vertical centrelines, respectively, of the cavity at different Reynolds numbers $Re \in \{400,\ 3200\}$.

Re	Method	Grid	u_{min}	y_{min}	v_{max}	x_{max}	v_{min}	x_{min}
400	Present	31×31	-0.316205	0.2833	0.293696	0.2236	-0.435578	0.8583
	Present	51×51	-0.323158	0.2814	0.297493	0.2248	-0.442770	0.8605
	Present	71×71	-0.325168	0.2804	0.300818	0.2252	-0.449146	0.8620
	FDM $(\psi - \omega)$ (Ghia et al., 1982)	129×129	-0.32726	0.2813	0.30203	0.2266	-0.44993	0.8594
	FVM $(\mathbf{u} - p)$, cpi. (Deng et al., 1994)	128×128	-0.32751	—	0.30271	—	-0.45274	—
	FVM $(\mathbf{u} - p)$ (Sahin and Owens, 2003)	257×257	-0.328375	0.2816	0.304447	0.2253	-0.456316	0.8621
3200	Present	91×91	-0.406818	0.0983	0.403852	0.1016	-0.528864	0.9451
	Present	111×111	-0.418545	0.0962	0.415776	0.0995	-0.544789	0.9462
	Present	129×129	-0.423061	0.0963	0.420565	0.0994	-0.551563	0.9466
	FDM $(\psi - \omega)$ (Ghia et al., 1982)	129×129	-0.41933	0.1016	0.42768	0.0938	-0.54053	0.9453
	FVM $(\mathbf{u} - p)$ (Sahin and Owens, 2003)	257×257	-0.435402	0.0921	0.432448	0.0972	-0.569145	0.9491



(a)



(b)

Figure 2.7: Recirculating cavity flow, Treatment 2, $Re = 100$: Variations of u along the vertical centreline (a) and v along the horizontal centreline (b) by the present scheme using a grid of 21×21 and the exact solution (Shih and Tan, 1989).

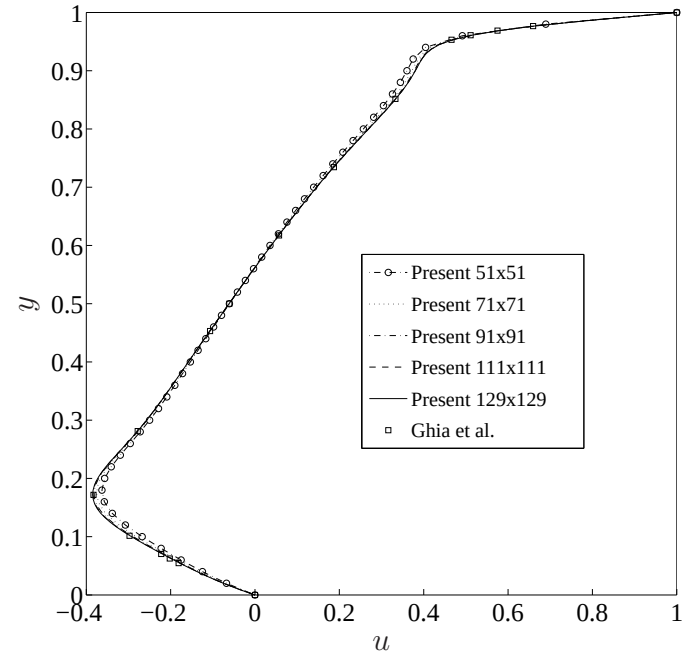
and 2.5 show the present results for the extrema of the velocity profiles along the centrelines of the cavity for several Reynolds numbers in comparison with some others (Ghia et al., 1982; Deng et al., 1994; Botella and Peyret, 1998; Sahin and Owens, 2003; Bruneau and Saad, 2006). For $Re = 100$ (Table 2.3) and $Re = 1000$ (Table 2.4), the "errors" are calculated relative to a "benchmark" solution (Botella and Peyret, 1998), which shows that the present results are very comparable with others.

Velocity profiles along the vertical and horizontal centrelines for different grid sizes at $Re = 1000$ are displayed in Figure 2.8, where a grid convergence of the IRBF solution is clearly observed (i.e. the present solution approaches the benchmark solution very fast as the grid density is increased). We virtually achieve the benchmark solution with only 91×91 grid in comparison with a grid of 129×129 used to obtain the benchmark solution in Ghia et al. (1982). In addition, those velocity profiles at $Re \in \{100, 400, 1000, 3200\}$ with the grid of 129×129 are also shown in Figure 2.9, where the present solutions match the benchmark ones very well.

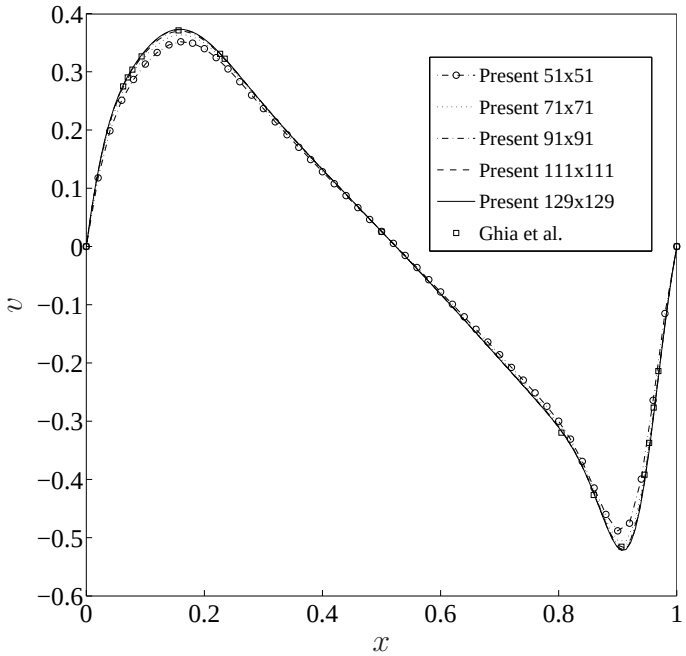
Figure 2.10 exhibits the distributions of the pressure for the flow at $Re \in \{100, 400, 1000, 3200\}$ which look feasible in comparison with those reported in the literature. We also show streamlines and iso-vorticity lines, which are derived from the velocity field, for the flow at $Re \in \{100, 400, 1000, 3200\}$ in Figure 2.11 and 2.12, where secondary vortices are well captured.

2.6 Concluding remarks

In this chapter, we propose a high-order compact local IRBF scheme for the discretisation of the pressure-velocity formulation in the Cartesian-grid point-collocation framework. Two boundary treatments for the pressure, one is based on values of the pressure and the other based on normal derivative values of the pressure, are studied. Like FDMs, the present approximations involve 3 nodes in each direction, resulting in a sparse system matrix. Numerical examples indicate that (i) the present results are superior to the FD results in terms of the solution accuracy and the convergence rate with grid refinement, and (ii) the imposition of boundary conditions for the pressure yields better results in the Dirichlet form than in the Neumann form.



(a)



(b)

Figure 2.8: Lid-driven cavity flow, $Re = 1000$: Profiles of the u-velocity along the vertical centreline (a) and the v-velocity along the horizontal centreline (b) using several grids. Note that curves for the last three grids are indistinguishable and agree well with the benchmark FD results.

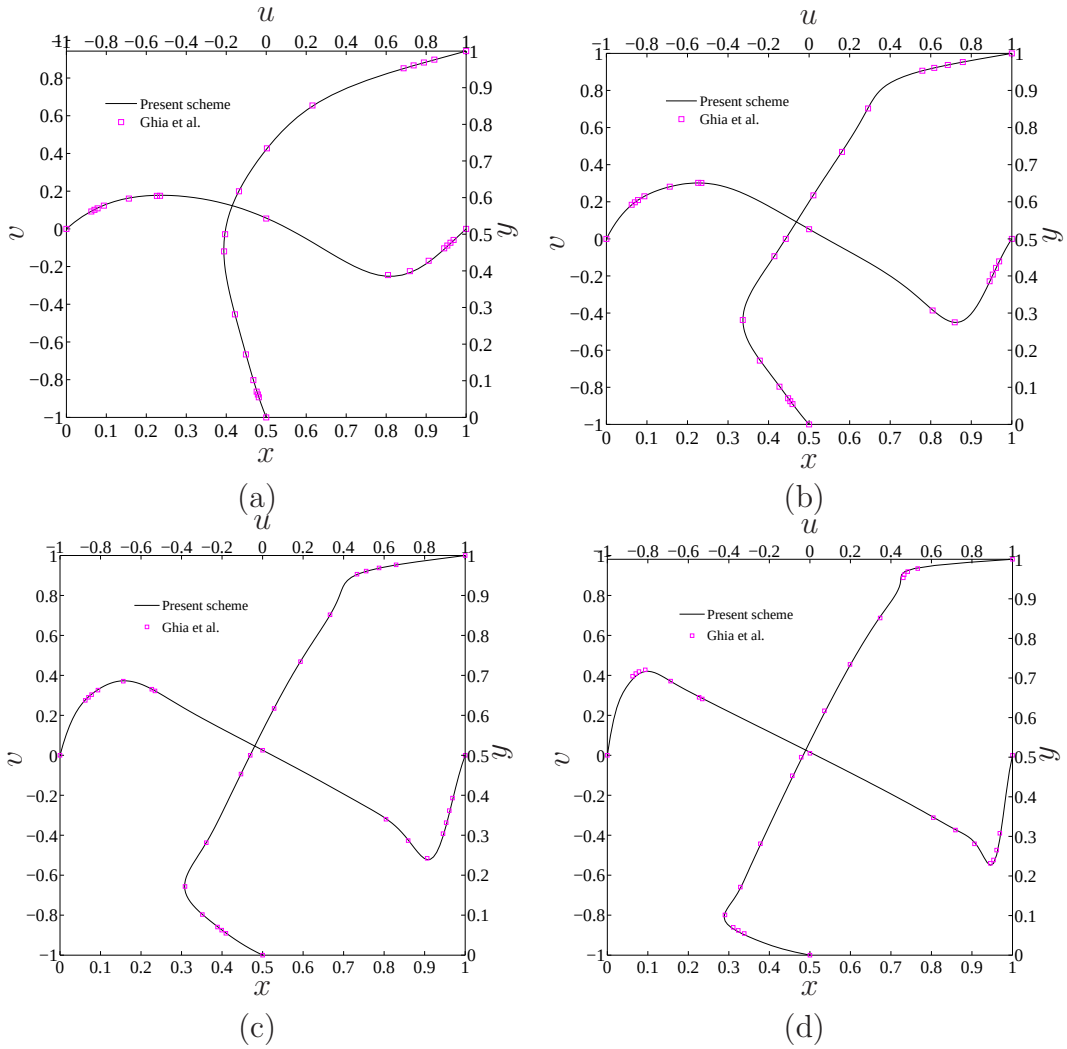
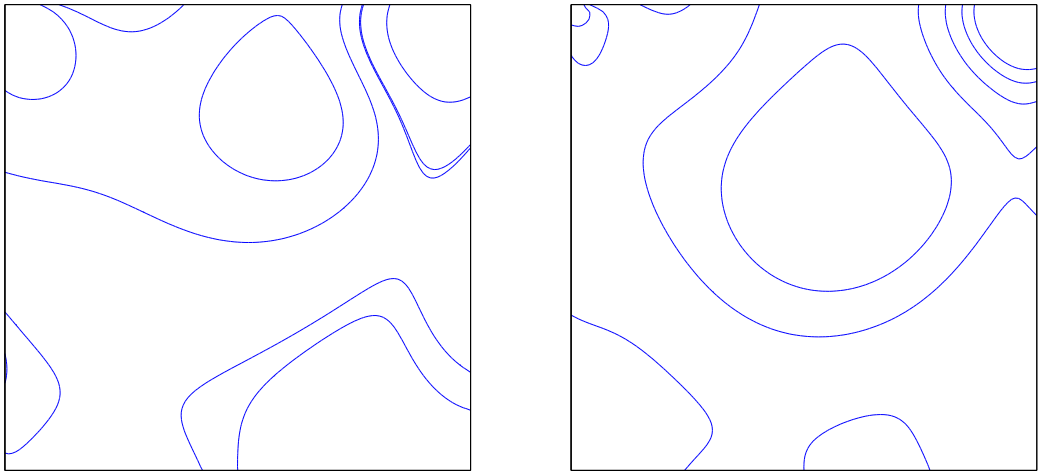
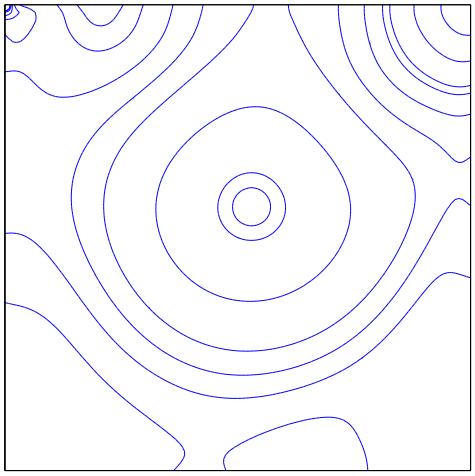


Figure 2.9: Lid-driven cavity flow, 129×129: Profiles of the u-velocity along the vertical centreline and the v-velocity along the horizontal centreline for $Re = 100$ (a), $Re = 400$ (b), $Re = 1000$ (c) and $Re = 3200$ (d).

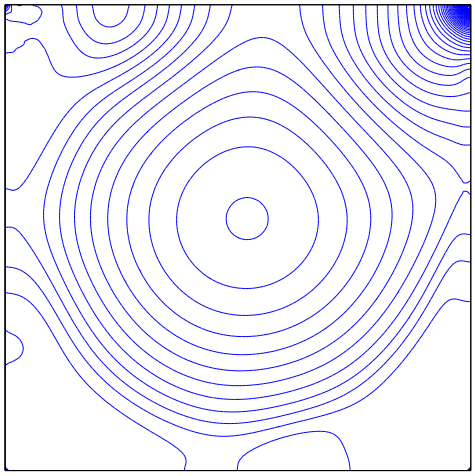


(a)

(b)

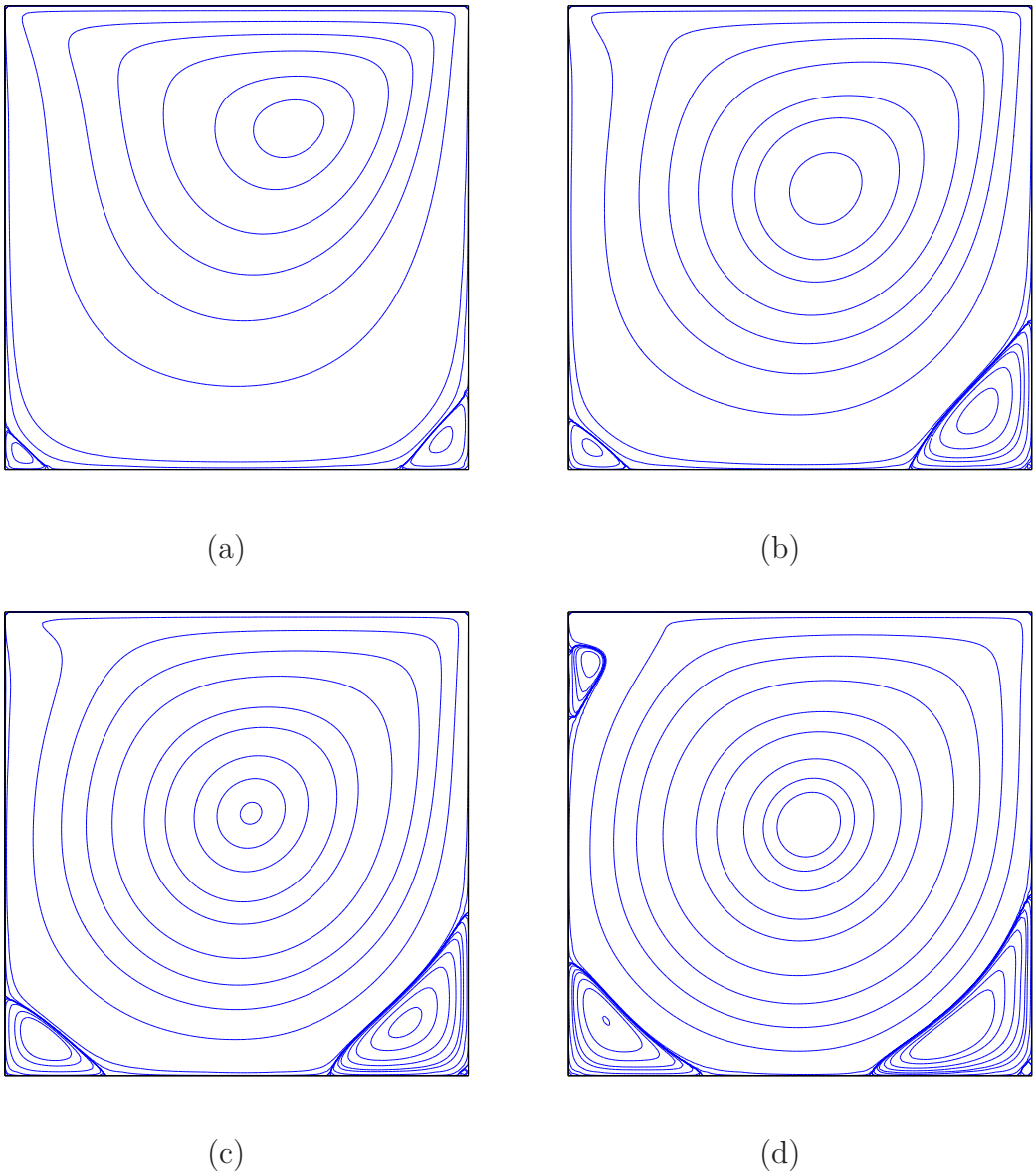


(c)



(d)

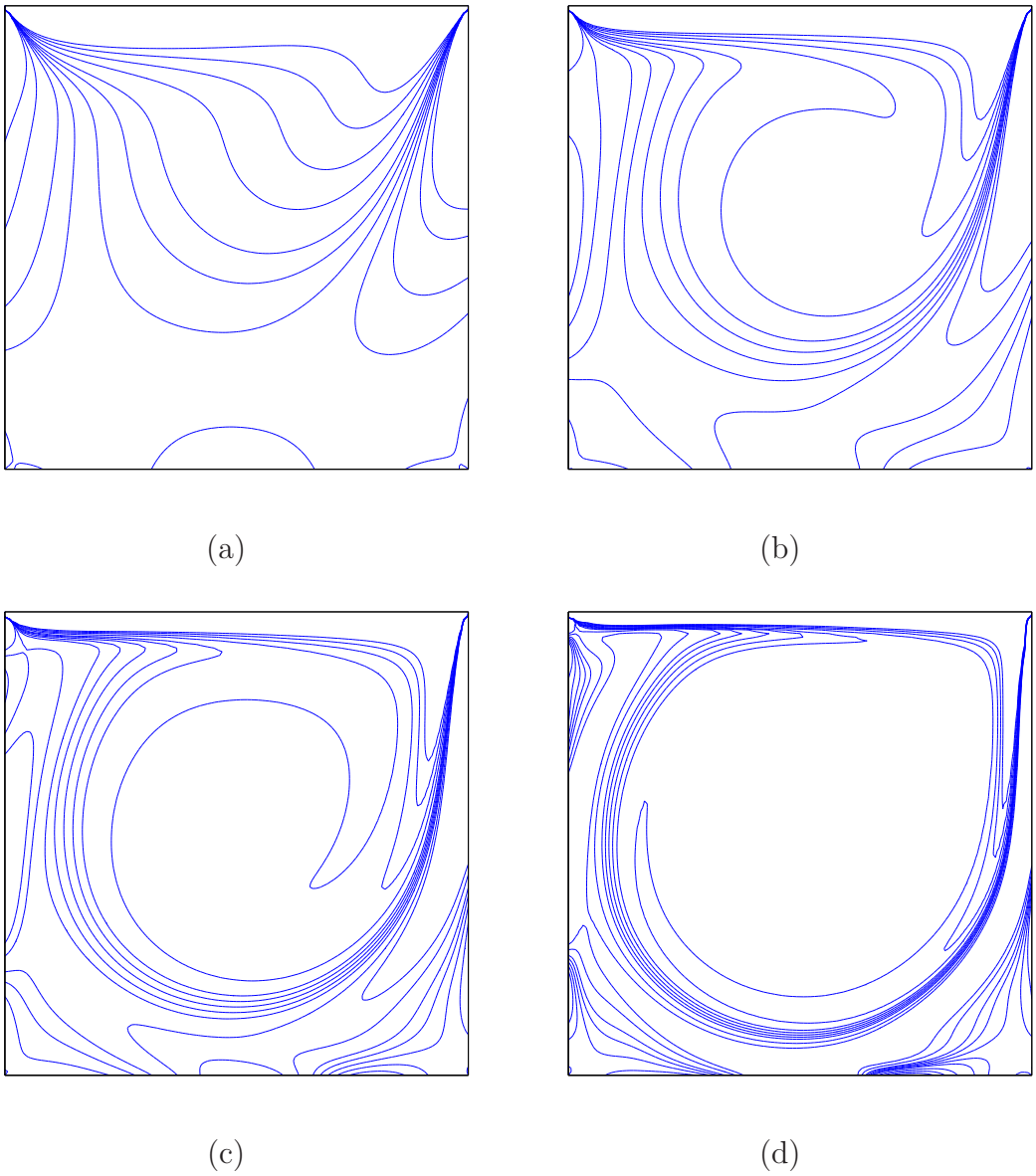
Figure 2.10: Lid-driven cavity flow, 129×129: Isobaric lines of the flow for $Re = 100$ (a), $Re = 400$ (b), $Re = 1000$ (c) and $Re = 3200$ (d). The contour values used here are taken to be the same as those in Abdallah (1987), Botella and Peyret (1998) and Bruneau and Saad (2006).



(a) (b)

(c) (d)

Figure 2.11: Lid-driven cavity flow, 129×129: Streamlines of the flow for $Re = 100$ (a), $Re = 400$ (b), $Re = 1000$ (c) and $Re = 3200$ (d). The contour values used here are taken to be the same as those in Ghia et al. (1982).



(a) (b)

(c) (d)

Figure 2.12: Lid-driven cavity flow, 129×129: Iso-vorticity lines of the flow for $Re = 100$ (a), $Re = 400$ (b), $Re = 1000$ (c) and $Re = 3200$ (d). The contour values used here are taken to be the same as those in Ghia et al. (1982).

Chapter 3

A compact IRBF scheme for transient flows

This chapter presents a high-order approximation scheme based on CIRBF stencils and second-order Adams-Bashforth/Crank-Nicolson algorithms for solving time-dependent problems in one and two space dimensions. We employ CIRBF stencils, where the RBF approximations are locally constructed through integration and expressed in terms of nodal values of the function and its derivatives, to discretise the spatial derivatives in the governing equation. We adopt the Adams-Bashforth and Crank-Nicolson algorithms, which are second-order accurate, to discretise the temporal derivatives. The performance of the proposed scheme is investigated numerically through the solution of several test problems, including heat transfer governed by the diffusion equation, shock wave propagation and shock-like solution governed by the Burgers' equation, and torsionally oscillating lid-driven cavity flow governed by the Navier-Stokes equation in the primitive variables. Numerical experiments show that the proposed scheme is stable and high-order accurate in reference to the exact solution of analytic test problems and achieves a good agreement with published results for other test problems.

3.1 Introduction

High-order approximation schemes have the ability to provide efficient solutions to time-dependent differential problems. A high level of accuracy can be achieved using a relatively coarse discretisation. Many types of high-order schemes have been reported in the literature. For example, in solving the diffusion equation, Kouatchou (2001) combined a high-order compact finite difference approximation and a collocation technique. Sun and Zhang (2003) pro-

posed a high-order compact boundary value method. Gupta et al. (2005) proposed a high-order FD approximation defined on a square mesh stencil using nine nodes. For Burgers' equation, introduced originally by Bateman (1915), Hassanien et al. (2005) proposed a fourth-order FD method based on two-level three-point approximations of order 2 in time and 4 in space while Zhu and Wang (2009) presented a method based on cubic B-spline quasi-interpolations. Recently, Hosseini and Hashemi (2011) presented a local-RBF meshless method dealing with several initial and boundary conditions. It is noted that the Burgers' equation is considered as a good means of verifying new numerical methods in CFD (Caldwell et al., 1987; Iskander and Mohsen, 1992) because the equation contains both the convection and diffusion terms. For the Navier-Stokes equation, Botella and Peyret (1998) developed a Chebyshev collocation method and provided the benchmark spectral results for the lid driven cavity flow problem. Ding et al. (2006) presented a local multiquadric differential quadrature method for the solution of three-dimensional incompressible flow problems in the velocity-pressure formulation while Mai-Duy et al. (2008), Mai-Duy and Tran-Cong (2008), Le-Cao et al. (2009) proposed an integrated RBF (IRBF) method to solve fluid flow and thermal problems in the stream function-vorticity formulation. Recently, Tian et al. (2011) proposed a fourth-order compact difference scheme constructed on 2D nine-point stencils and Fadel and Agouzoul (2011) used a standard Padé scheme to construct high-order approximations in a velocity-pressure-pressure gradient formulation. It is noted that the velocity ($\mathbf{u}$) and pressure (p) formulation has several advantages over both the stream function-vorticity formulation and the stream function formulation. The $\mathbf{u}$-p formulation can provide both the velocity field and the pressure field from solving the discretised equations and also works for 2D and 3D problems in a similar manner.

In this chapter, we present a high-order approximation scheme based on CIRBF stencils and second-order Adams-Bashforth/Crank-Nicolson algorithms for solving time-dependent problems, where emphasis is placed on the treatment of the extra information in the compact stencils. A series of test problems is considered to verify the proposed scheme. It will be shown that the present scheme is stable and high-order accurate. For example, the CIRBF solution converges very fast with grid refinement at a rate of up to 4.47 (i.e. error norm $\approx O(h^{4.47})$, h the grid size) in solving the Burgers' equation, and is more accurate than the compact FD method reported in Tian et al. (2011) in the study of Taylor decaying vortices. The remainder of the chapter is organised as follows. Section 3.2 briefly outlines the problem formulation, i.e. time-dependent equations considered in this chapter. The proposed CIRBF scheme, and its implementation and boundary treatment details are described in Section 3.3. In Section 3.4, numerical results are presented and compared with the exact solutions and some published approximate results, where appropriate. Section 3.5 concludes the chapter.

3.2 Problem formulations

We consider four types of time-dependent equations, namely the diffusion, Burgers, Stokes and Navier-Stokes equations. It is noted that the Navier-Stokes equation is taken in the velocity-pressure formulation, and the Stokes equation is just a simple case of the Navier-Stokes equation by neglecting the convection term and letting the Reynolds number be one.

3.2.1 Diffusion equation

$$\frac{\partial u}{\partial t} = \alpha \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) + f_b \qquad \text{in} \quad \Omega,\ t \geq 0, \tag{3.1}$$

$$u(x, y, 0) = u_0(x, y) \qquad \text{in} \quad \Omega,\ t = 0, \tag{3.2}$$

$$u(x, y, t) = u_\Gamma(x, y, t) \quad \text{on} \quad \Gamma,\ t \geq 0, \tag{3.3}$$

where u, α and f_b are the field variable, the heat conductivity (assumed constant here) and the driving function, respectively; u_0 and u_Γ prescribed functions; Γ the boundary of domain Ω.

3.2.2 Burgers' equation

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = \frac{1}{Re}\frac{\partial^2 u}{\partial x^2}, \quad a \leq x \leq b, \quad t \geq 0, \tag{3.4}$$

$$u(x, 0) = u_0(x), \quad a \leq x \leq b, \tag{3.5}$$

$$u(a, t) = u_{\Gamma_1}(t) \quad \text{and} \quad u(b, t) = u_{\Gamma_2}(t), \quad t > 0, \tag{3.6}$$

where $u_0(x)$, $u_{\Gamma_1}(t)$ and $u_{\Gamma_2}(t)$ are prescribed functions; and $[a, b]$ the segment of interest.

3.2.3 Navier-Stokes equation

$$\nabla.\mathbf{u} = 0 \qquad \text{in} \quad \Omega,\ t \geq 0, \tag{3.7}$$

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}.\nabla)\mathbf{u} = -\nabla p + \frac{1}{Re}\nabla^2 \mathbf{u} + \mathbf{f}_b \qquad \text{in} \quad \Omega,\ t \geq 0, \tag{3.8}$$

$$\mathbf{u}(x, y, 0) = \mathbf{u}_0(x, y) \qquad \text{in} \quad \Omega,\ t = 0, \tag{3.9}$$

$$\mathbf{u}(x, y, t) = \mathbf{u}_\Gamma(x, y, t) \qquad \text{on} \quad \Gamma,\ t \geq 0, \tag{3.10}$$

where $\mathbf{u}_0$ and $\mathbf{u}_\Gamma$ are prescribed functions. The other relevant field variables are defined before in chapter 2.

3.3 Numerical formulations

The proposed scheme discretises the temporal and spatial derivatives using the Adams-Bashforth/Crank-Nicolson algorithms and CIRBF approximations, respectively. For the latter, salient features include (i) the IRBF approximations are constructed over three-point stencils, (ii) values of the field variable and its second derivatives at a node are considered as two unknowns, and (iii) nodal derivative values at the boundary are approximated using the global 1D-IRBF approach.

The spatial discretisation simply and necessarily relies on a Cartesian grid on which overlapping 3-node stencils are defined. Thus, the method is not unstructured. Approximations using integrated RBFs are constructed over each stencil to represent the field variable and its derivatives. Owing to the nature of RBFs, the present method has the ability to work with "irregular" stencils, in which distances between nodes can be different, and thus can be extended to domains of non-rectangular shape in a straightforward manner, which was numerically demonstrated in some previous works (Mai-Duy and Tran-Cong, 2013) using the streamfunction-vorticity formulation of fluid flow. It was found that high order IRBF methods can capture very steep gradients without resorting to non-uniform discretisation, e.g. (Mai-Cao and Tran-Cong, 2005, 2008). This is further demonstrated with the solution of problems reported later in the present work. Thus, we only use uniform Cartesian grids in this work. Unlike FE-type meshes, the generation of a Cartesian grid, uniform or non-uniform, is a trivial task.

3.3.1 Temporal discretisation

Diffusion equation

The temporal discretisation of (3.1) with a Crank-Nicolson scheme (Crank and Nicolson, 1996) gives

$$\frac{u^n - u^{n-1}}{\Delta t} = \frac{1}{2}\alpha \left[\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right)^n + \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right)^{n-1} \right] + \frac{1}{2} \left({f_b}^n + {f_b}^{n-1} \right). \tag{3.11}$$

Burgers' equation

The temporal discretisation of (3.4) with an Adams-Bashforth scheme (Butcher, 2003) for the convection term and a Crank-Nicolson scheme for the diffusion term gives

$$\frac{u^n - u^{n-1}}{\Delta t} + \left[\frac{3}{2}\left(u\frac{\partial u}{\partial x}\right)^{n-1} - \frac{1}{2}\left(u\frac{\partial u}{\partial x}\right)^{n-2}\right] = \frac{1}{2Re}\left(\frac{\partial^2 u}{\partial x^2}^{n} + \frac{\partial^2 u}{\partial x^2}^{n-1}\right). \quad (3.12)$$

Navier-Stokes equation

The temporal discretisation of (3.8) with an Adams-Bashforth scheme for the convection term and a Crank-Nicolson scheme for the diffusion term gives

$$\frac{\mathbf{u}^n - \mathbf{u}^{n-1}}{\Delta t} + \left[\frac{3}{2}(\mathbf{u}^{n-1}.\nabla)\mathbf{u}^{n-1} - \frac{1}{2}(\mathbf{u}^{n-2}.\nabla)\mathbf{u}^{n-2}\right] = - \nabla p^{n-1/2} + \frac{1}{2Re}\left(\nabla^2\mathbf{u}^n + \nabla^2\mathbf{u}^{n-1}\right) + \frac{1}{2}\left(\mathbf{f}_b^n + \mathbf{f}_b^{n-1}\right). \quad (3.13)$$

We apply the pressure-free projection/fractional-step methods developed by Kim and Moin (1985) to solve the continuity equation (3.7) and momentum equation (3.13). The two governing equations are thus advanced in time according to the following two-step procedure, where one can project an arbitrary vector field into a divergence-free vector field via the pressure variable in (3.13)

$$\frac{\mathbf{u}^{*,n} - \mathbf{u}^{n-1}}{\Delta t} + \left[\frac{3}{2}(\mathbf{u}^{n-1}.\nabla)\mathbf{u}^{n-1} - \frac{1}{2}(\mathbf{u}^{n-2}.\nabla)\mathbf{u}^{n-2}\right] = \frac{1}{2Re}\left(\nabla^2\mathbf{u}^{*,n} + \nabla^2\mathbf{u}^{n-1}\right) + \frac{1}{2}\left(\mathbf{f}_b^n + \mathbf{f}_b^{n-1}\right), \quad (3.14)$$

$$\frac{\mathbf{u}^n - \mathbf{u}^{*,n}}{\Delta t} = -\nabla\phi^n, \quad (3.15)$$

with

$$\nabla.\mathbf{u}^n = 0, \quad (3.16)$$

where ϕ is the pseudo-pressure from which the pressure is derived through

$$p^{n-1/2} = \phi^n - \frac{\Delta t}{2Re}\nabla^2\phi^n. \quad (3.17)$$

The detailed calculation can be summarised as follows.

At first, the intermediate velocity vector is obtained by solving (3.14) in the

form of

$$\mathbf{u}^{*,n} - \frac{\Delta t}{2Re}\nabla^2\mathbf{u}^{*,n} = \mathbf{u}^{n-1} + \frac{\Delta t}{2Re}\nabla^2\mathbf{u}^{n-1} + \frac{\Delta t}{2}\left(\mathbf{f}_b^n + \mathbf{f}_b^{n-1}\right) - \Delta t\left[\frac{3}{2}(\mathbf{u}^{n-1}.\nabla)\mathbf{u}^{n-1} - \frac{1}{2}(\mathbf{u}^{n-2}.\nabla)\mathbf{u}^{n-2}\right], \quad (3.18)$$

subject to the following boundary conditions (Brown et al., 2001)

$$\mathbf{n}\ .\ \mathbf{u}^{*,n}|_\Gamma = \mathbf{n}\ .\ \mathbf{u}_b^n, \quad (3.19)$$

$$\mathbf{t}\ .\ \mathbf{u}^{*,n}|_\Gamma = \mathbf{t}\ .\ \left(\mathbf{u}_b^n + \Delta t\nabla\phi^{n-1}\right)|_\Gamma, \quad (3.20)$$

where $\mathbf{n} = (n_x, n_y)^T$ and $\mathbf{t} = (t_x, t_y)^T$ are the normal and tangential unit vectors. It can be seen that $\mathbf{u}^{*,n}$ does not satisfy the continuity equation (3.16).

Then, by applying the gradient operator to both sides of (3.15) and forcing $\mathbf{u}^n$ to satisfy (3.16), the Poisson equation for the pseudo-pressure ϕ is obtained

$$\nabla^2\phi^n = \frac{1}{\Delta t}\nabla.\mathbf{u}^{*,n}, \quad (3.21)$$

which can be solved with the Neumann boundary condition derived from (3.15)

$$\nabla\phi^n|_\Gamma = \frac{\mathbf{u}_b^{*,n} - \mathbf{u}_b^n}{\Delta t}. \quad (3.22)$$

However, as shown in Thai-Quang, Le-Cao, Mai-Duy and Tran-Cong (2012), Thai-Quang, Le-Cao, Mai-Duy and Tran-Cong (2012), a transformation of the boundary condition from the Neumann form into the Dirichlet form can lead to an improvement in the solution accuracy. We will use the Dirichlet form in solving (3.21).

After obtaining the pseudo-pressure ϕ^n, one can compute the velocity at the next time level through (3.15)

$$\mathbf{u}^n = \mathbf{u}^{*,n} - \Delta t\nabla\phi^n, \quad (3.23)$$

and then the pressure through (3.17).

3.3.2 Spatial discretisation

The approximation scheme employed in this chapter for the spatial discretisation is almost the same as the one used in chapter 2, i.e. equations (2.26)-(2.28). Here we use the implicit form of the extra information of nodal second-order derivative values rather than the explicit form as in chapter 2. Since this chapter is mainly concerned with the evolution of the solution, it can be seen that the

values of u and $\partial^2 u/\partial\eta^2$ used in (2.26)-(2.28) should be chosen at the same time level in order to achieve the best performance of the CIRBF stencils. In this regard, we propose to take values of u^n and $\frac{\partial^2 u^n}{\partial\eta^2}$ at each node as two independent unknowns.

Let a 2D problem domain Ω be represented by a Cartesian grid where the nodes are indexed by i in the x direction ($i \in \{1,\ 2, \ldots,\ n_x\}$) and j in the y direction ($j \in \{1,\ 2, \ldots,\ n_y\}$). The total number of grid points is therefore $N = n_x \times n_y$ for a rectangular domain. For non-rectangular domains, the grid points outside the domain are discarded and $N < n_x \times n_y$. For each interior grid point $\mathbf{x}_{i,j} = (x_{i,j}, y_{i,j})^T$ ($i \in \{2,\ 3, \ldots,\ n_x - 1\}$ and $j \in \{2,\ 3, \ldots,\ n_y - 1\}$), we form two 3-point stencils, namely $[x_{i-1,j}, x_{i,j}, x_{i+1,j}]$ and $[y_{i,j-1}, y_{i,j}, y_{i,j+1}]$. For the sake of convenience, we use η to denote x and y, thus having a generic stencil $[\eta_1, \eta_2, \eta_3]$ ($\eta_1 < \eta_2 < \eta_3,\ \eta_2 \equiv \eta_{i,j}$) as shown in Figure 2.2.

Consider a node $\mathbf{x}_{i,j} = (x_{i,j}, y_{i,j})^T$. The two algebraic equations associated with this node are established as follows.

The first equation is derived from (2.28) for a 3-point stencil associated with $\mathbf{x}_{i,j}$. Collocating (2.28) at the central node of a stencil yields

$$\frac{d^2 u_2^n}{d\eta^2} = \sum_{i=1}^{3} \frac{d^2\varphi_i(\eta_2)}{d\eta^2} u_i^n + \frac{d^2\varphi_4(\eta_2)}{d\eta^2}\frac{d^2 u_1^n}{d\eta^2} + \frac{d^2\varphi_5(\eta_2)}{d\eta^2}\frac{d^2 u_3^n}{d\eta^2}, \tag{3.24}$$

where $\eta_2 = x_{i,j}$ for $[x_{i-1,j}, x_{i,j}, x_{i+1,j}]$ and $\eta_2 = y_{i,j}$ for $[y_{i,j-1}, y_{i,j}, y_{i,j+1}]$. It is noted that, in chapter 2, values of the second-order derivative at two extreme nodes of a compact stencil are treated like known values using nodal function values at time step $(n-1)$ and nodal second-order derivative values at time step $(n-2)$.

The second equation is obtained from the governing equation. Collocating the diffusion equation (3.11), Burgers' equation (3.12), and Navier-Stokes equations (3.14) and (3.21), at $\mathbf{x} = \mathbf{x}_{i,j}$ yields

$$u_{i,j}^n - \frac{\alpha\Delta t}{2}\left(\frac{\partial^2 u_{i,j}^n}{\partial x^2} + \frac{\partial^2 u_{i,j}^n}{\partial y^2}\right) = u_{i,j}^{n-1} + \frac{\alpha\Delta t}{2}\left(\frac{\partial^2 u_{i,j}^{n-1}}{\partial x^2} + \frac{\partial^2 u_{i,j}^{n-1}}{\partial y^2}\right) + \frac{\Delta t}{2}\left(f_{b_{i,j}}^n + f_{b_{i,j}}^{n-1}\right), \tag{3.25}$$

$$u_{i,j}^n - \frac{\Delta t}{2Re}\frac{\partial^2 u_{i,j}^n}{\partial x^2} = u_{i,j}^{n-1} + \frac{\Delta t}{2Re}\frac{\partial^2 u_{i,j}^{n-1}}{\partial x^2} - \Delta t\left[\frac{3}{2}\left(u_{i,j}^{n-1}\frac{\partial u_{i,j}^{n-1}}{\partial x}\right) - \frac{1}{2}\left(u_{i,j}^{n-2}\frac{\partial u_{i,j}^{n-2}}{\partial x}\right)\right], \tag{3.26}$$

$$\mathbf{u}_{i,j}^{*,n} - \frac{\Delta t}{2Re}\nabla^2\mathbf{u}_{i,j}^{*,n} = \mathbf{u}_{i,j}^{n-1} + \frac{\Delta t}{2Re}\nabla^2\mathbf{u}_{i,j}^{n-1} + \frac{\Delta t}{2}\left(\mathbf{f}_{b_{i,j}}^{n} + \mathbf{f}_{b_{i,j}}^{n-1}\right) - \Delta t\left[\frac{3}{2}\left(\mathbf{u}_{i,j}^{n-1}.\nabla\right)\mathbf{u}_{i,j}^{n-1} - \frac{1}{2}\left(\mathbf{u}_{i,j}^{n-2}.\nabla\right)\mathbf{u}_{i,j}^{n-2}\right], \quad (3.27)$$

$$\nabla^2\phi_{i,j}^{n} = \frac{1}{\Delta t}\nabla.\mathbf{u}_{i,j}^{*,n}, \quad (3.28)$$

respectively.

The present final algebraic systems corresponding to the above equations are thus solved for the values of the field variable and its second-order derivative at the interior grid nodes. In the following, we will describe the solution procedure in detail for the case of Dirichlet boundary conditions.

Algebraic systems from the stencil equations

Rewriting (3.24) with reference to the global coordinate system leads to

$$-\frac{d^2\varphi_4(x_{i,j})}{dx^2}\frac{\partial^2 u_{i-1,j}^{n}}{\partial x^2} + \frac{\partial^2 u_{i,j}^{n}}{\partial x^2} - \frac{d^2\varphi_5(x_{i,j})}{dx^2}\frac{\partial^2 u_{i+1,j}^{n}}{\partial x^2} = \frac{d^2\varphi_1(x_{i,j})}{dx^2}u_{i-1,j}^{n} + \frac{d^2\varphi_2(x_{i,j})}{dx^2}u_{i,j}^{n} + \frac{d^2\varphi_3(x_{i,j})}{dx^2}u_{i+1,j}^{n}, \quad (3.29)$$

for the stencil $[x_{i-1,j}, x_{i,j}, x_{i+1,j}]$, and

$$-\frac{d^2\varphi_4(y_{i,j})}{dy^2}\frac{\partial^2 u_{i,j-1}^{n}}{\partial y^2} + \frac{\partial^2 u_{i,j}^{n}}{\partial y^2} - \frac{d^2\varphi_5(y_{i,j})}{dy^2}\frac{\partial^2 u_{i,j+1}^{n}}{\partial y^2} = \frac{d^2\varphi_1(y_{i,j})}{dy^2}u_{i,j-1}^{n} + \frac{d^2\varphi_2(y_{i,j})}{dy^2}u_{i,j}^{n} + \frac{d^2\varphi_3(y_{i,j})}{dy^2}u_{i,j+1}^{n}, \quad (3.30)$$

for the stencil $[y_{i,j-1}, y_{i,j}, y_{i,j+1}]$.

Equations (3.29) and (3.30) at the interior nodes adjacent to the boundary involve second derivatives of u at the boundaries. We utilise the global 1D-IRBF approach (Mai-Duy and Tran-Cong, 2008; Thai-Quang, Le-Cao, Mai-Duy and

Tran-Cong, 2012) to estimate these values, e.g. on an x-grid line,

$$\frac{\partial^2 u^n_{1,j}}{\partial x^2} = \underbrace{\begin{bmatrix} G_1(x_{1,j}) \\ \vdots \\ G_{n_x}(x_{1,j}) \\ 0 \\ 0 \end{bmatrix}^T \begin{bmatrix} \overline{H}_1(x_{1,j}) & \cdots & \overline{H}_{n_x}(x_{1,j}) & x_{1,j} & 1 \\ \overline{H}_1(x_{2,j}) & \cdots & \overline{H}_{n_x}(x_{2,j}) & x_{2,j} & 1 \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ \overline{H}_1(x_{n_x,j}) & \cdots & \overline{H}_{n_x}(x_{n_x,j}) & x_{n_x,j} & 1 \end{bmatrix}^{-1}}_{\mathcal{D}_{xx(1)}} \begin{pmatrix} u^n_{1,j} \\ u^n_{2,j} \\ \vdots \\ u^n_{n_x,j} \end{pmatrix}, \tag{3.31}$$

$$\frac{\partial^2 u^n_{n_x,j}}{\partial x^2} = \underbrace{\begin{bmatrix} G_1(x_{n_x,j}) \\ \vdots \\ G_{n_x}(x_{n_x,j}) \\ 0 \\ 0 \end{bmatrix}^T \begin{bmatrix} \overline{H}_1(x_{1,j}) & \cdots & \overline{H}_{n_x}(x_{1,j}) & x_{1,j} & 1 \\ \overline{H}_1(x_{2,j}) & \cdots & \overline{H}_{n_x}(x_{2,j}) & x_{2,j} & 1 \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ \overline{H}_1(x_{n_x,j}) & \cdots & \overline{H}_{n_x}(x_{n_x,j}) & x_{n_x,j} & 1 \end{bmatrix}^{-1}}_{\mathcal{D}_{xx(n_x)}} \begin{pmatrix} u^n_{1,j} \\ u^n_{2,j} \\ \vdots \\ u^n_{n_x,j} \end{pmatrix}, \tag{3.32}$$

where $j \in \{1,\ 2, \ldots,\ n_y\}$, or in matrix-vector form

$$\frac{\partial^2 u^n_{1,j}}{\partial x^2} = \mathcal{D}_{xx(1)} \left[u^n_{1,j}\ u^n_{2,j}\ \ldots\ u^n_{n_x,j}\right]^T, \tag{3.33}$$

$$\frac{\partial^2 u^n_{n_x,j}}{\partial x^2} = \mathcal{D}_{xx(n_x)} \left[u^n_{1,j}\ u^n_{2,j}\ \ldots\ u^n_{n_x,j}\right]^T. \tag{3.34}$$

On a y-grid line, in the same manner, one has

$$\frac{\partial^2 u^n_{i,1}}{\partial y^2} = \mathcal{D}_{yy(1)} \left[u^n_{i,1}\ u^n_{i,2}\ \ldots\ u^n_{i,n_y}\right]^T, \tag{3.35}$$

$$\frac{\partial^2 u^n_{i,n_y}}{\partial y^2} = \mathcal{D}_{yy(n_y)} \left[u^n_{i,1}\ u^n_{i,2}\ \ldots\ u^n_{i,n_y}\right]^T, \tag{3.36}$$

where

$$\mathcal{D}_{yy(1)} = \begin{bmatrix} G_1(y_{i,1}) \\ \vdots \\ G_{n_y}(y_{i,1}) \\ 0 \\ 0 \end{bmatrix}^T \begin{bmatrix} \overline{H}_1(y_{i,1}) & \cdots & \overline{H}_{n_y}(y_{i,1}) & y_{i,1} & 1 \\ \overline{H}_1(y_{i,2}) & \cdots & \overline{H}_{n_y}(y_{i,2}) & y_{i,2} & 1 \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ \overline{H}_1(y_{i,n_y}) & \cdots & \overline{H}_{n_y}(y_{i,n_y}) & y_{i,n_y} & 1 \end{bmatrix}^{-1}, \tag{3.37}$$

$$\mathcal{D}_{yy(n_y)} = \begin{bmatrix} G_1(y_{i,n_y}) \\ \vdots \\ G_{n_y}(y_{i,n_y}) \\ 0 \\ 0 \end{bmatrix}^T \begin{bmatrix} \overline{H}_1(y_{i,1}) & \cdots & \overline{H}_{n_y}(y_{i,1}) & y_{i,1} & 1 \\ \overline{H}_1(y_{i,2}) & \cdots & \overline{H}_{n_y}(y_{i,2}) & y_{i,2} & 1 \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ \overline{H}_1(y_{i,n_y}) & \cdots & \overline{H}_{n_y}(y_{i,n_y}) & y_{i,n_y} & 1 \end{bmatrix}^{-1}. \tag{3.38}$$

It can be seen from (3.33)-(3.34) and (3.35)-(3.36) that derivative values of u at the boundaries are expressed in terms of nodal values of u on the x- and y-grid lines, respectively.

The systems of equations corresponding to the stencil equation are thus obtained by letting $i \in \{2,\ 3, \ldots,\ n_x - 1\}$ and $j \in \{2,\ 3, \ldots,\ n_y - 1\}$ in (3.29)-(3.30), $j \in \{1,\ 2, \ldots,\ n_y\}$ in (3.33)-(3.34), and $i \in \{1,\ 2, \ldots,\ n_x\}$ in (3.35)-(3.36)

$$\mathcal{L}_{xx} \widehat{\frac{\partial^2 u}{\partial x^2}}^n = \mathcal{B}_{xx} \widehat{u}^n, \tag{3.39}$$

$$\mathcal{L}_{yy} \widehat{\frac{\partial^2 u}{\partial y^2}}^n = \mathcal{B}_{yy} \widehat{u}^n, \tag{3.40}$$

where $\mathcal{L}_{xx}$, $\mathcal{B}_{xx}$, $\mathcal{L}_{yy}$ and $\mathcal{B}_{yy}$ are $N \times N$ matrices,

$$\widehat{u}^n = \left(u^n_{1,1}, u^n_{1,2}, \ldots, u^n_{1,n_y}, u^n_{2,1}, u^n_{2,2}, \ldots, u^n_{2,n_y}, \ldots, u^n_{n_x,1}, u^n_{n_x,2}, \ldots, u^n_{n_x,n_y} \right)^T, \tag{3.41}$$

$$\widehat{\frac{\partial^2 u}{\partial x^2}}^n = \left(\frac{\partial^2 u^n_{1,1}}{\partial x^2}, \frac{\partial^2 u^n_{1,2}}{\partial x^2}, \cdots, \frac{\partial^2 u^n_{1,n_y}}{\partial x^2}, \frac{\partial^2 u^n_{2,1}}{\partial x^2}, \frac{\partial^2 u^n_{2,2}}{\partial x^2}, \cdots, \frac{\partial^2 u^n_{2,n_y}}{\partial x^2}, \right.$$
$$\left. \cdots, \frac{\partial^2 u^n_{n_x,1}}{\partial x^2}, \frac{\partial^2 u^n_{n_x,2}}{\partial x^2}, \cdots, \frac{\partial^2 u^n_{n_x,n_y}}{\partial x^2} \right)^T, \tag{3.42}$$

and

$$\widehat{\frac{\partial^2 u}{\partial y^2}}^n = \left(\frac{\partial^2 u^n_{1,1}}{\partial y^2}, \frac{\partial^2 u^n_{1,2}}{\partial y^2}, \cdots, \frac{\partial^2 u^n_{1,n_y}}{\partial y^2}, \frac{\partial^2 u^n_{2,1}}{\partial y^2}, \frac{\partial^2 u^n_{2,2}}{\partial y^2}, \cdots, \frac{\partial^2 u^n_{2,n_y}}{\partial y^2}, \right.$$
$$\left. \cdots, \frac{\partial^2 u^n_{n_x,1}}{\partial y^2}, \frac{\partial^2 u^n_{n_x,2}}{\partial y^2}, \cdots, \frac{\partial^2 u^n_{n_x,n_y}}{\partial y^2} \right)^T. \quad (3.43)$$

Algebraic systems from the governing equations

Collocating the governing equation, e.g. (3.25) for diffusion problems, at a grid node $\mathbf{x}_{i,j}$ results in

$$u^n_{i,j} - \frac{\alpha \Delta t}{2} \frac{\partial^2 u^n_{i,j}}{\partial x^2} - \frac{\alpha \Delta t}{2} \frac{\partial^2 u^n_{i,j}}{\partial y^2} = rhs^n_{i,j}, \quad (3.44)$$

where

$$rhs^n_{i,j} = u^{n-1}_{i,j} + \frac{\alpha \Delta t}{2} \left(\frac{\partial^2 u_{i,j}}{\partial x^2} + \frac{\partial^2 u_{i,j}}{\partial y^2} \right)^{n-1} + \frac{\Delta t}{2} \left(f^n_{b_{i,j}} + f^{n-1}_{b_{i,j}} \right). \quad (3.45)$$

It is straightforward to extend the present procedure to diffusion problems, where the heat conductivity is a function of space and time. It is noted that this collocation process can also be done in a similar way for fluid problems governed by (3.12), (3.14) and (3.21).

The system of equations corresponding to the governing equation is thus obtained by letting i and j in (3.44) be $\{2,\ 3, \ldots,\ n_x - 1\}$ and $\{2,\ 3, \ldots,\ n_y - 1\}$, respectively

$$\mathrm{I}\widehat{u}^n - \left(\frac{\alpha \Delta t}{2} \mathrm{I} \right) \widehat{\frac{\partial^2 u}{\partial x^2}}^n - \left(\frac{\alpha \Delta t}{2} \mathrm{I} \right) \widehat{\frac{\partial^2 u}{\partial y^2}}^n = \widehat{rhs}^n, \quad (3.46)$$

where I is the $N \times N$ identity matrix,

$$\widehat{rhs}^n = \left(rhs^n_{1,1}, rhs^n_{1,2}, \ldots, rhs^n_{1,n_y}, rhs^n_{2,1}, rhs^n_{2,2}, \ldots, rhs^n_{2,n_y}, \right.$$
$$\left. \ldots, rhs^n_{n_x,1}, rhs^n_{n_x,2}, \ldots, rhs^n_{n_x,n_y} \right)^T. \quad (3.47)$$

Final system of algebraic equations

The final set of equations is formed by combining (3.39), (3.40) and (3.46)

$$\left[\begin{array}{c|c|c} \mathcal{B}_{xx} & -\mathcal{L}_{xx} & \mathcal{O} \\ \hline \mathcal{B}_{yy} & \mathcal{O} & -\mathcal{L}_{yy} \\ \hline \mathrm{I}_{ip} & -\frac{\alpha\Delta t}{2}\mathrm{I}_{ip} & -\frac{\alpha\Delta t}{2}\mathrm{I}_{ip} \end{array}\right] \left(\begin{array}{c} \widehat{u}^n \\ \widehat{\frac{\partial^2 u}{\partial x^2}}^n \\ \widehat{\frac{\partial^2 u}{\partial y^2}}^n \end{array}\right) = \left(\begin{array}{c} \mathrm{O} \\ \mathrm{O} \\ \widehat{rhs}_{ip}^{\,n} \end{array}\right), \tag{3.48}$$

where $\mathcal{O}$ is a zero matrix of dimensions $N \times N$, O is a zero vector of length N, I_{ip} consists of rows of I that correspond to the interior nodes, and $\widehat{rhs}_{ip}^{\,n}$ are entries of $\widehat{rhs}^{\,n}$ that are associated with the interior nodes. After imposing the boundary conditions, the system matrix on the left side of (3.48) is of dimensions $(3N - (2n_x + 2n_y - 4)) \times (3N - (2n_x + 2n_y - 4))$.

The resultant system (3.48) is sparse and can be efficiently solved by, for example, the *LU* decomposition technique to obtain unknown nodal values of u, $\frac{\partial^2 u}{\partial x^2}$ and $\frac{\partial^2 u}{\partial y^2}$ at the current time level. The number of non-zero entries is given by

$$\begin{aligned} N_{entries} &= 2 \times [2 + (n_x - 1) \times (n_y - 2)] \\ &+ 2 \times 3\,(n_x - 2) + 6 \times (n_x - 2) \times (n_y - 2) - 2 \times (n_y - 2) \\ &+ 2 \times [2 + (n_y - 1) \times (n_x - 2)] \\ &+ 2 \times 3\,(n_y - 2) + 6 \times (n_x - 2) \times (n_y - 2) - 2 \times (n_x - 2) \\ &\qquad + 3 \times (n_x - 2) \times (n_y - 2)\,, \end{aligned}$$

or

$$N_{entries} = 19N - 64\sqrt{N} + 60,$$

for the case of $n_x = n_y = \sqrt{N}$. Furthermore, the extra storage required by the Adams-Bashforth scheme from equation (3.18) is $2N$. Thus, the storage requirement is of $O(N)$.

In the case of Neumann boundary conditions, one can appropriately add the following first derivative equations, which can be generated using the CIRBF (i.e. (2.24)/(2.27)) or the global 1D-IRBF approach (i.e. equations provided below) (Mai-Duy and Tran-Cong, 2008; Thai-Quang, Mai-Duy, Tran and Tran-Cong, 2012), to the final system of equations to determine unknown values of

u on the boundaries

$$\frac{\partial u^n_{1,j}}{\partial x} = \underbrace{\begin{bmatrix} H_1(x_{1,j}) \\ \vdots \\ H_{n_x}(x_{1,j}) \\ 1 \\ 0 \end{bmatrix}^T \begin{bmatrix} \overline{H}_1(x_{1,j}) & \cdots & \overline{H}_{n_x}(x_{1,j}) & x_{1,j} & 1 \\ \overline{H}_1(x_{2,j}) & \cdots & \overline{H}_{n_x}(x_{2,j}) & x_{2,j} & 1 \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ \overline{H}_1(x_{n_x,j}) & \cdots & \overline{H}_{n_x}(x_{n_x,j}) & x_{n_x,j} & 1 \end{bmatrix}^{-1}}_{\mathcal{D}_{x(1)}} \begin{pmatrix} u^n_{1,j} \\ u^n_{2,j} \\ \vdots \\ u^n_{n_x,j} \end{pmatrix}, \tag{3.49}$$

$$\frac{\partial u^n_{n_x,j}}{\partial x} = \underbrace{\begin{bmatrix} H_1(x_{n_x,j}) \\ \vdots \\ H_{n_x}(x_{n_x,j}) \\ 1 \\ 0 \end{bmatrix}^T \begin{bmatrix} \overline{H}_1(x_{1,j}) & \cdots & \overline{H}_{n_x}(x_{1,j}) & x_{1,j} & 1 \\ \overline{H}_1(x_{2,j}) & \cdots & \overline{H}_{n_x}(x_{2,j}) & x_{2,j} & 1 \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ \overline{H}_1(x_{n_x,j}) & \cdots & \overline{H}_{n_x}(x_{n_x,j}) & x_{n_x,j} & 1 \end{bmatrix}^{-1}}_{\mathcal{D}_{x(n_x)}} \begin{pmatrix} u^n_{1,j} \\ u^n_{2,j} \\ \vdots \\ u^n_{n_x,j} \end{pmatrix}, \tag{3.50}$$

for an x-grid line, and

$$\frac{\partial u^n_{i,1}}{\partial y} = \underbrace{\begin{bmatrix} H_1(y_{i,1}) \\ \vdots \\ H_{n_y}(y_{i,1}) \\ 1 \\ 0 \end{bmatrix}^T \begin{bmatrix} \overline{H}_1(y_{i,1}) & \cdots & \overline{H}_{n_y}(y_{i,1}) & y_{i,1} & 1 \\ \overline{H}_1(y_{i,2}) & \cdots & \overline{H}_{n_y}(y_{i,2}) & y_{i,2} & 1 \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ \overline{H}_1(y_{i,n_y}) & \cdots & \overline{H}_{n_y}(y_{i,n_y}) & y_{i,n_y} & 1 \end{bmatrix}^{-1}}_{\mathcal{D}_{y(1)}} \begin{pmatrix} u^n_{i,1} \\ u^n_{i,2} \\ \vdots \\ u^n_{i,n_y} \end{pmatrix}, \tag{3.51}$$

$$\frac{\partial u^n_{i,n_y}}{\partial y} = \underbrace{\begin{bmatrix} H_1(y_{i,n_y}) \\ \vdots \\ H_{n_y}(y_{i,n_y}) \\ 1 \\ 0 \end{bmatrix}^T \begin{bmatrix} \overline{H}_1(y_{i,1}) & \cdots & \overline{H}_{n_y}(y_{i,1}) & y_{i,1} & 1 \\ \overline{H}_1(y_{i,2}) & \cdots & \overline{H}_{n_y}(y_{i,2}) & y_{i,2} & 1 \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ \overline{H}_1(y_{i,n_y}) & \cdots & \overline{H}_{n_y}(y_{i,n_y}) & y_{i,n_y} & 1 \end{bmatrix}^{-1}}_{\mathcal{D}_{y(n_y)}} \begin{pmatrix} u^n_{i,1} \\ u^n_{i,2} \\ \vdots \\ u^n_{i,n_y} \end{pmatrix}, \tag{3.52}$$

for a y-grid line.

3.4 Numerical examples

Since the RBF approximations employed are local, the present scheme can work with a wide range of β ($2 \leq \beta \leq 100$) as shown in chapter 2. Thus, we simply pick $\beta = 20$ for CIRBF approximations in this chapter. The quality of the RBF approximations depends on the distance between RBF centres which are also grid nodes. Better approximation is generally obtained as more RBFs are employed. A grid refinement (i.e. increasing the number of RBFs) thus provides a way to enhance the accuracy of the RBF solution. We assess the performance of the proposed scheme through following measures:

- the RMS error defined as (2.48),
- the maximum absolute error (L_∞) defined as

$$L_\infty = \max_i |u_i - \overline{u}_i|, \tag{3.53}$$

- the average rate of convergence α determined by the method of least squares as described in chapter 2.

3.4.1 Diffusion equations

Diffusion problems are considered here to investigate the performance of the proposed scheme for the diffusion term only.

One-dimensional case

Consider a one-dimensional diffusion equation

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2}, \quad 0 \leq x \leq \pi, \quad t \geq 0, \tag{3.54}$$

with the following initial and boundary conditions

$$u(x,0) = \sin(2x),\ 0 < x < \pi \quad \text{and} \quad u(0,t) = u(\pi,t) = 0,\ t \geq 0. \tag{3.55}$$

The exact solution of this problem is known to be $\overline{u}(x,t) = \sin(2x)e^{-4t}$.

Firstly, the spatial accuracy of the proposed scheme is investigated on various grids $N \in \{11,\ 21, \ldots,\ 101\}$. We employ here a very small time step, i.e. $\Delta t = 0.001$, in order to minimise the effect of the approximate error in time. Figure 3.1 shows the effect of the grid size h on the accuracy of the solution computed at $t = 1$, from which one can see that the solution error behaves as $O(h^{3.4})$ (i.e. more than third-order accuracy in space).

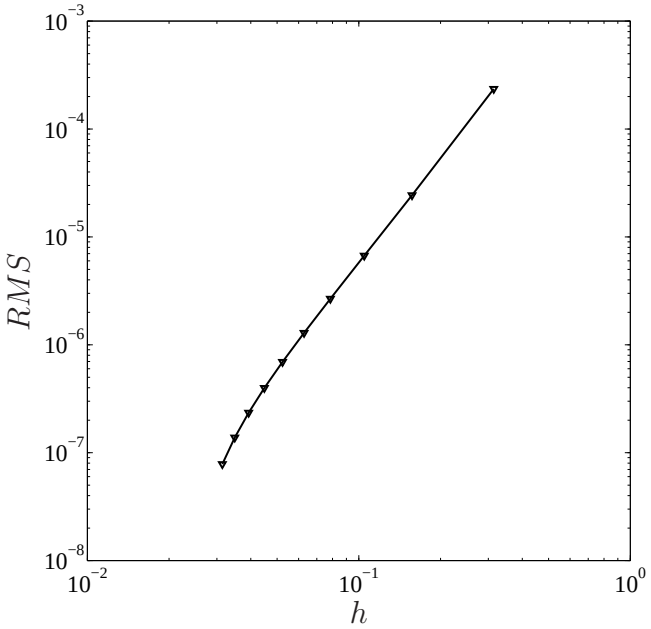


Figure 3.1: 1D diffusion equation, $N \in \{11,\ 21, \ldots,\ 101\}$, $\Delta t = 0.001$, $t = 1$: The effect of grid size h on the solution accuracy by the proposed scheme. The solution error behaves apparently as $Ne \approx O(h^{3.4})$.

Secondly, we investigate the temporal accuracy of our scheme through a set of time steps, namely $\Delta t \in \{\frac{1}{100},\ \frac{1}{90}, \ldots,\ \frac{1}{10}\}$. A fine grid of $h = 0.0157$ (i.e. 201 grid points) is taken so that the approximate error in space is much smaller than

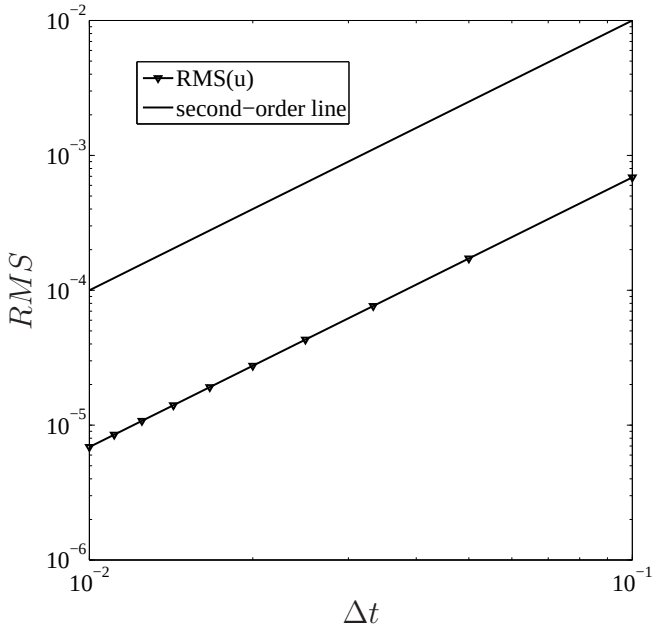


Figure 3.2: 1D diffusion equation, $N = 201$, $\Delta t \in \{\frac{1}{100}, \frac{1}{90}, \ldots, \frac{1}{10}\}$, $t = 1$: The effect of time step Δt on the solution accuracy by the proposed scheme. The solution error behaves apparently as $Ne \approx O(\Delta t^2)$.

the time splitting error. The error of the solution at $t = 1$ is shown in Figure 3.2 as a function of the time step. It can be seen that our scheme obtains second-order accuracy in time. This result is fully expected as second-order approximation algorithms for the discretisation of temporal derivative terms are adopted in the present scheme.

Two-dimensional case

We next consider equation (3.1) in $\Omega = [0, 1]\times[0, 1]$ with Dirichlet boundary conditions, and the following parameters $\alpha = 1$ and $f_b = \left(\frac{1}{\pi^2} + 2k^2t\right)\sin(k\pi x)\sin(k\pi y)$, where k is a given integer. The initial and boundary conditions are derived from the exact solution $\overline{u}(x, y, t) = \frac{1}{\pi^2}\sin(k\pi x)\sin(k\pi y)t + xy$. The higher the value of k is, the more complex the variation of the solution will be. We choose $k = 5$, $\Delta t = 0.01$ and employ a set of grids, namely $\{11 \times 11,\ 21 \times 21, \ldots,\ 51 \times 51\}$ to investigate the accuracy of the spatial discretisation of the proposed scheme. Figure 3.3 shows the effect of the grid size h on the accuracy of the solution computed at $t = 1$. One can see that the solution error behaves as $O(h^{3.31})$ (i.e. also more than third-order accuracy in space).

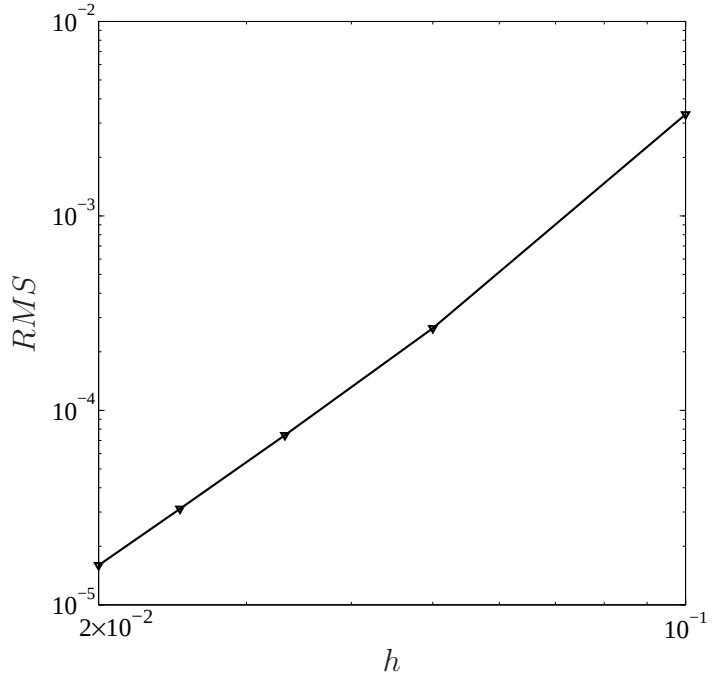


Figure 3.3: 2D diffusion equation, $\{11 \times 11,\ 21 \times 21, \ldots,\ 51 \times 51\}$, $\Delta t = 0.01$, $t = 1$: The effect of grid size h on the solution accuracy by the proposed scheme. The solution error behaves apparently as $Ne \approx O(h^{3.31})$.

3.4.2 Stokes flow

The Stokes equations are a simple case of the transient Navier-Stokes equations (3.7)-(3.8), where the non-linear convection term $(\mathbf{u}.\nabla)\mathbf{u}$ is neglected and the Reynolds number is set one. Through Stokes flows, we can test the performance of the proposed scheme for the effect of coupling the velocity with the pressure field without the involvement of the non-linear convection term $(\mathbf{u}.\nabla)\mathbf{u}$. Following Sheng et al. (2011), we consider the following exact solution

$$\overline{u}(x, y, t) = \pi \sin(2\pi y) \sin^2(\pi x) \sin t, \tag{3.56}$$

$$\overline{v}(x, y, t) = -\pi \sin(2\pi x) \sin^2(\pi y) \sin t, \tag{3.57}$$

$$\overline{p}(x, y, t) = \cos(\pi x) \sin(\pi y) \sin t, \tag{3.58}$$

on a square domain $\Omega = [0, 1] \times [0, 1]$, which corresponds to the source term $\mathbf{f}_b = \frac{\partial \overline{\mathbf{u}}}{\partial t} - \nabla^2 \overline{\mathbf{u}} + \nabla \overline{p}$. One can derive the initial solution and Dirichlet boundary conditions for the velocity from (3.56)-(3.57). The reference pressure $p = 0$ is specified at the centre of the domain.

The calculations are carried out with a very small time step, i.e. $\Delta t = 10^{-5}$, to minimise the effect of the approximate error in time. We employ a number

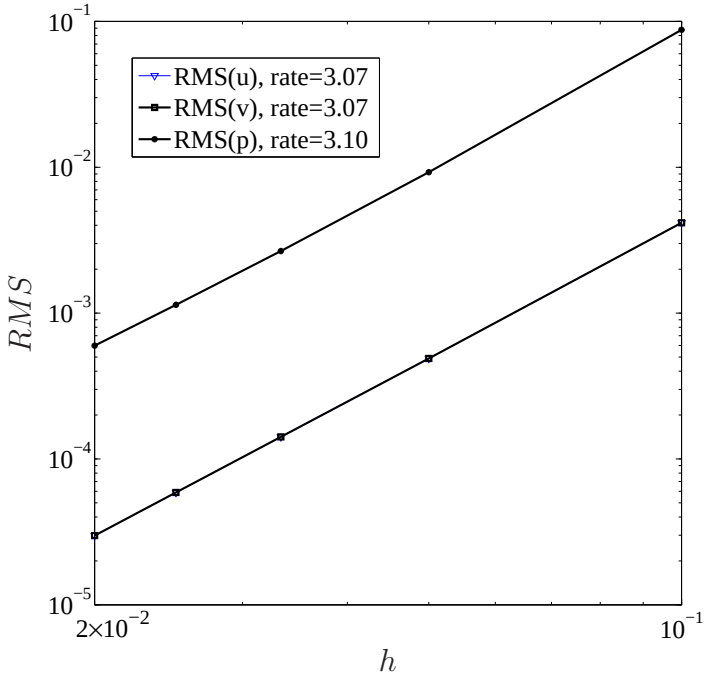


Figure 3.4: Stokes flow, $\{11 \times 11,\ 21 \times 21, \ldots,\ 51 \times 51\}$, $\Delta t = 10^{-5}$, $t = 1$: The effect of grid size h on the solution accuracy by the proposed scheme. The solution error behaves as $Ne \approx O(h^{3.07})$ and $Ne \approx O(h^{3.1})$ for the velocity (the two indistinguishable lower lines) and the pressure, respectively.

of grids, namely $\{11 \times 11,\ 21 \times 21, \ldots,\ 51 \times 51\}$ to study the grid convergence of the proposed scheme. Figure 3.4 shows the behaviour of the solution error computed at $t = 1$ against the grid size h. One can see that the solution error behaves apparently as $O(h^{3.07})$ for the velocity and $O(h^{3.1})$ for the pressure.

3.4.3 Burgers' equation

The Burgers' equation (3.4) is a good example to test a numerical scheme as it involves both the convection and diffusion terms.

Approximation of shock wave propagation

Consider the following analytic solution of the Burgers' equation (3.4) reported in Hassanien et al. (2005)

$$\overline{u}(x,t) = \frac{[\alpha_0 + \mu_0 + (\mu_0 - \alpha_0)\exp(\eta)]}{1 + \exp(\eta)}, \tag{3.59}$$

where $0 \leq x \leq 1$, $t \geq 0$, $\eta = \alpha_0 Re(x - \mu_0 t - \beta_0)$, $\alpha_0 = 0.4$, $\beta_0 = 0.125$ and $\mu_0 = 0.6$.

The initial and boundary conditions can be derived from (3.59). To facilitate comparisons with some published results, we also employ $Re = 100$, $t = 0.5$ and $N = 37$. Table 3.1 presents the CIRBF solution, the exact solution and those in Hassanien et al. (2005), Dag, Irk and Saka (2005), Dogan (2004), Ali et al. (2005), showing a close agreement achieved.

In order to study the convergence of the solution with grid refinement, the calculations are carried out on a set of grids, namely $N \in \{11,\ 21, \ldots,\ 101\}$. The time step Δt is required to be small enough at which the error caused by the temporal discretisation can be negligible. In the present study, the time step $\Delta t = 10^{-5}$ is chosen since experimentally there was no improvement with smaller time steps. The errors of the solution against different grid sizes at $t = 0.5$ are displayed in Figure 3.5, where the solution error behaves as $O(h^{4.47})$.

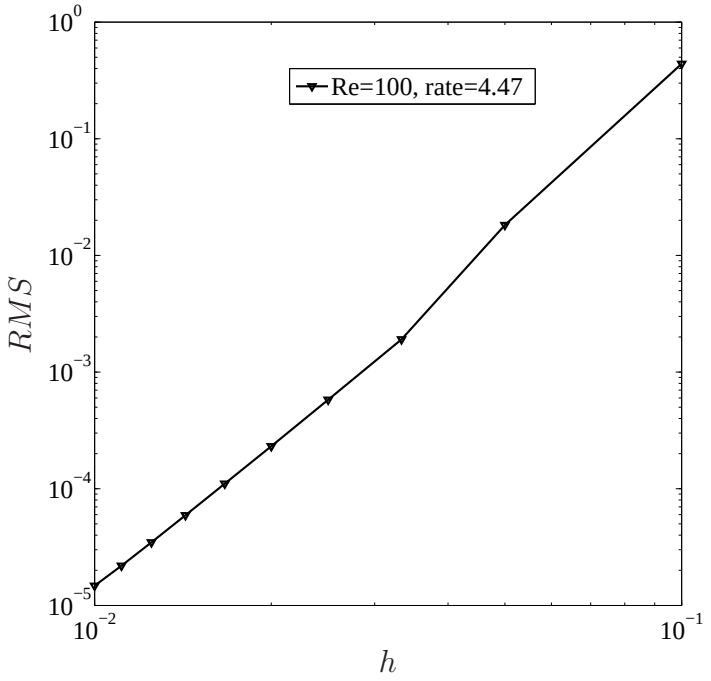


Figure 3.5: Shock wave propagation, $N \in \{11,\ 21, \ldots,\ 101\}$, $Re = 100$, $\Delta t = 10^{-5}$, $t = 0.5$: The effect of grid size h on the solution accuracy by the proposed scheme. The solution error behaves apparently as $Ne \approx O(h^{4.47})$.

Table 3.1: Shock wave propagation, grid $N = 37$, $Re = 100$, $t = 0.5$: the exact, present and some other numerical solutions.

x	Exact solution	Present $\Delta t = 0.05$	Ali et al. (2005) $\Delta t = 0.025$	Dogan (2004) $\Delta t = 0.05$	Dag, Irk and Saka (2005) $\Delta t = 0.025$	Hassanien et al. (2005) $\Delta t = 0.01$
0.000	1.0	1.0	1.0	1.0	1.0	1.0
0.056	1.0	1.0	1.0	1.0	1.0	1.0
0.111	1.0	1.0	1.0	1.0	1.0	1.0
0.167	1.0	1.0	1.0	1.0	1.0	1.0
0.222	1.0	1.0	1.0	1.0	1.0	1.0
0.278	0.998	0.998	0.999	0.999	0.999	0.998
0.333	0.980	0.980	0.985	0.994	0.986	0.982
0.389	0.847	0.850	0.847	0.848	0.850	0.849
0.444	0.452	0.449	0.452	0.407	0.448	0.455
0.500	0.238	0.238	0.238	0.232	0.236	0.239
0.556	0.204	0.204	0.204	0.204	0.204	0.204
0.611	0.2	0.2	0.2	0.2	0.2	0.2
0.667	0.2	0.2	0.2	0.2	0.2	0.2
0.722	0.2	0.2	0.2	0.2	0.2	0.2
0.778	0.2	0.2	0.2	0.2	0.2	0.2
0.833	0.2	0.2	0.2	0.2	0.2	0.2
0.889	0.2	0.2	0.2	0.2	0.2	0.2
0.944	0.2	0.2	0.2	0.2	0.2	0.2
1.000	0.2	0.2	0.2	0.2	0.2	0.2

Shock-like solution

Another analytic solution of the Burgers' equation is a shock-like one (Zhu and Wang, 2009)

$$\overline{u}(x,t) = \frac{x/t}{1+\sqrt{t/t_0}\exp(x^2 Re/4t)}, \quad 0 \le x \le 1, \quad t \ge 1, \tag{3.60}$$

where $t_0 = \exp(Re/8)$. This solution represents the propagation of the shock. We take the solution (3.60) at $t = 1$ as the initial condition, and conduct the calculation at $Re \in \{100,\ 200\}$ on a grid of $N = 51$ with $\Delta t = 0.01$ and at $Re = 2000$ on a grid of $N = 201$ with $\Delta t = 0.005$.

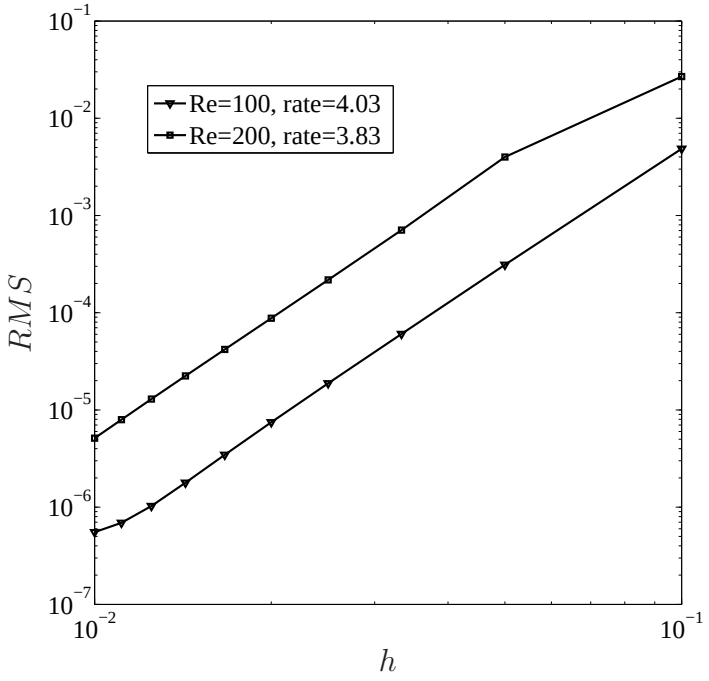


Figure 3.6: Shock-like solution, $N \in \{11,\ 21, \ldots,\ 101\}$, $Re \in \{100,\ 200\}$, $\Delta t = 10^{-5}$, $t = 1.7$: The effect of grid size h on the solution accuracy by the proposed scheme. The solution error behaves as $Ne \approx O(h^{4.03})$ for $Re = 100$ and $Ne \approx O(h^{3.83})$ for $Re = 200$.

The present errors in terms of RMS and L_∞ together with some other published results are given in Table 3.2. The latter was obtained by the quadratic (QBGM) and cubic (CBGM) B-spline based finite Galerkin methods (Dag, Saka and Boz, 2005), the septic B-spline method (Ramadanand et al., 2005), the quartic B-spline collocation methods (QBCM1 and QBCM2) (Saka and Dag, 2007) and the multiquadric quasi-interpolation (MQQI) method (Chen and Wu, 2006) (results presented here correspond to the best value of the shape parameter c).

Table 3.2: Shock-like solution: RMS and L_∞ errors by the present and some other numerical methods.

x	$RMS \times 10^3$	$RMS \times 10^3$	$RMS \times 10^3$	$L_\infty \times 10^3$	$L_\infty \times 10^3$	$L_\infty \times 10^3$
N=51, Re=100, $\Delta t = 0.01$	t=1.7	t=2.1	t=2.6	t=1.7	t=2.1	t=2.6
Present CIRBF	0.51333	0.39849	0.30645	1.29020	0.94819	0.70402
BSQI (Zhu and Wang, 2009)	0.82751	0.98595	1.58264	2.59444	2.35031	5.73827
MQQI (Chen and Wu, 2006) ($c = 10^{-3}$)	5.89555	6.64358	6.90385	14.7550	15.9892	16.3403
QBCM1 (Saka and Dag, 2007)	0.17014	0.20476	1.29951	0.40431	0.86363	6.69425
QBCM2 (Saka and Dag, 2007)	0.24003	0.30849	1.57548	0.48800	1.14760	8.06799
Ref. (Ramadanand et al., 2005)	0.69910	0.72976	1.74570	3.13476	2.66986	8.06798
N=51, Re=200, $\Delta t = 0.01$	t=1.8	t=2.4	t=3.2	t=1.8	t=2.4	t=3.2
Present CIRBF	0.79475	0.56208	0.39224	2.69630	1.77650	1.19470
BSQI (Zhu and Wang, 2009)	1.66464	2.06695	2.36889	5.12020	6.31491	6.85425
MQQI (Chen and Wu, 2006) ($c = 10^{-2}$)	6.88480	7.89738	8.56856	25.6767	27.2424	2.68122
QBCM1 (Saka and Dag, 2007)	0.19127	0.14246	0.93617	0.54058	0.39241	5.54899
QBCM2 (Saka and Dag, 2007)	0.49130	0.41864	1.28863	1.16930	0.93664	7.49147
Ref. (Ramadanand et al., 2005)	0.68761	0.67943	1.48559	2.47189	2.16784	7.49146
N=201, Re=2000, $\Delta t = 0.005$	t=1.7	t=2.4	t=3.1	t=1.7	t=2.4	t=3.1
Present CIRBF	1.86700	1.07860	0.74665	19.8620	12.3130	7.87850
BSQI (Zhu and Wang, 2009)	1.73914	1.60551	2.17489	4.57388	5.06708	6.21046
MQQI (Chen and Wu, 2006) ($c = 10^{-3}$)	85.5754	91.4861	92.0002	370.921	313.435	275.749
QBCM1 (Saka and Dag, 2007)	0.01705	0.01252	0.60199	0.06192	0.05882	4.43469
QBCM2 (Saka and Dag, 2007)	0.35891	0.25132	0.63052	1.21170	0.80777	4.79061
QBGM (Dag, Saka and Boz, 2005)	0.35133	0.24451	0.63335	1.20755	0.80187	4.79061
CBGM (Dag, Saka and Boz, 2005)	0.35126	0.24448	0.63340	1.20726	0.80176	4.79061

It can be seen that the proposed scheme outperforms the MQQI scheme and produces results that are comparative with those by the other schemes.

Figure 3.6 shows a fast rate of convergence with grid refinement of the proposed scheme, where a set of grids, namely $\{11,\ 21, \ldots,\ 101\}$ is employed and the solution is computed at $t = 1.7$ using $\Delta t = 10^{-5}$. The convergence rate is of $O(h^{4.03})$ for $Re = 100$ and $O(h^{3.83})$ for $Re = 200$.

3.4.4 Taylor decaying vortices

This problem is governed by the transient Navier-Stokes equations (3.7)-(3.8) in which the body-force vector $\mathbf{f}_b$ is neglected. Taylor decaying vortices are often employed to verify numerical schemes in CFD. The flow describes an initially periodical vortex structure convected by the flow field and exponentially decaying due to the viscous effects. The exact solution of this problem is known to be (Tian et al., 2011)

$$\overline{u}(x, y, t) = -\cos(kx)\sin(ky)\exp(-2k^2t/Re), \tag{3.61}$$

$$\overline{v}(x, y, t) = \sin(kx)\cos(ky)\exp(-2k^2t/Re), \tag{3.62}$$

$$\overline{p}(x, y, t) = -0.25(\cos(2kx) + \cos(2ky))\exp(-4k^2t/Re), \tag{3.63}$$

where k is an integer. The initial and boundary conditions for the velocity are derived from the exact solution (3.61)-(3.62) whereas the reference pressure $p = 0$ is specified at the centre of the domain.

As in Tian et al. (2011), we also employ $k = 2$, $\Omega = [0, \pi] \times [0, \pi]$, $t = 2$, $\Delta t = 0.002$ and $Re = 100$, and conduct the simulation on a set of grids, namely $\{11 \times 11,\ 21 \times 21, \ldots,\ 51 \times 51\}$. Numerical results indicate that the proposed scheme is more than third-order accurate in space (i.e. $Ne \approx O(h^{3.4})$) for the velocity (Table 3.3) and more than fourth-order accurate in space (i.e. $Ne \approx O(h^{4.11})$) for the pressure (Table 3.4). In comparison with the standard second-order FDM and the compact FDM in Tian et al. (2011), the present scheme performs better in terms of the accuracy, the average rate of convergence and the CPU time (for a given accuracy) (Tables 3.3 and 3.4). For example, to achieve the accuracy of about 1.3×10^{-3}, the grid density and CPU time required are about 21×21 and 2.517s for the present scheme, more than 31×31 and 2.625s for the compact FDM and more than 51×51 and 10.141s for the standard FDM (Table 3.4). It is noted that the simulations are carried out on a DELL computer which has 3.87 GB of RAM and one processor Intel (R) Core (TM)2 Duo CPU E8400 of 3.0 GHz. All codes are written using the MATLAB language.

Table 3.3: Taylor decaying vortices, $k = 2$, $\Delta t = 0.002$, $t = 2$, $Re = 100$: RMS errors and average rates of convergence for the velocity by the present and some other numerical methods.

Grid	Compact FDM (Tian et al., 2011) $RMS(u)$	Compact IRBF $RMS(u)$	Compact FDM (Tian et al., 2011) $RMS(v)$	Compact IRBF $RMS(v)$
11×11	7.0070E-02	6.3558E-03	7.0070E-02	6.0332E-03
21×21	9.0692E-03	2.5467E-04	9.0692E-03	2.5316E-04
31×31	2.8851E-03	8.0732E-05	2.8851E-03	8.0587E-05
41×41	1.2239E-03	3.9428E-05	1.2239E-03	3.9396E-05
51×51	6.3063E-04	2.5607E-05	6.3063E-04	2.5599E-05
	$Ne \approx O(h^{3.06})$	$Ne \approx O(h^{3.43})$	$Ne \approx O(h^{3.06})$	$Ne \approx O(h^{3.40})$

Table 3.4: Taylor decaying vortices, $k = 2$, $\Delta t = 0.002$, $t = 2$, $Re = 100$: RMS errors, average rates of convergence for the pressure and CPU time (seconds) by the present and some other numerical methods.

	Second-order FDM (Tian et al., 2011)		Compact FDM (Tian et al., 2011)		Compact IRBF	
Grid	$RMS(p)$	CPU (s)	$RMS(p)$	CPU (s)	$RMS(p)$	CPU (s)
11×11	1.405E-01	0.281	1.076E-01	0.344	2.988E-02	0.756
21×21	4.160E-02	0.640	1.057E-02	0.875	1.350E-03	2.517
31×31	1.897E-02	1.984	2.910E-03	2.625	2.569E-04	9.759
41×41	1.073E-02	5.156	1.136E-03	6.266	8.714E-05	24.298
51×51	6.869E-03	10.141	5.393E-04	12.015	4.156E-05	58.934
	$Ne \approx O(h^{1.97})$		$Ne \approx O(h^{3.44})$		$Ne \approx O(h^{4.11})$	

3.4.5 Torsionally oscillating lid-driven cavity flow

In chapter 2, the steady state lid-driven square cavity flow was studied with a similar 3-point stencil. The proposed CIRBF scheme is now applied to simulate the flow in a square cavity driven by a simple harmonic oscillation of the top lid (Figure 3.7). This cavity problem is studied in Chung and Denis (1999), Iwatsu et al. (1992), Soh and Goodrich (1988). The flow is governed by the transient Navier-Stokes equations (3.7)-(3.8) without the body force. The oscillating lid velocity is given as $u(t) = U_0 \sin(\overline{\omega} t)$, where $\overline{\omega}$ is the frequency, with period of $K = 2\pi/\overline{\omega}$.

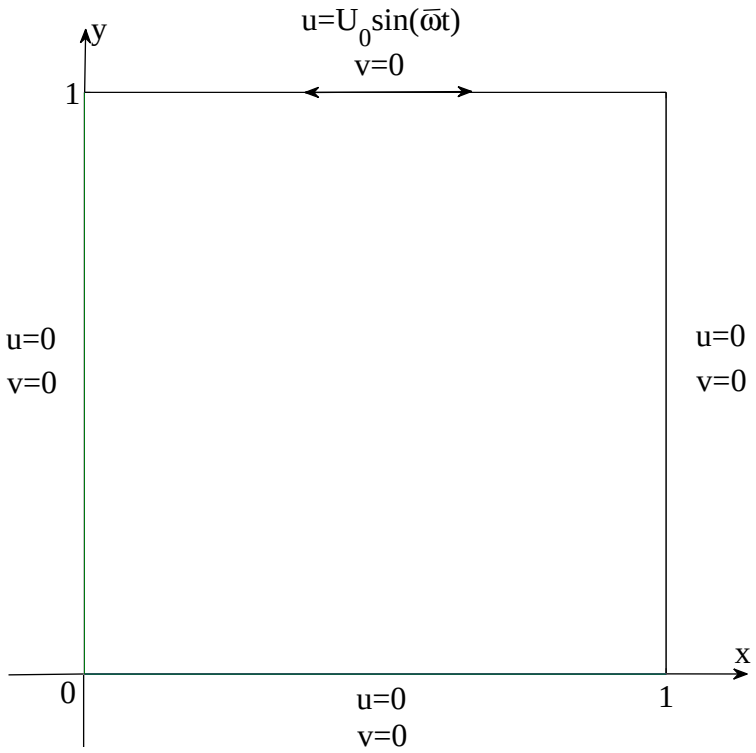


Figure 3.7: Torsionally oscillating lid-driven cavity flow: Geometry and boundary conditions.

In the present study, the flow starts from rest and the reference pressure $p = 0$ is specified at the centre of the cavity. It is observed that, after approximately 6 cycles, it reaches the periodic state. We are mainly interested in the simulation of the quasi-steady periodic motion of the fluid.

For this problem, the Reynolds number Re and the frequency $\overline{\omega}$ have a strong influence on the flow behaviour. We consider $Re \in \{100,\ 400,\ 1000\}$ based on the maximum lid velocity $U_0 = 1$ and $\overline{\omega} \in \{0.1,\ 1,\ 10\}$ which were used in Chung and Denis (1999), Iwatsu et al. (1992), Soh and Goodrich (1988). In the

case of $Re \in \{100,\ 400\}$, the calculations are carried out on a grid of 65×65 using $\Delta t = 2\pi/500$ for $\overline{\omega} \in \{0.1,\ 1\}$ and $\Delta t = 2\pi/5000$ for $\overline{\omega} = 10$. In the case of $Re = 1000$, the calculations are carried out on a grid of 129×129 using $\Delta t = 2\pi/1000$ for $\overline{\omega} \in \{0.1,\ 1\}$ and $\Delta t = 2\pi/10000$ for $\overline{\omega} = 10$. For steady flow simulations through transient equations, a time step should be chosen as large as possible to save the computational time. More detail can be found in chapter 2. The u-velocity profiles along the vertical centreline are shown in Figure 3.8 at

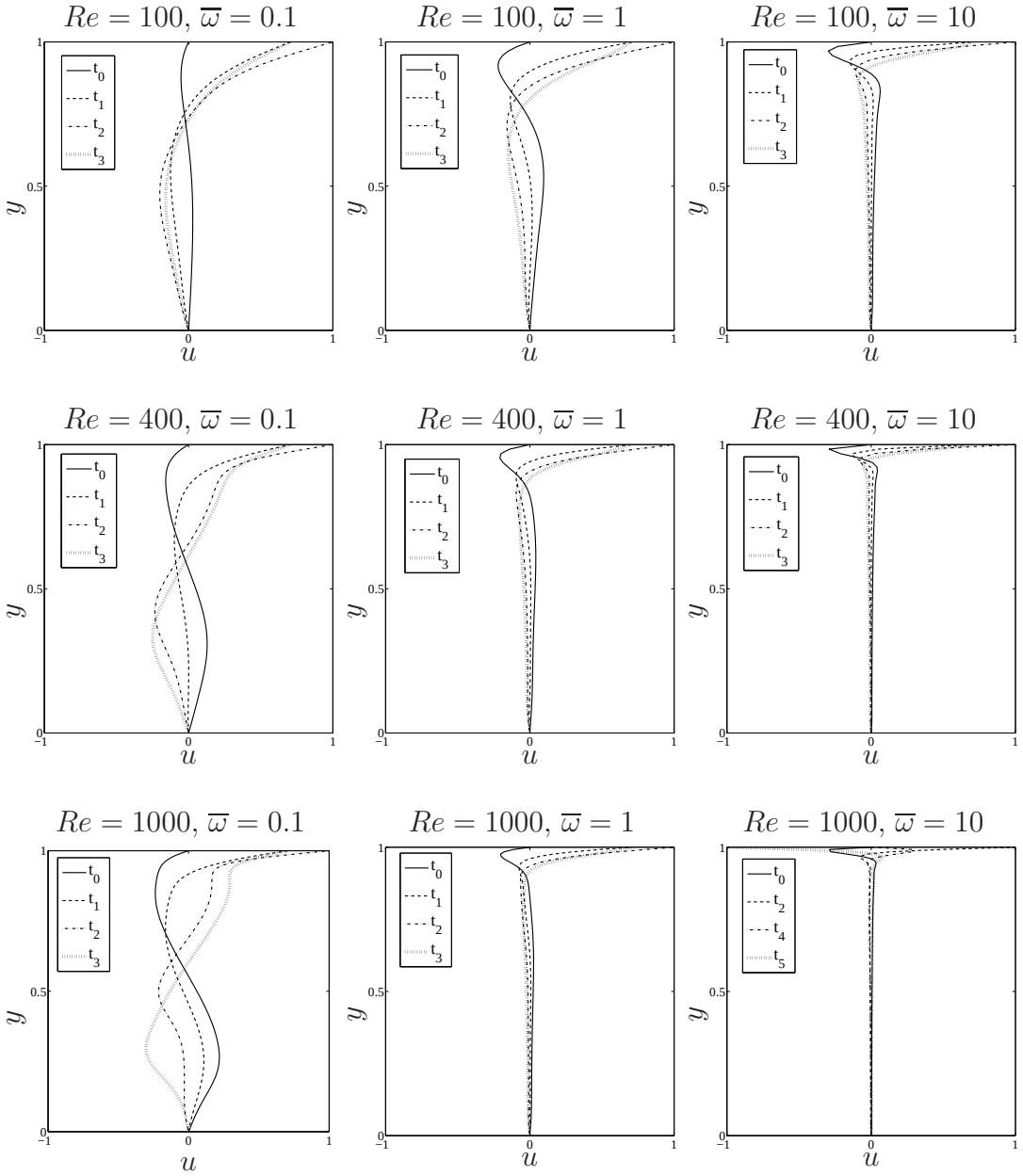


Figure 3.8: Torsionally oscillating lid-driven cavity flow: Profiles of u-velocity along the vertical centreline during a half cycle of the lid oscillation for three values of $\overline{\omega} \in \{0.1,\ 1,\ 10\}$ and three values of $Re \in \{100,\ 400,\ 1000\}$. Times used are $t_0 = 0$, $t_1 = K/8$, $t_2 = K/4$, $t_3 = 3K/8$, $t_4 = K/2$ and $t_5 = 3K/4$.

$Re \in \{100,\ 400,\ 1000\}$ and $\overline{\omega} \in \{0.1,\ 1,\ 10\}$. Firstly, we investigate the effect of the frequency on the flow by fixing the Reynolds number. It can be seen that the oscillating motion of the lid has an influence over a small region adjacent to the lid for high frequencies and more deeply into the cavity at lower-frequency oscillations. Secondly, we investigate the effect of the Reynolds number by fixing the frequency. The effect of the oscillating lid penetrates the cavity more deeply as the Reynolds number increases for $\overline{\omega} = 0.1$, but vice versa for $\overline{\omega} \in \{1,\ 10\}$. In the case of high frequency and high Reynolds number (i.e. $\overline{\omega} = 10$ and $Re = 1000$), the flow is seen to be confined to a very small depth of penetration from the top of the lid, forming a boundary layer. The v-velocity profiles along the horizontal centreline are shown in Figure 3.9 for aforementioned Reynolds numbers and frequencies. In the case of high frequency, the magnitude of the maximum v-velocity in the cavity is seen to decrease to a very small value as the Reynolds number increases. In the interior core, the fluid is nearly stagnant. These are in support of the previous statement that, in the high frequency range, the flows are predominantly in the x-direction and concentrated only in a narrow strip which is very near the top lid. The streamlines at $Re \in \{400,\ 1000\}$ and $\overline{\omega} \in \{1,\ 10\}$ are shown in Figures 3.10-3.13. In the case of $Re = 400$ and $\overline{\omega} = 1$, the counter rotating vortex appears at the upper right corner at $t = 0$. The primary vortex is surrounded by the counter-rotating vortex at $t = K/8$, and becomes very weak at $t = 3K/8$. However, at $\overline{\omega} = 10$, the primary vortex exists in a short period from 0 to $K/8$ and then dies out at $t = K/4$. In the case of $Re = 1000$ and $\overline{\omega} = 1$, the size of the counter-rotating vortex is relatively large at $t = 0$. The primary vortex is reduced in size and strength as t increases, and vanishes at the end of the first half-cycle. At $\overline{\omega} = 10$, the flow structure is very similar to that in the case of $Re = 400$.

Our observations here are similar to those reported in the literature such as Iwatsu et al. (1992) (Figures 2-7 and their descriptions) and Chung and Denis (1999) (Figures 8-9 and 10-13 and their descriptions). It is noted that a quantitative comparison is not possible since numerical values were not reported in Chung and Denis (1999), Iwatsu et al. (1992).

3.5 Concluding remarks

In this chapter, a new Cartesian-grid collocation discretisation scheme for time-dependent problems is presented. The present approximations are based on 3-point stencils, resulting in a sparse system matrix, while Adams-Bashforth/Crank Nicolson algorithms and IRBF-based compact approximations are employed to yield a high-order accurate solution in time and space, respectively. The proposed scheme is verified successfully with a series of linear and nonlinear problems, including Taylor decaying vortices and torsionally oscillating lid-driven cavity flows.

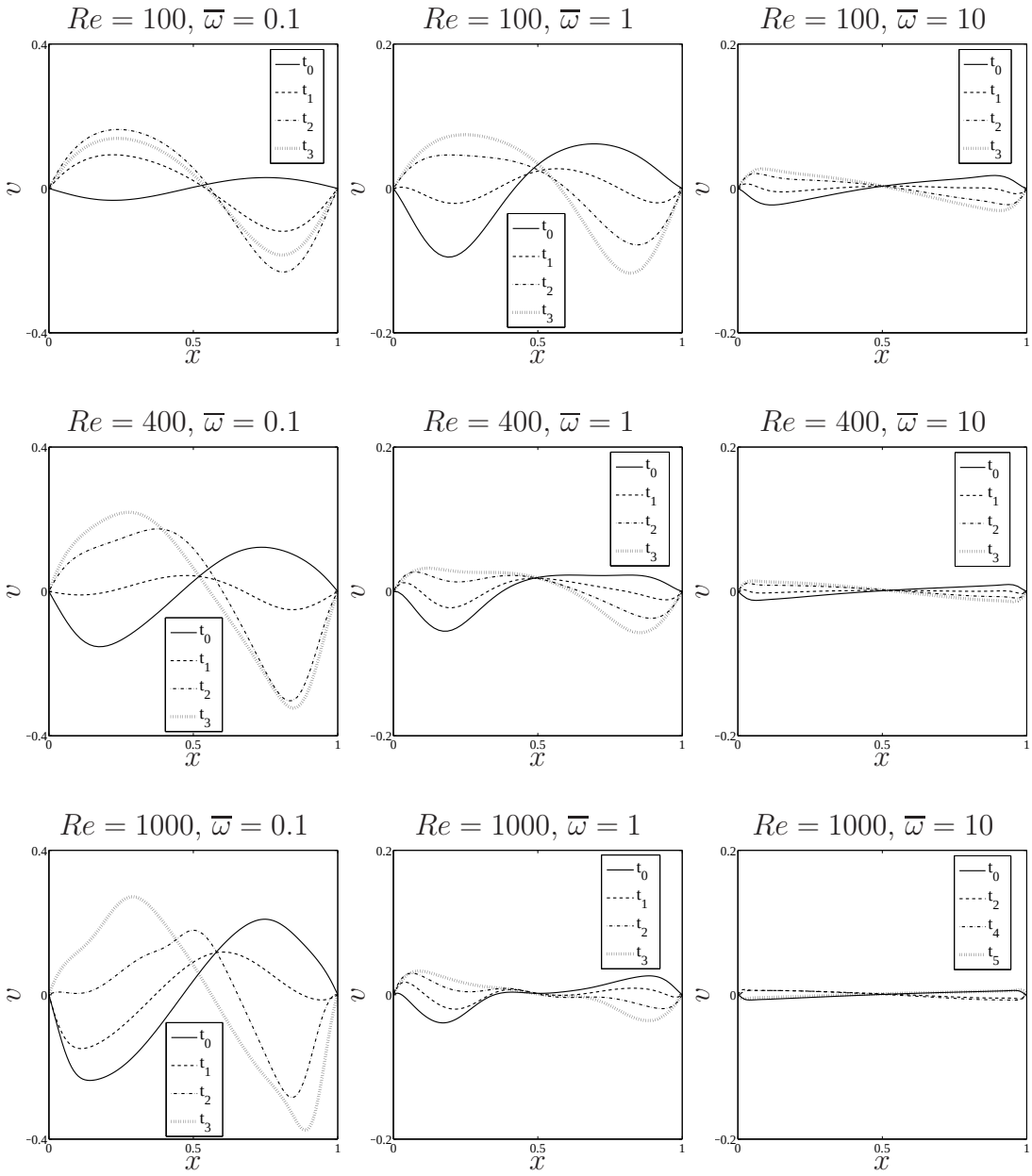


Figure 3.9: Torsionally oscillating lid-driven cavity flow: Profiles of v-velocity along the horizontal centreline during a half cycle of the lid oscillation for three values of $\overline{\omega} \in \{0.1,\ 1,\ 10\}$ and three values of $Re \in \{100,\ 400,\ 1000\}$. Times used are $t_0 = 0$, $t_1 = K/8$, $t_2 = K/4$, $t_3 = 3K/8$, $t_4 = K/2$ and $t_5 = 3K/4$.

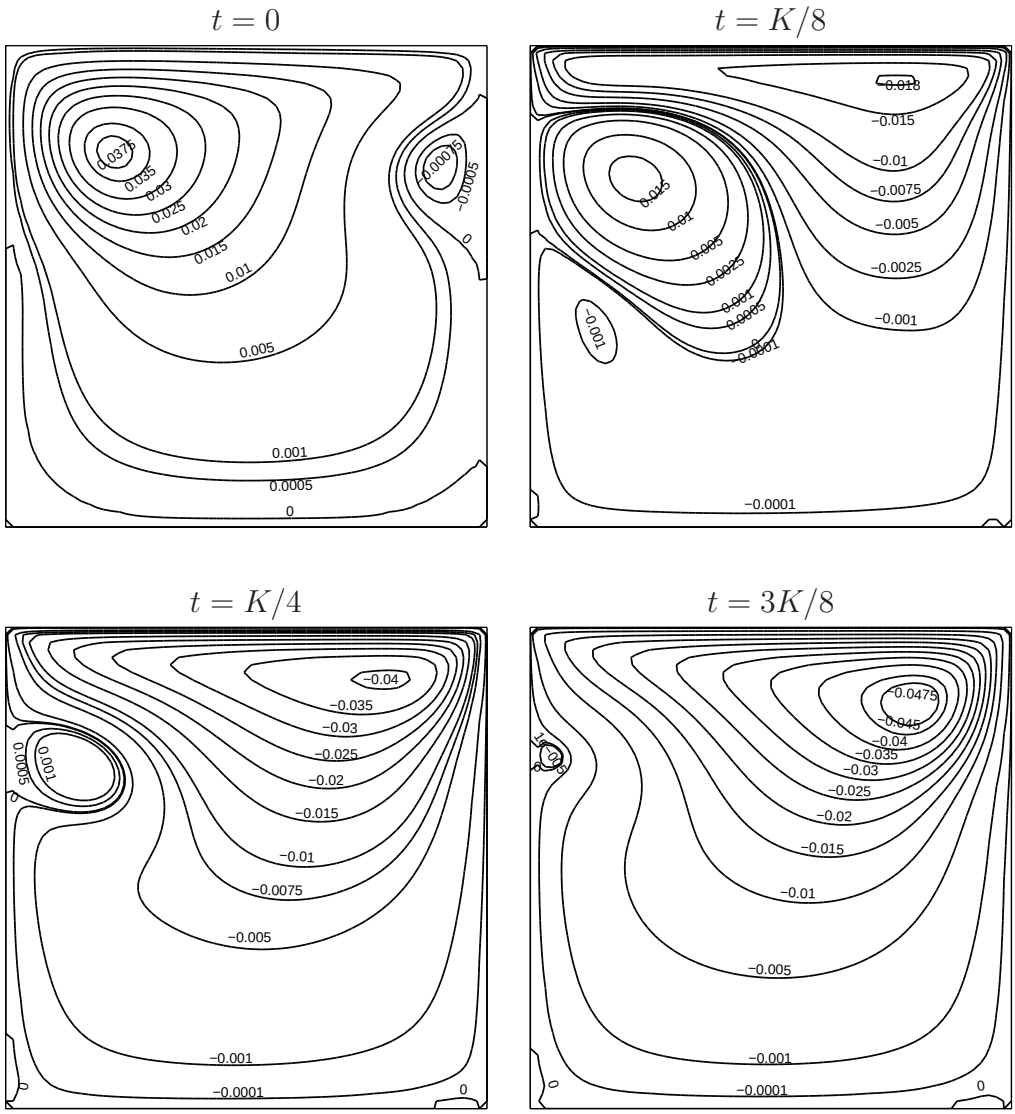


Figure 3.10: Torsionally oscillating lid-driven cavity flow, 65×65: Evolution of streamlines during a half-cycle of the lid motion at $Re = 400$ and $\overline{\omega} = 1$.

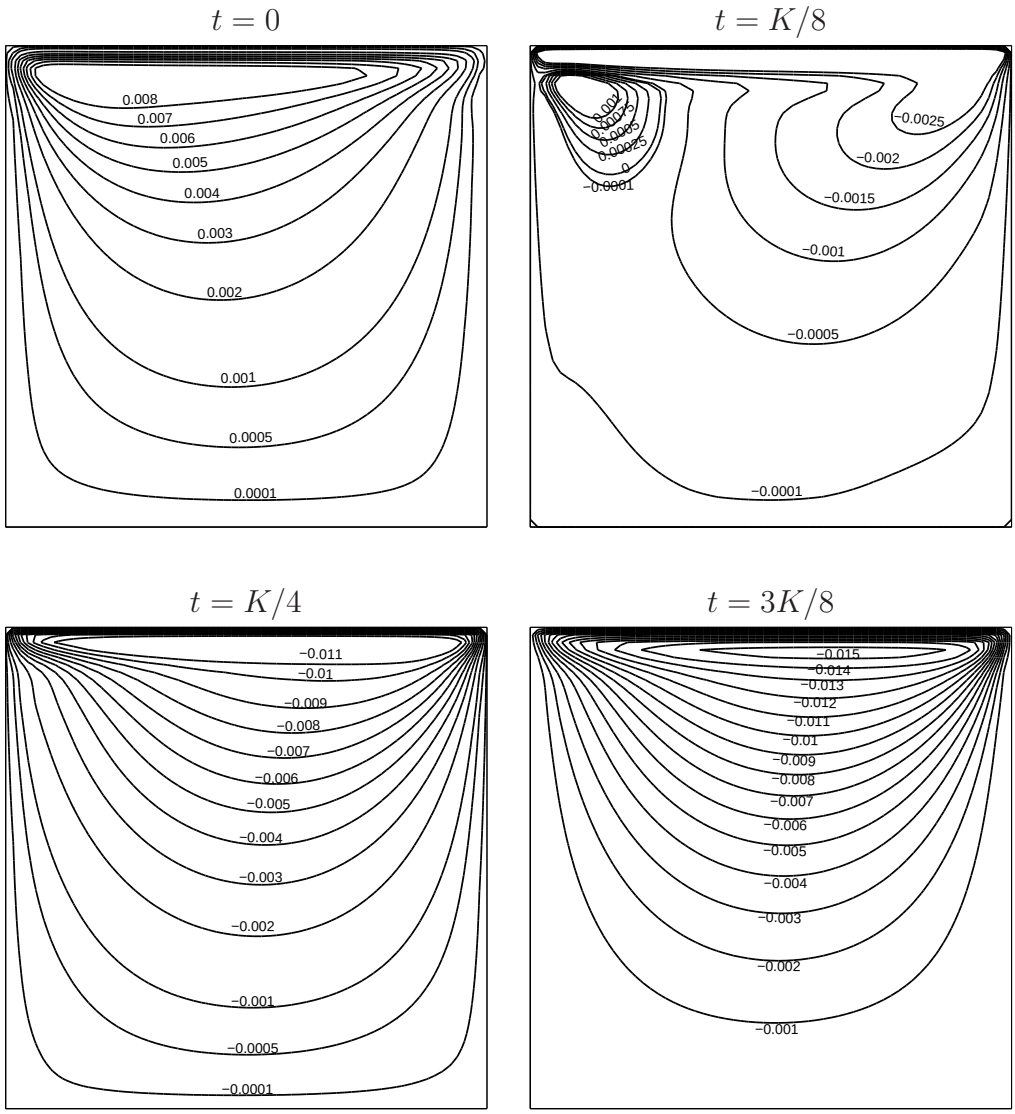


Figure 3.11: Torsionally oscillating lid-driven cavity flow, 65×65: Evolution of streamlines during a half-cycle of the lid motion at $Re = 400$ and $\overline{\omega} = 10$.

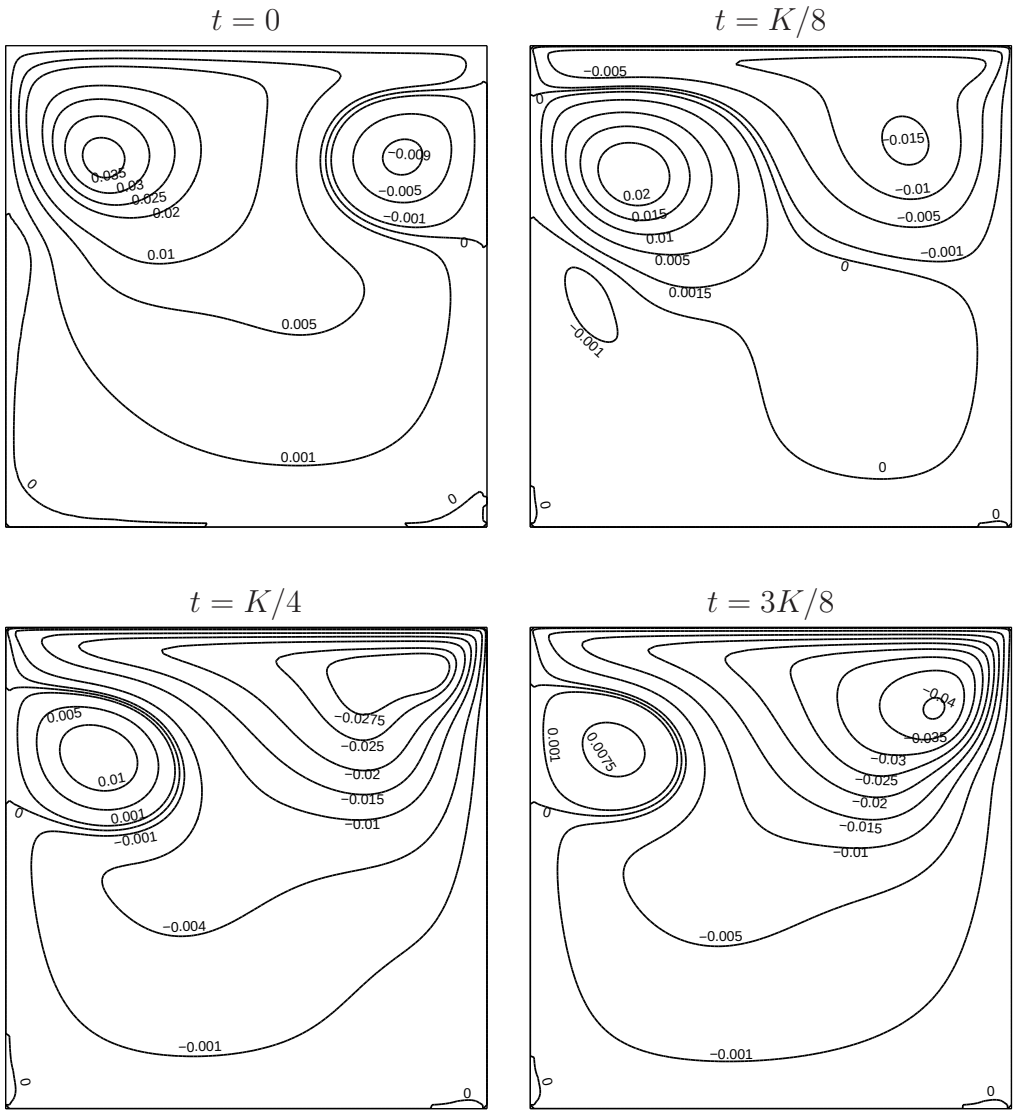


Figure 3.12: Torsionally oscillating lid-driven cavity flow, 129×129: Evolution of streamlines during a half-cycle of the lid motion at $Re = 1000$ and $\overline{\omega} = 1$.

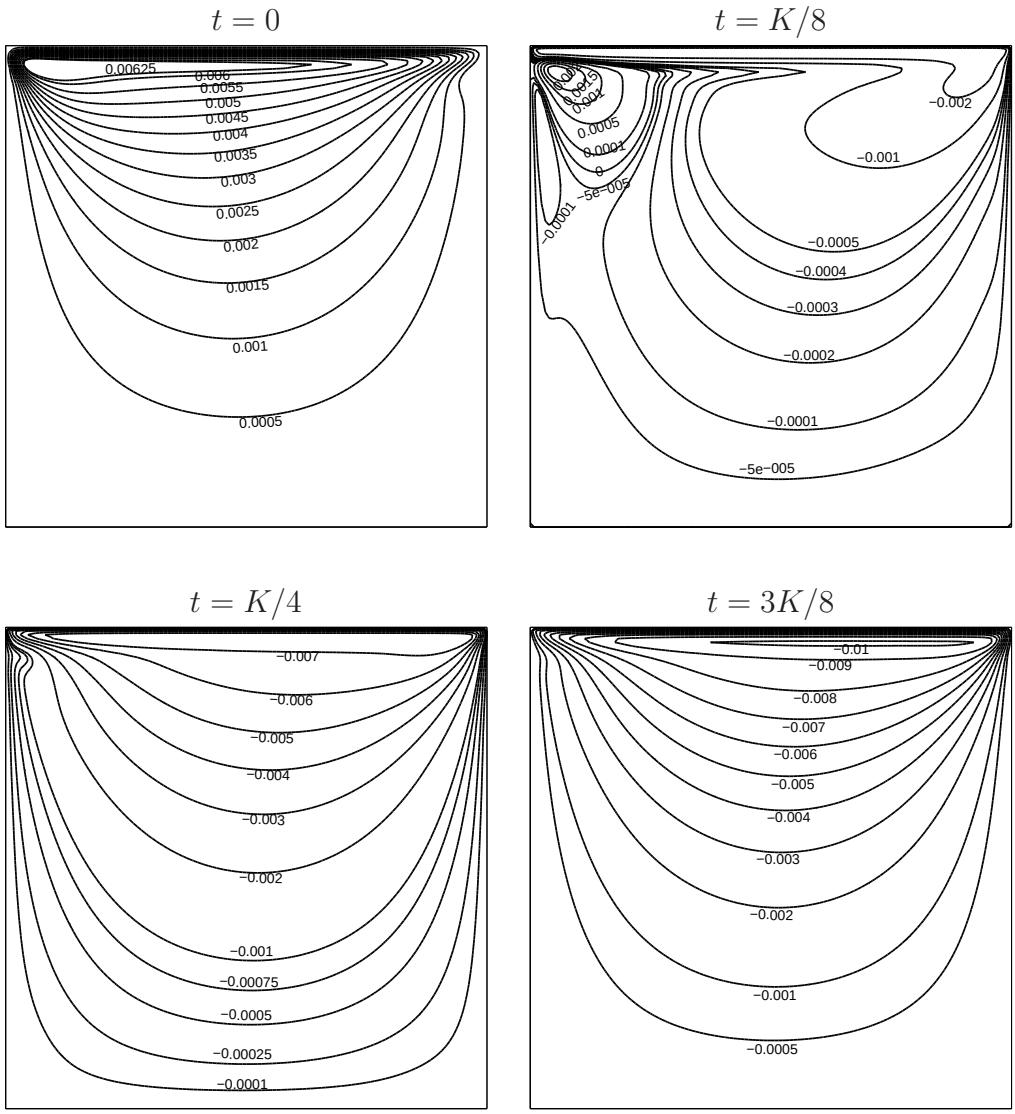


Figure 3.13: Torsionally oscillating lid-driven cavity flow, 129×129: Evolution of streamlines during a half-cycle of the lid motion at $Re = 1000$ and $\overline{\omega} = 10$.

Chapter 4

Incorporation of Alternating Direction Implicit (ADI) algorithm into compact IRBF scheme

In this chapter, the ADI algorithm reported in You (2006) for the convection-diffusion equation is implemented in the context of CIRBF approximations. The CIRBF approximations are constructed over 3-point stencils, where extra information is incorporated via two forms: only nodal second-order derivative values (Scheme 1), and both nodal first- and second-order derivative values (Scheme 2). The resultant algebraic systems are sparse, especially for Scheme 2 (tridiagonal matrices). Several steady and non-steady problems are considered to verify the present schemes and to compare their accuracy with some other ADI schemes. Numerical results show that highly accurate results are obtained with the proposed methods.

4.1 Introduction

In this chapter, we consider a 2D unsteady convection-diffusion equation for a variable u

$$\frac{\partial u}{\partial t} + c_x \frac{\partial u}{\partial x} + c_y \frac{\partial u}{\partial y} = d_x \frac{\partial^2 u}{\partial x^2} + d_y \frac{\partial^2 u}{\partial y^2} + f_b, \quad (x, y) \in \Omega,\ t \geq 0, \tag{4.1}$$

subject to the initial condition

$$u(x, y) = u_0(x, y), \quad (x, y) \in \Omega,\ t = 0, \tag{4.2}$$

and Dirichlet boundary condition

$$u(x, y, t) = u_\Gamma(x, y, t), \quad (x, y) \in \Gamma,\ t \geq 0, \tag{4.3}$$

where relevant variables are defined before in chapters 2, 3. In equation (4.1), c_x and c_y are the convective velocities, and d_x and d_y are the positive diffusion coefficients, respectively.

For the steady-state case, equation (4.1) reduces to

$$c_x \frac{\partial u}{\partial x} + c_y \frac{\partial u}{\partial y} = d_x \frac{\partial^2 u}{\partial x^2} + d_y \frac{\partial^2 u}{\partial y^2} + f_b. \tag{4.4}$$

Equations (4.1) and (4.4) are known as a simplified version of the Navier-Stokes equation. They are widely used in CFD and physical sciences to describe the transport of mass, momentum, vorticity, heat and energy, the modeling of semiconductors, etc. For example, by means of (4.1), one can describe the heat transfer in a draining film (Isenberg and Gutfinger, 1972), water transfer in soils (Parlange, 1980) and the chemical separation processes (Dehghan, 2004).

It is desirable to have accurate and stable methods for solving the convection-diffusion equation. The upwind and central finite differences are among popular discretisation schemes for the approximation of spatial derivative terms because they are simple and easy to implement. These finite-difference schemes generally yield good results on sufficiently fine meshes. However, poor results may be obtained if the mesh used is relatively coarse. To improve the accuracy order, larger stencils can be used. The drawback of this approach is that the bandwidth of their coefficient matrices is increased, and thus it is time-consuming to solve such systems either by using direct solvers, e.g. Gaussian elimination and LU decomposition technique, or iterative methods, e.g. a generalized minimal residual algorithm (GMRES) and biconjugate gradient stabilised method (BICGSTAB). This leads to the development of compact finite difference methods, where small matrix bandwidth and high-order accuracy can be achieved together (Kalita and Chhabra, 2006; Kalita et al., 2002; Noye and Tan, 1989).

The ADI methods are highly efficient procedures for solving parabolic and hyperbolic problems (Thomas, 1995). As shown in Thomas (1995), the efficiency of ADI methods is based on reducing problems in several space variables to a number of one-dimensional problems. The standard PR-ADI method (Peaceman and H. H. Rachford, 1955) has been popular because of its computational cost-effectiveness. However, due to its low-order accuracy, the method often produces significant dissipation and phase errors (Karaa and Zhang, 2004). To enhance spatial accuracy, Karaa and Zhang (2004) has developed a high-order compact ADI (HOC-ADI) scheme, which possesses fourth-order accuracy and still retains the tridiagonal algorithm of the standard PR-ADI. For solving the convection-dominated diffusion (CDD) equation (i.e. high Peclet numbers), HOC-based schemes may suffer from excessive numerical dissipation (Ma et al.,

2012; You, 2006). A high-order ADI method proposed in You (2006) was designed to overcome this problem, where its factorisation involves four terms and spatial derivatives are approximated using the Padé algorithm.

In this chapter, we implement the ADI method proposed in You (2006) in the context of CIRBF approximations for the convection-diffusion equation. Two compact 3-point schemes, namely ADI-CIRBF-1 and ADI-CIRBF-2, for the spatial discretisation are proposed. Scheme 1 incorporates nodal values of the second-order derivatives into the approximations, while Scheme 2 includes not only nodal second-order derivative values but also nodal first-derivative values. The resultant algebraic systems are sparse, especially for Scheme 2 (tridiagonal matrices). The performances of the two present schemes are investigated numerically through the solution of several analytic test problems governed by the unsteady and steady 2D convection-diffusion equations. Results obtained are also compared with those obtained by the standard PR-ADI scheme and some other high-order compact ADI schemes. The remainder of the chapter is organised as follows. Section 4.2 gives a brief review of some ADI methods. Section 4.3 describes the two proposed schemes. In section 4.4, numerical results are presented and compared with some published solutions. Section 4.5 concludes the chapter.

4.2 A brief review of ADI methods

4.2.1 The Peaceman-Rachford method

The PR-ADI method splits equation (4.1) into two

$$\frac{u^{n-1/2}-u^{n-1}}{\Delta t/2}+c_x\frac{\partial u^{n-1/2}}{\partial x}+c_y\frac{\partial u^{n-1}}{\partial y}=d_x\frac{\partial^2 u^{n-1/2}}{\partial x^2}+d_y\frac{\partial^2 u^{n-1}}{\partial y^2}+f_b^{n-1/2}, \quad (4.5)$$

$$\frac{u^{n}-u^{n-1/2}}{\Delta t/2}+c_x\frac{\partial u^{n-1/2}}{\partial x}+c_y\frac{\partial u^{n}}{\partial y}=d_x\frac{\partial^2 u^{n-1/2}}{\partial x^2}+d_y\frac{\partial^2 u^{n}}{\partial y^2}+f_b^{n-1/2}, \quad (4.6)$$

where the derivatives with respect to x and y are treated implicitly in the first and second equations, respectively. The PR-ADI method often leads to significant dissipation and phase errors due to its low-order accuracy in the spatial discretisation.

4.2.2 The Douglas-Rachford method

The Douglas-Rachford method (Douglas and Rachford, 1956) is a variant of the Peaceman-Rachford method. Applying this method to (4.1), one obtains

$$\frac{u^{*,n}-u^{n-1}}{\Delta t}+c_x\frac{\partial u^{*,n}}{\partial x}+c_y\frac{\partial u^{n-1}}{\partial y}=d_x\frac{\partial^2 u^{*,n}}{\partial x^2}+d_y\frac{\partial^2 u^{n-1}}{\partial y^2}+f_b^n, \tag{4.7}$$

$$\frac{u^{n}-u^{n-1}}{\Delta t}+c_x\frac{\partial u^{*,n}}{\partial x}+c_y\frac{\partial u^{n}}{\partial y}=d_x\frac{\partial^2 u^{*,n}}{\partial x^2}+d_y\frac{\partial^2 u^{n}}{\partial y^2}+f_b^n. \tag{4.8}$$

Unlike the Peaceman-Rachford method, this scheme is very easy to generalise to operator decompositions involving more than two operators (Glowinski et al., 2003). However, it still retains the drawbacks of the standard Peaceman-Rachford method.

4.2.3 Karaa's method

This method factorises equation (4.1) as

$$\left(L_x+\frac{\Delta t}{2}A_x\right)\left(L_y+\frac{\Delta t}{2}A_y\right)u^n=\left(L_x-\frac{\Delta t}{2}A_x\right)\left(L_y-\frac{\Delta t}{2}A_y\right)u^{n-1}, \tag{4.9}$$

where

$$L_x=1+\frac{\Delta x^2}{12}\left(\delta_x^2-\frac{c_x}{d_x}\delta_x\right),\quad A_x=-\left(d_x+\frac{c_x^2\Delta x^2}{12d_x}\right)\delta_x^2+c_x\delta_x, \tag{4.10}$$

$$L_y=1+\frac{\Delta y^2}{12}\left(\delta_y^2-\frac{c_y}{d_y}\delta_y\right),\quad A_y=-\left(d_y+\frac{c_y^2\Delta y^2}{12d_y}\right)\delta_y^2+c_y\delta_y, \tag{4.11}$$

δ_η and δ_η^2 are the first- and second-order central difference operators for η-direction; and Δx and Δy the mesh size.

Introducing an intermediate variable u^*, equation (4.9) can be solved by the following two steps

$$\left(L_x+\frac{\Delta t}{2}A_x\right)u^*=\left(L_x-\frac{\Delta t}{2}A_x\right)\left(L_y-\frac{\Delta t}{2}A_y\right)u^{n-1}, \tag{4.12}$$

$$\left(L_y+\frac{\Delta t}{2}A_y\right)u^n=u^*. \tag{4.13}$$

4.2.4 You's method

You (2006) proposed the following factorisation to equation (4.1)

$$\left(1+\frac{\Delta t}{2}c_x\frac{\partial}{\partial x}\right)\left(1-\frac{\Delta t}{2}d_x\frac{\partial^2}{\partial x^2}\right)\left(1+\frac{\Delta t}{2}c_y\frac{\partial}{\partial y}\right)\left(1-\frac{\Delta t}{2}d_y\frac{\partial^2}{\partial y^2}\right)u^n = \left(1-\frac{\Delta t}{2}c_x\frac{\partial}{\partial x}\right)\left(1+\frac{\Delta t}{2}d_x\frac{\partial^2}{\partial x^2}\right)\left(1-\frac{\Delta t}{2}c_y\frac{\partial}{\partial y}\right)\left(1+\frac{\Delta t}{2}d_y\frac{\partial^2}{\partial y^2}\right)u^{n-1} + \Delta t f_b^{n-1/2}. \quad (4.14)$$

In the matrix-vector notation, equation (4.14) becomes

$$L_x^{-1}T_x^+L_{xx}^{-1}T_{xx}^-L_y^{-1}T_y^+L_{yy}^{-1}T_{yy}^-u^n = L_x^{-1}T_x^-L_{xx}^{-1}T_{xx}^+L_y^{-1}T_y^-L_{yy}^{-1}T_{yy}^+u^{n-1}, \quad (4.15)$$

where

$$T_x^\pm = \left(L_x \pm \frac{\Delta t}{2}c_xA_x\right), \qquad T_{xx}^\pm = \left(L_{xx} \pm \frac{\Delta t}{2}d_xB_{xx}\right), \quad (4.16)$$
$$T_y^\pm = \left(L_y \pm \frac{\Delta t}{2}c_yA_y\right), \qquad T_{yy}^\pm = \left(L_{yy} \pm \frac{\Delta t}{2}d_yB_{yy}\right),$$

L_x, L_{xx}, A_x, B_{xx}, L_y, L_{yy}, A_y and B_{yy} are coefficient matrices derived from the standard fourth-order Padé schemes.

The equation (4.15) can be solved by the following two steps

$$L_x^{-1}T_x^+L_{xx}^{-1}T_{xx}^-u^* = L_x^{-1}T_x^-L_{xx}^{-1}T_{xx}^+L_y^{-1}T_y^-L_{yy}^{-1}T_{yy}^+u^{n-1}, \quad (4.17)$$

$$L_y^{-1}T_y^+L_{yy}^{-1}T_{yy}^-u^n = u^*. \quad (4.18)$$

The last two ADI methods (section 4.2.3 and 4.2.4) are preferable to the first two methods (section 4.2.1 and 4.2.2) in solving the convection-dominated diffusion (CDD) equation.

4.3 Proposed schemes

We propose two high-order numerical schemes, which are based on CIRBF approximations, for the spatial discretisation, and incorporated them into the ADI framework proposed in You (2006).

4.3.1 Spatial discretisation

Consider a 2D domain Ω as described in section 3.3.2. At an interior grid point $\mathbf{x}_{i,j}$, the associated stencils to be considered here are (i) two local stencils: $[x_{i-1,j}, x_{i,j}, x_{i+1,j}]$ in the x-direction and $[y_{i,j-1}, y_{i,j}, y_{i,j+1}]$ in the y-direction, and (ii) two global stencils: $[x_{1,j}, x_{2,j}, \ldots, x_{n_x,j}]$ ($x_{1,j}$ and $x_{n_x,j}$ are the two boundary nodes) in the x-direction and $\left[y_{i,1}, y_{i,2}, \ldots, y_{i,n_y}\right]$ ($y_{i,1}$ and y_{i,n_y} the two boundary nodes) in the y-direction. Hereafter, for brevity, we use η to denote x and y, and thus to have a generic local stencil $[\eta_1, \eta_2, \eta_3]$ ($\eta_1 < \eta_2 < \eta_3,\ \eta_2 \equiv \eta_{i,j}$) (Figure 2.2) and a generic global stencil $\left[\eta_1, \eta_2, \ldots, \eta_{n_\eta}\right]$ (Figure 4.1).

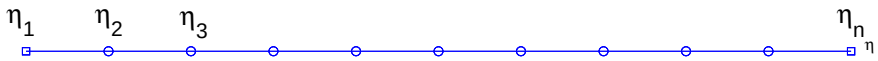


Figure 4.1: Global 1D-IRBF stencil.

Below are two proposed schemes whose constructions are based on (2.9)-(2.11). The difference between the two lies in (i) types of nodal derivatives used in their compact forms (i.e. second-order derivatives only for the first proposed scheme, and both first- and second-order derivatives for the second proposed scheme); and (ii) approximations for the boundary derivative values (i.e. global approximations for the first scheme and compact local approximations for the second scheme). The value of m is taken to be 3 for local stencils, and n_x or n_y for global stencils.

CIRBF-1

In this scheme 1, the compact approximations for the nodal first- and second-order derivative values at the interior nodes are exactly the same as described in section 2.4.1, 3.3.2, i.e. equations (2.27)-(2.28).

First-order derivative compact approximations:

At the current time level n, equation (2.27) is treated in an implicit manner as

$$\frac{du^n(\eta)}{d\eta} = \sum_{i=1}^{3} \frac{d\varphi_i(\eta)}{d\eta} u_i^n + \frac{d\varphi_4(\eta)}{d\eta} \frac{d^2u_1^n}{d\eta^2} + \frac{d\varphi_5(\eta)}{d\eta} \frac{d^2u_3^n}{d\eta^2}, \tag{4.19}$$

where the nodal second-order derivative values are also considered as unknowns.

Collocating (4.19) at $\eta = \eta_2$ results in

$$\frac{du_2^n}{d\eta} = \frac{d\varphi_1(\eta_2)}{d\eta}u_1^n + \frac{d\varphi_2(\eta_2)}{d\eta}u_2^n + \frac{d\varphi_3(\eta_2)}{d\eta}u_3^n + \frac{d\varphi_4(\eta_2)}{d\eta}\frac{d^2u_1^n}{d\eta^2} + \frac{d\varphi_5(\eta_2)}{d\eta}\frac{d^2u_3^n}{d\eta^2}, \quad (4.20)$$

or in matrix-vector form

$$\begin{bmatrix} 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \frac{du_1^n}{d\eta} \\ \frac{du_2^n}{d\eta} \\ \frac{du_3^n}{d\eta} \end{bmatrix} = \begin{bmatrix} \frac{d\varphi_1(\eta_2)}{d\eta} & \frac{d\varphi_2(\eta_2)}{d\eta} & \frac{d\varphi_3(\eta_2)}{d\eta} \end{bmatrix} \begin{bmatrix} u_1^n \\ u_2^n \\ u_3^n \end{bmatrix} + \begin{bmatrix} \frac{d\varphi_4(\eta_2)}{d\eta} & 0 & \frac{d\varphi_5(\eta_2)}{d\eta} \end{bmatrix} \begin{bmatrix} \frac{d^2u_1^n}{d\eta^2} \\ \frac{d^2u_2^n}{d\eta^2} \\ \frac{d^2u_3^n}{d\eta^2} \end{bmatrix}. \quad (4.21)$$

At the boundary nodes, nodal values of the first-order derivatives are approximated using the global 1D-IRBF approach (Mai-Duy and Tran-Cong, 2008)

$$\begin{pmatrix} \frac{du_1^n}{d\eta} \\ \frac{du_{n_\eta}^n}{d\eta} \end{pmatrix} = \begin{bmatrix} H_1(\eta_1) & H_1(\eta_{n_\eta}) \\ \vdots & \vdots \\ H_{n_\eta}(\eta_1) & H_{n_\eta}(\eta_{n_\eta}) \\ 1 & 1 \\ 0 & 0 \end{bmatrix}^T \begin{bmatrix} \overline{H}_1(\eta_1) & \cdots & \overline{H}_{n_\eta}(\eta_1) & \eta_1 & 1 \\ \overline{H}_1(\eta_2) & \cdots & \overline{H}_{n_\eta}(\eta_2) & \eta_2 & 1 \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ \overline{H}_1(\eta_{n_\eta}) & \cdots & \overline{H}_{n_\eta}(\eta_{n_\eta}) & \eta_{n_\eta} & 1 \end{bmatrix}^{-1} \begin{pmatrix} u_1^n \\ u_2^n \\ \vdots \\ u_{n_\eta}^n \end{pmatrix}, \quad (4.22)$$

where $u_i = u(\eta_i)$ $(i \in \{1,\ 2, \ldots,\ n_\eta\})$; and $\frac{du_i}{d\eta} = \frac{du}{d\eta}(\eta_i)$ $(i \in \{1,\ n_\eta\})$.

The IRBF system on a grid line for the first-order derivative of u is obtained by letting the interior node taking value from 2 to $(n_\eta - 1)$ in (4.21) and making use of (4.22)

$$\mathcal{L}_\eta \widehat{u}_\eta^n = \mathcal{A}_\eta \widehat{u}^n + \mathcal{A}_{\eta\eta}\widehat{u}_{\eta\eta}^n, \quad (4.23)$$

where $\mathcal{L}_\eta$, $\mathcal{A}_\eta$ and $\mathcal{A}_{\eta\eta}$ are $n_\eta \times n_\eta$ matrices, and

$$\widehat{u}^n = \left[u_1^n, u_2^n, \ldots, u_{n_\eta}^n\right]^T, \quad (4.24)$$

$$\widehat{u}_\eta^n = \left[\frac{du_1^n}{d\eta}, \frac{du_2^n}{d\eta}, \ldots, \frac{du_{n_\eta}^n}{d\eta}\right]^T, \quad (4.25)$$

$$\widehat{u}_{\eta\eta}^n = \left[\frac{d^2u_1^n}{d\eta^2}, \frac{d^2u_2^n}{d\eta^2}, \ldots, \frac{d^2u_{n_\eta}^n}{d\eta^2}\right]^T. \tag{4.26}$$

Second-order derivative compact approximations:

Collocating (2.28) at $\eta = \eta_2$ and treating in an implicit manner at the current time level n, leads to

$$-\frac{d^2\varphi_4(\eta_2)}{d\eta^2}\frac{d^2u_1^n}{d\eta^2} + \frac{d^2u_2^n}{d\eta^2} - \frac{d^2\varphi_5(\eta_2)}{d\eta^2}\frac{d^2u_3^n}{d\eta^2} = \frac{d^2\varphi_1(\eta_2)}{d\eta^2}u_1^n + \frac{d^2\varphi_2(\eta_2)}{d\eta^2}u_2^n + \frac{d^2\varphi_3(\eta_2)}{d\eta^2}u_3^n, \tag{4.27}$$

or in matrix-vector form

$$\begin{bmatrix} -\frac{d^2\varphi_4(\eta_2)}{d\eta^2} & 1 & -\frac{d^2\varphi_5(\eta_2)}{d\eta^2} \end{bmatrix} \begin{bmatrix} \frac{d^2u_1^n}{d\eta^2} \\ \frac{d^2u_2^n}{d\eta^2} \\ \frac{d^2u_3^n}{d\eta^2} \end{bmatrix} = \begin{bmatrix} \frac{d^2\varphi_1(\eta_2)}{d\eta^2} & \frac{d^2\varphi_2(\eta_2)}{d\eta^2} & \frac{d^2\varphi_3(\eta_2)}{d\eta^2} \end{bmatrix} \begin{bmatrix} u_1^n \\ u_2^n \\ u_3^n \end{bmatrix}. \tag{4.28}$$

At the boundary nodes, nodal values of the second-order derivatives are approximated as (Mai-Duy and Tran-Cong, 2008; Thai-Quang, Le-Cao, Mai-Duy and Tran-Cong, 2012)

$$\begin{pmatrix} \frac{d^2u_1^n}{d\eta^2} \\ \frac{d^2u_{n_\eta}^n}{d\eta^2} \end{pmatrix} = \begin{bmatrix} G_1(\eta_1) & G_1(\eta_{n_\eta}) \\ \vdots & \vdots \\ G_{n_\eta}(\eta_1) & G_{n_\eta}(\eta_{n_\eta}) \\ 0 & 0 \\ 0 & 0 \end{bmatrix}^T \begin{bmatrix} \overline{H}_1(\eta_1) & \cdots & \overline{H}_{n_\eta}(\eta_1) & \eta_1 & 1 \\ \overline{H}_1(\eta_2) & \cdots & \overline{H}_{n_\eta}(\eta_2) & \eta_2 & 1 \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ \overline{H}_1(\eta_{n_\eta}) & \cdots & \overline{H}_{n_\eta}(\eta_{n_\eta}) & \eta_{n_\eta} & 1 \end{bmatrix}^{-1} \begin{pmatrix} u_1^n \\ u_2^n \\ \vdots \\ u_{n_\eta}^n \end{pmatrix}, \tag{4.29}$$

where $u_i = u(\eta_i)$ $(i \in \{1,\ 2, \ldots,\ n_\eta\})$; and $\frac{d^2u_i}{d\eta^2} = \frac{d^2u}{d\eta^2}(\eta_i)$ $(i \in \{1,\ n_\eta\})$.

The IRBF system on a grid line for the second derivative of u is obtained by letting the interior node taking value from 2 to $(n_\eta - 1)$ in (4.28) and making use of (4.29)

$$\mathcal{L}_{\eta\eta}\widehat{u}_{\eta\eta}^n = \mathcal{B}_{\eta\eta}\widehat{u}^n, \tag{4.30}$$

where $\mathcal{L}_{\eta\eta}$, $\mathcal{B}_{\eta\eta}$ are $n_\eta \times n_\eta$ matrices.

CIRBF-2

First-order derivative compact approximations:

Unlike Scheme 1, nodal derivative values (i.e. extra information) used in the compact approximation of first derivatives are chosen here as $\frac{du_1}{d\eta}$ and $\frac{du_3}{d\eta}$. We construct the conversion system over a 3-point stencil associated with an interior node in the form of

$$\begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ \frac{du_1}{d\eta} \\ \frac{du_3}{d\eta} \end{pmatrix} = \underbrace{\begin{pmatrix} \overline{\mathcal{H}} \\ \mathcal{H} \end{pmatrix}}_{\mathcal{C}_2} \begin{pmatrix} w_1 \\ w_2 \\ w_3 \\ c_1 \\ c_2 \end{pmatrix}, \tag{4.31}$$

where $\frac{du_i}{d\eta} = \frac{du}{d\eta}(\eta_i)$ $(i \in \{1,\ 3\})$; $\mathcal{C}_2$ is the conversion matrix; $\overline{\mathcal{H}}$ is defined as (2.20); and

$$\mathcal{H} = \begin{bmatrix} H_1(\eta_1) & H_2(\eta_1) & H_3(\eta_1) & 1 & 0 \\ H_1(\eta_3) & H_2(\eta_3) & H_3(\eta_3) & 1 & 0 \end{bmatrix}. \tag{4.32}$$

Solving (4.31) yields

$$\begin{pmatrix} w_1 \\ w_2 \\ w_3 \\ c_1 \\ c_2 \end{pmatrix} = \mathcal{C}_2^{-1} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ \frac{du_1}{d\eta} \\ \frac{du_3}{d\eta} \end{pmatrix}, \tag{4.33}$$

which maps the vector of nodal values of the function and of its first-order derivative to the vector of RBF coefficients including the two integration constants. Approximate expressions for the first-order derivatives in the physical space are obtained by substituting (4.33) into (2.10)

$$\frac{du(\eta)}{d\eta} = \begin{bmatrix} H_1(\eta) & H_2(\eta) & H_3(\eta) & 1 & 0 \end{bmatrix} \mathcal{C}_2^{-1} \begin{pmatrix} \widehat{u} \\ \frac{\widehat{du}}{d\eta} \end{pmatrix}, \tag{4.34}$$

where $\eta_1 \leq \eta \leq \eta_3$; $\widehat{u} = (u_1, u_2, u_3)^T$ and $\frac{\widehat{du}}{d\eta} = (\frac{du_1}{d\eta}, \frac{du_3}{d\eta})^T$.

It can be rewritten as

$$\frac{du(\eta)}{d\eta} = \sum_{i=1}^{3} \frac{d\phi_i(\eta)}{d\eta} u_i + \frac{d\phi_4(\eta)}{d\eta}\frac{du_1}{d\eta} + \frac{d\phi_5(\eta)}{d\eta}\frac{du_3}{d\eta}, \tag{4.35}$$

where $\{\phi_i(\eta)\}_{i=1}^{5}$ is the set of IRBFs in the physical space.

At the current time level, equation (4.35) is taken as

$$\frac{du^n(\eta)}{d\eta} = \sum_{i=1}^{3} \frac{d\phi_i(\eta)}{d\eta} u_i^n + \frac{d\phi_4(\eta)}{d\eta}\frac{du_1^n}{d\eta} + \frac{d\phi_5(\eta)}{d\eta}\frac{du_3^n}{d\eta}, \tag{4.36}$$

where nodal values of the first-order derivatives on the right hand side are treated as unknowns.

Collocating (4.36) at $\eta = \eta_2$ results in

$$-\frac{d\phi_4(\eta_2)}{d\eta}\frac{du_1^n}{d\eta} + \frac{du_2^n}{d\eta} - \frac{d\phi_5(\eta_2)}{d\eta}\frac{du_3^n}{d\eta} = \frac{d\phi_1(\eta_2)}{d\eta}u_1^n + \frac{d\phi_2(\eta_2)}{d\eta}u_2^n + \frac{d\phi_3(\eta_2)}{d\eta}u_3^n, \tag{4.37}$$

or in matrix-vector form

$$\begin{bmatrix} -\frac{d\phi_4(\eta_2)}{d\eta} & 1 & -\frac{d\phi_5(\eta_2)}{d\eta} \end{bmatrix} \begin{bmatrix} \frac{du_1^n}{d\eta} \\ \frac{du_2^n}{d\eta} \\ \frac{du_3^n}{d\eta} \end{bmatrix} = \begin{bmatrix} \frac{d\phi_1(\eta_2)}{d\eta} & \frac{d\phi_2(\eta_2)}{d\eta} & \frac{d\phi_3(\eta_2)}{d\eta} \end{bmatrix} \begin{bmatrix} u_1^n \\ u_2^n \\ u_3^n \end{bmatrix}. \tag{4.38}$$

Figure 4.2: Special compact 4-point 1D-IRBF stencils for left and right boundary nodes.

At the boundary nodes, instead of using the global 1D-IRBF approach as in Scheme 1, we compute the first derivative here using special compact local stencils (Figure 4.2). These proposed stencils are constructed as follows. Consider the boundary node η_1. Its associated stencil is $[\eta_1, \eta_2, \eta_3, \eta_4]$. The conversion system over this stencil is presented as the following matrix-vector multiplication

$$\begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ \frac{du_2}{d\eta} \end{pmatrix} = \underbrace{\begin{pmatrix} \overline{\mathcal{H}_{sp}} \\ \mathcal{H}_{sp} \end{pmatrix}}_{\mathcal{C}_{sp_1}} \begin{pmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \\ c_1 \\ c_2 \end{pmatrix}, \tag{4.39}$$

where $\mathcal{C}_{sp_1}$ is the conversion matrix and $\overline{\mathcal{H}_{sp}}$, $\mathcal{H}_{sp}$ are matrices defined as

$$\overline{\mathcal{H}_{sp}} = \begin{bmatrix} \overline{H}_1(\eta_1) & \overline{H}_2(\eta_1) & \overline{H}_3(\eta_1) & \overline{H}_4(\eta_1) & \eta_1 & 1 \\ \overline{H}_1(\eta_2) & \overline{H}_2(\eta_2) & \overline{H}_3(\eta_2) & \overline{H}_4(\eta_2) & \eta_2 & 1 \\ \overline{H}_1(\eta_3) & \overline{H}_2(\eta_3) & \overline{H}_3(\eta_3) & \overline{H}_4(\eta_3) & \eta_3 & 1 \\ \overline{H}_1(\eta_4) & \overline{H}_2(\eta_4) & \overline{H}_3(\eta_4) & \overline{H}_4(\eta_4) & \eta_4 & 1 \end{bmatrix}, \tag{4.40}$$

$$\mathcal{H}_{sp} = \begin{bmatrix} H_1(\eta_2) & H_2(\eta_2) & H_3(\eta_2) & H_4(\eta_2) & 1 & 0 \end{bmatrix}. \tag{4.41}$$

Solving (4.39) yields

$$\begin{pmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \\ c_1 \\ c_2 \end{pmatrix} = \mathcal{C}_{sp_1}^{-1} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ \frac{du_2}{d\eta} \end{pmatrix}. \tag{4.42}$$

The boundary value of the first-order derivative of u is thus obtained by substituting (4.42) into (2.10) and taking $\eta = \eta_1$

$$\frac{du(\eta_1)}{d\eta} = \begin{bmatrix} H_1(\eta_1) & H_2(\eta_1) & H_3(\eta_1) & H_4(\eta_1) & 1 & 0 \end{bmatrix} \mathcal{C}_{sp_1}^{-1} \begin{pmatrix} u_1 & u_2 & u_3 & u_4 & \frac{du_2}{d\eta} \end{pmatrix}^T, \tag{4.43}$$

or

$$\frac{du_1^n}{d\eta} - \frac{d\phi_{sp_5}(\eta_1)}{d\eta}\frac{du_2^n}{d\eta} = \frac{d\phi_{sp_1}(\eta_1)}{d\eta}u_1^n + \frac{d\phi_{sp_2}(\eta_1)}{d\eta}u_2^n + \frac{d\phi_{sp_3}(\eta_1)}{d\eta}u_3^n + \frac{d\phi_{sp_4}(\eta_1)}{d\eta}u_4^n, \tag{4.44}$$

where $\{\phi_{sp_i}(\eta)\}_{i=1}^5$ is the set of IRBFs in the physical space. We rewrite equation (4.44) in matrix-vector form

$$\begin{bmatrix} 1 & -\frac{d\phi_{sp_5}(\eta_1)}{d\eta} & 0 & 0 \end{bmatrix} \begin{bmatrix} \frac{du_1^n}{d\eta} \\ \frac{du_2^n}{d\eta} \\ \frac{du_3^n}{d\eta} \\ \frac{du_4^n}{d\eta} \end{bmatrix} = \begin{bmatrix} \frac{d\phi_{sp_1}(\eta_1)}{d\eta} & \frac{d\phi_{sp_2}(\eta_1)}{d\eta} & \frac{d\phi_{sp_3}(\eta_1)}{d\eta} & \frac{d\phi_{sp_4}(\eta_1)}{d\eta} \end{bmatrix} \begin{bmatrix} u_1^n \\ u_2^n \\ u_3^n \\ u_4^n \end{bmatrix}. \tag{4.45}$$

In a similar manner, one is able to calculate the first derivative of u at the boundary node η_{n_η}.

The IRBF system on a grid line for the first derivative of u is obtained by letting the interior node taking value from 2 to $(n_\eta - 1)$ in (4.38) and making use of (4.45),

$$\mathcal{L}_\eta \widehat{u}_\eta^n = \mathcal{A}_\eta \widehat{u}^n. \tag{4.46}$$

Second-order derivative compact approximations:

Nodal derivative values (i.e. extra information) used in the compact approximation of second-order derivatives are chosen here as $\frac{d^2u_1}{d\eta^2}$ and $\frac{d^2u_3}{d\eta^2}$. The corresponding formulation is thus exactly the same as described in Scheme 1 (i.e. (4.28)). However, at the boundary nodes, instead of using the global 1D-IRBF approach as in Scheme 1, we compute the second-order derivative here using special compact local stencils (Figure 4.2). Consider the boundary node, e.g. η_1. The conversion system over the associated stencil is presented as the following matrix-vector multiplication

$$\begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ \frac{d^2u_2}{d\eta^2} \end{pmatrix} = \underbrace{\begin{pmatrix} \overline{\mathcal{H}_{sp}} \\ \mathcal{G}_{sp} \end{pmatrix}}_{\mathcal{C}_{sp_2}} \begin{pmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \\ c_1 \\ c_2 \end{pmatrix}, \tag{4.47}$$

where $\mathcal{C}_{sp_2}$ is the conversion matrix; $\overline{\mathcal{H}_{sp}}$ is defined as before; and

$$\mathcal{G}_{sp} = \begin{bmatrix} G_1(\eta_2) & G_2(\eta_2) & G_3(\eta_2) & G_4(\eta_2) & 0 & 0 \end{bmatrix}. \tag{4.48}$$

Solving (4.47) yields

$$\begin{pmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \\ c_1 \\ c_2 \end{pmatrix} = \mathcal{C}_{sp_2}^{-1} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ \frac{d^2u_2}{d\eta^2} \end{pmatrix}. \tag{4.49}$$

The boundary value of the second-order derivative of u is thus obtained by substituting (4.49) into (2.9) and taking $\eta = \eta_1$

$$\frac{d^2u(\eta_1)}{d\eta^2} = \begin{bmatrix} G_1(\eta_1) & G_2(\eta_1) & G_3(\eta_1) & G_4(\eta_1) & 0 & 0 \end{bmatrix} \mathcal{C}_{sp_2}^{-1} \begin{pmatrix} u_1 & u_2 & u_3 & u_4 & \frac{d^2u_2}{d\eta^2} \end{pmatrix}^T, \tag{4.50}$$

or

$$\frac{d^2u_1^n}{d\eta^2} - \frac{d^2\varphi_{sp_5}(\eta_1)}{d\eta^2}\frac{d^2u_2^n}{d\eta^2} = \frac{d^2\varphi_{sp_1}(\eta_1)}{d\eta^2}u_1^n + \frac{d^2\varphi_{sp_2}(\eta_1)}{d\eta^2}u_2^n + \frac{d^2\varphi_{sp_3}(\eta_1)}{d\eta^2}u_3^n + \frac{d^2\varphi_{sp_4}(\eta_1)}{d\eta^2}u_4^n, \quad (4.51)$$

where $\{\varphi_{sp_i}(\eta)\}_{i=1}^5$ is the set of IRBFs in the physical space. We rewrite equation (4.51) in matrix-vector form

$$\begin{bmatrix} 1 & -\frac{d^2\varphi_{sp_5}(\eta_1)}{d\eta^2} & 0 & 0 \end{bmatrix} \begin{bmatrix} \frac{d^2u_1^n}{d\eta^2} \\ \frac{d^2u_2^n}{d\eta^2} \\ \frac{d^2u_3^n}{d\eta^2} \\ \frac{d^2u_4^n}{d\eta^2} \end{bmatrix} = \begin{bmatrix} \frac{d^2\varphi_{sp_1}(\eta_1)}{d\eta^2} & \frac{d^2\varphi_{sp_2}(\eta_1)}{d\eta^2} & \frac{d^2\varphi_{sp_3}(\eta_1)}{d\eta^2} & \frac{d^2\varphi_{sp_4}(\eta_1)}{d\eta^2} \end{bmatrix} \begin{bmatrix} u_1^n \\ u_2^n \\ u_3^n \\ u_4^n \end{bmatrix}. \quad (4.52)$$

The IRBF system on a grid line for the second derivative of u is obtained by letting the interior node taking value from 2 to $(n_\eta - 1)$ in (4.28) and making use of (4.52),

$$\mathcal{L}_{\eta\eta}\widehat{u}_{\eta\eta}^n = \mathcal{B}_{\eta\eta}\widehat{u}^n, \quad (4.53)$$

where $\mathcal{L}_{\eta\eta}$, $\mathcal{B}_{\eta\eta}$ are $n_\eta \times n_\eta$ matrices.

It is noted that, for brevity, we use the same notations to represent the RBF coefficients and the coefficient matrices for the two schemes and also for the approximation of first and second derivatives in Scheme 2. In fact, for example, the entries of $\mathcal{L}_\eta$, $\mathcal{L}_{\eta\eta}$, $\mathcal{A}_\eta$ and $\mathcal{B}_{\eta\eta}$ in (4.46) and (4.53) are different from those of $\mathcal{L}_\eta$, $\mathcal{L}_{\eta\eta}$, $\mathcal{A}_\eta$ and $\mathcal{B}_{\eta\eta}$ in (4.23) and (4.30); and the coefficient set (w_1, w_2, w_3, w_4, c_1, c_2) in (4.39) is not the same as that in (4.47).

4.3.2 Temporal discretisation

The temporal discretisation of (4.1) with a Crank-Nicolson scheme (Crank and Nicolson, 1996) gives

$$u^n + \frac{\Delta t}{2}c_x\frac{\partial u^n}{\partial x} - \frac{\Delta t}{2}d_x\frac{\partial^2 u^n}{\partial x^2} + \frac{\Delta t}{2}c_y\frac{\partial u^n}{\partial y} - \frac{\Delta t}{2}d_y\frac{\partial^2 u^n}{\partial y^2} = u^{n-1} - \frac{\Delta t}{2}c_x\frac{\partial u^{n-1}}{\partial x} + \frac{\Delta t}{2}d_x\frac{\partial^2 u^{n-1}}{\partial x^2} - \frac{\Delta t}{2}c_y\frac{\partial u^{n-1}}{\partial y} + \frac{\Delta t}{2}d_y\frac{\partial^2 u^{n-1}}{\partial y^2} + \Delta t f_b^{n-1/2} + \mathcal{O}(\Delta t^2). \quad (4.54)$$

We apply the ADI factorisation to (4.54), resulting in

$$\left\{1 + \frac{\Delta t}{2}c_x\frac{\partial}{\partial x} - \frac{\Delta t}{2}d_x\frac{\partial^2}{\partial x^2}\right\}\left\{1 + \frac{\Delta t}{2}c_y\frac{\partial}{\partial y} - \frac{\Delta t}{2}d_y\frac{\partial^2}{\partial y^2}\right\}u^n = \left\{1 - \frac{\Delta t}{2}c_x\frac{\partial}{\partial x} + \frac{\Delta t}{2}d_x\frac{\partial^2}{\partial x^2}\right\}\left\{1 - \frac{\Delta t}{2}c_y\frac{\partial}{\partial y} + \frac{\Delta t}{2}d_y\frac{\partial^2}{\partial y^2}\right\}u^{n-1} + \Delta t f_b^{n-1/2} + \mathcal{O}(\Delta t^2). \quad (4.55)$$

Equation (4.55) can be further factorised as

$$\left(1 + \frac{\Delta t}{2}c_x\frac{\partial}{\partial x}\right)\left(1 - \frac{\Delta t}{2}d_x\frac{\partial^2}{\partial x^2}\right)\left(1 + \frac{\Delta t}{2}c_y\frac{\partial}{\partial y}\right)\left(1 - \frac{\Delta t}{2}d_y\frac{\partial^2}{\partial y^2}\right)u^n = \left(1 - \frac{\Delta t}{2}c_x\frac{\partial}{\partial x}\right)\left(1 + \frac{\Delta t}{2}d_x\frac{\partial^2}{\partial x^2}\right)\left(1 - \frac{\Delta t}{2}c_y\frac{\partial}{\partial y}\right)\left(1 + \frac{\Delta t}{2}d_y\frac{\partial^2}{\partial y^2}\right)u^{n-1} + \Delta t f_b^{n-1/2} + \mathcal{O}(\Delta t^2). \quad (4.56)$$

It is noted that equations (4.55) and (4.56) have the same order accuracy in time (i.e. second order) as equation (4.54).

4.3.3 Spatial - temporal discretisation

Incorporation of the CIRBF approximations derived in section 4.3.1 (i.e. CIRBF-1) and 4.3.1 (i.e. CIRBF-2) into the ADI equation (4.56) leads to, respectively, the following two schemes.

ADI-CIRBF-1

From (4.23) and (4.30), nodal values of the first- and second-order derivatives of u can be derived in terms of nodal variable values

$$\widehat{u}_\eta = \mathcal{L}_\eta^{-1}\left(\mathcal{A}_\eta + \mathcal{A}_{\eta\eta}\mathcal{L}_{\eta\eta}^{-1}\mathcal{B}_{\eta\eta}\right)\widehat{u}, \tag{4.57}$$

$$\widehat{u}_{\eta\eta} = \mathcal{L}_{\eta\eta}^{-1}\mathcal{B}_{\eta\eta}\widehat{u}. \tag{4.58}$$

Substituting (4.57) and (4.58) into (4.56) results in

$$\mathcal{L}_x^{-1}\mathcal{T}_x^{+}\mathcal{L}_{xx}^{-1}\mathcal{T}_{xx}^{-}\mathcal{L}_y^{-1}\mathcal{T}_y^{+}\mathcal{L}_{yy}^{-1}\mathcal{T}_{yy}^{-}\widehat{u}^n = \mathcal{L}_x^{-1}\mathcal{T}_x^{-}\mathcal{L}_{xx}^{-1}\mathcal{T}_{xx}^{+}\mathcal{L}_y^{-1}\mathcal{T}_y^{-}\mathcal{L}_{yy}^{-1}\mathcal{T}_{yy}^{+}\widehat{u}^{n-1} + \Delta t\widehat{f}_b^{n-1/2}, \tag{4.59}$$

where

$$\mathcal{T}_x^{\pm} = \left(\mathcal{L}_x \pm \frac{\Delta t}{2}c_x\left\{\mathcal{A}_x + \mathcal{A}_{xx}\mathcal{L}_{xx}^{-1}\mathcal{B}_{xx}\right\}\right), \quad \mathcal{T}_{xx}^{\pm} = \left(\mathcal{L}_{xx} \pm \tfrac{\Delta t}{2}d_x\mathcal{B}_{xx}\right), \tag{4.60}$$
$$\mathcal{T}_y^{\pm} = \left(\mathcal{L}_y \pm \frac{\Delta t}{2}c_y\left\{\mathcal{A}_y + \mathcal{A}_{yy}\mathcal{L}_{yy}^{-1}\mathcal{B}_{yy}\right\}\right), \quad \mathcal{T}_{yy}^{\pm} = \left(\mathcal{L}_{yy} \pm \tfrac{\Delta t}{2}d_y\mathcal{B}_{yy}\right).$$

ADI-CIRBF-2

From (4.46) and (4.53), nodal values of the first- and second-order derivatives of u can be derived in terms of nodal variable values

$$\widehat{u}_\eta = \mathcal{L}_\eta^{-1}\mathcal{A}_\eta\widehat{u}, \tag{4.61}$$

$$\widehat{u}_{\eta\eta} = \mathcal{L}_{\eta\eta}^{-1}\mathcal{B}_{\eta\eta}\widehat{u}. \tag{4.62}$$

Substituting (4.61) and (4.62) into (4.56) results in

$$\mathcal{L}_x^{-1}\mathcal{T}_x^{+}\mathcal{L}_{xx}^{-1}\mathcal{T}_{xx}^{-}\mathcal{L}_y^{-1}\mathcal{T}_y^{+}\mathcal{L}_{yy}^{-1}\mathcal{T}_{yy}^{-}\widehat{u}^n = \mathcal{L}_x^{-1}\mathcal{T}_x^{-}\mathcal{L}_{xx}^{-1}\mathcal{T}_{xx}^{+}\mathcal{L}_y^{-1}\mathcal{T}_y^{-}\mathcal{L}_{yy}^{-1}\mathcal{T}_{yy}^{+}\widehat{u}^{n-1} + \Delta t\widehat{f}_b^{n-1/2}, \tag{4.63}$$

where

$$\mathcal{T}_x^{\pm} = \left(\mathcal{L}_x \pm \frac{\Delta t}{2}c_x\mathcal{A}_x\right), \quad \mathcal{T}_{xx}^{\pm} = \left(\mathcal{L}_{xx} \pm \tfrac{\Delta t}{2}d_x\mathcal{B}_{xx}\right), \tag{4.64}$$
$$\mathcal{T}_y^{\pm} = \left(\mathcal{L}_y \pm \frac{\Delta t}{2}c_y\mathcal{A}_y\right), \quad \mathcal{T}_{yy}^{\pm} = \left(\mathcal{L}_{yy} \pm \tfrac{\Delta t}{2}d_y\mathcal{B}_{yy}\right).$$

Calculation procedure

Equation (4.59)/(4.63) is equivalent to

$$\mathcal{L}_x^{-1}\mathcal{T}_x^{+}\mathcal{L}_{xx}^{-1}\mathcal{T}_{xx}^{-}\widehat{u}^{*} = \mathcal{L}_x^{-1}\mathcal{T}_x^{-}\mathcal{L}_{xx}^{-1}\mathcal{T}_{xx}^{+}\mathcal{L}_y^{-1}\mathcal{T}_y^{-}\mathcal{L}_{yy}^{-1}\mathcal{T}_{yy}^{+}\widehat{u}^{n-1} + \Delta t\widehat{f}_b^{n-1/2}, \quad (4.65)$$

$$\mathcal{L}_y^{-1}\mathcal{T}_y^{+}\mathcal{L}_{yy}^{-1}\mathcal{T}_{yy}^{-}\widehat{u}^{n} = \widehat{u}^{*}, \quad (4.66)$$

which can be solved by the following two steps.

Step 1: This step involves two substeps

- Substep 1: Compute the nodal values of u^* at the left and right boundaries of the computational domain via equation (4.66) for $x = x_1$ and $x = x_{n_x}$ with the given boundary condition (4.3).
- Substep 2: Solve (4.65) on the x-grid lines ($y = y_j,\ j \in \{2,\ 3, \ldots,\ n_y - 1\}$) for the values of u^* at the interior nodes.

Step 2: Solve (4.66) on the y-grid lines ($x = x_i,\ i \in \{2,\ 3, \ldots,\ n_x - 1\}$) for the values of u^n at the interior nodes.

Owing to the ADI technique, the computational costs for two present solution procedures are low. Scheme 2 (i.e. ADI-CIRBF-2) is more efficient as only local stencils are involved.

4.4 Numerical examples

For global stencils, value of $\beta = 1$ is usually picked whereas value of $\beta = 50$ is simply picked for local stencils in Scheme 1 and Scheme 2 in this chapter. We assess the performance of the proposed scheme through the following measures:

- the average absolute error (L_1) defined as

$$L_1 = \frac{1}{N}\sum_{i=1}^{N}|u_i - \overline{u}_i|, \quad (4.67)$$

- the RMS error defined as (2.48),
- the maximum absolute error (L_∞) defined as (3.53),

- the average rate of convergence α determined by the method of least squares as described in chapter 2. The local rate of convergence is defined as (2.49).

A flow is considered to reach its steady state when the criterion (2.50) is satisfied.

4.4.1 Unsteady diffusion equation

Consider a diffusion equation by setting the parameters in equation (4.1) as $c_x = c_y = 0$, $d_x = d_y = 1$ and $f_b = 0$. The analytic solution is taken here as (Tian and Ge, 2007)

$$\overline{u}(x, y, t) = e^{-2\pi^2 t} \sin(\pi x) \sin(\pi y). \tag{4.68}$$

The problem domain is chosen to be a unit square $\Omega = [0, 1] \times [0, 1]$ and the initial and Dirichlet boundary conditions are derived from (4.68).

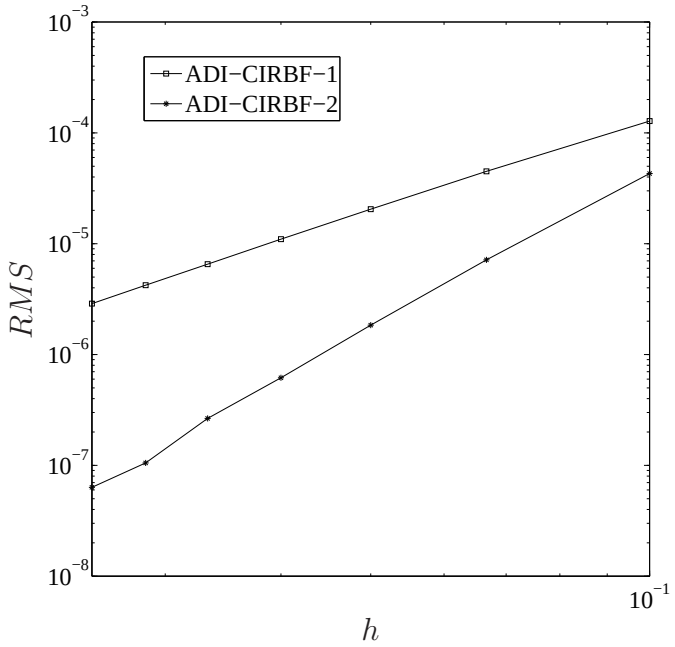


Figure 4.3: Unsteady diffusion equation, $\{11 \times 11,\ 16 \times 16, \ldots,\ 41 \times 41\}$, $\Delta t = 10^{-5}$, $t = 0.0125$: The effect of grid size h on the solution accuracy for the two present schemes. The solution converges as $O(h^{2.74})$ for ADI-CIRBF-1 and $O(h^{4.76})$ for ADI-CIRBF-2.

We employ a set of uniform grids to study the convergence of the solution with grid refinement. Results obtained at $t = 0.0125$ using $\Delta t = 10^{-5}$ and $\{11 \times 11,\ 16 \times 16, \ldots,\ 41 \times 41\}$ are displayed in Figure 4.3, showing that the

approximate solution converges apparently as $O(h^{2.74})$ for ADI-CIRBF-1, and $O(h^{4.76})$ for ADI-CIRBF-2.

We employ a set of time steps, namely $\Delta t \in \{0.05,\ 0.025,\ 0.0125,\ 0.00625\}$ to test the temporal accuracy. Results obtained at $t = 1.25$ using a uniform grid of 81×81 are shown in Table 4.1. Two present schemes are about second-order accurate in time as expected (temporal derivative terms are presently discretised with a second-order Crank-Nicolson scheme). It is noted that we employ a fine grid of 81×81 to ensure that the approximate error in space is small enough so that its effects can be neglected.

Table 4.1: Unsteady diffusion equation, $t = 1.25$, 81×81: Solution accuracy of the two present schemes against time step.

	ADI-CIRBF-1		ADI-CIRBF-2	
Δt	RMS	Local Rate	RMS	Local Rate
0.05	3.8700E-12	—-	3.8518E-12	—-
0.025	1.1432E-12	1.76	1.1276E-12	1.77
0.0125	2.9542E-13	1.95	2.9337E-13	1.94
0.00625	7.3686E-14	2.00	7.4054E-14	1.99

To facilitate a comparison with the exponential high-order compact ADI scheme (EHOC-ADI) (Tian and Ge, 2007), we now choose $\Delta t = h^2$ and $t = 0.125$. Table 4.2 indicates that the present ADI-CIRBF-2 scheme and the EHOC-ADI scheme yield similar local rates of convergence of about 4.

Table 4.2: Unsteady diffusion equation, $t = 0.125$, $\Delta t = h^2$: Effect of grid size on the solution accuracy.

	EHOC-ADI		ADI-CIRBF-2	
Grid ($n_x \times n_y$)	RMS	Local Rate	RMS	Local Rate
11×11	8.55134E-05	—-	9.39417E-05	—-
21×21	5.19160E-06	4.041	5.81951E-06	4.013
41×41	3.17475E-07	4.031	4.02907E-07	3.852

Figure 4.4 plots the RMS error against time with $\Delta t = 10^{-4}$. It can be seen that ADI-CIRBF-2 is the most accurate scheme, followed by ADI-CIRBF-1 and then PR-ADI. It also shows that the present results using a grid of 21×21 are already more accurate than the standard ADI results using a grid of 41×41.

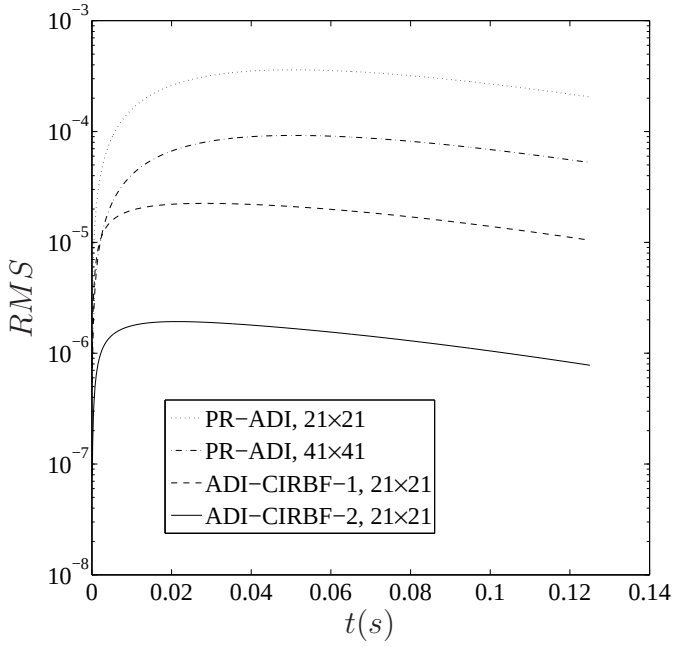


Figure 4.4: Unsteady diffusion equation, $\Delta t = 10^{-4}$: The solution accuracy of the standard PR-ADI and the two present schemes against time.

4.4.2 Unsteady convection-diffusion equation

Consider the unsteady convection-diffusion equation (4.1), where $f_b = 0$, in a square $\Omega = [0, 2] \times [0, 2]$ with the following analytic solution (Noye and Tan, 1989)

$$\overline{u}(x, y, t) = \frac{1}{4t + 1} \exp\left[-\frac{(x - c_x t - 0.5)^2}{d_x(4t + 1)} - \frac{(y - c_y t - 0.5)^2}{d_y(4t + 1)}\right], \qquad (4.69)$$

and subject to Dirichlet boundary conditions. From (4.69), one can derive the initial and boundary conditions.

Figure 4.5 shows the accuracy of the spatial discretisation of the two present schemes. The calculations are carried out on a set of uniform grids, namely $\{31 \times 31,\ 41 \times 41, \ldots,\ 81 \times 81\}$ and a time step of 10^{-4} with the following parameters: $c_x = c_y = 0.8$, $d_x = d_y = 0.01$. The accuracy of the solution is measured at $t = 1.25$. It is noted that the time step is chosen small enough to minimise the effect of the approximate error in time. It can be seen that the solution converges very fast with grid refinement: $O(h^{4.07})$ for ADI-CIRBF-1 and $O(h^{4.32})$ for ADI-CIRBF-2. Figure 4.6 shows the initial pulse and the pulse at $t = 1.25$ using a grid of 81×81 and $\Delta t = 0.00625$ by the two present schemes. The initial pulse is a Gaussian pulse centred at $(0.5, 0.5)$ with the pulse height 1. At $t = 1.25$, the pulse moves to a position centred at $(1.5, 1.5)$ with the pulse

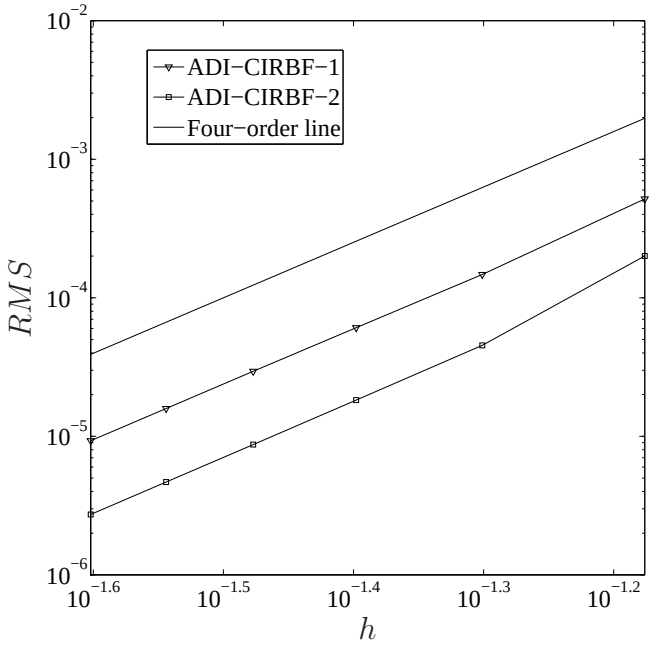
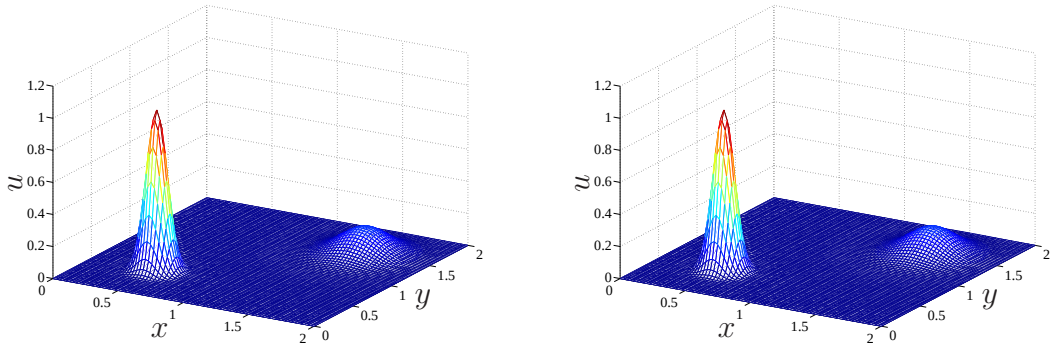


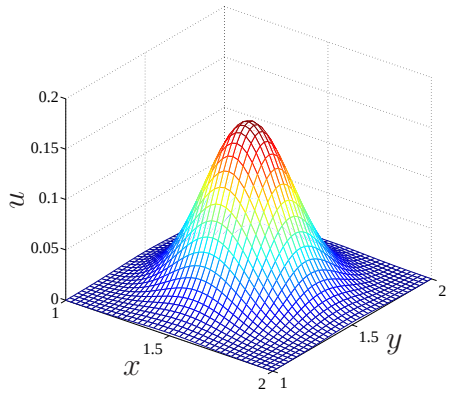
Figure 4.5: Unsteady convection-diffusion equation, $\{31\times31,\ 41\times41,\ldots,\ 81\times81\}$, $\Delta t = 10^{-4}$, $t = 1.25$: The effect of grid size h on the solution accuracy for the two present schemes. The solution converges as $O(h^{4.07})$ for ADI-CIRBF-1 and $O(h^{4.32})$ for ADI-CIRBF-2.



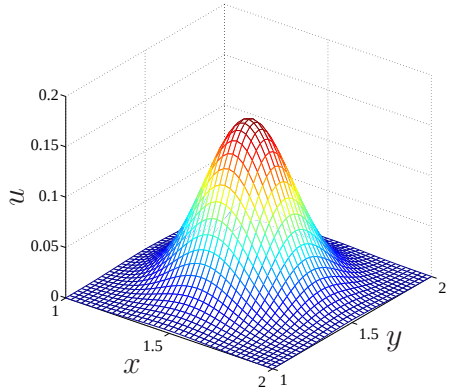
(a) (b)

Figure 4.6: Unsteady convection-diffusion equation, 81×81, $\Delta t = 0.00625$: The initial and the computed pulses at $t = 1.25$ by ADI-CIRBF-1 (a) and ADI-CIRBF-2 (b).

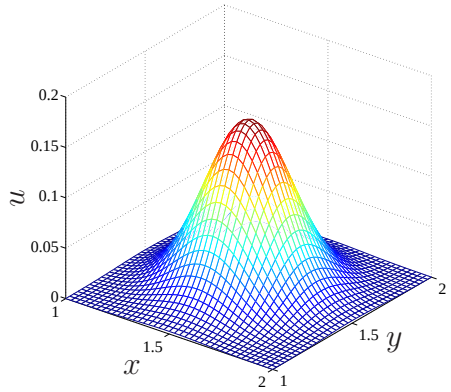
height of 1/6. Figure 4.7 displays the surface plots of the solution obtained by the analytic solution and the two present schemes in a sub-region $1 \le x, y \le 2$ - these plots are almost identical.



(a)



(b)



(c)

Figure 4.7: Unsteady convection-diffusion equation, 81×81, $\Delta t = 0.00625$: Surface plots of the pulse in the sub-region $1 \leq x, y \leq 2$ at $t = 1.25$ by the analytic solution (a), ADI-CIRBF-1 (b) and ADI-CIRBF-2 (c).

Table 4.3 shows a comparison of L_1, RMS and L_∞ errors between the two present schemes and the standard PR-ADI scheme, third-order nine-point compact scheme (Noye and Tan, 1989), fourth-order nine-point compact scheme (Kalita et al., 2002), HOC-ADI scheme (Karaa and Zhang, 2004) and exponential high-order compact ADI (EHOC-ADI) (Tian and Ge, 2007). It can be seen

Table 4.3: Unsteady convection-diffusion equation, 81×81, $t = 1.25$, $\Delta t = 0.00625$: Comparison of the solution accuracy between the present schemes and some other techniques.

Method	$L_1(u)$	$RMS(u)$	$L_\infty(u)$
PR-ADI (Peaceman and H. H. Rachford, 1955)	3.109E-04	2.025E-03	7.778E-03
3^{rd}-order 9-point compact (Noye and Tan, 1989)	1.971E-05	1.280E-04	6.509E-04
4^{th}-order 9-point compact (Kalita et al., 2002)	1.597E-05	1.024E-04	4.477E-04
HOC-ADI (Karaa and Zhang, 2004)	9.218E-06	5.931E-05	2.500E-04
EHOC-ADI (Tian and Ge, 2007)	9.663E-06	6.194E-05	2.664E-04
ADI-CIRBF-1	8.457E-06	2.808E-05	2.250E-04
ADI-CIRBF-2	6.742E-06	2.197E-05	1.703E-04

Table 4.4: Unsteady convection-diffusion equation, 81×81, $t = 1.25$, $\Delta t = 2.5E-4$: Comparison of the solution accuracy between the present schemes and some other techniques for case I.

Method	$RMS(u)$	$L_\infty(u)$
PR-ADI (Peaceman and H. H. Rachford, 1955)	1.11E-03	8.92E-03
HOC-ADI (Karaa and Zhang, 2004)	2.73E-05	2.46E-04
PDE-ADI (You, 2006)	2.20E-05	1.71E-04
HPD-ADI (Ma et al., 2012)	6.38E-05	6.54E-04
ADI-CIRBF-1	9.32E-06	7.80E-05
ADI-CIRBF-2	2.75E-06	2.37E-05

Table 4.5: Unsteady convection-diffusion equation, 81×81, $t = 0.0125$, $\Delta t = 2.5E-6$: Comparison of the solution accuracy between the present schemes and some other techniques for case II.

Method	$RMS(u)$	$L_\infty(u)$
PR-ADI (Peaceman and H. H. Rachford, 1955)	2.69E-02	3.74E-01
HOC-ADI (Karaa and Zhang, 2004)	1.47E-02	2.42E-01
PDE-ADI (You, 2006)	5.49E-04	1.22E-02
HPD-ADI (Ma et al., 2012)	5.49E-04	1.24E-02
ADI-CIRBF-1	1.71E-03	3.32E-02
ADI-CIRBF-2	5.45E-04	1.06E-02

that the present schemes yield solutions with higher accuracy. In addition, in Figure 4.8, we plot RMS against time for these schemes, except for EHOC-ADI (the data for this scheme is not available). It shows that all of these curves have similar shapes and the present schemes have smaller error for every time step. Figure 4.9 displays contour plots of the pulse in the region $1 \leq x, y \leq 2$ by the analytic solution, PR-ADI, ADI-CIRBF-1 and ADI-CIRBF-2. Contour plots of other mentioned schemes can be found in Tian and Ge (2007). It can be seen that the computed pulses by the two present schemes are visually indistinguishable from the analytic one, while PR-ADI produces a pulse that is distorted in both x- and y-directions. For the latter, the reason was explained in Noye and

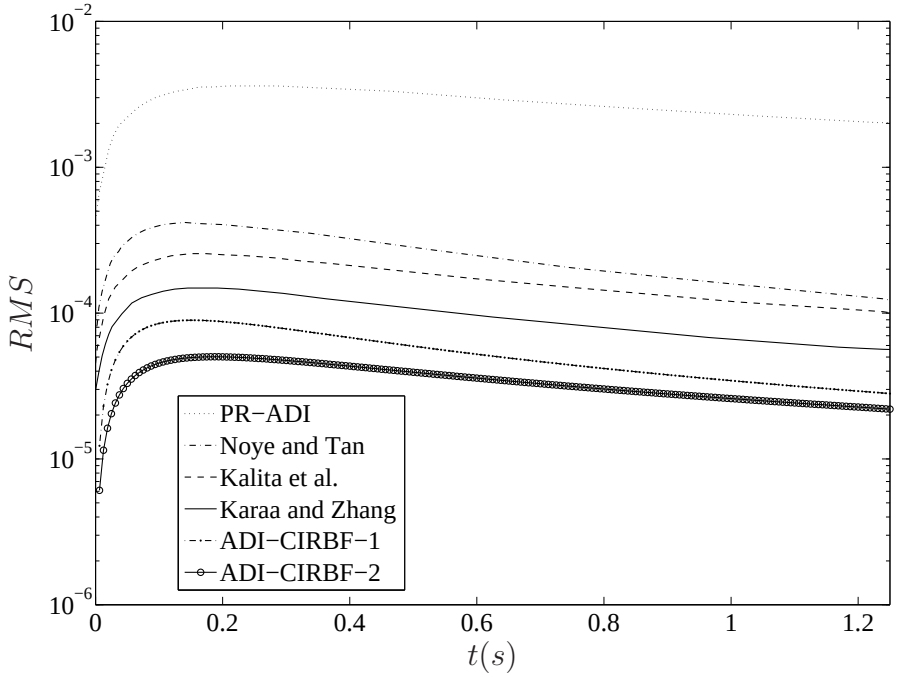


Figure 4.8: Unsteady convection-diffusion equation, 81×81, $\Delta t = 0.00625$: The solution accuracy of the present schemes and some other techniques against time.

Tan (1989) (the second-order error terms of the standard PR-ADI scheme are related to the wave numbers in both directions).

Recently, Ma et al. (2012) proposed a high-order hybrid Padé ADI (HPD-ADI) method for convection-dominated diffusion problems and also examined the performance of their method via this example. We consider two sets of parameters used in their article.

Case I: $c_x = c_y = 0.8$, $d_x = d_y = 0.01$, $h = 0.025$, $t = 1.25$, $\Delta t = 2.5E - 4$.

Case II: $c_x = c_y = 80$, $d_x = d_y = 0.01$, $h = 0.025$, $t = 0.0125$, $\Delta t = 2.5E - 6$.

The corresponding Peclet number is thus $Pe = 2$ for Case I and $Pe = 200$ for Case II. Results concerning RMS and L_∞ errors are presented in Tables 4.4-4.5. In the case of low P_e, the two present schemes are superior to HPD-ADI and also other schemes (Table 4.4). In the case of high P_e (i.e. convection dominated), ADI-CIRBF-2 yields the best performance: higher degrees of accuracy (Table 4.5) and higher rates of convergence (Table 4.6).

Table 4.6: Unsteady convection-diffusion equation, $t = 0.0125$, $\Delta t = 2.5E-6$: The solution accuracy of the present schemes and some other techniques against grid size for case II. LCR stands for "local convergence rate".

	PDE-ADI		HPD-ADI		ADI-CIRBF-1		ADI-CIRBF-2	
Grid ($n_x \times n_y$)	$RMS(n_x)$	LCR	$RMS(n_x)$	LCR	$RMS(n_x)$	LCR	$RMS(n_x)$	LCR
31×31	1.93E-02	—-	1.91E-02	—-	3.22E-02	—-	2.42E-02	—-
41×41	8.41E-03	2.98	8.30E-03	2.97	1.69E-02	2.29	8.45E-03	3.45
51×51	3.74E-03	3.30	3.70E-03	3.29	9.14E-03	2.52	3.74E-03	3.58
61×61	1.80E-03	3.51	1.78E-03	3.50	5.00E-03	2.75	1.79E-03	3.71
71×71	9.51E-04	3.63	9.48E-04	3.62	2.85E-03	2.92	9.47E-04	3.80
81×81	5.49E-04	3.69	5.49E-04	3.69	1.71E-03	3.05	5.45E-04	3.86
101×101	2.21E-04	3.78	2.23E-04	3.76	7.14E-04	3.22	2.18E-04	3.91
121×121	1.07E-04	3.81	1.10E-04	3.79	3.46E-04	3.33	1.04E-04	3.94

LCR=-log[RMS(n_x)/RMS(31)]/log[n_x/31].

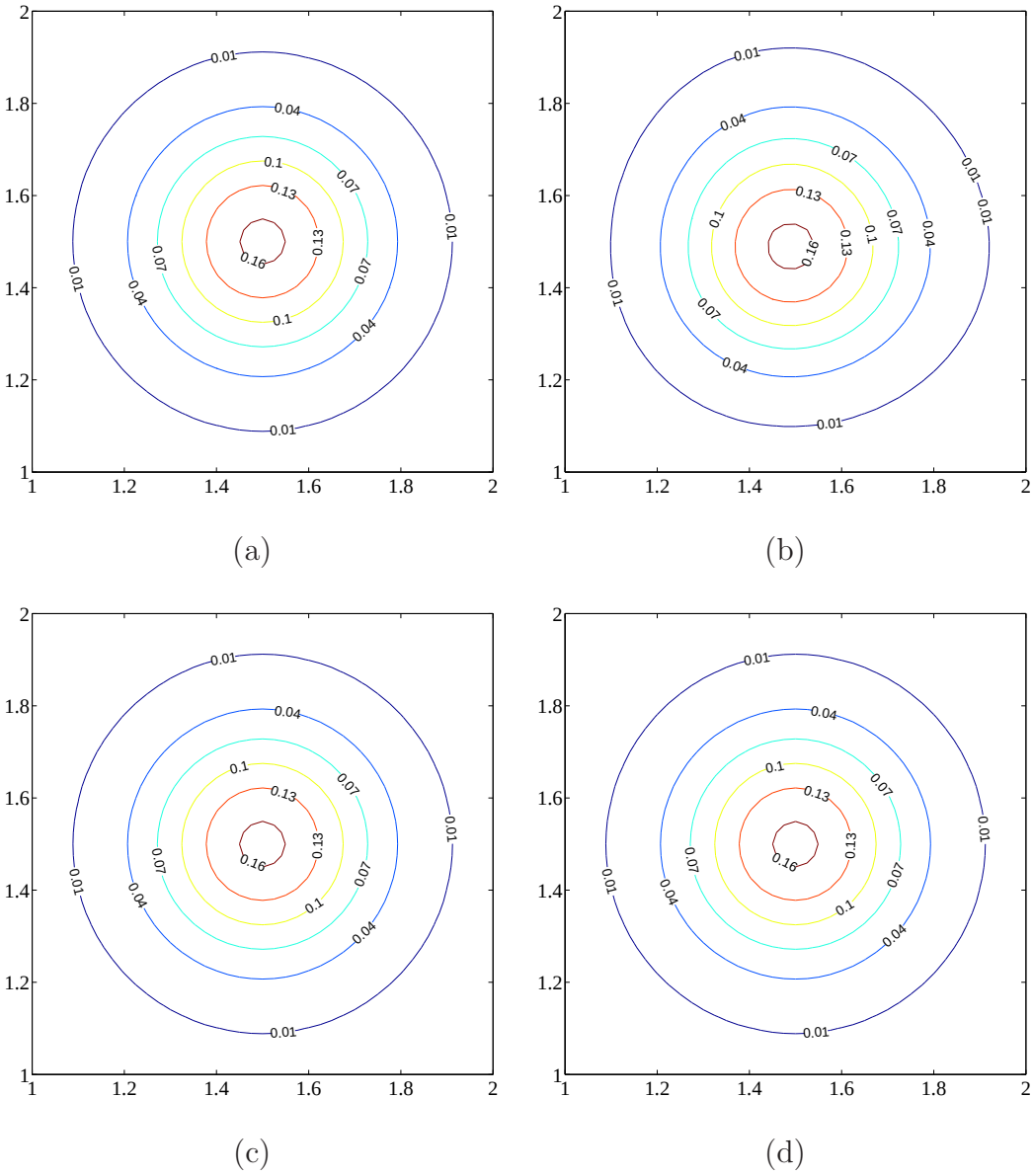


Figure 4.9: Unsteady convection-diffusion equation, 81×81, $\Delta t = 0.00625$: Contour plots of the pulse in the sub-region $1 \leq x, y \leq 2$ at $t = 1.25$ by the analytic solution (a), standard PR-ADI (b), ADI-CIRBF-1 (c) and ADI-CIRBF-2 (d).

4.4.3 Steady convection-diffusion equation

Consider equation (4.4) with $c_x = c_y = 0.1$, $d_x = d_y = 1$ in a square $\Omega = [0, L] \times [0, L]$ and subject to Dirichlet boundary condition. The analytic solution

takes the form (Sheu et al., 2011)

$$\overline{u} = \frac{u_0}{e^{r_+} - e^{r_-}} e^{\delta_x/2} \sin(\pi x) \left(e^{r+y} - e^{r-y}\right), \tag{4.70}$$

where $u_0 = 1$, $\delta_x = c_x L/d_x$, $\delta_y = c_y L/d_y$, $L = 1$, and

$$r_\pm = \frac{1}{2}\delta_y \pm \frac{1}{2}\sqrt{\left(\delta_y^2 + 4W\right)}, \quad W = 4\pi^2 + \delta_x^2/4. \tag{4.71}$$

The driving function f_b is given by

$$f_b = c_x \frac{\partial \overline{u}}{\partial x} + c_y \frac{\partial \overline{u}}{\partial y} - d_x \frac{\partial^2 \overline{u}}{\partial x^2} - d_y \frac{\partial^2 \overline{u}}{\partial y^2}. \tag{4.72}$$

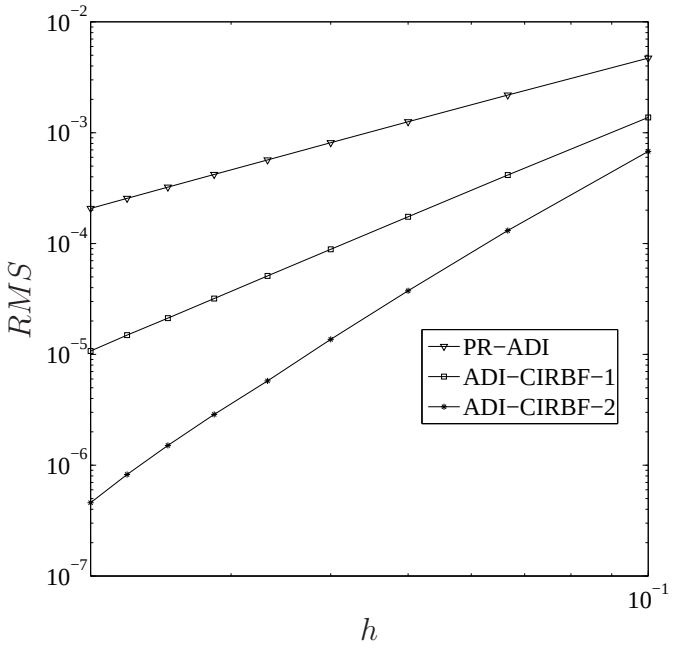


Figure 4.10: Steady convection-diffusion equation, $\{11 \times 11,\ 16 \times 16, \ldots,\ 51 \times 51\}$: The effect of grid size h on the solution accuracy for the standard PR-ADI and two present schemes. The solution converges as $O(h^{1.94})$, $O(h^{3.02})$ and $O(h^{4.53})$ for PR-ADI, ADI-CIRBF-1 and ADI-CIRBF-2, respectively.

To solve the steady equation (4.4), a pseudo time-derivative term $\frac{\partial u}{\partial t}$ is added to its left side to facilitate an iterative calculation. The steady equation (4.4) thus has the same form as the unsteady equation (4.1). When the difference of u between two successive time levels is small, i.e. less than a given tolerance, the obtained solution is the solution to (4.4).

In the present calculation, we employ a set of uniform grids, namely $\{11 \times$

11, $16 \times 16, \ldots, 51 \times 51\}$ and a time step of 0.0005. Figure 4.10 displays the solution accuracy against the grid size, which shows the superiority of the two present schemes over the standard PR-ADI scheme. The solution converges apparently as $O(h^{1.94})$, $O(h^{3.02})$ and $O(h^{4.53})$ for PR-ADI, ADI-CIRBF-1 and ADI-CIRBF-2, respectively. Additionally, Figure 4.11 show that profiles of u along the centrelines by ADI-CIRBF-1 and ADI-CIRBF-2 agree very well with the analytic solutions.

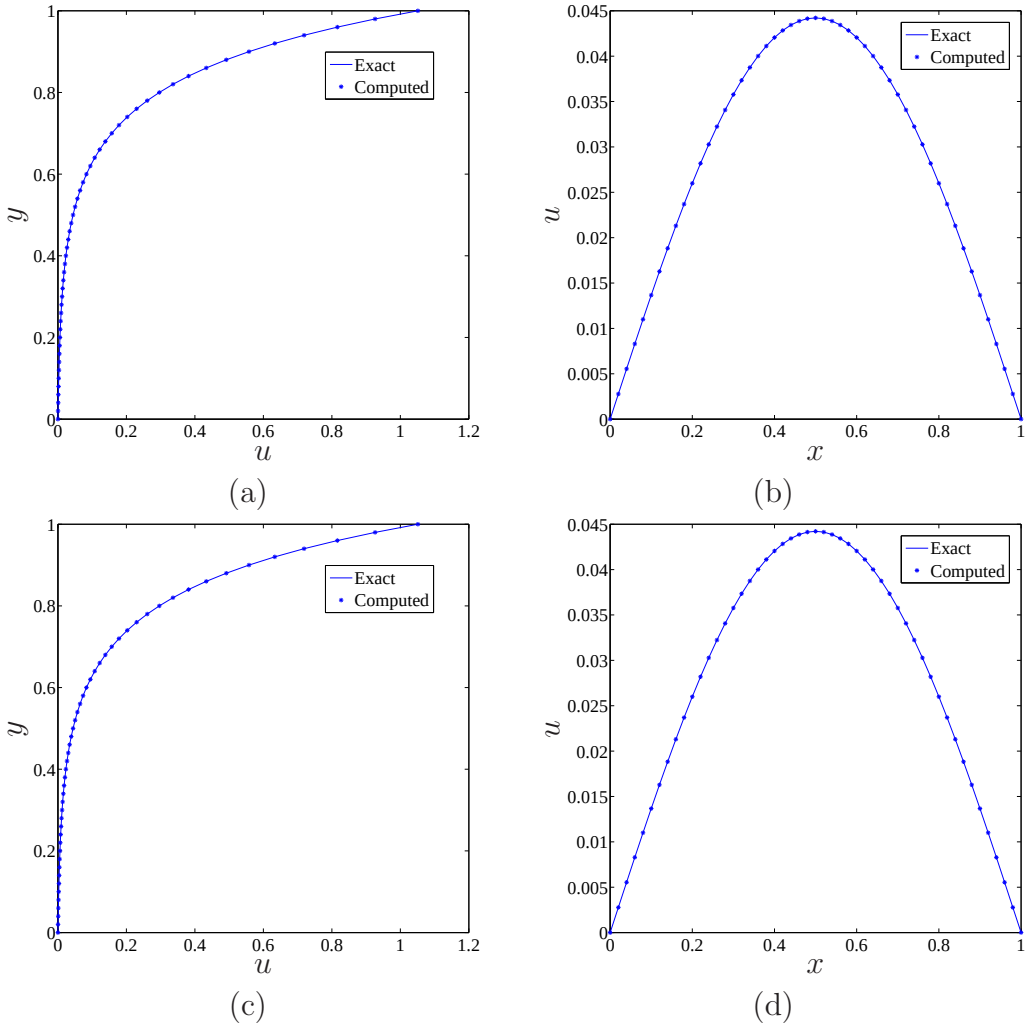


Figure 4.11: Steady convection-diffusion equation, 51×51: Profiles of the solution u along the vertical and horizontal centrelines by ADI-CIRBF-1 (a)-(b) and ADI-CIRBF-2 (c)-(d).

4.5 Concluding remarks

This chapter presents new high-order approximation schemes for the discretisation of convection-diffusion equations in two dimensions. The ADI algorithm is adopted in the form in which the operator is factorised into four separate terms rather than the usual two, while compact integrated-RBFs are implemented to represent the variable and its derivatives over 3-point stencils. Two CIRBF schemes are proposed, which lead to a significant improvement in accuracy over the central-finite-difference-based ADI method. CIRBF-2 scheme, where first-order and second-order derivatives are approximated separately, is found to be more efficient and effective than CIRBF-1 scheme, where first-order and second-order derivatives are approximated simultaneously. We note that the assumption of rectangular domain here is not necessary and the present methods can easily treat non-rectangular domains.

Chapter 5

Incorporation of direct forcing immersed boundary (DFIB) method into compact IRBF scheme

In this chapter, we present a numerical scheme, based on the DFIB method and CIRBF approximations, for solving the Navier-Stokes equations in two dimensions. The problem domain of complicated shape is embedded in a Cartesian grid containing Eulerian nodes. Non-slip conditions on the inner boundaries, represented by Lagrangian nodes, are imposed by means of the DFIB method, in which a smoothed version of the discrete delta functions is utilised to transfer the physical quantities between two types of nodes. The velocities and pressure variables are approximated locally on Eulerian nodes using 3-node CIRBF stencils, where first- and second-order derivative values of the field variables are also included in the RBF approximations. The present DFIB-CIRBF method is verified through the solution of several test problems including Taylor-Green vortices, rotational flow, lid-driven cavity flow with multiple non-moving suspended solid bodies, flow between rotating circular and fixed square cylinders, and natural convection in an eccentric annulus between two circular cylinders. Numerical results obtained using relatively coarse grids are in good agreement with available data in the literature.

5.1 Introduction

Flows past solid bodies of arbitrary shapes are widely encountered in engineering applications. Body-fitted grid methods, where the governing equations are

discretised on a curvilinear grid conforming to the boundary, have been applied to solve such problems. Their main advantage is that the boundary conditions can be imposed in a simple and accurate way. However, generating a high quality mesh/grid is difficult and time-consuming. As a result, a lot of research effort has been spent on the development of non-body-conforming methods. Among them, the immersed boundary methods (IBMs) have received much attention in recent years. In IBMs, one joins the fluid and solid regions together to make a single domain that is discretised using an appropriate method. This approach greatly simplifies the process of mesh generation and also retains the relative simplicity of the governing equations. The basis of IBMs lies in the way to introduce forces into the governing equations to impose prescribed values on the immersed boundary.

The IBM was originally introduced by Peskin (1972, 1977) to investigate the fluid dynamics of blood flow in human heart. The idea is to use an equivalent body force field to represent bodies immersed in a fluid and hence the whole domain, with the immersed bodies, is considered as fluid. Since then, many variants of the Peskin's method have been proposed. Goldstein et al. (1993) developed a feedback forcing approach to iteratively determine the magnitude of the force required to obtain a desired velocity on the immersed boundary. Saiki and Biringen (1996) implemented this approach with the virtual boundary method to compute the flow past a stationary, rotating and oscillating circular cylinder. However, the feedback forcing approach induces some oscillations and places some restriction on the computational time step. To overcome these drawbacks, Fadlun et al. (2000) proposed an approach, namely the direct forcing (DF) technique, to evaluate the interactive forces between the immersed boundary (IB) and the fluid, which is equivalent to applying a forcing term to the Navier-Stokes equations. In comparison with the feedback forcing approach, the DF approach can work with larger computational time steps. Kim et al. (2001) proposed a combined IB finite-volume method, where a mass source/sink and a momentum forcing are introduced, for simulating flows over complex geometries. To transfer the physical quantities smoothly between Eulerian and Lagrangian nodes and avoid strong restrictions on the time step, Uhlmann (2005) presented a method to incorporate the regularised delta functions into a direct formulation of the fluid-solid interactive force. Wang et al. (2008) developed an explicit multi-direct forcing approach and obtained a better satisfaction of the non-slip boundary condition than the original DF approach. Recently, Ji et al. (2012) proposed an iterative IBM in which the body force updating is incorporated into the pressure iterations for 2D and 3D numerical simulations of laminar and turbulent flows. The reader is referred to, e.g. Mittal and Iaccarino (2005) for a comprehensive review of IBMs.

High-order approximation schemes for the Navier-Stokes equations have the ability to provide efficient solutions to steady/unsteady fluid flow problems. A high level of accuracy can be achieved using a relatively coarse discretisation. Many types of high-order schemes for the Navier-Stokes equations have

been reported in the literature. Botella and Peyret (1998) developed a Chebyshev collocation method and provided the benchmark results for the lid-driven cavity flow problem. Ding et al. (2006) presented a local multiquadric differential quadrature method for the solution of 3D incompressible flow problems in the velocity-pressure formulation, while Mai-Duy and Tran-Cong (2001b), Mai-Duy et al. (2008), Mai-Duy and Tran-Cong (2008), Le-Cao et al. (2009) proposed an IRBF method to solve heat transfer and fluid flow problems in the stream function-vorticity formulation. Recently, Tian et al. (2011) proposed a fourth-order compact difference scheme constructed on 2D nine-point stencils, and Fadel and Agouzoul (2011) used the standard Padé scheme to construct high-order approximations for the velocity-pressure-pressure gradient formulation. It is noted that the velocity ($\mathbf{u}$) and pressure (p) formulation has several advantages over the stream function-vorticity formulation and the stream function formulation. The $\mathbf{u}$-p formulation can provide the velocity and pressure fields directly from solving the discretised equations and also work for 2D and 3D problems in a similar manner.

In this chapter, we present a numerical scheme, namely DFIB-CIRBF, for solving unsteady/steady fluid flow problems in 2D. The present scheme combines DFIB method and the high-order CIRBF approximations for the spatial discretisation and utilises the second-order Adams-Bashforth/Crank-Nicolson algorithms for the temporal discretisation. An interactive force, representing the effect of the solid bodies on the fluid region, is added directly to the governing equations (i.e. direct forcing) on the fluid-solid regions to satisfy their boundary conditions. This interactive force is evaluated explicitly from the pressure gradient, the convection and diffusion terms in the previous time level. Because the Eulerian grid nodes do not generally coincide with the nodes on the interfaces represented by Lagrangian nodes, a smoothed version of the discrete delta functions is employed to transfer the quantities between two types of nodes. The CIRBF approximations are constructed over 3-point stencils, where nodal first- and second-order derivative values of the field variables are included in the RBF approximations (Thai-Quang, Mai-Duy, Tran and Tran-Cong, 2013, 2012). A series of test problems, including Taylor-Green vortices, rotational flow, flow between rotating circular and fixed square cylinders, and natural convection in an eccentric annulus between two circular cylinders, is considered to verify the present scheme. The remainder of the chapter is organised as follows. Section 5.2 outlines the equations which govern the fluid flow phenomena. The numerical formulation including the derivation of interactive forces, and the temporal and spatial discretisations is described in detail in Section 5.3. In Section 5.4, in order to evaluate the efficiency of the present method, several numerical results are presented and compared with the analytic solutions and some approximate results available in the literature, where appropriate. Section 5.5 concludes the chapter.

5.2 Governing equations

The idea of using an equivalent body force field to represent bodies immersed in a fluid gives rise to the immersed boundary (IB) method as reviewed in section 5.1 above. We adapt the IB method in which the non-slip fluid-solid interfaces are replaced by an equivalent body force field $\mathbf{F}_I(\mathbf{s},t)$ distributed over the immersed surfaces. The body forces can then be converted to equivalent body force field $\mathbf{f}_I(\mathbf{x},t)$ distributed over the whole domain, with the immersed bodies now considered as fluid. Similarly, for non-isothermal interfaces one can introduce a thermal interactive force density f_{IT} over the whole domain. Once this step is achieved the entire domain is considered as containing only the fluid. The flow field variables can then be Eulerian while the fluid-solid interface ones are Lagrangian. The singular forces on the boundaries can be computed and converted to an Eulerian body force field via regularised Dirac delta functions. In the context of the IBM, the governing equations for thermal flows in the dimensionless form are as follows.

$$\nabla.\mathbf{u} = 0 \qquad \text{in} \quad \Omega,\ t \geq 0, \tag{5.1}$$

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}.\nabla)\mathbf{u} = -\nabla p + \sqrt{\frac{Pr}{Ra}}\nabla^2\mathbf{u} + \mathbf{f}_b + \mathbf{f}_I \qquad \text{in} \quad \Omega,\ t \geq 0, \tag{5.2}$$

$$\frac{\partial T}{\partial t} + (\mathbf{u}.\nabla)T = \frac{1}{\sqrt{PrRa}}\nabla^2 T + f_{IT} \qquad \text{in} \quad \Omega,\ t \geq 0, \tag{5.3}$$

subject to the initial and boundary conditions:

$$\mathbf{u}(x,y,0) = \mathbf{u}_0(x,y) \qquad \text{in} \quad \Omega,\ t = 0, \tag{5.4}$$

$$T(x,y,0) = T_0(x,y) \qquad \text{in} \quad \Omega,\ t = 0, \tag{5.5}$$

$$\mathbf{u}(x,y,t) = \mathbf{u}_\Gamma(x,y,t) \qquad \text{on} \quad \Gamma,\ t \geq 0, \tag{5.6}$$

$$T(x,y,t) = T_\Gamma(x,y,t) \qquad \text{on} \quad \Gamma,\ t \geq 0, \tag{5.7}$$

where T is the temperature; $\mathbf{f}_I = \left(f_{Ix}, f_{Iy}\right)^T$ the fluid body interactive force density vector; f_{IT} the thermal interactive force; T_0 and T_Γ prescribed functions; Pr and Ra the Prandtl and Rayleigh numbers defined as $Pr = \nu/\alpha$ and $Ra = \beta g \Delta T L^3/\alpha\nu$, respectively, in which α is thermal diffusivity, β is the thermal expansion coefficient, g is the gravity and ΔT is the temperature difference, respectively. In the dimensionless form, the characteristic velocity is taken as $U_0 = \sqrt{gL\beta\Delta T}$ for the purpose of balancing the buoyancy and inertial forces.

In (5.1), (5.2) and (5.3), the field variables are made dimensionless according to the following definitions

$$x = \frac{x'}{L},\ y = \frac{y'}{L},\ u = \frac{u'}{U_0},\ v = \frac{v'}{U_0},\ p = \frac{p'}{\rho U_0^2},\ T = \frac{T' - T_c}{T_h - T_c}, \tag{5.8}$$

where x', y', u', v', p', T' are the corresponding dimensional variables; and T_h and T_c the hot and cold temperatures, respectively.

The interactive forces $\mathbf{f}_I$ and f_{IT} represent the influence of the immersed solid bodies on the fluid by the viscous and thermal effects, while the body force $\mathbf{f}_b$ is a function of the temperature, for instance, $\mathbf{f}_b = (0, T)^T$ for the thermal problem considered in section 5.4. For isothermal flows, the term $\mathbf{f}_b$ in (5.2) is set to null, equation (5.3) is deactivated and the term $\sqrt{\frac{Pr}{Ra}}$ in (5.2) is replaced by $\frac{1}{Re}$.

5.3 Numerical formulation

Consider a domain Ω comprised of the fluid region Ω_f and solid region Ω_s. The latter is composed of N_{esb} embedded solid bodies S_k $\left(\Omega_s = \bigcup_{k=1}^{N_{esb}} S_k\right)$ as shown in Figure 5.1. Let Γ and ∂S_k be the boundaries of Ω and kth solid body S_k, respectively. While the entire domain Ω is discretised using a fixed uniform Cartesian grid g_h containing Eulerian grid nodes $\mathbf{x}_{i,j} = (x_{i,j}, y_{i,j})^T$ ($i \in \{1, 2, \ldots, n_x\}$ and $j \in \{1, 2, \ldots, n_y\}$), each ∂S_k is described by a set of N_L^k Lagrangian nodes

$$\mathbf{X}_l^k = \left(X_l^k, Y_l^k\right)^T \in \partial S_k \quad 1 \leq l \leq N_L^k,\ 1 \leq k \leq N_{esb}. \tag{5.9}$$

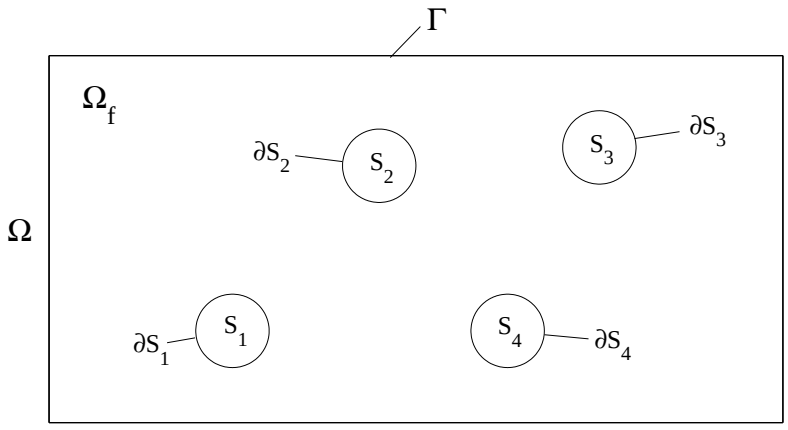


Figure 5.1: A schematic outline for the problem domain.

5.3.1 Direct forcing (DF) method

The DF method (Fadlun et al., 2000), a variant of the IB approach, proposed the so-called direct forcing technique for the computation of the body force field. It can be seen that the Lagrangian nodes, representing the immersed boundaries, do not generally coincide with the Eulerian nodes on the computational domain Ω. An interpolation process is therefore necessary to transfer data between the selected Eulerian nodes and the Lagrangian nodes on the immersed body surfaces. Below are the details for computing the momentum interactive force $\mathbf{f}_I$ in (5.2). One can calculate the thermal interactive force f_{IT} in (5.3) in a similar manner.

Derivation of the momentum interactive force

A temporal discretisation of the momentum equation (5.2) is given by (Uhlmann, 2005)

$$\frac{\mathbf{u}^n - \mathbf{u}^{n-1}}{\Delta t} = \mathbf{rhs}^{n-1/2} + \mathbf{f}_I^{n-1/2}, \tag{5.10}$$

where the convection, pressure, diffusion and body-force terms at a time $t^{n-1/2}$ are lumped together in $\mathbf{rhs}^{n-1/2}$.

The interactive force term yielding the desired velocity $\mathbf{u}^{(d)}$ can thus be defined as (Fadlun et al., 2000)

$$\mathbf{f}_I^{n-1/2} = \frac{\mathbf{u}^{(d),n} - \mathbf{u}^{n-1}}{\Delta t} - \mathbf{rhs}^{n-1/2}, \tag{5.11}$$

at some selected nodes (and zero elsewhere). The corresponding interactive force at the Lagrangian nodes will be

$$\mathbf{F}_I^{n-1/2} = \frac{\mathbf{U}^{(d),n} - \mathbf{U}^{n-1}}{\Delta t} - \mathbf{RHS}^{n-1/2}. \tag{5.12}$$

Hereafter, we use upper-case letters for quantities evaluated at the Lagrangian nodes $\mathbf{X}_l^k$.

The desired velocity at a node on the fluid-solid interface in (5.12) is computed from the rigid-body motion of the solid body as follow.

$$\mathbf{U}^{(d)}(\mathbf{X}_l^k) = \mathbf{U}_c^k + \boldsymbol{\omega}_c^k \times (\mathbf{X}_l^k - \mathbf{X}_c^k), \tag{5.13}$$

where $\mathbf{U}_c^k = \left(U_c^k, V_c^k\right)^T$ and $\mathbf{X}_c^k = \left(X_c^k, Y_c^k\right)^T$ are the translational velocity and the position vectors of the mass centre of the kth solid body, respectively, and $\boldsymbol{\omega}_c^k$ is the rotational velocity of the solid body - all is defined in the Cartesian

coordinate system.

When the interactive force is absent, equation (5.12) leads to

$$\widetilde{\mathbf{U}}^n = \mathbf{U}^{n-1} + \mathbf{RHS}^{n-1/2}\Delta t, \tag{5.14}$$

where $\widetilde{\mathbf{U}}^n$ is a preliminary velocity. Its Eulerian counterpart is

$$\widetilde{\mathbf{u}}^n = \mathbf{u}^{n-1} + \mathbf{rhs}^{n-1/2}\Delta t. \tag{5.15}$$

In the present work, we employ the Adams-Bashforth scheme for the temporal discretisation. The term $\mathbf{rhs}^{n-1/2}$ in (5.15) is computed explicitly as (Butcher, 2003)

$$\begin{aligned}\mathbf{rhs}^{n-1/2} = &-\left[\frac{3}{2}\nabla p^{n-1} - \frac{1}{2}\nabla p^{n-2}\right] - \left[\frac{3}{2}(\mathbf{u}^{n-1}.\nabla)\mathbf{u}^{n-1} - \frac{1}{2}(\mathbf{u}^{n-2}.\nabla)\mathbf{u}^{n-2}\right] \\ &+ \sqrt{\frac{Pr}{Ra}}\left[\frac{3}{2}\nabla^2\mathbf{u}^{n-1} - \frac{1}{2}\nabla^2\mathbf{u}^{n-2}\right] + \left[\frac{3}{2}\mathbf{f}_b^{n-1} - \frac{1}{2}\mathbf{f}_b^{n-2}\right].\end{aligned} \tag{5.16}$$

Then, the interactive force at the Lagrangian nodes is computed now as

$$\mathbf{F}_I^{n-1/2} = \frac{\mathbf{U}^{(d),n} - \widetilde{\mathbf{U}}^n}{\Delta t}. \tag{5.17}$$

In order to complete the evaluation of the interactive force term in (5.10), a mechanism for transferring the preliminary velocities $(\widetilde{\mathbf{u}}^n, \widetilde{\mathbf{U}}^n)$ and the forces $(\mathbf{F}_I^{n-1/2}, \mathbf{f}_I^{n-1/2})$ between the two Eulerian and Lagrangian node systems is required.

Transfer of quantities between Eulerian and Lagrangian nodes

Peskin (2002) employed the class of regularised delta functions

$$\delta_h(\mathbf{x} - \mathbf{x}_0) = \frac{1}{h^2}\phi\left(\frac{x - x_0}{h}\right)\phi\left(\frac{y - y_0}{h}\right), \tag{5.18}$$

as kernels in a transfer step, where $\phi(r)$ is the 1D discrete delta functions (r can be $(x - x_0)/h$ or $(y - y_0)/h$); and h the Eulerian grid size. The relation of the velocity and force between the two types of nodes can be given by (Uhlmann, 2005)

$$\widetilde{\mathbf{U}}(\mathbf{X}_l^k) = \sum_{\mathbf{x}\in g_h} \widetilde{\mathbf{u}}(\mathbf{x})\delta_h(\mathbf{x} - \mathbf{X}_l^k)h^2 \quad \forall 1 \le l \le N_L,\ 1 \le k \le N_{esb}, \tag{5.19}$$

$$\mathbf{f}_I(\mathbf{x}) = \sum_{k=1}^{N_{esb}} \sum_{l=1}^{N_L} \mathbf{F}_I(\mathbf{X}_l^k)\delta_h(\mathbf{x} - \mathbf{X}_l^k)\Delta V_l^k \quad \forall \mathbf{x} \in g_h, \tag{5.20}$$

where the temporal superscript is dropped for brevity and ΔV_l^k is the volume covering the lth Lagrangian node of the kth solid body. For 2D problems, this volume is simply taken as $\Delta V_l^k = \Delta s^2$ (Uhlmann, 2005), where Δs is a Lagrangian grid size that is chosen so that $\Delta s \approx h$.

In Peskin (2002), several axioms, including momentum conditions and a quadratic condition, are described. These axioms lead to the unique definition of a particular smoothed delta function with finite support. A family of such functions may be generated by imposing additional moment conditions and correspondingly broadening the support. Commonly used discrete delta functions include the 2-point hat function $\delta_{2h}(r)$ (Leveque and Li, 1994), the 3-point discrete delta function $\delta_{3h}(r)$ (Roma et al., 1999) and the 4-point piecewise function $\delta_{4h}(r)$ (Peskin, 2002). Their 1D forms are given below

$$\phi_2(r) = \begin{cases} 1 - |r|, & |r| \leq 1, \\ 0, & 1 \leq |r|, \end{cases} \tag{5.21}$$

$$\phi_3(r) = \begin{cases} \frac{1}{3}\left(1 + \sqrt{-3r^2 + 1}\right), & |r| \leq 0.5, \\ \frac{1}{6}\left(5 - 3|r| - \sqrt{-3(1 - |r|)^2 + 1}\right), & 0.5 \leq |r| \leq 1.5, \\ 0, & 1.5 \leq |r|, \end{cases} \tag{5.22}$$

$$\phi_4(r) = \begin{cases} \frac{1}{8}\left(3 - 2|r| + \sqrt{1 + 4|r| - 4r^2}\right), & |r| \leq 1, \\ \frac{1}{8}\left(5 - 2|r| - \sqrt{-7 + 12|r| - 4r^2}\right), & 1 \leq |r| \leq 2, \\ 0, & 2 \leq |r|. \end{cases} \tag{5.23}$$

In the present study, we employ the 3-point discrete delta function $\delta_{3h}(r)$ (Roma et al., 1999).

5.3.2 Spatial discretisation

In this chapter, the spatial derivatives are discretised using the CIRBF-2 scheme described in section 4.3.1 and modified as follows. At the boundary nodes, the compact 4-point stencils are replaced with a newly derived compact 2-point stencil (Figure 5.2) in order to make all coefficient matrices tridiagonal. The present scheme is named CIRBF-3.

Figure 5.2: Special compact 2-point IRBF stencils for the left and right boundary nodes.

First-order derivative compact approximations

At the interior nodes, the compact approximations for the first-order derivatives are described as (4.36), leading to (4.38).

At the boundary nodes, we compute the first derivative here using special compact local stencils (Figure 5.2). These proposed stencils are constructed as follows. Consider a boundary node η_1. Its associated stencil is $[\eta_1, \eta_2]$. The conversion system of this stencil is presented as the following matrix-vector multiplication

$$\begin{pmatrix} u_1 \\ u_2 \\ \frac{du_2}{d\eta} \end{pmatrix} = \underbrace{\begin{pmatrix} \overline{\mathcal{H}_{sp}} \\ \mathcal{H}_{sp} \end{pmatrix}}_{\mathcal{C}_{sp_1}} \begin{pmatrix} w_1 \\ w_2 \\ c_1 \\ c_2 \end{pmatrix}, \tag{5.24}$$

where $\mathcal{C}_{sp_1}$ is the conversion matrix; and $\overline{\mathcal{H}_{sp}}$, $\mathcal{H}_{sp}$ matrices defined as

$$\overline{\mathcal{H}_{sp}} = \begin{bmatrix} \overline{H}_1(\eta_1) & \overline{H}_2(\eta_1) & \eta_1 & 1 \\ \overline{H}_1(\eta_2) & \overline{H}_2(\eta_2) & \eta_2 & 1 \end{bmatrix}, \tag{5.25}$$

$$\mathcal{H}_{sp} = \begin{bmatrix} H_1(\eta_2) & H_2(\eta_2) & 1 & 0 \end{bmatrix}. \tag{5.26}$$

Solving (5.24) yields

$$\begin{pmatrix} w_1 \\ w_2 \\ c_1 \\ c_2 \end{pmatrix} = \mathcal{C}_{sp_1}^{-1} \begin{pmatrix} u_1 \\ u_2 \\ \frac{du_2}{d\eta} \end{pmatrix}. \tag{5.27}$$

The boundary value of the first-order derivative of u is thus obtained by substituting (5.27) into (2.10) and taking $\eta = \eta_1$

$$\frac{du(\eta_1)}{d\eta} = \begin{bmatrix} H_1(\eta_1) & H_2(\eta_1) & 1 & 0 \end{bmatrix} \mathcal{C}_{sp_1}^{-1} \begin{pmatrix} u_1 & u_2 & \frac{du_2}{d\eta} \end{pmatrix}^T, \tag{5.28}$$

or

$$\frac{du_1^n}{d\eta} - \frac{d\phi_{sp_3}(\eta_1)}{d\eta}\frac{du_2^n}{d\eta} = \frac{d\phi_{sp_1}(\eta_1)}{d\eta}u_1^n + \frac{d\phi_{sp_2}(\eta_1)}{d\eta}u_2^n, \tag{5.29}$$

where $\{\phi_{sp_i}(\eta)\}_{i=1}^3$ is the set of IRBFs in the physical space. We rewrite equation

(5.29) in matrix-vector form

$$\left[\begin{array}{cc} 1 & -\frac{d\phi_{sp_3}(\eta_1)}{d\eta} \end{array}\right] \left[\begin{array}{c} \frac{du_1^n}{d\eta} \\ \frac{du_2^n}{d\eta} \end{array}\right] = \left[\begin{array}{cc} \frac{d\phi_{sp_1}(\eta_1)}{d\eta} & \frac{d\phi_{sp_2}(\eta_1)}{d\eta} \end{array}\right] \left[\begin{array}{c} u_1^n \\ u_2^n \end{array}\right]. \tag{5.30}$$

In a similar manner, one can calculate the first derivative of u at the other boundary node η_{n_η}.

The IRBF system on a grid line for the first derivative of u is obtained by letting the interior node taking value from 2 to $(n_\eta - 1)$ in (4.38) and making use of (5.30),

$$\mathcal{L}_\eta \widehat{u}_\eta^n = \mathcal{A}_\eta \widehat{u}^n. \tag{5.31}$$

Second-order derivative compact approximations

At the interior nodes, the compact approximations for the second-order derivative leads to equation (4.28).

At the boundary nodes, we compute the second derivative here using special compact local stencils (Figure 5.2). Consider a boundary node, e.g., η_1. The conversion system of its associated 2-node stencil is presented as the following matrix-vector multiplication

$$\left(\begin{array}{c} u_1 \\ u_2 \\ \frac{d^2u_2}{d\eta^2} \end{array}\right) = \underbrace{\left(\begin{array}{c} \overline{\mathcal{H}_{sp}} \\ \mathcal{G}_{sp} \end{array}\right)}_{\mathcal{C}_{sp_2}} \left(\begin{array}{c} w_1 \\ w_2 \\ c_1 \\ c_2 \end{array}\right), \tag{5.32}$$

where $\mathcal{C}_{sp_2}$ is the conversion matrix; $\overline{\mathcal{H}_{sp}}$ defined as before; and

$$\mathcal{G}_{sp} = \left[\begin{array}{cccc} G_1(\eta_2) & G_2(\eta_2) & 0 & 0 \end{array}\right]. \tag{5.33}$$

Solving (5.32) yields

$$\left(\begin{array}{c} w_1 \\ w_2 \\ c_1 \\ c_2 \end{array}\right) = \mathcal{C}_{sp_2}^{-1} \left(\begin{array}{c} u_1 \\ u_2 \\ \frac{d^2u_2}{d\eta^2} \end{array}\right). \tag{5.34}$$

The boundary value of the second-order derivative of u is thus obtained by substituting (5.34) into (2.9) and taking $\eta = \eta_1$

$$\frac{d^2u(\eta_1)}{d\eta^2} = \left[\begin{array}{cccc} G_1(\eta_1) & G_2(\eta_1) & 0 & 0 \end{array}\right] \mathcal{C}_{sp_2}^{-1} \left(\begin{array}{ccc} u_1 & u_2 & \frac{d^2u_2}{d\eta^2} \end{array}\right)^T, \tag{5.35}$$

or

$$\frac{d^2u_1^n}{d\eta^2} - \frac{d^2\varphi_{sp_3}(\eta_1)}{d\eta^2}\frac{d^2u_2^n}{d\eta^2} = \frac{d^2\varphi_{sp_1}(\eta_1)}{d\eta^2}u_1^n + \frac{d^2\varphi_{sp_2}(\eta_1)}{d\eta^2}u_2^n, \tag{5.36}$$

where $\{\varphi_{sp_i}(\eta)\}_{i=1}^3$ is the set of IRBFs in the physical space. We rewrite equation (5.36) in matrix-vector form

$$\begin{bmatrix} 1 & -\frac{d^2\varphi_{sp_3}(\eta_1)}{d\eta^2} \end{bmatrix} \begin{bmatrix} \frac{d^2u_1^n}{d\eta^2} \\ \frac{d^2u_2^n}{d\eta^2} \end{bmatrix} = \begin{bmatrix} \frac{d^2\varphi_{sp_1}(\eta_1)}{d\eta^2} & \frac{d^2\varphi_{sp_2}(\eta_1)}{d\eta^2} \end{bmatrix} \begin{bmatrix} u_1^n \\ u_2^n \end{bmatrix}. \tag{5.37}$$

The IRBF system on a grid line for the second derivative of u is obtained by letting the interior node taking value from 2 to $(n_\eta - 1)$ in (4.28) and making use of (5.37),

$$\mathcal{L}_{\eta\eta}\widehat{u}_{\eta\eta}^n = \mathcal{B}_{\eta\eta}\widehat{u}^n, \tag{5.38}$$

where $\mathcal{L}_{\eta\eta}$, $\mathcal{B}_{\eta\eta}$ are $n_\eta \times n_\eta$ matrices.

5.3.3 Temporal discretisation

The temporal discretisation of (5.1)-(5.3) using the Adams-Bashforth scheme (Butcher, 2003) for the convection term and the Crank-Nicolson scheme (Crank and Nicolson, 1996) for the diffusion term yields

$$\nabla.\mathbf{u}^n = 0, \tag{5.39}$$

$$\frac{\mathbf{u}^n - \mathbf{u}^{n-1}}{\Delta t} + \left[\frac{3}{2}(\mathbf{u}^{n-1}.\nabla)\mathbf{u}^{n-1} - \frac{1}{2}(\mathbf{u}^{n-2}.\nabla)\mathbf{u}^{n-2}\right] = - \nabla p^{n-1/2} + \frac{1}{2}\sqrt{\frac{Pr}{Ra}}\left(\nabla^2\mathbf{u}^n + \nabla^2\mathbf{u}^{n-1}\right) + \mathbf{f}_b^{n-1/2} + \mathbf{f}_I^{n-1/2}, \tag{5.40}$$

$$\frac{T^n - T^{n-1}}{\Delta t} + \left[\frac{3}{2}(\mathbf{u}^{n-1}.\nabla)T^{n-1} - \frac{1}{2}(\mathbf{u}^{n-2}.\nabla)T^{n-2}\right] = \frac{1}{2\sqrt{PrRa}}\left(\nabla^2T^n + \nabla^2T^{n-1}\right) + f_{I_T}^{n-1/2}. \tag{5.41}$$

We apply the pressure-free projection/fractional-step method developed by Kim and Moin (1985) to solve (5.40). This equation is advanced in time according

to the following two-step procedure

$$\frac{\mathbf{u}^{*,n}-\mathbf{u}^{n-1}}{\Delta t}+\left[\frac{3}{2}(\mathbf{u}^{n-1}.\nabla)\mathbf{u}^{n-1}-\frac{1}{2}(\mathbf{u}^{n-2}.\nabla)\mathbf{u}^{n-2}\right]= \frac{1}{2}\sqrt{\frac{Pr}{Ra}}\left(\nabla^2\mathbf{u}^{*,n}+\nabla^2\mathbf{u}^{n-1}\right)+\mathbf{f}_b^{n-1/2}+\mathbf{f}_I^{n-1/2}, \quad (5.42)$$

$$\frac{\mathbf{u}^{n}-\mathbf{u}^{*,n}}{\Delta t}=-\nabla\phi^n, \quad (5.43)$$

where ϕ is the pseudo-pressure. It is noted that $\mathbf{u}^{*,n}$ does not satisfy the continuity equation (5.39) and the actual pressure p is derived as

$$p^{n-1/2}=\phi^n-\left(\frac{\Delta t}{2}\sqrt{\frac{Pr}{Ra}}\right)\nabla^2\phi^n. \quad (5.44)$$

5.3.4 Algorithm of the computational procedure

- Step 0: Start with the given initial and boundary conditions. In this study, the initial conditions are zero for the velocity and temperature fields.
- Step 1: Compute thermal Eulerian counterpart $\widetilde{\mathbf{t}}^n$, using a formula similar to (5.15), which is then transferred to Lagrangian nodes to obtain $\widetilde{\mathbf{T}}^n$ using a formula similar to (5.19).
- Step 2: Compute ${F_I}_T^{n-1/2}$, using a formula similar to (5.17), which is then transferred to Eulerian nodes to obtain ${f_I}_T^{n-1/2}$ using a formula similar to (5.20).
- Step 3: Solve (5.41) for the solution T^n with known ${f_I}_T^{n-1/2}$ and prescribed boundary condition T_Γ^n.
- Step 4: Compute the body force $\mathbf{f}_b^{n-1/2}$ from the temperature field as

$$\mathbf{f}_b^{n-1/2}=\left(0,T^{n-1/2}\right)^T=\left(0,\frac{T^n+T^{n-1}}{2}\right)^T. \quad (5.45)$$

- Step 5: Compute momentum Eulerian counterpart $\widetilde{\mathbf{u}}^n$ from (5.15), which is then transferred to Lagrangian nodes to obtain $\widetilde{\mathbf{U}}^n$ via (5.19).
- Step 6: Compute $\mathbf{F}_I^{n-1/2}$ from (5.17), which is then transferred to Eulerian nodes to obtain $\mathbf{f}_I^{n-1/2}$ via (5.20).
- Step 7: Solve (5.42) for $\mathbf{u}^{*,n}$ subject to the following boundary condition (Kim and Moin, 1985)

$$\mathbf{u}^{*,n}|_\Gamma=\mathbf{u}_b^n+\Delta t\left(\nabla\phi^{n-1}\right)|_\Gamma. \quad (5.46)$$

For a more efficient solution, one can apply the alternating direction implicit (ADI) algorithm to solve (5.42) and (5.41) as shown in Thai-Quang, Mai-Duy, Tran and Tran-Cong (2012).

- Step 8: Equations (5.43) and (5.39) are then solved in a coupled manner for $\mathbf{u}^n$ and ϕ^n in which the boundary condition for the pseudo-pressure ϕ is not required. The values of ϕ^n are obtained for the interior nodes only. After that, the values of ϕ at the boundary nodes are extrapolated from known values at the interior nodes and known Neumann boundary values derived from (5.43) (i.e., $\nabla\phi^n|_\Gamma = (\mathbf{u}_b^{*,n} - \mathbf{u}_b^n)/\Delta t$) (Thai-Quang, Le-Cao, Mai-Duy and Tran-Cong, 2012):

$$\begin{pmatrix} \phi^n_{1,j} \\ \phi^n_{nx,j} \end{pmatrix} = \begin{bmatrix} \overline{H}_1(x_{1,j}) & \cdots & \overline{H}_{n_x}(x_{1,j}) & x_{1,j} & 1 \\ \overline{H}_1(x_{n_x,j}) & \cdots & \overline{H}_{n_x}(x_{n_x,j}) & x_{n_x,j} & 1 \end{bmatrix} \begin{bmatrix} \overline{H}_1(x_{2,j}) & \cdots & \overline{H}_{n_x}(x_{2,j}) & x_{2,j} & 1 \\ \overline{H}_1(x_{3,j}) & \cdots & \overline{H}_{n_x}(x_{3,j}) & x_{3,j} & 1 \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ \overline{H}_1(x_{n_x-1,j}) & \cdots & \overline{H}_{n_x}(x_{n_x-1,j}) & x_{n_x-1,j} & 1 \\ H_1(x_{1,j}) & \cdots & H_{n_x}(x_{1,j}) & 1 & 0 \\ H_1(x_{n_x,j}) & \cdots & H_{n_x}(x_{n_x,j}) & 1 & 0 \end{bmatrix}^{-1} \begin{pmatrix} \phi^n_{2,j} \\ \phi^n_{3,j} \\ \vdots \\ \phi^n_{n_x-1,j} \\ \partial\phi^n_{1,j}/\partial x \\ \partial\phi^n_{n_x,j}/\partial x \end{pmatrix}, \tag{5.47}$$

for an x-grid line, and

$$\begin{pmatrix} \phi^n_{i,1} \\ \phi^n_{i,n_y} \end{pmatrix} = \begin{bmatrix} \overline{H}_1(y_{i,1}) & \cdots & \overline{H}_{n_y}(y_{i,1}) & y_{i,1} & 1 \\ \overline{H}_1(y_{i,n_y}) & \cdots & \overline{H}_{n_y}(y_{i,n_y}) & y_{i,n_y} & 1 \end{bmatrix} \begin{bmatrix} \overline{H}_1(y_{i,2}) & \cdots & \overline{H}_{n_y}(y_{i,2}) & y_{i,2} & 1 \\ \overline{H}_1(y_{i,3}) & \cdots & \overline{H}_{n_y}(y_{i,3}) & y_{i,3} & 1 \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ \overline{H}_1(y_{i,n_y-1}) & \cdots & \overline{H}_{n_y}(y_{i,n_y-1}) & y_{i,n_y-1} & 1 \\ H_1(y_{i,1}) & \cdots & H_{n_y}(y_{i,1}) & 1 & 0 \\ H_1(y_{i,n_y}) & \cdots & H_{n_y}(y_{i,n_y}) & 1 & 0 \end{bmatrix}^{-1} \begin{pmatrix} \phi^n_{i,2} \\ \phi^n_{i,3} \\ \vdots \\ \phi^n_{i,n_y-1} \\ \partial\phi^n_{i,1}/\partial y \\ \partial\phi^n_{i,n_y}/\partial y \end{pmatrix}, \tag{5.48}$$

for a y-grid line. It is noted that for flows with irregular outer boundaries, instead of solving (5.43) and (5.39), we solve (5.39)-(5.40) simultaneously for $\mathbf{u}^n$ and $p^{n-1/2}$ in which $p^{n-1/2}$ involves the interior nodes only (the boundary condition for $p^{n-1/2}$ is not required here).

- Step 9: Go back to step 1 and iterate for the next time level.

5.4 Numerical examples

For the calculations in this chapter, values of $\beta = 25$ and $\beta = 50$ are simply picked. We assess the performance of the present scheme through following measures:

- the RMS error defined as (2.48),
- maximum absolute error (L_∞) defined as (3.53),
- the average rate of convergence α determined by the method of least squares as described in chapter 2,
- the convergence measure based on the velocity magnitude (CM_{vel}) in the whole analysis domain is defined as (given two successive grids)

$$CM_{vel} = \frac{\sqrt{\sum_{i=1}^{N} \left(vel_i^{ctfg} - vel_i^{fg} \right)^2}}{\sqrt{\sum_{i=1}^{N} \left(vel_i^{fg} \right)^2}}, \tag{5.49}$$

  where vel^{fg} is the velocity magnitude field computed using the finer grid; vel^{ctfg} is the velocity magnitude field obtained at the finer grid by interpolating the solution computed using the coarser grid. The present results is considered to be grid convergent if CM_{vel} is less than 10^{-3}.

A flow is considered to reach its steady state when the criterion (2.50) is satisfied.

Since the approximations are presently based on RBFs, distances between two neighbouring nodes in the stencil can be different. This capability is exploited here to handle non-rectangular outer boundaries in a direct manner (i.e. body-fitted grid). We can thus retain a body-conforming treatment for rectangular and non-rectangular outer boundaries. We numerically demonstrate this ability with the following example

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = -8\pi^2 \sin(2\pi x) \sin(2\pi y), \tag{5.50}$$

defined on a circular domain of radius $R = 1.5$ and subject to Dirichlet boundary condition. Its exact solution is $\overline{u} = \sin(2\pi x) \sin(2\pi y)$. A number of grids, namely $\{12 \times 12,\ 22 \times 22, \ldots,\ 102 \times 102\}$, are employed to study the grid-convergence behaviour of the solution (Figure 5.3). Those interior nodes that fall very close to the boundary (within a distance of $h/8$) are removed from the set of of nodal points. Figure 5.4 shows the matrix condition number and the RMS error of the interior solution against grid size. Results by a Cartesian-grid FDM (Sanmiguel-Rojas et al., 2005) are also included for comparison purposes.

The solution converges as $O(h^{2.03})$ for FDM and quite fast as $O(h^{3.17})$ for the present method. The two methods have similar condition numbers of the system matrix.

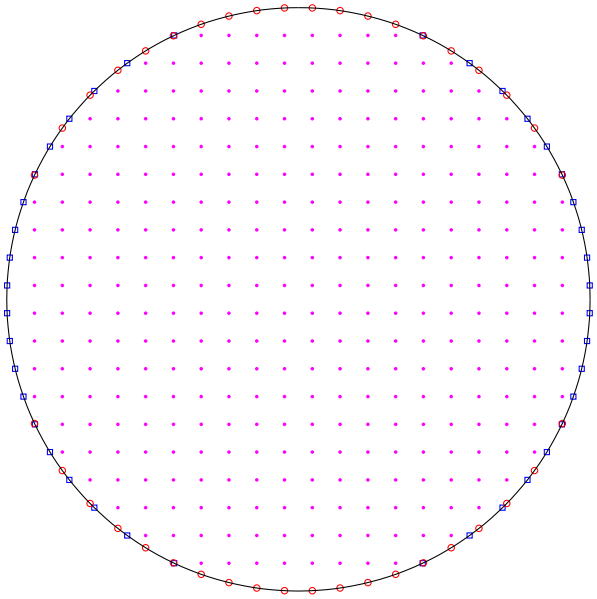


Figure 5.3: Poisson equation, circular domain: Computational domain and its discretisation.

5.4.1 Taylor-Green vortices

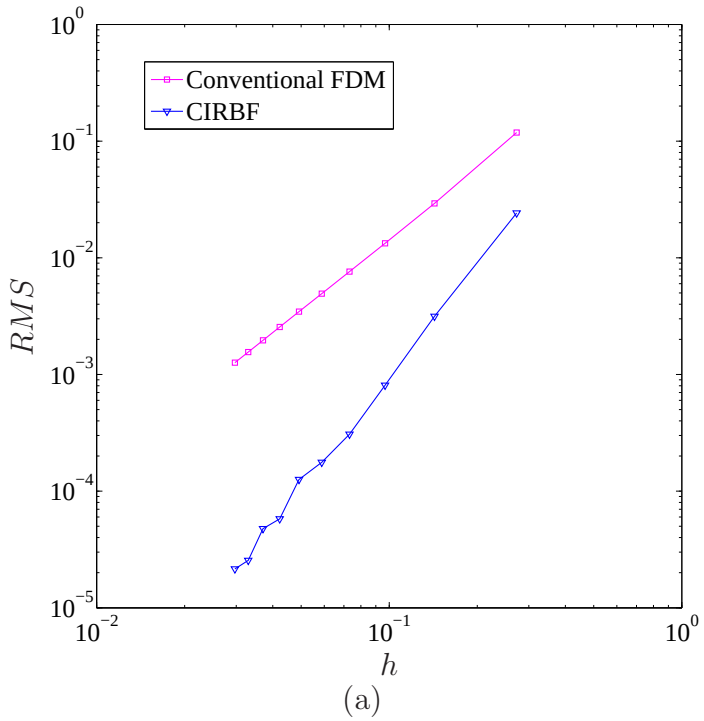
This problem is taken from Uhlmann (2005), where the analytic solution is given by

$$\overline{u}(x,y,t) = \sin(\pi x)\cos(\pi y)e^{-2\pi^2 t/Re}, \tag{5.51}$$

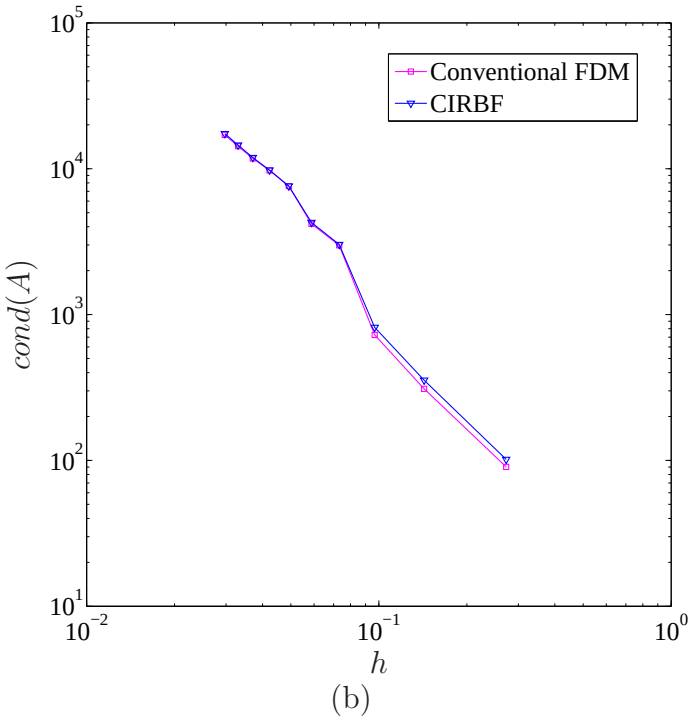
$$\overline{v}(x,y,t) = -\sin(\pi y)\cos(\pi x)e^{-2\pi^2 t/Re}, \tag{5.52}$$

$$\overline{p}(x,y,t) = 0.5\left(\cos^2(\pi y) - \sin^2(\pi x)\right)e^{-4\pi^2 t/Re}, \tag{5.53}$$

from which one can derive the initial solution, the time-dependent boundary conditions and the time-dependent desired velocities $\mathbf{U}^{(d)}$ on the inner immersed boundaries. The solution is computed at $Re = 5$ and $t = 0.3$ using a time step $\Delta t = 0.001$ and $\beta = 25$ for the following two domains.



(a)



(b)

Figure 5.4: Poisson equation, circular domain, $\{12\times12,\ 22\times22, \ldots,\ 102\times102\}$: The solution accuracy (a) and the matrix condition number (b) against grid size by FDM and the present method. The solution converges as $O(h^{2.03})$ and $O(h^{3.17})$ while the matrix condition grows as $O(h^{-2.52})$ and $O(h^{-2.46})$ for FDM and the present method, respectively.

Circular domain

A circular domain of unit radius is chosen here to investigate the performance of the present scheme in dealing with non-rectangular outer boundaries. Several grids, namely $\{12 \times 12,\ 22 \times 22, \ldots,\ 52 \times 52\}$ are employed. Figure 5.5 shows the RMS errors of the velocity components and the pressure against the grid size h. The solutions converge as $O(h^{3.31})$, $O(h^{3.29})$ and $O(h^{2.87})$ for the x-component velocity, y-component velocity and pressure, respectively. It can be seen that fast rates of convergence (about third order) are achieved with the present method. Figure 5.6 shows the analytic and computed vorticity isolines using a grid of 52×52, which are graphically indistinguishable.

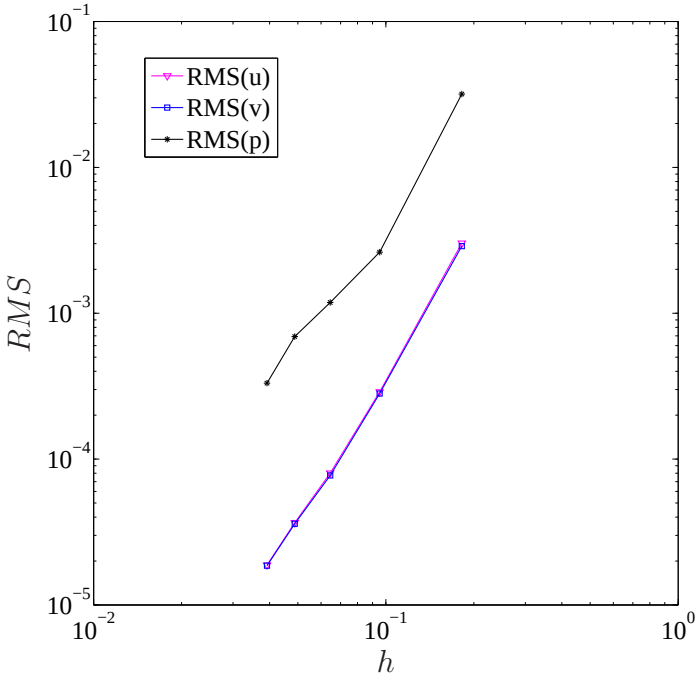
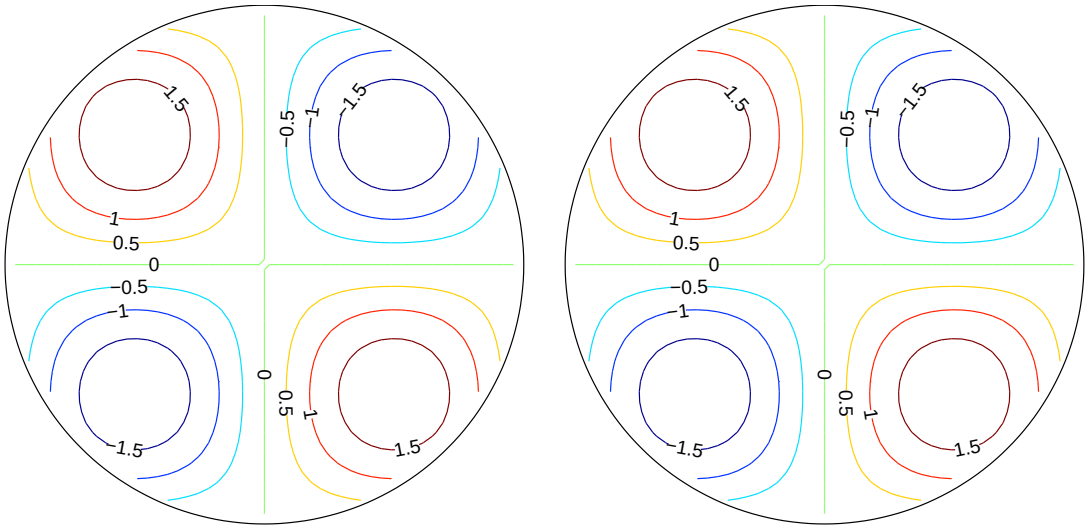


Figure 5.5: Taylor-Green vortices, circular domain, $\{12 \times 12,\ 22 \times 22, \ldots,\ 52 \times 52\}$: The solution accuracy of the velocity components and pressure against grid size. The solution converges as $O(h^{3.31})$, $O(h^{3.29})$ and $O(h^{2.87})$ for x-component velocity, y-component velocity and pressure, respectively.

Concentric annulus between two circular cylinders

The outer and inner radii of this domain are taken as $R_o = 1$ and $R_i = 0.5$, respectively. We employ a set of grids, namely $\{22 \times 22,\ 32 \times 32, \ldots,\ 52 \times 52\}$ to represent the problem domain. Figure 5.7 shows the Eulerian nodes dis-



(a) (b)

Figure 5.6: Taylor-Green vortices, circular domain, 52×52, $\Delta t = 0.001$: the analytic (a) and computed (b) isolines of the vorticity field at $t = 0.3$.

tributed inside and on the outer boundary, and Lagrangian nodes distributed on the inner boundary, for instance, by a grid of 22×22. Figure 5.8 shows the analytic and computed vorticity isolines using a grid of 52×52, where an excellent agreement can be seen. The L_∞ errors of the velocity components and pressure against the grid size h are presented in Figure 5.9. The solutions converge as $O(h^{2.02})$, $O(h^{2.03})$ and $O(h^{2.02})$ for u, v and p, respectively. The rates of convergence are reduced due to the effect of using regularised δ_h functions, which are second-order accurate (Uhlmann, 2005), in the IB approach.

5.4.2 Rotational flow

The present scheme is further verified with a rotational flow, where a circular ring (zero thickness) of $R = 0.3$ is embedded in a square domain $\Omega = [-1, 1] \times [-1, 1]$. The solid ring rotates about its centre with an angular velocity $\omega = 2$. The simulation is conducted for $Re = 18$ using a grid of 65×65 and $\Delta t = h/4$ as in Le et al. (2006). Plots of the velocity u and velocity vector in a subdomain $[-0.5, 0.5] \times [-0.5, 0.5]$ at $t = 10$ are shown in Figure 5.10, in which the flow behaviours observed here are very similar to those reported in Le et al. (2006).

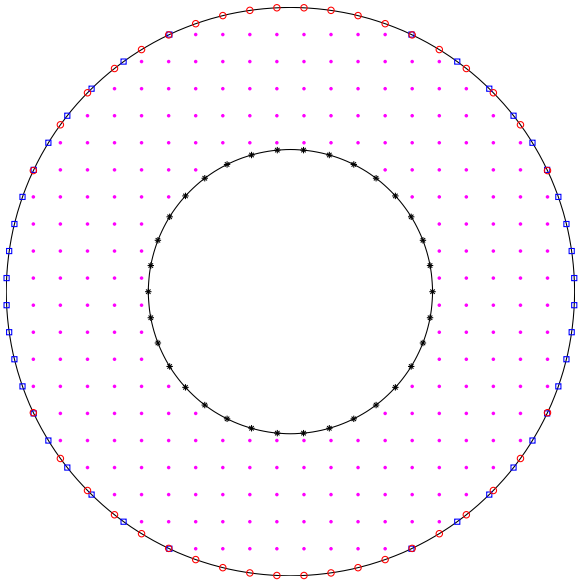
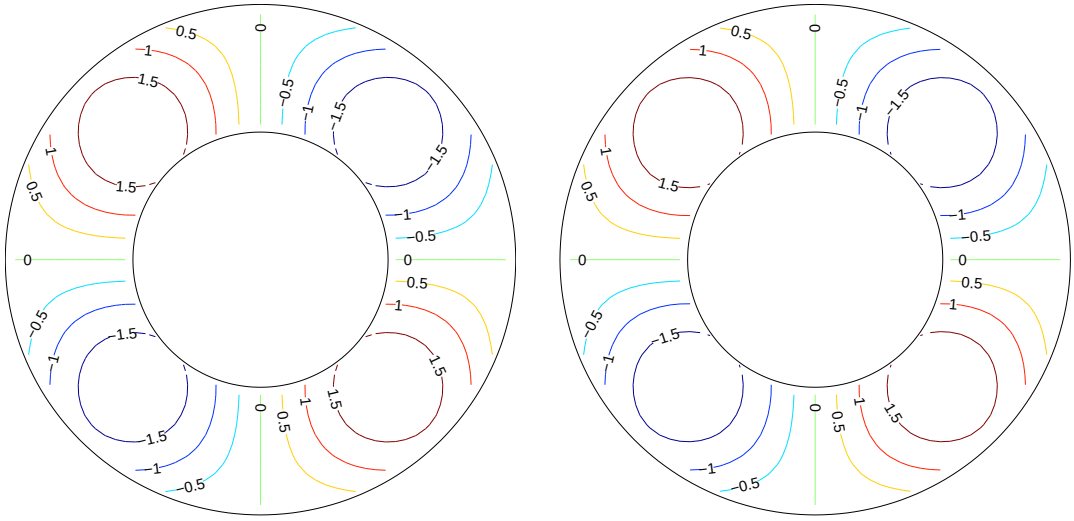


Figure 5.7: Taylor-Green vortices, concentric annulus: Computational domain and its discretisation (Eulerian nodes inside the annulus and on the outer boundary, Lagrangian nodes on the inner boundary with a grid of 22×22).



(a) (b)

Figure 5.8: Taylor-Green vortices, concentric annulus, 52×52, $\Delta t = 0.001$: the analytic (a) and computed (b) isolines of the vorticity field at $t = 0.3$.

5.4.3 Lid-driven cavity flow with multiple solid bodies

This test problem is concerned with the lid-driven cavity flow in a square domain $\Omega = [-1, 1] \times [-1, 1]$ containing five fixed rigid circular cylinders (Figure 5.11).

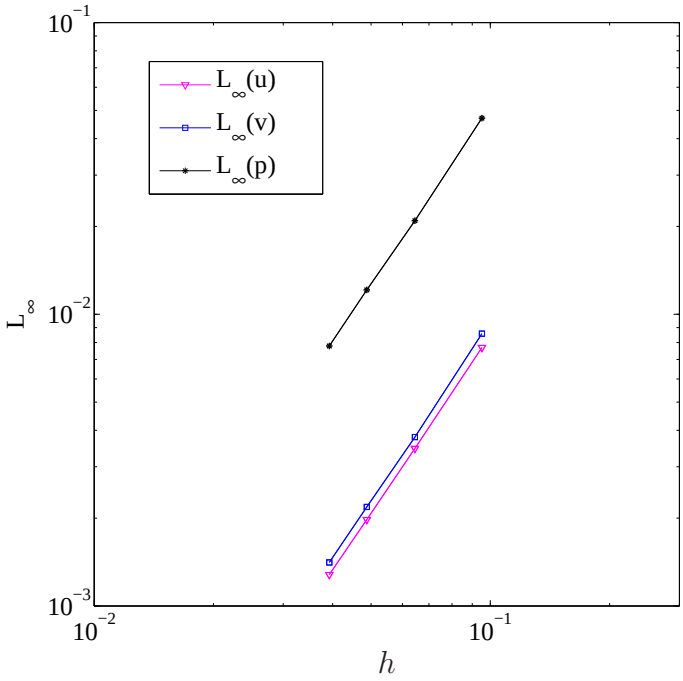
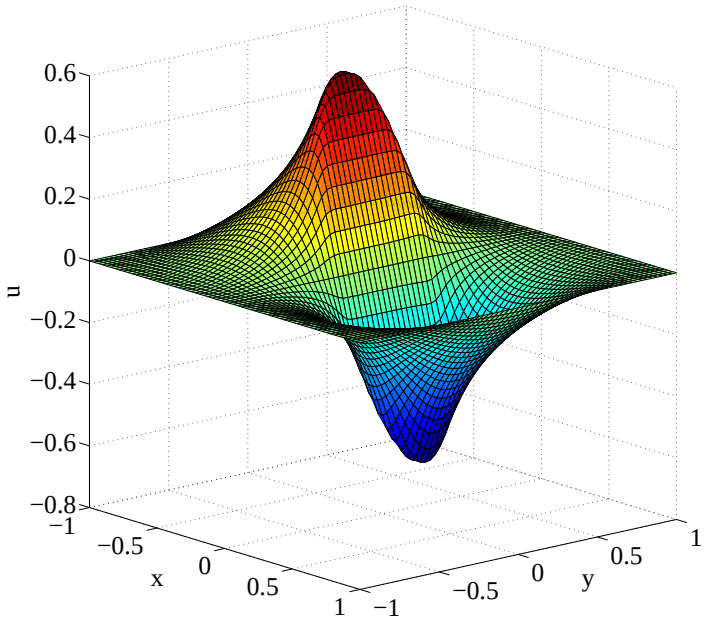
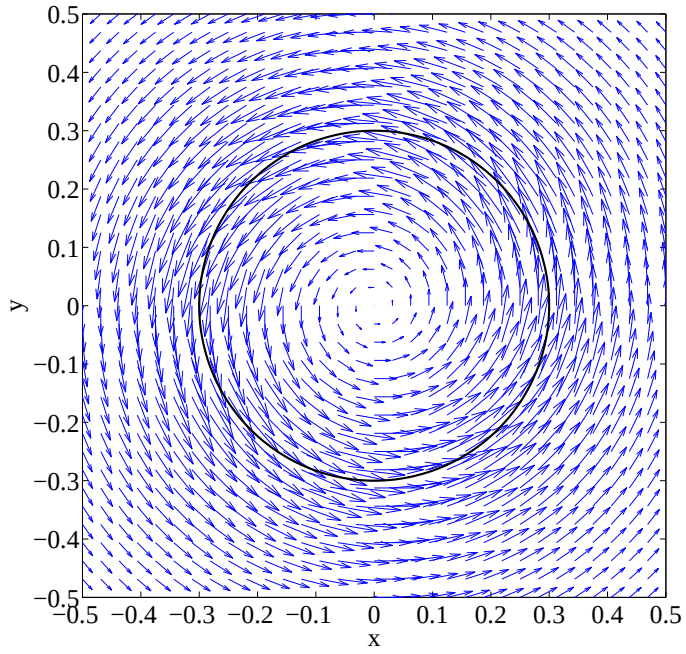


Figure 5.9: Taylor-Green vortices, concentric annulus, $\{22 \times 22,\ 32 \times 32, \ldots,\ 52 \times 52\}$: The solution accuracy of the velocity components and pressure against grid size. The solution converges as $O(h^{2.02})$, $O(h^{2.03})$ and $O(h^{2.02})$ for x-component velocity, y-component velocity and pressure, respectively.

The radius of the cylinders is $R = 0.15$ and their centres are located at $(0, 0)$, $(0, -0.6)$, $(-0.6, 0)$, $(0, 0.6)$ and $(0.6, 0)$, respectively. The top wall is driven from left to right by a unit velocity whereas the other walls are stationary. The Lagrangian nodes are distributed on the boundaries with a grid spacing ratio $\Delta s/h = 0.85$. These parameters are taken from Su and Lai (2007). The grid convergence study for this problem is carried out at $Re = 100$ on a set of uniform grids, namely $\{41 \times 41,\ 61 \times 61,\ 81 \times 81,\ 101 \times 101,\ 121 \times 121,\ 141 \times 141\}$, using a time step of $\Delta t = 0.001$. The present solutions converge at the grid of 121×121. The velocity field obtained with the grid 121×121 is presented in Figure 5.12, showing that the primary vortex is captured very well around the top-right corner. The flow field looks feasible and similar in comparison with those shown in Su and Lai (2007). (To avoid cluttering, the velocity vectors are plotted at every third grid point, i.e. at 41×41 points as in Su and Lai (2007)). Figure 5.13 shows the u-velocity profile along the diagonal $x = y$ for different grid sizes.



(a)



(b)

Figure 5.10: Rotational flow generated by a circular ring rotating about its centre in a fluid filled square cavity, $Re = 18$, 65×65, $t = 10$, $\Delta t = h/4$: Distributions of the x-component velocity (a) and velocity vector (b) over the computational domain.

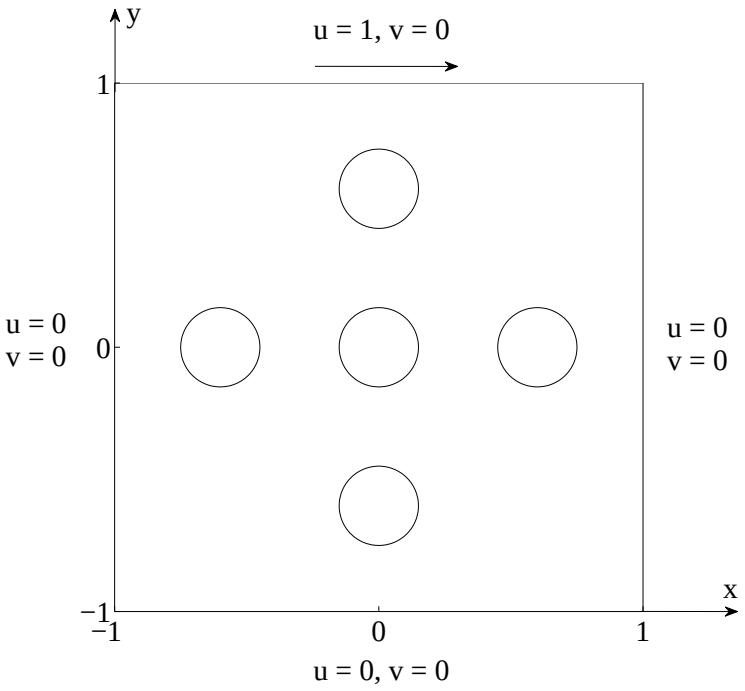


Figure 5.11: Lid-driven cavity flow with multiple solid bodies: Geometry and boundary condition.

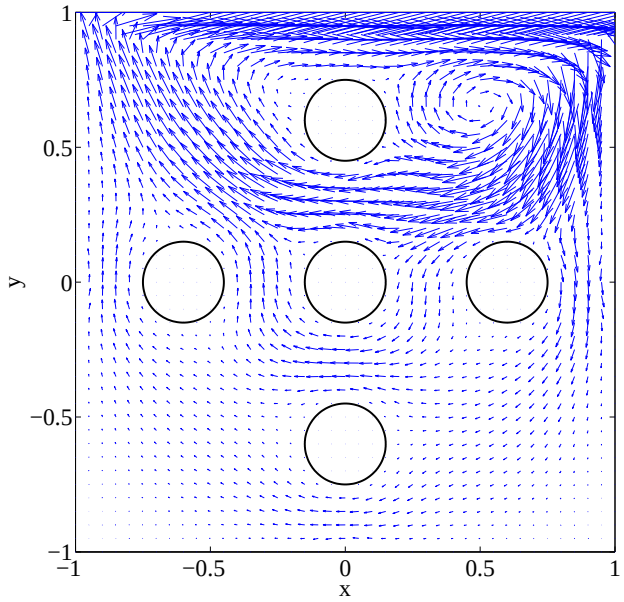


Figure 5.12: Lid-driven cavity flow with multiple solid bodies: Velocity vector field.

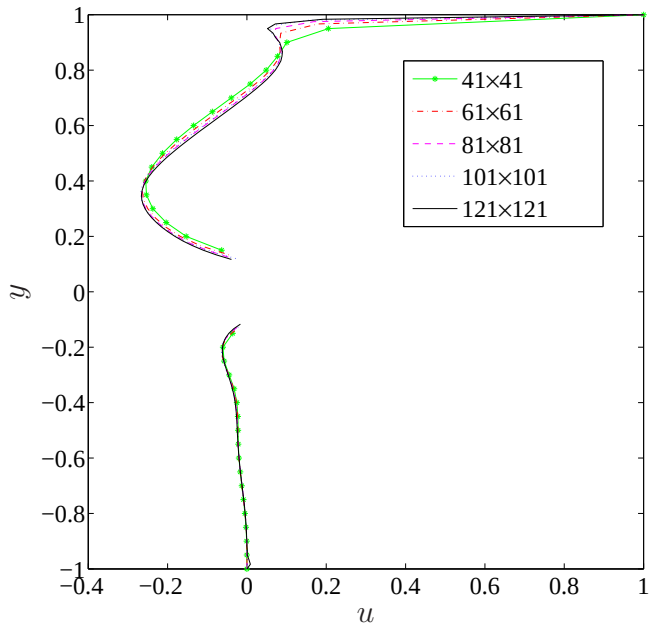


Figure 5.13: Lid-driven cavity flow with multiple solid bodies: The effect of the grid size on the u-velocity profile along the diagonal $x = y$. The curves are discontinuous due to the presence of a circular body on the diagonal around $x = y = 0$.

5.4.4 Flow between a rotating circular and a fixed square cylinder

Consider a flow in a concentric annulus between a square cylinder $\Omega = [-2, 2] \times [-2, 2]$ and a circular cylinder of $R = 1$ (Figure 5.14). The inner cylinder rotates with an angular velocity $\omega = 1$ while the outer cylinder is stationary. This problem is taken from Lewis (1979). The boundary conditions are as follows.

$$\mathbf{u} = 0 \quad \text{on} \quad x = \pm 2,\ y = \pm 2, \tag{5.54}$$

$$u = -\omega y,\ v = \omega x \quad \text{on} \quad R = 1. \tag{5.55}$$

The calculations are carried out on a set of uniform grids, namely $N \in \{61 \times 61,\ 81 \times 81,\ 101 \times 101,\ 121 \times 121,\ 131 \times 131,\ 141 \times 141\}$ and a set of time steps, namely $\Delta t \in \{0.001,\ 0.0005,\ 0.00025,\ 0.0001\}$ for various values of the Reynolds number, namely $Re \in \{1,\ 100,\ 200,\ 500,\ 1000,\ 1400\}$. Smaller time step is utilised for denser grid and higher Reynolds number. The maximum values of the stream function and vorticity (ψ_{max} and ζ_{max}), the values of the stream function on the circular cylinder (ψ_c) and minimum values of the stream function (ψ_{min}) are presented in Table 5.1. The present results, convergent at a grid of 131×131, agree well with those reported in Lewis (1979) using a

161 × 161 grid.

The streamlines of the flow field using a grid of 131 × 131 is shown in Figure 5.15, in which the vortices at the corners are well captured and in agreement with the results of Lewis (1979).

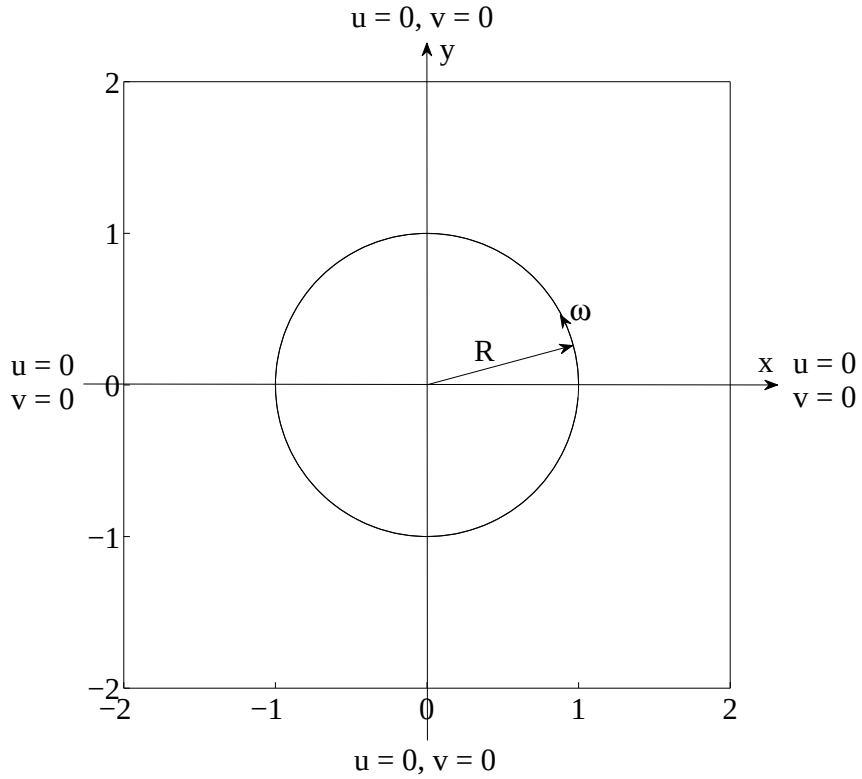


Figure 5.14: Flow between a rotating circular and a fixed square cylinder: Geometry and boundary conditions.

Table 5.1: Flow between rotating circular and fixed square cylinders: Maximum values of the stream function (ψ_{max}) and vorticity (ζ_{max}), and values of the stream function on the circular cylinder (ψ_c) by the present method and FDM.

Re	Method	Grid	ψ_{min}	ψ_{max}	ζ_{max}	ψ_c
1	Present ($\mathbf{u}-p$)	61×61	-1.4203E-4	0.4785	1.0472	0.4785
		81×81	-1.3415E-4	0.4699	1.0233	0.4699
		101×101	-1.3588E-4	0.4712	1.0325	0.4712
		121×121	-1.3523E-4	0.4701	1.0249	0.4701
		131×131	-1.3478E-4	0.4695	1.0216	0.4695
		141×141	-1.3472E-4	0.4691	1.0209	0.4691
	FDM ($\psi-\zeta$) (Lewis, 1979)	161×161	-1.4000E-4	0.4656	1.0186	0.4656
100	Present ($\mathbf{u}-p$)	61×61	-1.2527E-3	0.4808	1.2433	0.4808
		81×81	-1.1994E-3	0.4747	1.2374	0.4747
		101×101	-1.1830E-3	0.4711	1.2265	0.4711
		121×121	-1.1788E-3	0.4679	1.2216	0.4679
		131×131	-1.1760E-3	0.4658	1.2198	0.4658
		141×141	-1.1758E-3	0.4652	1.2193	0.4652
	FDM ($\psi-\zeta$) (Lewis, 1979)	161×161	—	—	—	0.4577
200	Present ($\mathbf{u}-p$)	61×61	-2.0812E-3	0.4777	1.3110	0.4777
		81×81	-1.9988E-3	0.4715	1.3095	0.4715
		101×101	-1.9882E-3	0.4678	1.2992	0.4678
		121×121	-1.9796E-3	0.4652	1.2916	0.4652
		131×131	-1.9721E-3	0.4629	1.2897	0.4629
		141×141	-1.9716E-3	0.4625	1.2893	0.4625
	FDM ($\psi-\zeta$) (Lewis, 1979)	161×161	—	0.4539	1.2559	0.4539
500	Present ($\mathbf{u}-p$)	61×61	-3.0170E-3	0.4738	1.3957	0.4738
		81×81	-2.9114E-3	0.4676	1.4143	0.4676
		101×101	-2.8354E-3	0.4599	1.3732	0.4599
		121×121	-2.7762E-3	0.4526	1.3719	0.4526
		131×131	-2.7298E-3	0.4512	1.3708	0.4512
		141×141	-2.7291E-3	0.4511	1.3702	0.4511
	FDM ($\psi-\zeta$) (Lewis, 1979)	161×161	-2.7100E-3	0.4465	1.3430	0.4465
1000	Present ($\mathbf{u}-p$)	61×61	-3.2525E-3	0.4714	1.4321	0.4714
		81×81	-3.1714E-3	0.4648	1.4899	0.4648
		101×101	-3.1014E-3	0.4502	1.4264	0.4502
		121×121	-3.0326E-3	0.4429	1.3925	0.4429
		131×131	-3.0048E-3	0.4397	1.3767	0.4397
		141×141	-3.0042E-3	0.4394	1.3761	0.4394
	FDM ($\psi-\zeta$) (Lewis, 1979)	161×161	—	—	—	0.4375
1400	Present ($\mathbf{u}-p$)	61×61	-3.2105E-3	0.4707	1.4329	0.4707
		81×81	-3.1543E-3	0.4637	1.5223	0.4637
		101×101	-3.0785E-3	0.4461	1.4279	0.4461
		121×121	-3.0241E-3	0.4379	1.4117	0.4379
		131×131	-2.9953E-3	0.4324	1.4026	0.4324
		141×141	-2.9947E-3	0.4320	1.4024	0.4320
	FDM ($\psi-\zeta$) (Lewis, 1979)	161×161	—	—	—	0.4314

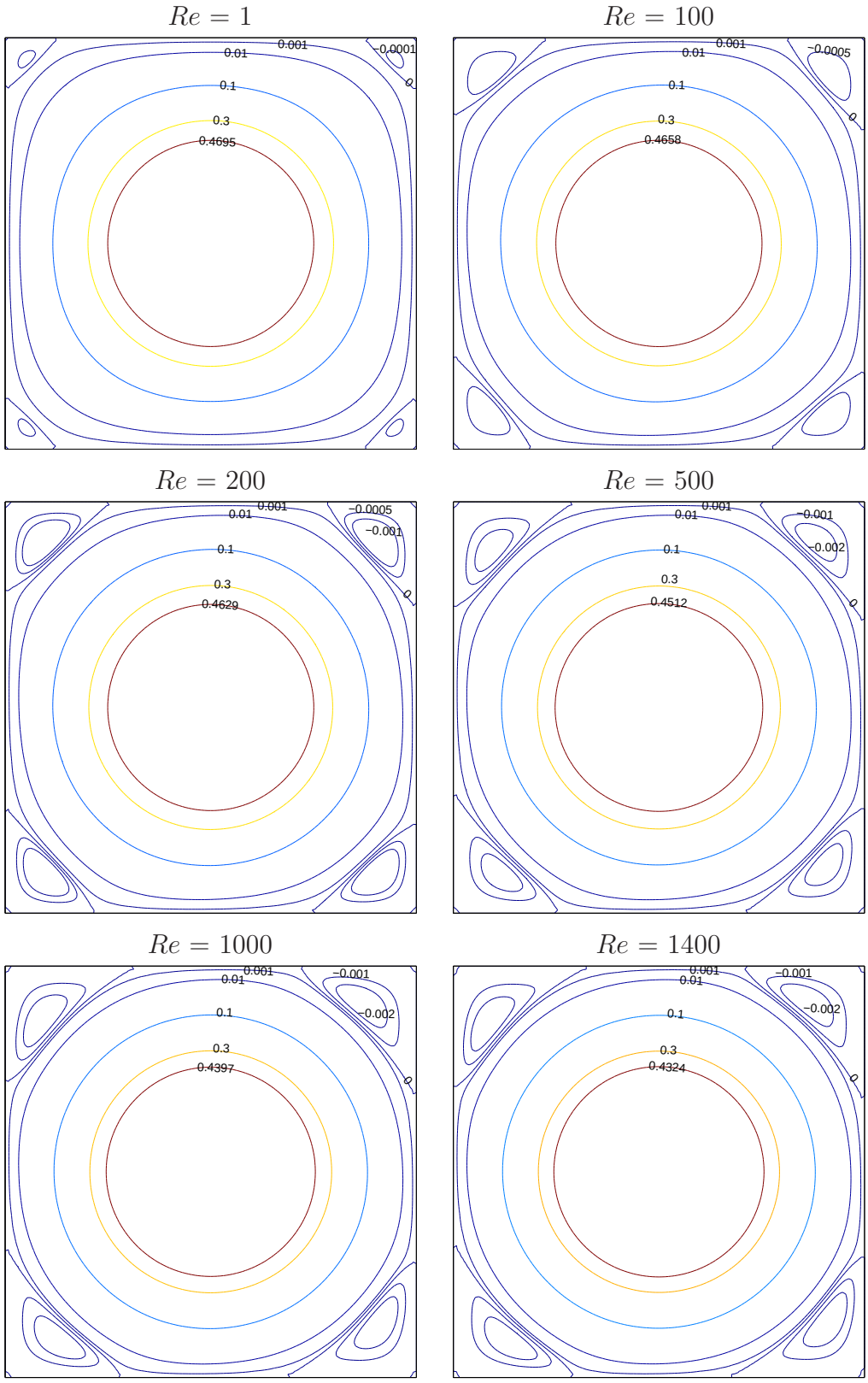


Figure 5.15: Flow between a rotating circular and a fixed square cylinder: Streamlines of the flow for several Reynolds numbers using a grid of 131×131. The contour values used here are taken to be the same as those in Lewis (1979), except those on the circular boundary.

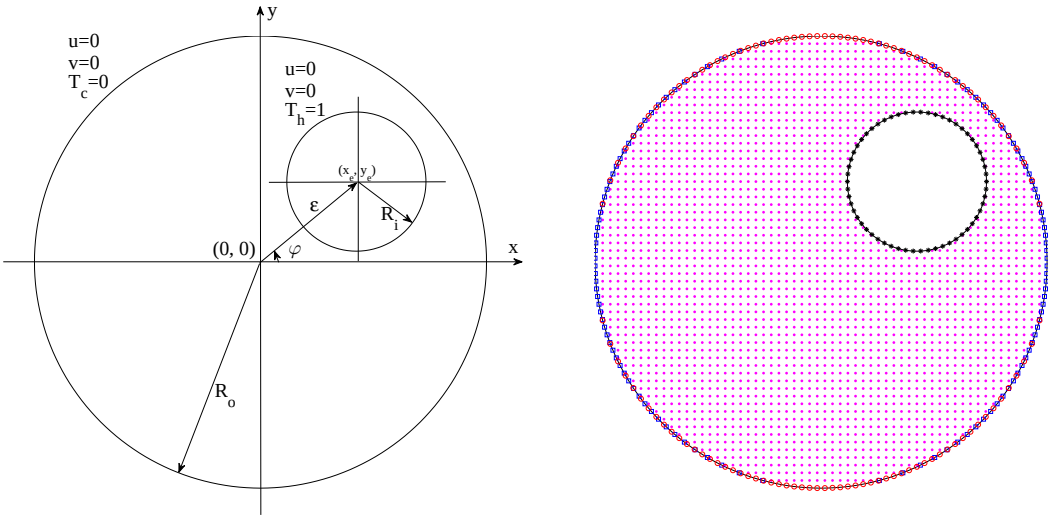
5.4.5 Natural convection in an eccentric annulus between two circular cylinders

The geometry of this problem can be defined through the following parameters: the eccentricity ε, angular position φ, radius of the outer cylinder R_o and radius of the inner cylinder R_i (Figure 5.16). The inner and outer cylinders are heated ($T_h = 1$) and cooled ($T_c = 0$), respectively. Calculation is carried out for $Pr = 0.71$, $R_o/R_i = 2.6$ and $Ra = 10^4$ using a set of uniform grids, namely $\{60 \times 60,\ 70 \times 70,\ 80 \times 80,\ 90 \times 90,\ 100 \times 100\}$ and a set of time steps, namely $\Delta t \in \{0.001,\ 0.0005,\ 0.00025,\ 0.0001\}$. Smaller time steps are used for higher grid densities. A distribution of nodes and the boundary conditions are shown in Figure 5.16.

For symmetrical flows, where the centres of the inner and outer cylinders lie on the vertical symmetrical axis, several values of eccentricity, namely $\varepsilon \in \{0.25,\ 0.50,\ 0.75,\ 0.95\}$ and angular direction, namely $\varphi \in \{-90°,\ 90°\}$ are considered. Table 5.2 compares the maximum value of the stream function (ψ_{max}) between the present scheme, one-dimensional integrated radial basis function (1D-IRBF) scheme (Le-Cao et al., 2011) and differential quadrature method (DQM) (Shu et al., 2002). It can be seen that good agreement is achieved. The present solutions are convergent at the grid of 90×90.

For unsymmetrical flows, the stream function at the inner wall (ψ_w) is no longer zero and its value varies with the location of the inner cylinder. Values of the eccentricity and angular direction are taken as $\{0.25,\ 0.50,\ 0.75\}$ and $\{-45°,\ 0°,\ 45°\}$, respectively. In Table 5.3, values of ψ_w are presented and agree satisfactorily with those obtained by the 1D-IRBF scheme (Le-Cao et al., 2011), DQM (Shu et al., 2002) and domain-free discretisation method (DFD) (Shu and Wu, 2002). It is noted that the present governing equations (5.1)-(5.3) are different from those used in Shu et al. (2002) and Shu and Wu (2002) by a factor $\sqrt{PrRa}$. Therefore, to facilitate a comparison, our results in the table, which are computed in the average sense from the values of ψ at the Lagrangian nodes, are multiplied by this factor. The present solutions are convergent at the grid of 90×90.

Figures 5.17-5.18 and Figures 5.19-5.21 show the isotherms and streamlines of the flow for symmetrical and unsymmetrical flows, respectively, where several combinations of eccentricity and angular direction are considered. Each plot contains 22 contour lines whose levels vary linearly from the minimum to maximum values. All plots look very feasible when compared with those obtained by the 1D-IRBF scheme (Le-Cao et al., 2011), DQM (Shu et al., 2002) and (DFD) (Shu and Wu, 2002).



(a) (b)

Figure 5.16: Natural convection in eccentric circular-circular annulus: Geometry and boundary conditions (a) and distribution of nodes (b) (Eulerian nodes inside the annulus and on the outer boundary, Lagrangian nodes on the inner boundary with a grid of 60×60).

Table 5.2: Natural convection in eccentric circular-circular annulus, symmetrical flows: the maximum values of the stream function (ψ_{max}) for two special cases $\varphi \in \{-90°,\ 90°\}$ by the present and some other numerical schemes.

φ	ε	ψ_{max}						
		DQM[a]	1D-IRBF[b]	DFIB-CIRBF[c]				
				60×60	70×70	80×80	90×90	100×100
$-90°$	0.25	15.50	15.71	15.26	15.30	15.35	15.36	15.36
	0.50	18.32	18.50	18.10	18.39	18.44	18.47	18.47
	0.75	20.62	20.72	20.10	20.41	20.47	20.49	20.49
	0.95	22.16	22.19	21.91	22.35	22.44	22.49	22.50
$90°$	0.25	11.13	11.26	11.07	11.11	11.13	11.14	11.14
	0.50	9.55	9.64	9.51	9.55	9.57	9.58	9.58
	0.75	8.12	8.25	8.17	8.18	8.20	8.21	8.21
	0.95	7.17	7.28	7.21	7.23	7.24	7.24	7.24

[a] (Shu et al., 2002)
[b] (Le-Cao et al., 2011)
[c] Present

5.5 Concluding remarks

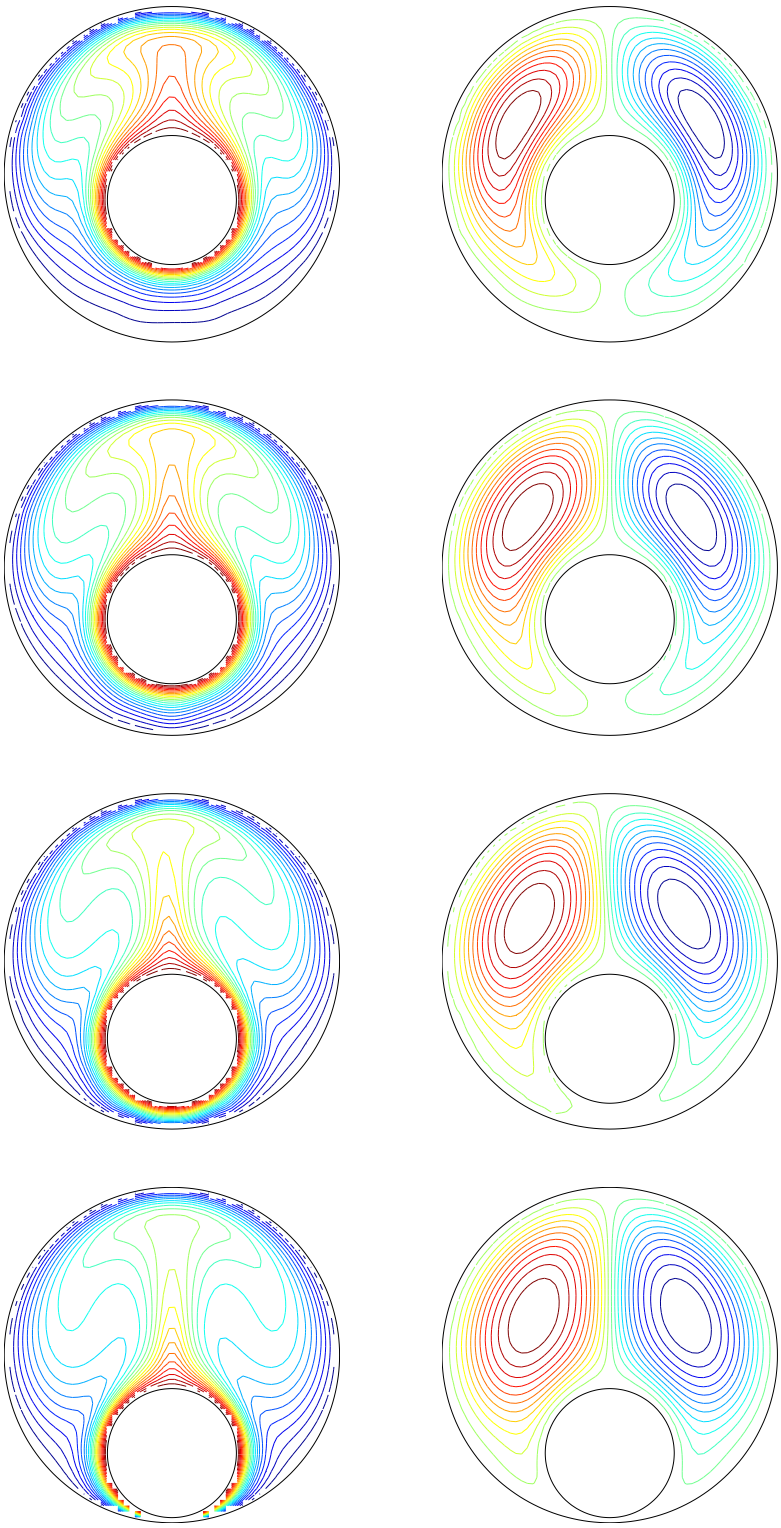
In this chapter, we introduce compact integrated RBF approximations into the immersed boundary and point-collocation framework to simulate viscous flows in two dimensions. The direct forcing immersed boundary method is utilised for the handling of inner boundaries, while high-order approximation

Table 5.3: Natural convection in eccentric circular-circular annulus, unsymmetrical flows: the stream function values at the inner cylinders (ψ_w) for $\varepsilon \in \{0.25,\ 0.50,\ 0.75\}$ and $\varphi \in \{-45°,\ 0°,\ 45°\}$ by the present and some other numerical schemes.

φ	ε	ψ_w							
		DFD[a]	DQM[b]	1D-IRBF[c]	DFIB-CIRBF[d]				
					60×60	70×70	80×80	90×90	100×100
$-45°$	0.25	0.51	0.51	0.48	0.46	0.48	0.49	0.50	0.50
	0.50	0.77	0.92	0.80	0.80	0.81	0.82	0.82	0.82
	0.75	0.77	0.99	1.05	1.10	1.13	1.14	1.15	1.15
$0°$	0.25	0.72	0.72	0.60	0.67	0.68	0.68	0.68	0.68
	0.50	1.10	1.15	1.28	1.07	1.07	1.07	1.07	1.07
	0.75	1.26	1.30	1.18	1.25	1.29	1.31	1.32	1.32
$45°$	0.25	0.54	0.52	0.52	0.56	0.57	0.57	0.57	0.57
	0.50	1.29	1.31	1.25	1.23	1.23	1.23	1.23	1.23
	0.75	1.09	1.07	1.01	0.98	1.01	1.02	1.03	1.03

[a] (Shu and Wu, 2002)
[b] (Shu et al., 2002)
[c] (Le-Cao et al., 2011)
[d] Present

schemes (Adams-Bashforth/Crank-Nicolson and compact 3-point IRBFs) are employed to represent temporal and spatial derivatives. The proposed method is verified successfully with a series of problems of fluid flow in multiply-connected domains. Very good results are obtained using relatively coarse Cartesian grids.



(a) (b)

Figure 5.17: Natural convection in an eccentric circular-circular annulus, symmetrical flows: Contour plots for the temperature (a) and stream function (b) fields for $\varepsilon \in \{0.25,\ 0.50,\ 0.75,\ 0.95\}$ (from top to bottom) and $\varphi = -90°$. Each plot contains 22 contour lines whose levels vary linearly from the minimum to maximum values.

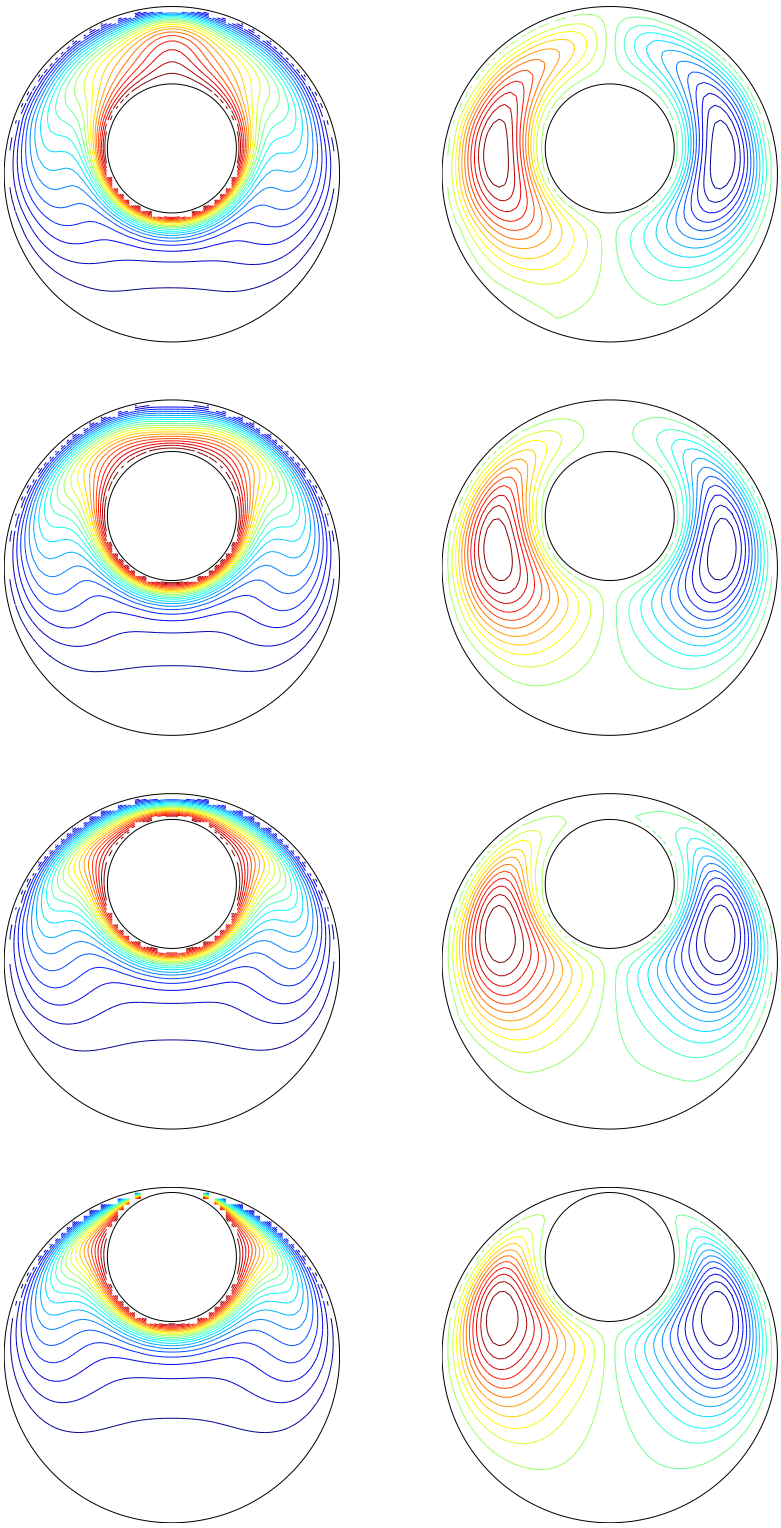
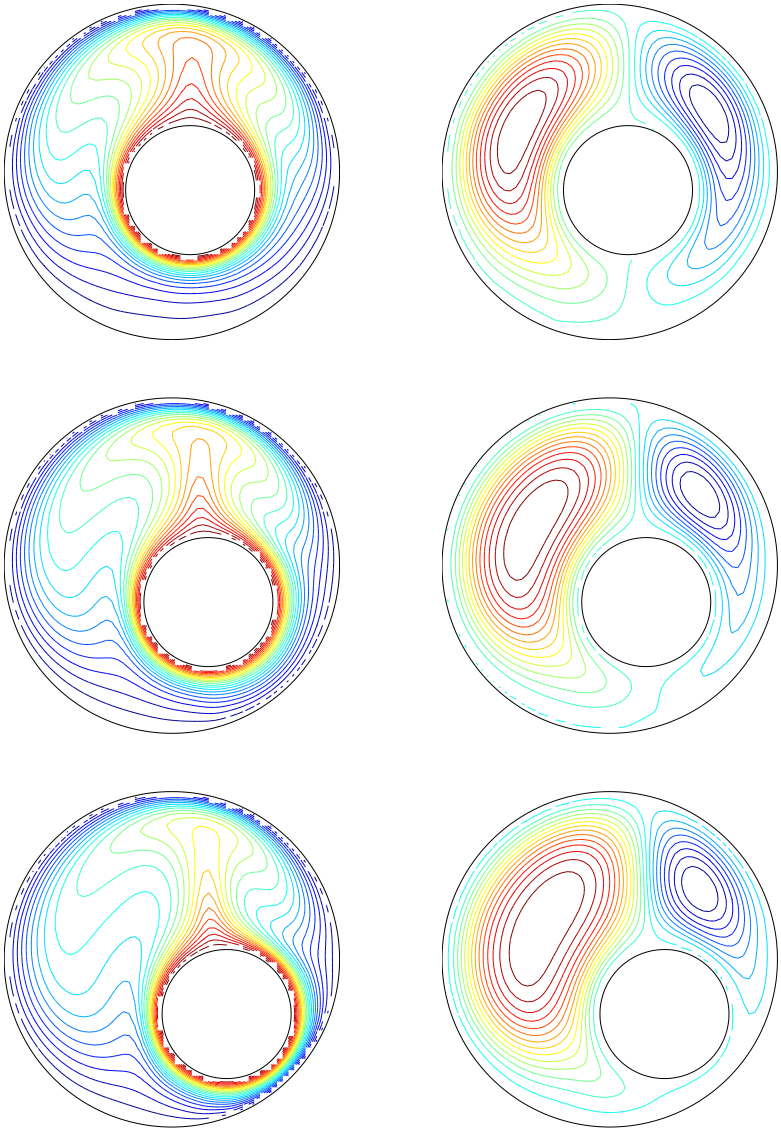
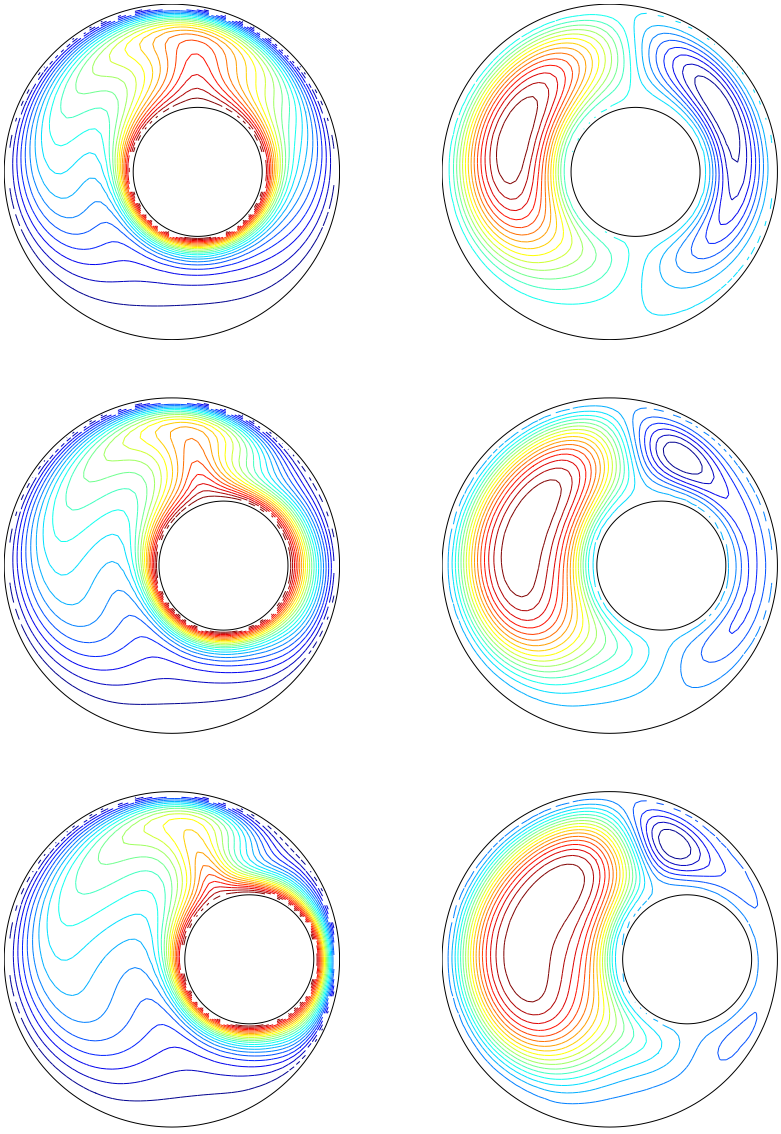


Figure 5.18: Natural convection in an eccentric circular-circular annulus, symmetrical flows: Contour plots for the temperature (a) and stream function (b) fields for $\varepsilon \in \{0.25,\ 0.50,\ 0.75,\ 0.95\}$ (from top to bottom) and $\varphi = 90°$. Each plot contains 22 contour lines whose levels vary linearly from the minimum to maximum values.



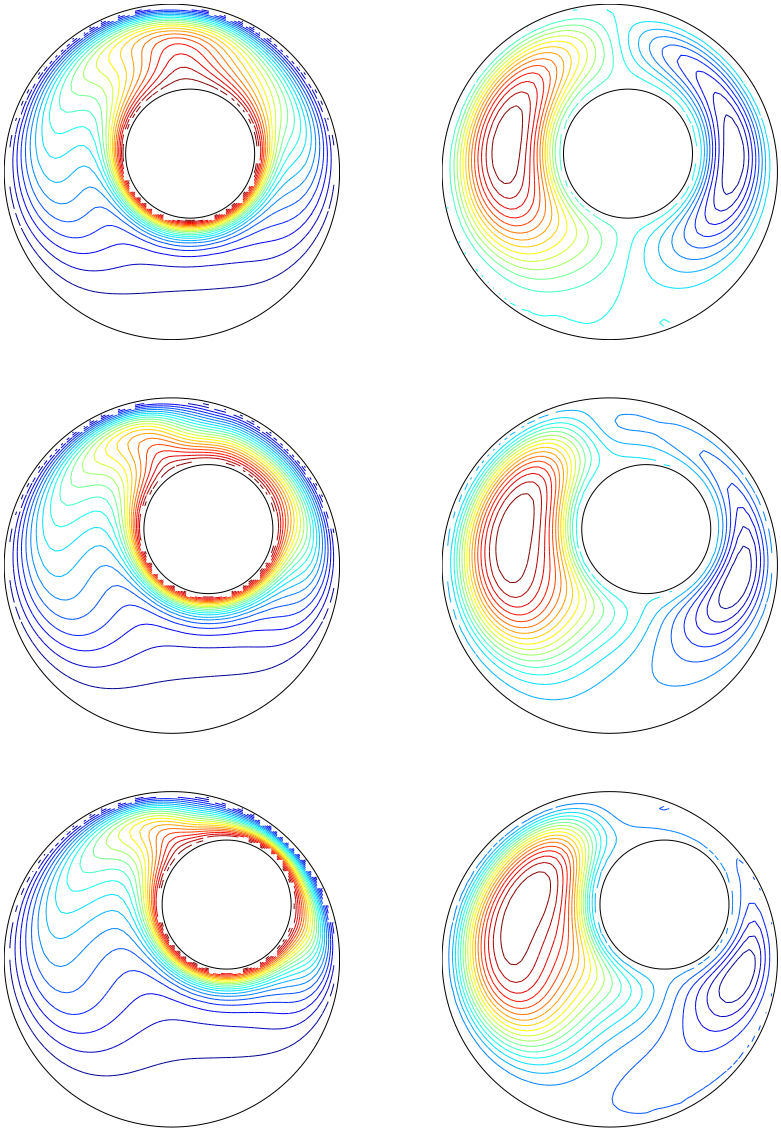
(a) (b)

Figure 5.19: Natural convection in an eccentric circular-circular annulus, unsymmetrical flows: Contour plots for the temperature (a) and stream function (b) fields for $\varepsilon \in \{0.25,\ 0.50,\ 0.75\}$ (from top to bottom) and $\varphi = -45°$. Each plot contains 22 contour lines whose levels vary linearly from the minimum to maximum values.



(a) (b)

Figure 5.20: Natural convection in an eccentric circular-circular annulus, unsymmetrical flows: Contour plots for the temperature (a) and stream function (b) fields for $\varepsilon \in \{0.25,\ 0.50,\ 0.75\}$ (from top to bottom) and $\varphi = 0°$. Each plot contains 22 contour lines whose levels vary linearly from the minimum to maximum values.



(a) (b)

Figure 5.21: Natural convection in an eccentric circular-circular annulus, unsymmetrical flows: Contour plots for the temperature (a) and stream function (b) fields for $\varepsilon \in \{0.25,\ 0.50,\ 0.75\}$ (from top to bottom) and $\varphi = 45°$. Each plot contains 22 contour lines whose levels vary linearly from the minimum to maximum values.

Chapter 6

A DFIB-CIRBF approach for fluid-solid interactions in particulate fluids

In this chapter, the DFIB-CIRBF method presented in chapter 5 is further developed to model fluid-particle interaction in particulate fluids. The fluid flow is computed by solving the Navier-Stokes equations in an Eulerian frame of reference (FOR) with fractional-step/projection method based on CIRBF approximations whereas the rigid particles are allowed to move freely through the computational domain and their motions are computed by solving the Newton-Euler equations in a Lagrangian FOR. The interaction between fluid and particles is taken into account by mean of direct forcing immersed boundary method. The particle-particle and particle-wall interactions are handled by available collision models reported in the literature. The capability of the present method for particulate flows is depicted by simulating several problems, including Taylor-Green vortices, induced flow by an oscillating circular cylinder, single particle sedimentation and drafting-kissing-tumbling behaviour of two settling particles. Numerical experiments show that the present method provides an accurate approach to directly simulate particulate flows.

6.1 Introduction

The numerical simulation of flows of a viscous liquid containing several small particles (i.e. multiphase flows or particulate flows) is an important area of research and remains a challenge in many engineering applications. These types of problems, depending on the application area, include sedimenting and fluidised suspensions, lubricated transport, hydraulic fracturing of reservoirs, slurries and

so on. It is very desirable to understand the phenomenon of fluid-particle interaction through either experimental measurements or numerical simulations for engineering analysis and design purposes.

In comparison with experimental measurements, numerical simulations have noticeable advantages because relevant parameters can be studied separately in investigating the fluid-particle interactions of particulate flows. But, the results obtained by the numerical simulations must be believable. Therefore, developing numerical methods with high accuracy is very important to simulate particulate flows effectively. At the present, in continuum mechanics, direct numerical simulation appears to offer the highest-resolution. There are two types of DNS approaches for the particulate flows currently. In the first approach, the fluid is simulated using DNS but the particle is assumed to be a massless/volumeless point and the main force acted on it is modeled by the Stokes drag force. This approach is valid only when the particle scale is smaller than the grid scale and the suspension is dilute. This approach is known as traditional DNS and it has been extensively applied in the past decades (Luo et al., 2006; Marchioli et al., 2003; Soltani et al., 1998; Squires and Eaton, 1990). In the second approach, the actual physical characteristics of particle are considered and the hydrodynamic force acted on it is computed by integrating the fluid stress tensor over the particle surface. The flow around the particle such as the wake can be captured well and hence, the particulate flows are fully resolved in this approach. In this chapter, we are interested in the latter approach because it is truly DNS of particulate flows.

Several DNS methods for the particulate flows have been developed so far. Hu et al. (1992), Hu et al. (2001) developed a FEM based on unstructured grids to simulate the motion of a large number of rigid objects in Newtonian and viscoelastic fluids. This method is based on an ALE technique. Both fluid and solid equations of motion are incorporated into a single coupled variational equation, and the hydrodynamic forces and torques acting on the particles are eliminated in the formulation. The nodes on the particle surface are allowed to move with the particle, while the nodes in the interior of the fluid are smoothly updated by solving a Laplace's equation. At each time step, a new mesh is generated when the old one becomes too distorted and the flow field is projected onto the new mesh. The positions of the particles and grid nodes are updated explicitly, while the velocity of the fluid and particles are determined implicitly. In 2D problems, the remeshing of the body-fitted meshes in this method can be done by available grid generation software. But in 3D problems, efficiently generating the body-fitted mesh is very difficult and it has not been solved in a satisfactory manner yet. Therefore, many researchers have tried to propose methods that do not require remeshing. For example, Glowinski et al. (1999), Glowinski et al. (2001) proposed a distributed Lagrange multiplier/fictitious domain method (DLM/FDM) for the DNS of suspensions containing a large number of rigid particles. In this method, the whole domain is assumed to be fluid and the volumes corresponding to particles are constrained to move as rigid bodies.

The fluid-particle motion is treated implicitly using a combined weak formulation in which the mutual forces are canceled. This formulation permits the use of the fixed structured grid, thus eliminating the requirement for remeshing the domain. Later, Wan and Turek (2006), Wan and Turek (2007a) presented a similar, but different multigrid FBM for the simulation of particulate flows. The method is based on a fixed FEM background grid. The motion of the rigid particles is modeled by the Newton-Euler equations. The boundary conditions applied at the fluid-particle interface are considered as an additional constraint to the governing Navier-Stokes equations. Hence, the fluid domain is extended into the whole domain covering both fluid and particle volumes. The advantage of this FDM over the previous ALE-FEM is that it allows a fixed grid to be used, eliminating remeshing procedure, and it can be handled independently from the flow features. As an improvement on the DLM/FDM formulation of Glowinski et al. (1999), Patankar et al. (2000); Patankar and Sharma (2005) presented a new DLM formulation (both strong and weak forms) for the simulation of particulate flows where a simple and efficient method of computing the particle velocities was derived from the conservation of momenta of a rigid particle. The efficiency of the method is based on a simple projection scheme for rigid motion and makes it suitable for handling a large number of particles and for 3D applications. A similar technique, which is also non-Lagrange multiplier based, was considered in Veeramani et al. (2007). Later, Blasco et al. (2009) used the fast computational technique of Patankar and Sharma (2005) and developed a method based on FDM with parallel computation.

Other known approaches for solving the particulate flows are based on Lattice Boltzmann method (LBM) in combination with IBM. The first attempts were made by Feng and Michaelides (2004), Feng and Michaelides (2005). The simulation of 2D and 3D particle sedimentation were carried out. Niu et al. (2006) put forward a momentum exchange-based IB-LBM to simulate flows around fixed and moving particles. The IBM, originally developed by Peskin (1972) as reviewed in chapter 5, has attracted considerable interest in the last few years (Mittal and Iaccarino, 2005). To compute the interactive force between the fluid and solids, Goldstein et al. (1993) proposed a feedback forcing approach to iteratively determine the magnitude of the force required to obtain a desired velocity on fluid-solid interface. Saiki and Biringen (1996) successfully implemented this feedback forcing approach to compute the flow past a circular cylinder with virtual boundary method. But the feedback forcing approach induces oscillations and the computational time step is restricted for numerical stability. In order to overcome this drawback, Fadlun et al. (2000) proposed a different approach, so-called direct forcing, to evaluate the interactive force. An advantage of the direct over the feedback forcing approach is that the DF approach does not require a smaller computational time step for numerical stability. To transfer the quantities smoothly between Eulerian and Lagrangian nodes and avoid strong restrictions of the time step, Uhlmann (2005) proposed an improved method to incorporate the regularised delta function approach into a direct formulation of the fluid-solid interactive force. Wang et al. (2008) pro-

posed an explicit multi-direct forcing approach based on the Peskin's original IBM to obtain better satisfaction of the non-slip boundary condition at the fluid-solid interface than that of the original DF approach. Recently, Ji et al. (2012) has proposed an iterative IBM in which the body force updating is incorporated into the pressure iterations for the 2D and 3D numerical simulations of laminar and turbulent flows.

In this chapter, the DFIB-CIRBF method presented in chapter 5 is further developed to model fluid-particle interaction in particulate fluids. A series of test problems, including Taylor-Green vortices, induced flow by an oscillating circular cylinder, single particle sedimentation and drafting-kissing-tumbling behaviour of two settling particles are carried out to depict the capability of the present method for particulate flow problems. The remainder of the chapter is organised as follows. Section 6.2 outlines the equations which govern the phenomenon of fluid-particle interaction and mathematical equations for the DF approach in IBM. Section 6.3 provides detailed descriptions of the numerical formulation including the temporal and spatial discretisations in the governing equations. In Section 6.4, numerical results are presented and compared with the analytic solutions and some published approximate results, where appropriate. Finally, Section 6.5 concludes the chapter.

6.2 Mathematical formulation

In this section, we describe the equations which govern the motion of rigid particles immersed in a Newtonian fluid in two dimensions. In IBM, the existence of the particle is represented by its effect on the fluid and this effect is enforced by introducing a fluid body force density into the momentum equations as reviewed in chapter 5. Let us consider the entire domain Ω including the fluid region Ω_f and solid region Ω_s occupied by the N_p particles P_k $\left(\Omega_s = \bigcup_{k=1}^{N_p} P_k\right)$ as shown in Figure 6.1. The outer boundary of the entire domain is Γ and the boundary surface of kth particle P_k is ∂P_k. The particle boundary surface is represented by the Lagrangian parametric coordinate $\mathbf{s}$ and the flow domain is represented by the Eulerian coordinate $\mathbf{x}$. Hence, any position on the particle surface may be written as $\mathbf{x} = \mathbf{X}(\mathbf{s}, t)$. The non-slip boundary condition is satisfied by enforcing the velocity at all particle boundaries to be equal to the velocity of the fluid at the same location

$$\mathbf{U}(\mathbf{s}, t) = \frac{\partial \mathbf{X}(\mathbf{s}, t)}{\partial t} = \mathbf{u}(\mathbf{X}(\mathbf{s}, t), t) = \mathbf{u}(\mathbf{x}, t). \tag{6.1}$$

In (6.1) and hereafter, we use upper-case and lower-case letters for the quantities evaluated at the Lagrangian nodes $\mathbf{X}$ and Eulerian nodes $\mathbf{x}$, respectively.

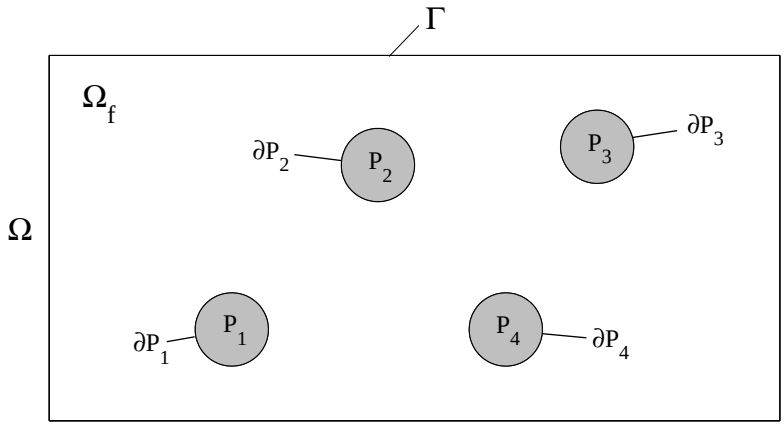


Figure 6.1: Configuration with several rigid particles and interstitial fluid domain.

6.2.1 Governing equations for fluid motion

In the context of the IBM, the suspended rigid particles can be replaced by an equivalent body force field $\mathbf{f}_I(\mathbf{x}, t)$. The whole domain is then considered to be fluid only whose motion is governed by the transient incompressible Navier-Stokes equations, together with appropriate initial and boundary conditions as follows (Wu and Shu, 2010; Peskin, 2002).

$$\nabla.\mathbf{u} = 0 \qquad \text{in} \quad \Omega,\ t \geq 0, \tag{6.2}$$

$$\rho_f \left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}.\nabla)\,\mathbf{u} \right) = -\nabla p + \mu \nabla^2 \mathbf{u} + \mathbf{f}_I \qquad \text{in} \quad \Omega,\ t \geq 0, \tag{6.3}$$

$$\mathbf{u}(x, y, 0) = \mathbf{u}_0(x, y) \qquad \text{in} \quad \Omega,\ t = 0, \tag{6.4}$$

$$\mathbf{u}(x, y, t) = \mathbf{u}_\Gamma(x, y, t) \qquad \text{on} \quad \Gamma,\ t \geq 0, \tag{6.5}$$

where ρ_f and μ are the density and the dynamic viscosity of fluid, respectively.

6.2.2 Direct forcing method

The readers are referred to section 5.3.1 for a detailed derivation of the interactive force $\mathbf{f}_I$ from non-dimensionalised Navier-Stokes equations. One can derive the interactive force $\mathbf{f}_I$ before non-dimensionalising equation (6.3) as follows.

Derivation of the interactive force

Dividing (6.3) by ρ_f, one obtains

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}.\nabla)\,\mathbf{u} = -\frac{1}{\rho_f}\nabla p + \nu\nabla^2\mathbf{u} + \frac{1}{\rho_f}\mathbf{f}_I, \tag{6.6}$$

where $\nu = \mu/\rho_f$ is the kinematic viscosity of fluid.

The temporal discretisation of the momentum equation (6.6) is written in the form

$$\frac{\mathbf{u}^n - \mathbf{u}^{n-1}}{\Delta t} = \mathbf{rhs}^{n-1/2} + \frac{1}{\rho_f}\mathbf{f}_I^{n-1/2}, \tag{6.7}$$

where the convection, pressure and diffusion terms at a time $t^{n-1/2}$ are lumped together in $\mathbf{rhs}^{n-1/2}$.

The interactive force term yielding the desired velocity $\mathbf{u}^{(d)}$ can thus be defined as (Fadlun et al., 2000)

$$\mathbf{f}_I^{n-1/2} = \rho_f \left(\frac{\mathbf{u}^{(d),n} - \mathbf{u}^{n-1}}{\Delta t} - \mathbf{rhs}^{n-1/2} \right), \tag{6.8}$$

at some selected nodes (and zeros elsewhere). The corresponding interactive force at the Lagrangian nodes on the particle surfaces will be

$$\mathbf{F}_I^{n-1/2} = \rho_f \left(\frac{\mathbf{U}^{(d),n} - \mathbf{U}^{n-1}}{\Delta t} - \mathbf{RHS}^{n-1/2} \right). \tag{6.9}$$

The desired velocity at a node on the fluid-particle interface in (6.9) is simply computed from the rigid-body motion of the particles

$$\mathbf{U}^{(d),n}(\mathbf{X}) = \mathbf{U}_p^n + \boldsymbol{\omega}_p^n \times (\mathbf{X}^n - \mathbf{X}_p^n), \tag{6.10}$$

where $\mathbf{U}_p = (U_p, V_p)^T$ and $\mathbf{X}_p = (X_p, Y_p)^T$ are the translational velocity and the position vectors of the particle centre, respectively, and $\boldsymbol{\omega}_p$ is the rotational velocity of the particle - all is defined in the Cartesian coordinate system.

When the interactive force is absent, equation (6.9) leads to a preliminary velocity $\widetilde{\mathbf{U}}^n$ given by equation (5.14). Its Eulerian counterpart is $\widetilde{\mathbf{u}}^n$ given by equation (5.15).

In this chapter, the term $\mathbf{rhs}^{n-1/2}$ in (5.15) is computed explicitly as (Butcher,

2003)

$$\mathbf{rhs}^{n-1/2} = -\frac{1}{\rho_f}\left[\frac{3}{2}\nabla p^{n-1} - \frac{1}{2}\nabla p^{n-2}\right] - \left[\frac{3}{2}(\mathbf{u}^{n-1}.\nabla)\mathbf{u}^{n-1} - \frac{1}{2}(\mathbf{u}^{n-2}.\nabla)\mathbf{u}^{n-2}\right] + \nu\left[\frac{3}{2}\nabla^2\mathbf{u}^{n-1} - \frac{1}{2}\nabla^2\mathbf{u}^{n-2}\right]. \quad (6.11)$$

Hence, the interactive force at Lagrangian nodes on the particle surfaces is computed now as

$$\mathbf{F}_I^{n-1/2} = \rho_f \frac{\mathbf{U}^{(d),n} - \widetilde{\mathbf{U}}^n}{\Delta t}. \quad (6.12)$$

In order to complete the evaluation of the interactive force term in (6.7), a mechanism for transferring the preliminary velocity $(\widetilde{\mathbf{u}}^n, \widetilde{\mathbf{U}}^n)$ and the forces $(\mathbf{F}_I^{n-1/2}, \mathbf{f}_I^{n-1/2})$ between the two Eulerian and Lagrangian node system is required. For brevity, from here on we use "Lagrangian nodes" for "Lagrangian nodes on the particle surfaces".

Transfer of quantities between Eulerian and Lagrangian nodes

The fluid body interactive force at Eulerian nodes is computed from the particle surface interactive force at Lagrangian nodes, and the particle surface velocities at Lagrangian nodes is computed from the fluid velocity at the Eulerian nodes as follows (Feng and Michaelides, 2005).

$$\mathbf{f}_I(\mathbf{x}, t) = \oint_{\partial P} \mathbf{F}_I(\mathbf{s}, t)\, \delta(\mathbf{x} - \mathbf{X}(\mathbf{s}, t))\, dV, \quad (6.13)$$

$$\mathbf{U}(\mathbf{s}, t) = \int_{\Omega} \mathbf{u}(\mathbf{x}, t)\, \delta(\mathbf{x} - \mathbf{X}(\mathbf{s}, t))\, dv, \quad (6.14)$$

where $\delta(\mathbf{x} - \mathbf{X}(\mathbf{s}, t))$ is the Dirac delta function, dV and dv are the infinitesimal volumes in Lagrangian parametric and Eulerian coordinates, respectively.

In this chapter, we employ the class of regularised delta functions (Peskin, 2002) as defined in (5.18). It is preferable to write (6.13)-(6.14) in a discrete form. For convenience, an even number of N_L Lagrangian nodes are distributed over the fluid-particle interface on each particle and their locations are defined by $\mathbf{X}_l^k = \left(X_l^k, Y_l^k\right)^T$, where

$$\mathbf{X}_l^k \in \partial P_k, \quad \forall 1 \le l \le N_L,\ 1 \le k \le N_p. \quad (6.15)$$

For the sake of simplicity, all particles are assumed to have the same shape and size. Therefore, the number of N_L Lagrangian nodes is the same for all particles. Equations (6.13)-(6.14) is rewritten in the discrete form as follows

(Uhlmann, 2005).

$$\widetilde{\mathbf{U}}(\mathbf{X}_l^k) = \sum_{\mathbf{x}\in\Omega} \widetilde{\mathbf{u}}(\mathbf{x})\delta_h(\mathbf{x}-\mathbf{X}_l^k)h^2, \quad \forall 1 \le l \le N_L,\ 1 \le k \le N_p. \tag{6.16}$$

$$\mathbf{f}_I(\mathbf{x}) = \sum_{k=1}^{N_p}\sum_{l=1}^{N_L} \mathbf{F}_I(\mathbf{X}_l^k)\delta_h(\mathbf{x}-\mathbf{X}_l^k)\Delta V_l^k, \quad \forall \mathbf{x} \in \Omega, \tag{6.17}$$

A family of Peskin's delta functions are summarised in section 5.3.1. The readers are referred to that section for various common delta functions, e.g. 2-point hat function, 3-point discrete delta function and 4-point piecewise function. In this chapter, we also employ the 3-point discrete delta function $\delta_{3h}(r)$ (Roma et al., 1999) as in chapter 5.

6.2.3 Governing equations for particle motion

Based on above DF approach and IBM, the hydrodynamic force $\mathbf{F}^H = \left(F_x^H, F_y^H\right)^T$ and torque $\mathbf{T}^H$ exerted on a moving particle immersed in the flowing fluid can be expressed as (Wang et al., 2008)

$$\mathbf{F}^H = -\int_1^{N_p}\int_1^{N_L} \mathbf{F}_{Il}^k\left(\mathbf{X}_l^k\right) dV_l^k = -\int_\Omega \mathbf{f}_I(\mathbf{x})dv = -\sum_{\mathbf{x}\in\Omega}\mathbf{f}_I(\mathbf{x})h^2, \tag{6.18}$$

$$\begin{aligned}\mathbf{T}^H &= -\int_1^{N_p}\int_1^{N_L} (\mathbf{X}_l^k - \mathbf{X}_p^k) \times \mathbf{F}_{Il}^k(\mathbf{X}_l^k)dV_l^k \\ &= -\int_\Omega (\mathbf{x}-\mathbf{X}_p)\times\mathbf{f}_I(\mathbf{x})dv = -\sum_{\mathbf{x}\in\Omega}(\mathbf{x}-\mathbf{X}_p)\times\mathbf{f}_I(\mathbf{x})h^2,\end{aligned} \tag{6.19}$$

where $\mathbf{X}_p^k = \left(X_p^k, Y_p^k\right)^T$ is the centre coordinates of the kth particle. Then, the motion of a particle is governed by the following Newton-Euler equations (Wang et al., 2008; Blasco et al., 2009).

$$M_p\frac{d\mathbf{U}_p}{dt} = \mathbf{F}^H + (\rho_p - \rho_f)V_p\mathbf{g} + \mathbf{F}^{col}, \tag{6.20}$$

$$I_p\frac{d\boldsymbol{\omega}_p}{dt} = \mathbf{T}^H, \tag{6.21}$$

$$\frac{d\mathbf{X}_p}{dt} = \mathbf{U}_p, \tag{6.22}$$

$$\frac{d\boldsymbol{\theta}_p}{dt} = \boldsymbol{\omega}_p, \tag{6.23}$$

where ρ_p, V_p and I_p are the density, the volume and the moment of inertia of the particle, respectively; $M_p = \rho_p V_p$ the mass of a particle; $\boldsymbol{\theta}_p$ the angular position of the particle; $\mathbf{g} = (g_x, g_y)^T$ the gravitational acceleration vector; $\mathbf{F}^{col} = \left(F_x^{col}, F_y^{col}\right)^T$ the collision force vector acting on a particle by other particles or walls when they come close to each other. For 2D problems, the volume and the moment of inertia for a circular particle are $V_p = \pi R^2$ and $I_p = \frac{1}{2}\rho_p \pi R^4$, respectively, where R is the radius of the particle.

The discretisation forms of these governing equations are given as follows (Wang et al., 2008).

$$\mathbf{U}_p^n = \mathbf{U}_p^{n-1} + \frac{\mathbf{F}^{H,n-1} + \mathbf{F}^{col,n-1}}{M_p}\Delta t + \left(1 - \frac{1}{\rho_r}\right)\mathbf{g}\Delta t, \tag{6.24}$$

$$\boldsymbol{\omega}_p^n = \boldsymbol{\omega}_p^{n-1} + \frac{\mathbf{T}^{H,n-1}}{I_p}\Delta t, \tag{6.25}$$

$$\mathbf{X}_p^n = \mathbf{X}_p^{n-1} + \frac{\mathbf{U}_p^n + \mathbf{U}_p^{n-1}}{2}\Delta t, \tag{6.26}$$

$$\boldsymbol{\theta}_p^n = \boldsymbol{\theta}_p^{n-1} + \frac{\boldsymbol{\omega}_p^n + \boldsymbol{\omega}_p^{n-1}}{2}\Delta t, \tag{6.27}$$

where $\rho_r = \frac{\rho_p}{\rho_f}$ is the particle-fluid density ratio.

When spreading the effect of the interactive force from the Lagrangian nodes to the Eulerian nodes with DF method, the interactive force acted on the Lagrangian nodes containing the desired velocity $\mathbf{U}^{(d),n}$ is unknown at the current time level n. A simple way to deal with this problem is to apply a first-order explicit scheme with $\mathbf{U}^{(d),n-1}$ instead of $\mathbf{U}^{(d),n}$. This way has been employed in previous numerical simulations (Wang et al., 2008; Wan and Turek, 2007a; Uhlmann, 2005) and it is known as a "weak coupling" of fluid-particle equations (Uhlmann, 2005). We also employ this explicit scheme in the present work. The interactive force exerted at the Lagrangian nodes on the immersed boundary (i.e. equation (6.12)) is thus be changed into

$$\mathbf{F}_I^{n-1/2} = \rho_f \frac{\mathbf{U}^{(d),n} - \widetilde{\mathbf{U}}^n}{\Delta t} \approx \rho_f \frac{\mathbf{U}^{(d),n-1} - \widetilde{\mathbf{U}}^n}{\Delta t}, \tag{6.28}$$

where

$$\mathbf{U}^{(d),n-1} = \mathbf{U}_p^{n-1} + \boldsymbol{\omega}_p^{n-1} \times (\mathbf{X}_l^{n-1} - \mathbf{X}_p^{n-1}). \tag{6.29}$$

6.2.4 Particle-particle and particle-wall collision models

In particulate flows, particle-particle or particle-wall collisions are unavoidable and handling these collisions is an important task. Generally, an artificial mechanism is introduced in the numerical scheme to implement the repulsive force for the collision process. In this chapter, we employed two different collision models, namely Model-1 and Model-2, for comparison purposes in a given problem. Model-1 was originally proposed by Glowinski et al. (2001) and has been used in Feng and Michaelides (2005), Blasco et al. (2009). The repulsive force between particles is determined as

$$\mathbf{F}_{ij}^{P} = \begin{cases} 0 & \text{for} \quad d_{ij} > R_i + R_j + \xi, \\ \frac{C_{ij}}{\epsilon_P}\left(\frac{R_i+R_j+\xi-d_{ij}}{\xi}\right)^2\left(\frac{\mathbf{X_i}-\mathbf{X_j}}{d_{ij}}\right) & \text{for} \quad d_{ij} \leq R_i + R_j + \xi, \end{cases} \tag{6.30}$$

where C_{ij} and ϵ_P are the force scale and the stiffness parameter specified in a problem; R_i and R_j the radius of the ith and jth particle; $\mathbf{X}_i$ and $\mathbf{X}_j$ the coordinates of the centres; $d_{ij} = |\mathbf{X}_i - \mathbf{X}_j|$ the distance between the mass centres; ξ the range of the repulsive force. Similarly, the repulsive force between a particle and a wall is given by

$$\mathbf{F}_{i}^{W} = \begin{cases} 0 & \text{for} \quad d_{iw} > 2R_i + \xi, \\ \frac{C_{iw}}{\epsilon_W}\left(\frac{2R_i+\xi-d_{iw}}{\xi}\right)^2\left(\frac{\mathbf{X_i}-\mathbf{X}_{iw}}{d_{iw}}\right) & \text{for} \quad d_{iw} \leq 2R_i + \xi, \end{cases} \tag{6.31}$$

where $\mathbf{X}_{iw}$ is the coordinate vector of the centre of the nearest imaginary particle P_{iw} located on the boundary wall Γ with respect to the ith particle (Figure 6.2); $d_{iw} = |\mathbf{X}_i - \mathbf{X}_{iw}|$ the distance between the mass centres of the ith particle and the centre of the imaginary particle P_{iw}; and C_{iw} and ϵ_W corresponding force scale and positive stiffness parameter.

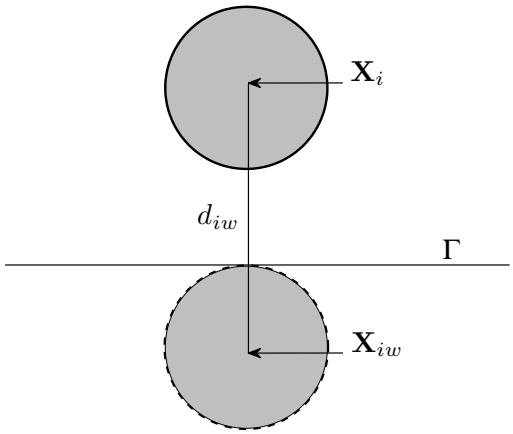


Figure 6.2: Single particle sedimentation: Imaginary particle.

Model-2 was proposed by Wan and Turek (2007a) and has been used in Wang et al. (2008). The repulsive force between particles is determined as

$$\mathbf{F}_{ij}^{P} = \begin{cases} 0 & \text{for} \quad d_{ij} > R_i + R_j + \xi, \\ \frac{1}{\epsilon_P} \left(\mathbf{X_i} - \mathbf{X}_j\right) \left(R_i + R_j + \xi - d_{ij}\right)^2 & \text{for} \quad R_i + R_j \leq d_{ij} \leq R_i + R_j + \xi, \\ \frac{1}{\epsilon'_P} \left(\mathbf{X}_i - \mathbf{X}_j\right) \left(R_i + R_j - d_{ij}\right) & \text{for} \quad d_{ij} \leq R_i + R_j, \end{cases} \tag{6.32}$$

where ϵ'_P is another positive stiffness parameter for particle-particle collisions. For the particle-wall collisions, the corresponding repulsive force is given by

$$\mathbf{F}_{i}^{W} = \begin{cases} 0 & \text{for} \quad d_{iw} > 2R_i + \xi, \\ \frac{1}{\epsilon_W} \left(\mathbf{X}_i - \mathbf{X}_{iw}\right) \left(2R_i + \xi - d_{iw}\right)^2 & \text{for} \quad 2R_i \leq d_{iw} \leq 2R_i + \xi, \\ \frac{1}{\epsilon'_W} \left(\mathbf{X}_i - \mathbf{X}_{iw}\right) \left(2R_i - d_{iw}\right) & \text{for} \quad d_{iw} \leq 2R_i, \end{cases} \tag{6.33}$$

where ϵ'_W is another positive stiffness parameter for particle-wall collisions.

The total repulsive forces (i.e. collision forces) exerted on the ith particle by the other particles and the walls are expressed as

$$\mathbf{F}_{i}^{col} = \sum_{j=1, j\neq i}^{N_p} \mathbf{F}_{ij}^{P} + \mathbf{F}_{i}^{W}. \tag{6.34}$$

6.3 Numerical formulation

For the discretisations of the spatial derivatives in the governing equations, the approximation scheme CIRBF-3 described in section 5.3.2 is employed in this chapter. For temporal discretisation, the procedure is as follows.

The temporal discretisation of (6.6) with an Adams-Bashforth scheme (Butcher, 2003) for the convection term and a Crank-Nicolson scheme (Crank and Nicolson, 1996) for the diffusion term gives

$$\begin{aligned} \frac{\mathbf{u}^n - \mathbf{u}^{n-1}}{\Delta t} + \left[\frac{3}{2}(\mathbf{u}^{n-1}.\nabla)\mathbf{u}^{n-1} - \frac{1}{2}(\mathbf{u}^{n-2}.\nabla)\mathbf{u}^{n-2}\right] = \\ - \frac{1}{\rho_f}\nabla p^{n-1/2} + \frac{\nu}{2}\left(\nabla^2\mathbf{u}^n + \nabla^2\mathbf{u}^{n-1}\right) + \frac{1}{\rho_f}\mathbf{f}_I^{n-1/2}, \end{aligned} \tag{6.35}$$

or

$$\mathbf{u}^n - \frac{\nu\Delta t}{2}\nabla^2\mathbf{u}^n = \mathbf{u}^{n-1} + \frac{\nu\Delta t}{2}\nabla^2\mathbf{u}^{n-1} - \frac{\Delta t}{\rho_f}\nabla p^{n-1/2} - \Delta t\left[\frac{3}{2}(\mathbf{u}^{n-1}.\nabla)\mathbf{u}^{n-1} - \frac{1}{2}(\mathbf{u}^{n-2}.\nabla)\mathbf{u}^{n-2}\right] + \frac{\Delta t}{\rho_f}\mathbf{f}_I^{n-1/2}. \quad (6.36)$$

We apply the pressure-free projection/fractional-step methods developed by Kim and Moin (1985) to solve the continuity equation (6.2) and momentum equation (6.36). The two governing equations are advanced in time according to the following two-step procedure, where one can project an arbitrary vector field into a divergence-free vector field via the pressure variable in (6.36)

$$\frac{\mathbf{u}^{*,n} - \mathbf{u}^{n-1}}{\Delta t} + \left[\frac{3}{2}(\mathbf{u}^{n-1}.\nabla)\mathbf{u}^{n-1} - \frac{1}{2}(\mathbf{u}^{n-2}.\nabla)\mathbf{u}^{n-2}\right] = \frac{\nu}{2}\left(\nabla^2\mathbf{u}^{*,n} + \nabla^2\mathbf{u}^{n-1}\right) + \frac{1}{\rho_f}\mathbf{f}_I^{n-1/2}, \quad (6.37)$$

$$\frac{\mathbf{u}^n - \mathbf{u}^{*,n}}{\Delta t} = -\frac{1}{\rho_f}\nabla\phi^n, \quad (6.38)$$

with

$$\nabla.\mathbf{u}^n = 0, \quad (6.39)$$

where ϕ is the pseudo-pressure from which the actual pressure is derived through

$$p^{n-1/2} = \phi^n - (\nu\Delta t/2)\nabla^2\phi^n. \quad (6.40)$$

The detailed calculation can be summarised as follows.

At first, an intermediate velocity vector $\mathbf{u}^{*,n}$ is obtained by solving (6.37) in the form of

$$\mathbf{u}^{*,n} - \frac{\nu\Delta t}{2}\nabla^2\mathbf{u}^{*,n} = \mathbf{u}^{n-1} + \frac{\nu\Delta t}{2}\nabla^2\mathbf{u}^{n-1} - \Delta t\left[\frac{3}{2}(\mathbf{u}^{n-1}.\nabla)\mathbf{u}^{n-1} - \frac{1}{2}(\mathbf{u}^{n-2}.\nabla)\mathbf{u}^{n-2}\right] + \frac{\Delta t}{\rho_f}\mathbf{f}_I^{n-1/2}, \quad (6.41)$$

subject to the following boundary condition (Kim and Moin, 1985)

$$\mathbf{u}^{*,n}|_\Gamma = \mathbf{u}_b^n + \frac{\Delta t}{\rho_f}\left(\nabla\phi^{n-1}\right)|_\Gamma. \quad (6.42)$$

It can be seen that $\mathbf{u}^{*,n}$ does not satisfy the continuity equation (6.39). For a more efficient solution, one can apply the alternating direction implicit (ADI) algorithm to solve (6.41) as shown in Thai-Quang, Mai-Duy, Tran and Tran-Cong (2012).

Equations (6.38)-(6.39) are then solved in a coupled manner for $\mathbf{u}^n$ and ϕ^n in which the boundary condition for the pseudo-pressure ϕ is not required. The values of ϕ^n are solved for the interior nodes only. Then, the values of ϕ^n at the boundary nodes are extrapolated from known values at the interior nodes and known Neumann boundary values derived from (6.38) (i.e. $\nabla\phi^n|_\Gamma = \rho_f\left(\mathbf{u}_b^{*,n} - \mathbf{u}_b^n\right)/\Delta t$) (Thai-Quang, Le-Cao, Mai-Duy and Tran-Cong, 2012):

$$\begin{pmatrix} \phi^n_{1,j} \\ \phi^n_{nx,j} \end{pmatrix} = \begin{bmatrix} \overline{H}_1(x_{1,j}) & \cdots & \overline{H}_{n_x}(x_{1,j}) & x_{1,j} & 1 \\ \overline{H}_1(x_{n_x,j}) & \cdots & \overline{H}_{n_x}(x_{n_x,j}) & x_{n_x,j} & 1 \end{bmatrix} \begin{bmatrix} \overline{H}_1(x_{2,j}) & \cdots & \overline{H}_{n_x}(x_{2,j}) & x_{2,j} & 1 \\ \overline{H}_1(x_{3,j}) & \cdots & \overline{H}_{n_x}(x_{3,j}) & x_{3,j} & 1 \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ \overline{H}_1(x_{n_x-1,j}) & \cdots & \overline{H}_{n_x}(x_{n_x-1,j}) & x_{n_x-1,j} & 1 \\ H_1(x_{1,j}) & \cdots & H_{n_x}(x_{1,j}) & 1 & 0 \\ H_1(x_{n_x,j}) & \cdots & H_{n_x}(x_{n_x,j}) & 1 & 0 \end{bmatrix}^{-1} \begin{pmatrix} \phi^n_{2,j} \\ \phi^n_{3,j} \\ \vdots \\ \phi^n_{n_x-1,j} \\ \partial\phi^n_{1,j}/\partial x \\ \partial\phi^n_{n_x,j}/\partial x \end{pmatrix}, \tag{6.43}$$

for an x-grid line ($j \in \{1,\ 2, \ldots,\ n_y\}$), and

$$\begin{pmatrix} \phi^n_{i,1} \\ \phi^n_{i,n_y} \end{pmatrix} = \begin{bmatrix} \overline{H}_1(y_{i,1}) & \cdots & \overline{H}_{n_y}(y_{i,1}) & y_{i,1} & 1 \\ \overline{H}_1(y_{i,n_y}) & \cdots & \overline{H}_{n_y}(y_{i,n_y}) & y_{i,n_y} & 1 \end{bmatrix} \begin{bmatrix} \overline{H}_1(y_{i,2}) & \cdots & \overline{H}_{n_y}(y_{i,2}) & y_{i,2} & 1 \\ \overline{H}_1(y_{i,3}) & \cdots & \overline{H}_{n_y}(y_{i,3}) & y_{i,3} & 1 \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ \overline{H}_1(y_{i,n_y-1}) & \cdots & \overline{H}_{n_y}(y_{i,n_y-1}) & y_{i,n_y-1} & 1 \\ H_1(y_{i,1}) & \cdots & H_{n_y}(y_{i,1}) & 1 & 0 \\ H_1(y_{i,n_y}) & \cdots & H_{n_y}(y_{i,n_y}) & 1 & 0 \end{bmatrix}^{-1} \begin{pmatrix} \phi^n_{i,2} \\ \phi^n_{i,3} \\ \vdots \\ \phi^n_{i,n_y-1} \\ \partial\phi^n_{i,1}/\partial y \\ \partial\phi^n_{i,n_y}/\partial y \end{pmatrix}, \tag{6.44}$$

for a y-grid line ($i \in \{1,\ 2, \ldots,\ n_x\}$).

It is noted that the governing equations (6.2)-(6.3) can be non-dimensionalised and written in a non-dimensionalised form with the definition of Reynolds number Re. The readers are referred to sections 5.3.3 and 5.3.1 for detailed descriptions of the derivations of temporal discretisation and DF method, respectively, in the non-dimensionalised form.

6.4 Numerical examples

As shown in chapter 2, our methods work well with a wide range of values for β. The value of $\beta = 50$ is simply picked for all calculations in this chapter. We assess the performance of the proposed method through the following measures:

- maximum absolute error (L_∞) defined by (3.53),
- the average rate of convergence α determined by the method of least squares as described in chapter 2.

6.4.1 Taylor-Green vortices

To investigate the performance of the present DFIB-CIRBF method in the IB approach, the problem of Taylor-Green vortices with following analytic solutions (Uhlmann, 2005) is considered

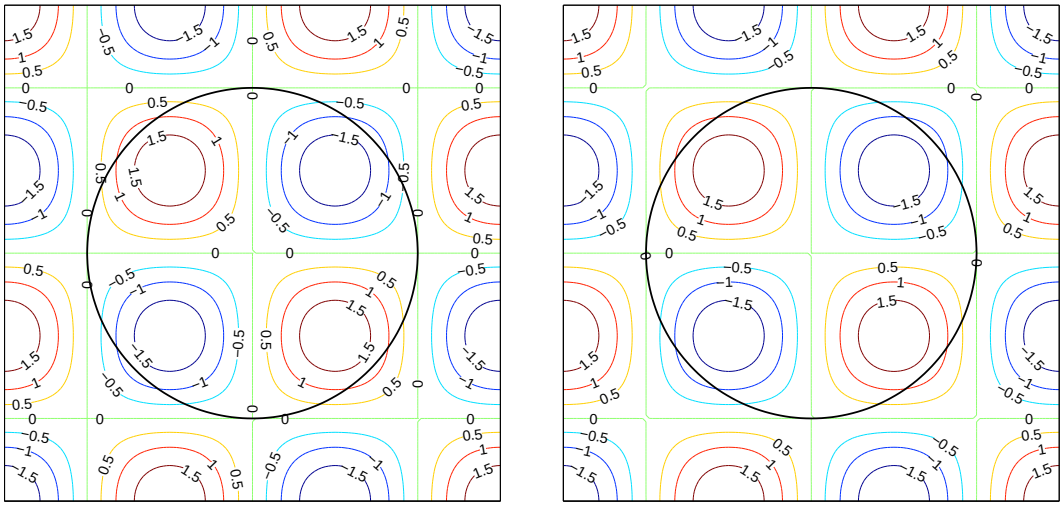
$$\overline{u}(x,y,t) = \sin(k_x x)\cos(k_y y)e^{-(k_x^2+k_y^2)t/Re}, \tag{6.45}$$

$$\overline{v}(x,y,t) = -k_x/k_y \sin(k_y y)\cos(k_x x)e^{-(k_x^2+k_y^2)t/Re}, \tag{6.46}$$

$$\overline{p}(x,y,t) = 0.5\left(\cos^2(k_y y)\frac{k_x^2}{k_y^2} - \sin^2(k_x x)\right)e^{-2(k_x^2+k_y^2)t/Re}, \tag{6.47}$$

where $k_x = k_y = \pi$. This problem is simulated in an embedded circular domain with radius unity and centred at the origin of the computational domain $\Omega = [-1.5, 1.5] \times [-1.5, 1.5]$. The Reynolds number is set to be $Re = 5$ and the solutions are computed at a final time $t = 0.3$ using a time step $\Delta t = 0.001$ (Uhlmann, 2005). The initial field at $t = 0$, the time-dependent boundary conditions at the domain boundary Γ and the time-dependent desired velocity values $\mathbf{U}^{(d)}$ at the circumference of the embedded circle are derived from (6.45)-(6.47).

Figure 6.3 shows the position of the embedded circle in the computational domain, and computed and analytic vorticity isolines using a uniform grid of 151×151. The two results are almost identical and it can be inferred that the problem is solved accurately by the present DFIB-CIRBF method. Indeed, Figure 6.4 shows the maximum absolute error L_∞ of the velocity for grid nodes located inside the embedded domain, plotted as a function of the grid size h. The result published by Uhlmann (2005) is also included for comparison purpose. The accuracy of the solution obtained by the present method is slightly higher than that obtained by Uhlmann's method whereas almost same convergence rate, i.e. $O(h^2)$, is observed for both methods. The overall accuracy of the present solution is higher because the high-order approximation scheme (CIRBF-3) is employed for the fluid solver. From those good agreements between the present and Uhlmann's results, the second-order accuracy of the interpolation/extrapolation with the regularised δ_h function is confirmed again in this study.



(a) (b)

Figure 6.3: Taylor-Green vortices, 151×151, $\Delta t = 0.001$: Position of the embedded circle and the vorticity isolines at $t = 0.3$ for the analytic (a) and present (b) solutions.

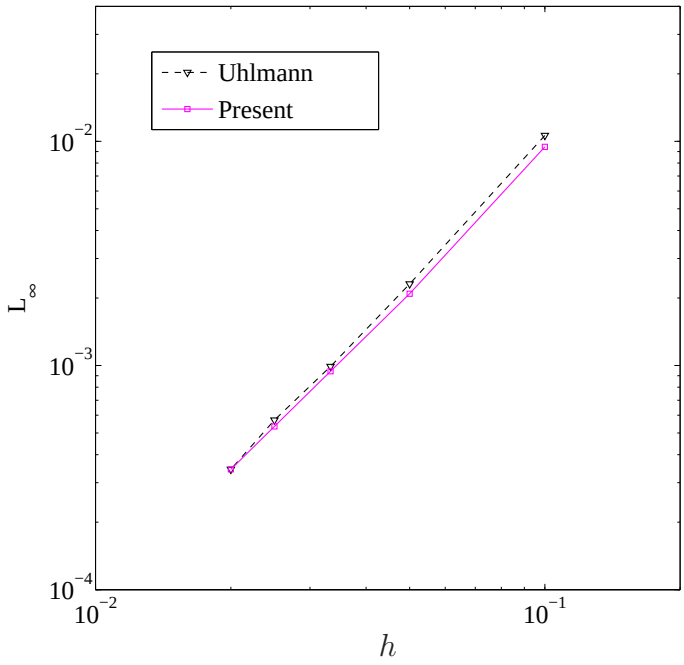


Figure 6.4: Taylor-Green vortices, $\{31\times31,\ 61\times61, \ldots,\ 151\times151\}$, $\Delta t = 0.001$, $t = 0.3$: The effect of grid size h on the solution accuracy for the velocity. The solutions converge as about $O(h^2)$ for both the present and referential results (Uhlmann, 2005).

6.4.2 Induced flow by an oscillating circular cylinder

In this problem, a 2D viscous flow in a stationary square domain $\Omega = [-0.5, 0.5]\times[-0.5, 0.5]$ induced by an oscillating circular cylinder is considered (Figure 6.5).

The diameter D of the cylinder is 0.5, and the motion is prescribed by the horizontal translation of the centre $X(t) = X_0 \cos(2\pi ft)$, where the initial position X_0 is equal to 0.1 and the oscillation frequency f of the cylinder is equal to 1 Hz (Liao et al., 2010). The corresponding horizontal velocity is $U(t) = U_0 \sin(2\pi ft)$ where the maximum velocity is $U_0 = -2\pi f X_0$. Non-slip boundary conditions are imposed on all fluid-solid interfaces. Based on D, U_0 and constant density, the Reynolds numbers of 100 and 800 are chosen by varying viscosity to simulate the problem for verification purposes.

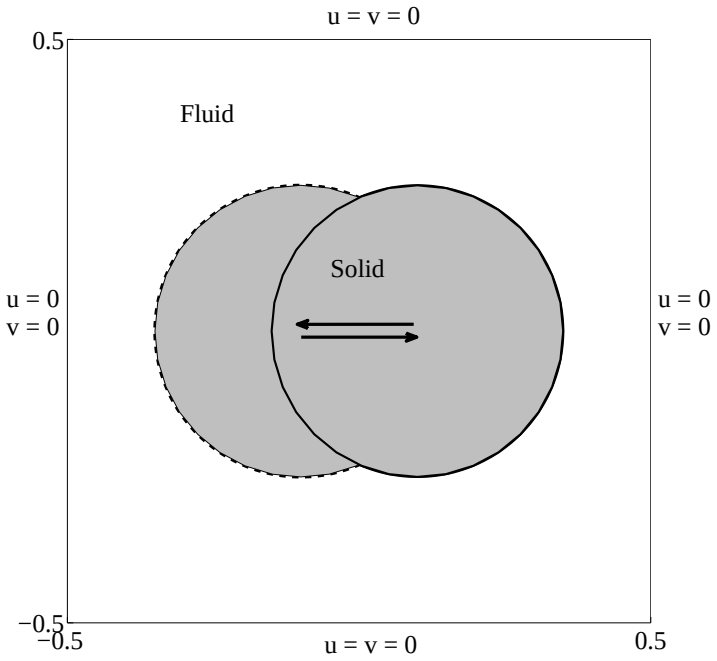


Figure 6.5: Induced flow by an oscillating circular cylinder: Configuration of the domain and boundary conditions.

The drag F_D and lift F_L forces generated by the cylinder can be obtained by integrating the stress distributions along the cylinder wall or by computing the volume integral of the Navier-Stokes equation. In this chapter, an alternative method for computing the drag and lift induced by the presence of the cylinder is used as in Lai and Peskin (2000), Su and Lai (2007), Liao et al. (2010)

$$F_D = -\int_\Omega f_{Ix}(\mathbf{x})d\mathbf{x} \cong -\sum_{\mathbf{x}} f_{Ix}(\mathbf{x})h^2, \tag{6.48}$$

$$F_L = -\int_\Omega f_{Iy}(\mathbf{x})d\mathbf{x} \cong -\sum_{\mathbf{x}} f_{Iy}(\mathbf{x})h^2. \tag{6.49}$$

Figure 6.6 shows the streamlines of the flow field at two different Reynolds numbers using a uniform grid of 151×151. The same values of the contour lines are used for comparison purpose. It is seen that the vortices inside the

domain is stronger for the larger Re number as vortices behind the cylinder are formed in the case of $Re = 800$, whereas no vortices exist in the case of $Re = 100$.

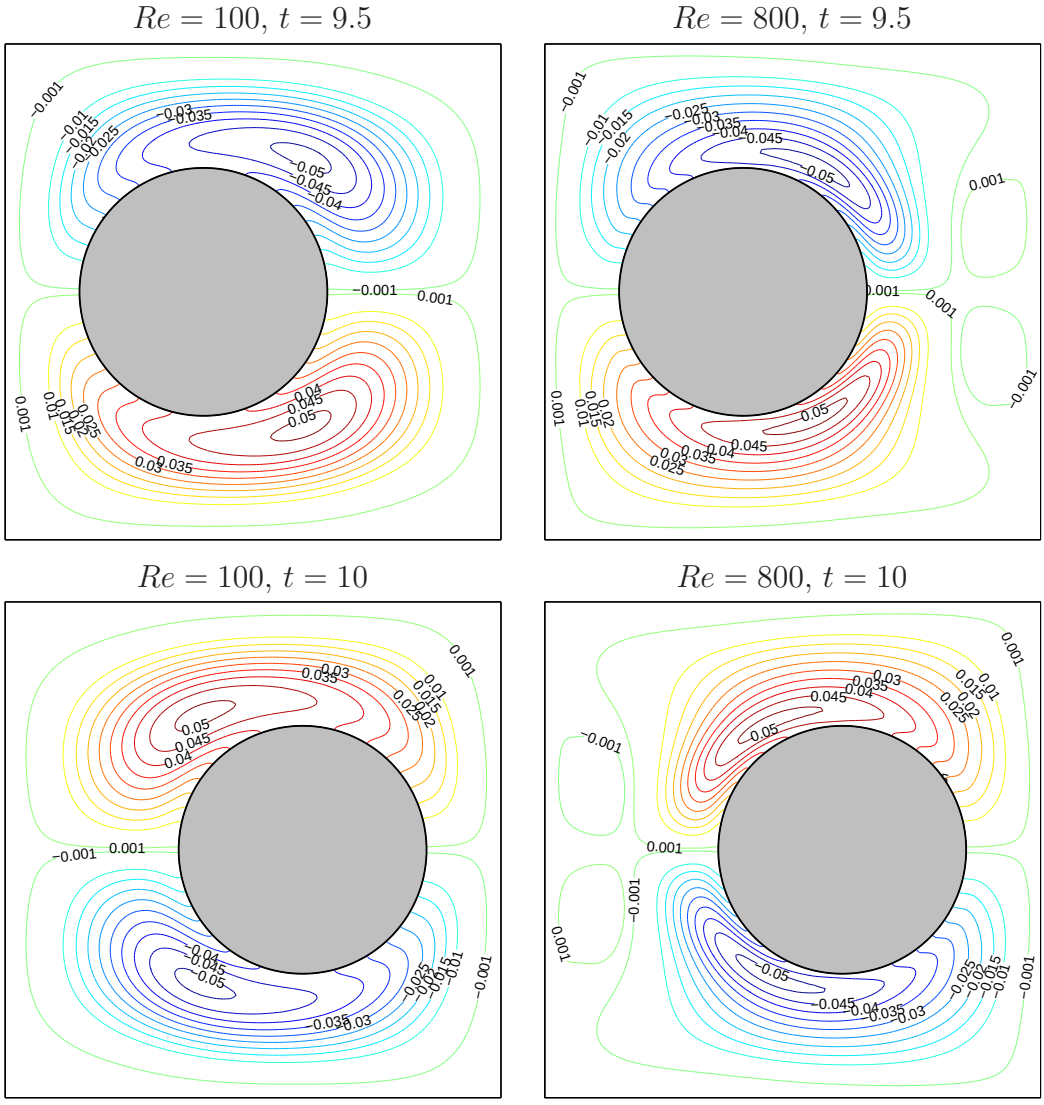


Figure 6.6: Induced flow by an oscillating circular cylinder, 151×151, $\Delta t = 0.001$: Streamlines of the flow field for different Reynolds numbers at different times.

The influence of the reduction of the viscosity can be clearly observed in Figure 6.7, where weaker drag force is observed at $Re = 800$ in contrast to the stronger drag force obtained at $Re = 100$. After approximately one period, the drag force appears to have reached the periodic state. Also shown in Figure 6.7, the evolution of the drag forces correlate well with the oscillation frequency of the cylinder. Similar results can be observed in Liao et al. (2010) for the case of $Re = 800$ where a much finer grid of 300×300 was used.

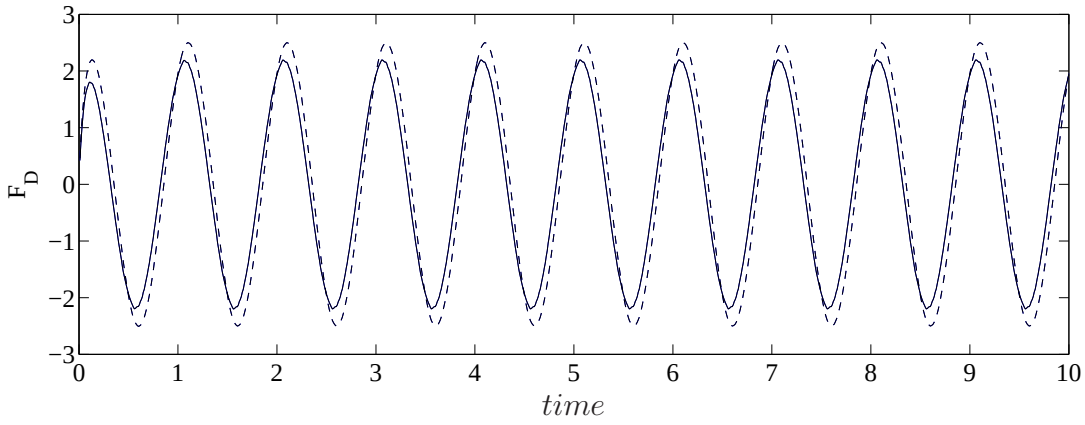


Figure 6.7: Induced flow by an oscillating circular cylinder, 151×151, $\Delta t = 0.001$: The evolution of the drag force for different Reynolds numbers. $Re = 100$ (dash line) and $Re = 800$ (solid line).

6.4.3 Single particle sedimentation

Next, we consider the problem of the sedimentation of a rigid circular particle due to gravity in a bounded cavity filled with a viscous incompressible Newtonian fluid, when the particle is released from rest from a position on the centreline of the cavity.

The computational domain is a rectangular box of dimensions 2×6 cm (Figure 6.8). The particle radius is $R = 0.125$ cm and its density is $\rho_p = 1.25$ g/cm^3. The density and the dynamic viscosity of the fluid are $\rho_f = 1$ g/cm^3 and $\mu = 0.1$ $g/cm.s$, respectively. The gravitational acceleration is $g = 980$ cm/s^2 and it acts in the negative y-direction. Initially, the particle centre is set at the position of (1,4) cm. Both the fluid and particle are initially at rest. All of the walls are stationary as shown in Figure 6.8. These parameter values are taken to be the same as those in Wu and Shu (2010), Blasco et al. (2009), Wan and Turek (2006) whose results will be compared with ours.

Two different uniform grids, namely $\{51 \times 151,\ 101 \times 301\}$ are considered to study the effect of the grid size. A time step of $\Delta t = 0.001$ is used in all calculations and the solution is computed until the final time of $t = 1$. Model-1 is employed for the collision model in this example and the values used for the parameters of the repulsive forces are $\xi = h$, $\epsilon_w = 2$ and $C_{pw} = (\rho_p - \rho_f)\,\pi R^2 g$ (Hayashi et al., 2012).

Figures 6.9, 6.10 and 6.11 shows the contours of the vertical velocity, streamlines, contours of the vorticity and the corresponding position of the particle at different times. As observed, the particle path is rectilinear and the symmetric

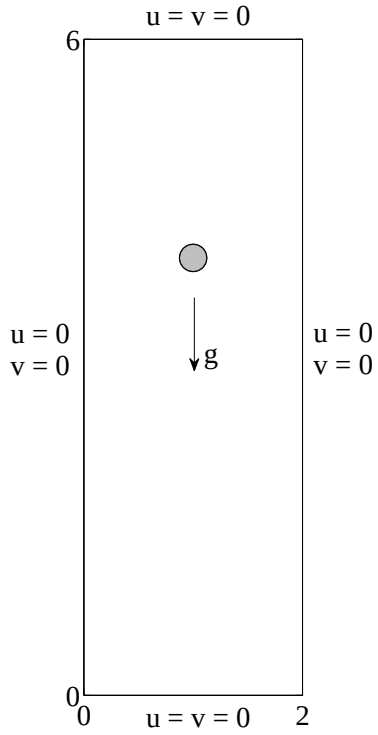


Figure 6.8: Single particle sedimentation: Schematic view and boundary conditions.

flow field with respect to the vertical centreline is formed. It is noted that the contours of the vorticity shown in Figure 6.11 are very comparable with the ones observed in Wu and Shu (2010). Figure 6.12a,b shows the evolution of the x and the y coordinates of the particle centre. As observed, the finer mesh yields the results which are in better agreement with the published ones. Additionally, symmetric solutions were reported in Blasco et al. (2009), Wan and Turek (2007a) for these values with meshes of the order of 1/100, which is also in agreement with our results as observed in Figure 6.12a. Figure 6.12c,d plots the evolution of the two components of translational velocity of the particle centre, where the same tendency and good agreement with those in Wu and Shu (2010), Wan and Turek (2006) are achieved.

For a quantitative comparison, the terminal settling velocity V_t, defined as the maximum of the particle translational velocity $\|\mathbf{U}_p(t)\|$, and the maximum particle Reynolds number $Re_{p,max}$ are investigated. These are known as ones of the main quantities of interest in this problem. The particle Reynolds number and translational energy are defined as

$$Re_p = \frac{\rho_p D \sqrt{U_p^2 + V_p^2}}{\mu}, \tag{6.50}$$

$$E_t = \frac{1}{2} M_p \left(U_p^2 + V_p^2 \right), \tag{6.51}$$

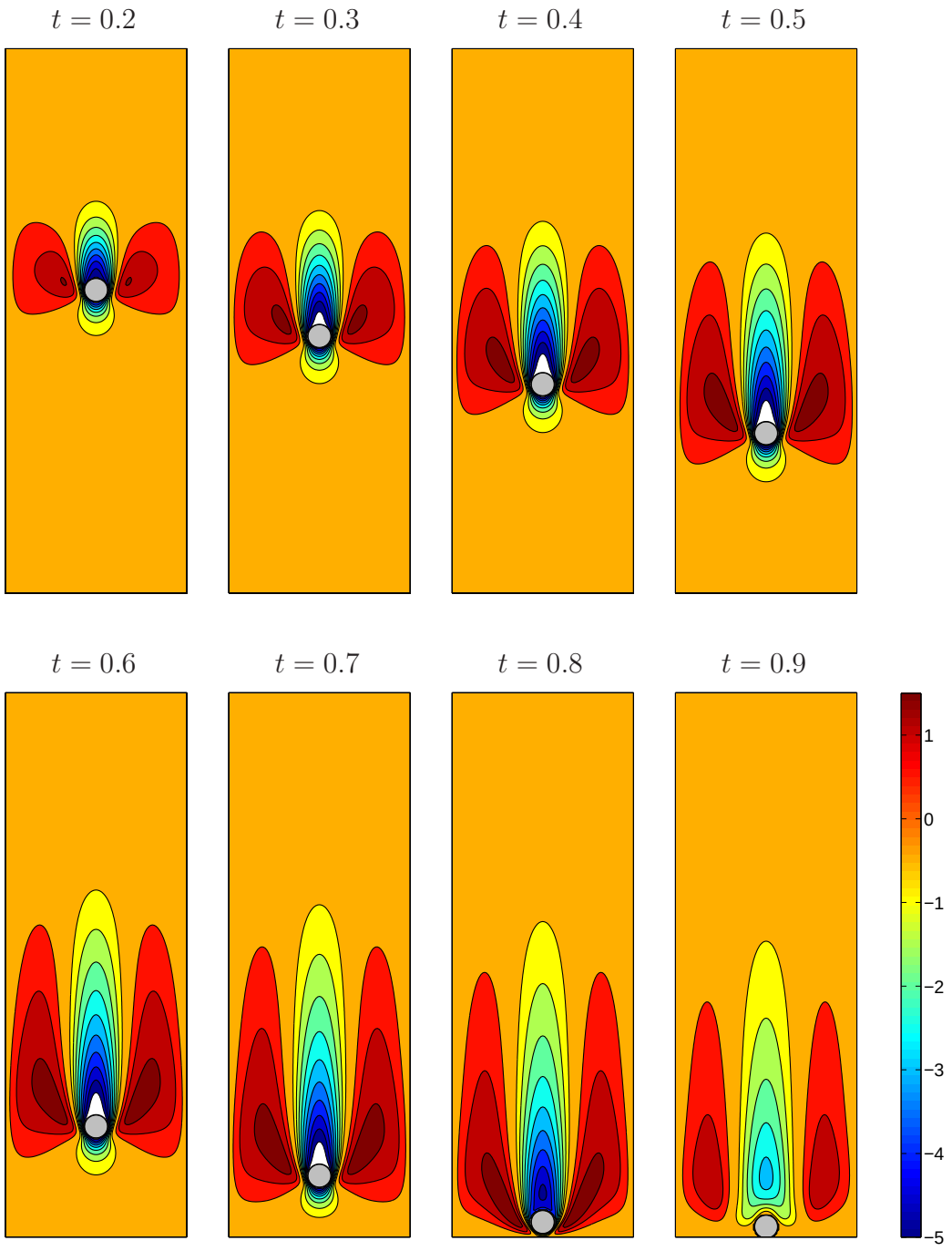


Figure 6.9: Single particle sedimentation, 101×301, $\Delta t = 0.001$: Contours of the vertical velocity at different times. Values of the contour lines: $\pm\{-0.5 : -0.5 : -5,\ 0.5 : 0.5 : 1.5\}$.

where D is the diameter of the particle. Table 6.1 provides the present and referential results, showing that good agreement is achieved. Figure 6.12e,f also displays the evolution of the particle Reynolds number and translational energy, showing that good agreements with referential results are obtained.

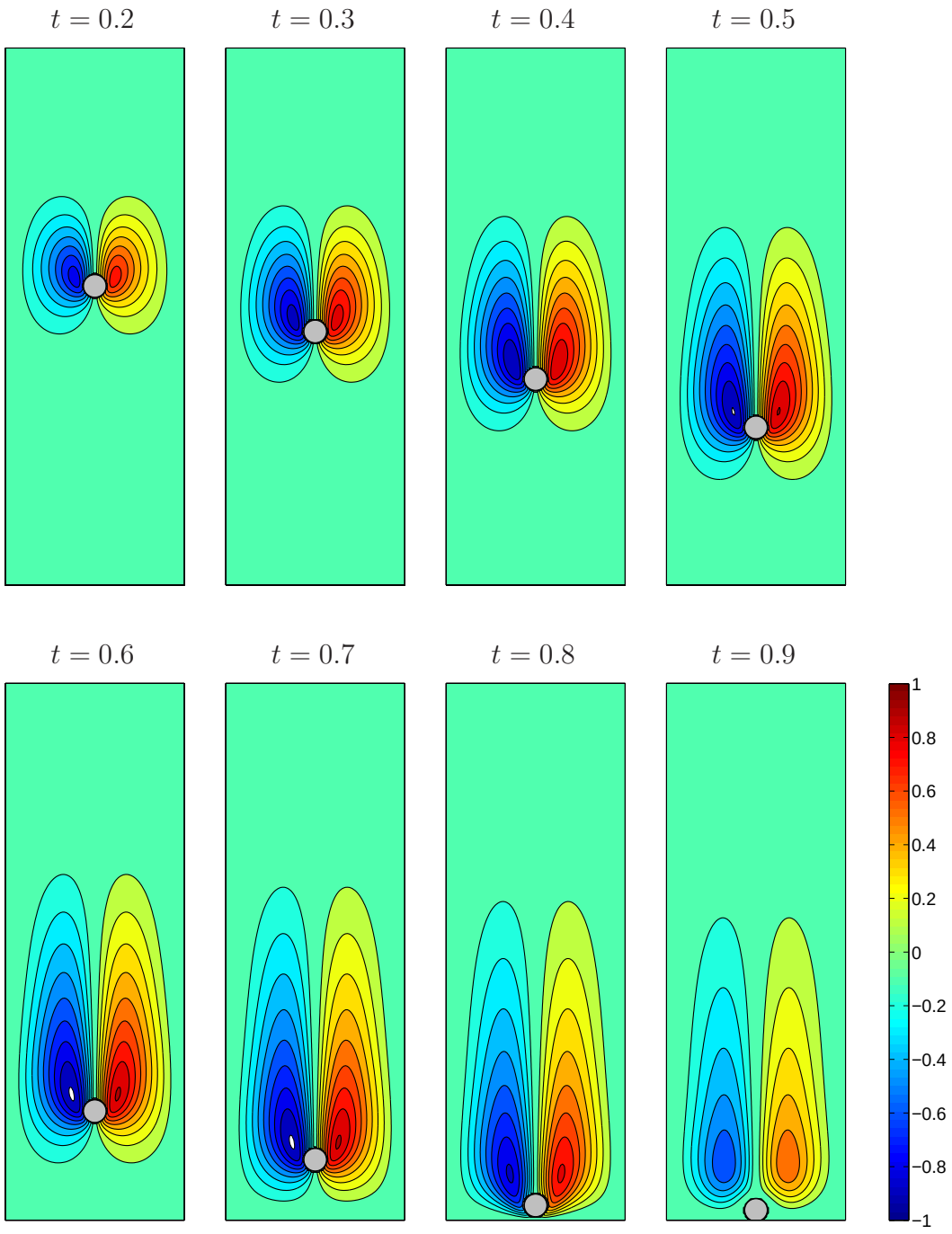


Figure 6.10: Single particle sedimentation, 101×301, $\Delta t = 0.001$: Streamlines of the flow field at different times. Values of the contour lines: $\pm\{0.1 : 0.1 : 0.9\}$.

6.4.4 Drafting-kissing-tumbling behaviour of two settling particles

This problem is usually known as a *drafting, kissing and tumbling* (DKT) phenomenon (Fortes et al., 1987). In this problem, we simulate the motion and

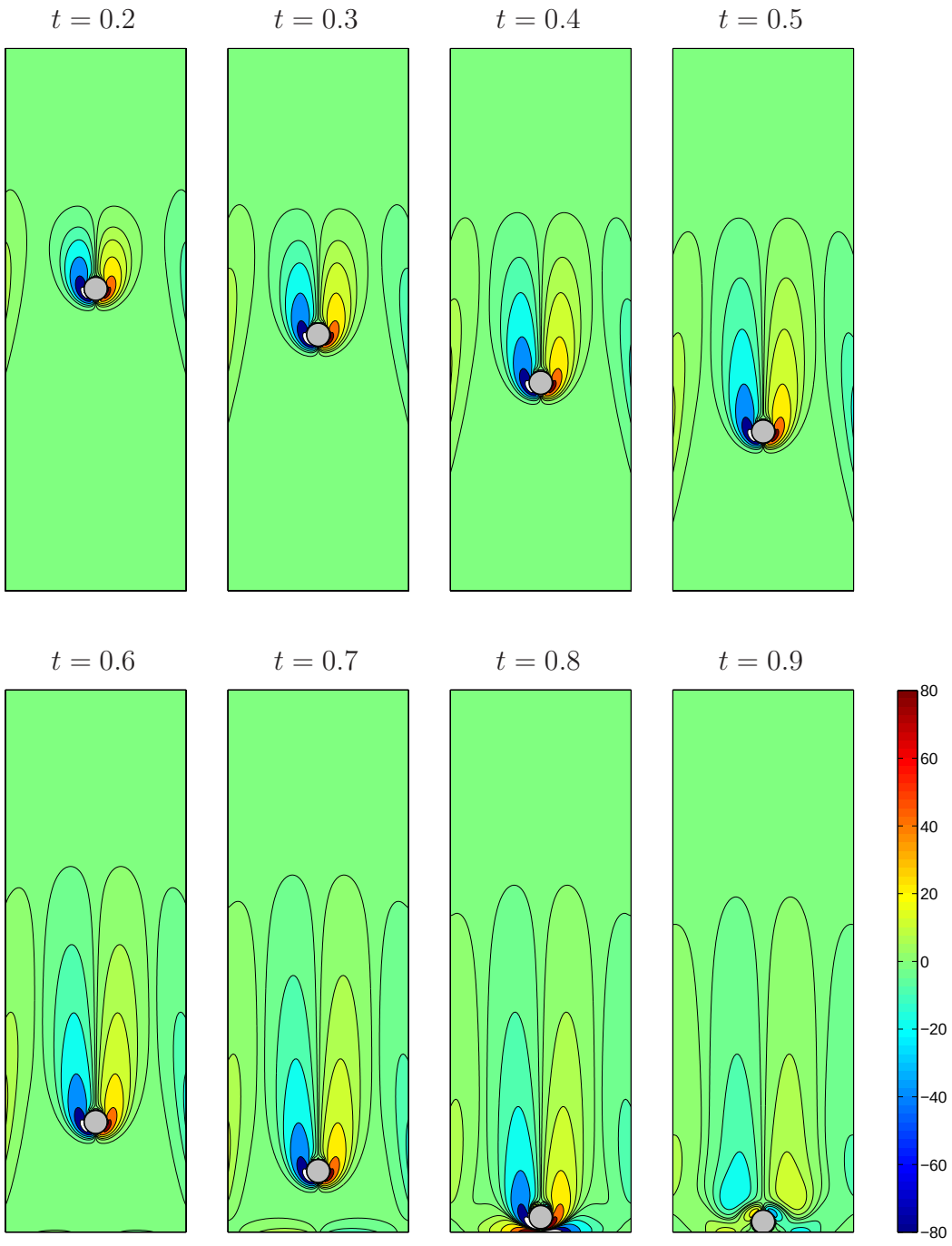


Figure 6.11: Single particle sedimentation, 101×301, $\Delta t = 0.001$: Contours of the vorticity of the flow field at different times. Values of the contour lines: $\pm\{1,\ 5,\ 10,\ 20,\ 40,\ 80\}$.

interaction of two rigid circular particles sedimenting in a vertical channel filled with a viscous incompressible Newtonian fluid as a further check for the performance of the present method.

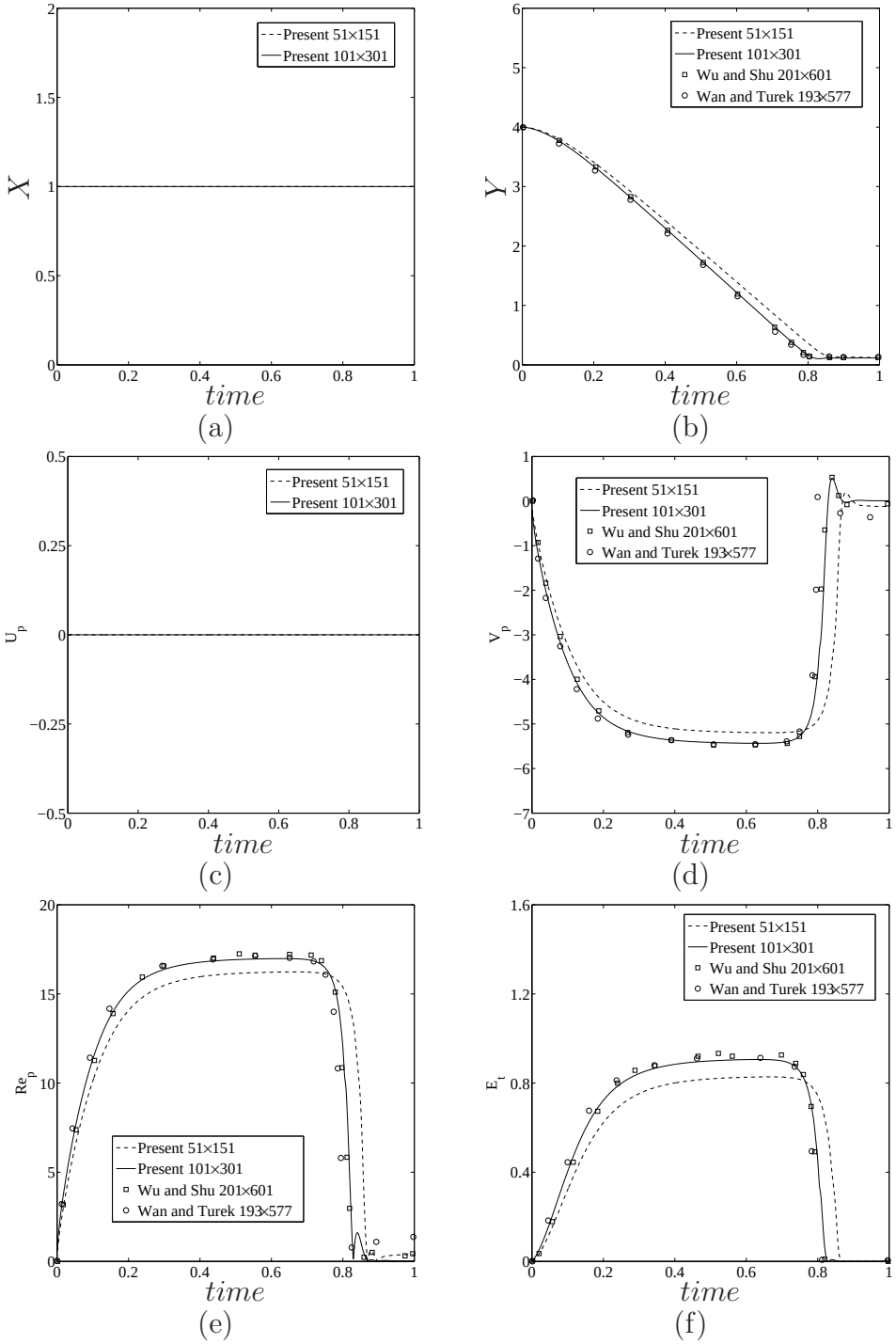


Figure 6.12: Single particle sedimentation: Time histories of some quantities including the x-coordinate of the particle centre (a), the y-coordinate of the particle centre (b), the x-component of the translational particle velocity (c), the y-component of the translational particle velocity (d), the Reynolds number for the particle (e), and the translational kinetic energy (f).

Table 6.1: Single particle sedimentation, $\Delta t = 0.001$: Comparison of the terminal settling velocity and maximum particle Reynolds number.

Method	h	V_t	$Re_{p,max}$
Present	1/25	5.197	16.24
	1/50	5.434	16.98
Wu and Shu (2010)	1/100	—	17.08
Blasco et al. (2009)	1/25	4.474	13.98
	1/50	5.302	16.57
	1/100	5.515	17.23
	1/150	5.550	17.34
Wan and Turek (2007a)	1/48	—	17.42
	1/96	—	17.15
Glowinski et al. (2001)	1/192	—	17.27
	1/256	—	17.31

The computational domain is a channel of width 2 cm and height 6 cm (Figure 6.13). The particle density is $\rho_p = 1.5\ g/cm^3$ and its radius is 0.125 cm. The density and dynamic viscosity of the fluid are $\rho_f = 1\ g/cm^3$ and $\mu = 0.01\ g/cm.s$, respectively. Gravity acts in the negative y-direction and gravitational acceleration is $g = 980\ cm/s^2$. All walls of the channel are stationary as shown in Figure 6.13. The simulation is started at $t = 0$ by releasing from rest two particles located at (1-0.001,5) cm (particle P_1) and (1+0.001,4.5) cm (particle P_2). The solution was computed until the final time of $t = 0.35$ and the time step $\Delta t = 6.25 \times 10^{-5}$ is used in the calculations. Model-2 is employed for the collision model in this example and the values used for the parameters of the repulsive forces are $\xi = 2/256$, $\epsilon_P = \epsilon'_P = 1.0 \times 10^{-7}$, $\epsilon'_W = \epsilon_W = 0.5\epsilon_P$. Those parameters are taken to be the same as used in Wang et al. (2008). A uniform grid of 71×211 is used in the calculations. The initial horizontal positions of the two particles are offset from the vertical axis of symmetry to encourage horizontal motion and hence tumbling behaviour. Otherwise, only drafting and kissing can be observed and the tumbling could not happen as pointed out in Wang et al. (2008).

The evolutions of horizontal and vertical positions of the two particles are shown in Figure 6.14. The particles tend to separate horizontally and, interestingly, the upper particle eventually overtakes the lower one, which indicates the DKT phenomenon of the two settling particles. Also, the evolutions of the horizontal and vertical velocities of the two particles are depicted in Figure 6.15.

Figures 6.16-6.17 display the contours of the velocity magnitude, the contours of the vorticity in the flow and the positions of the two particles sedimenting at different times. From Figure 6.17, it is seen that the tendency of the development of the vorticity is similar to that in Wang et al. (2008). It is noted that

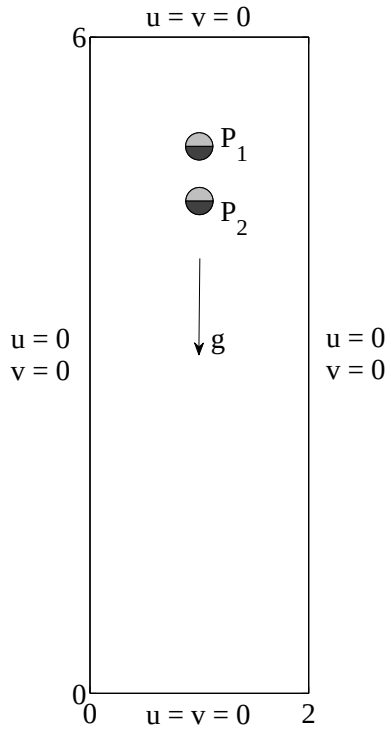


Figure 6.13: Drafting-kissing-tumbling of two settling particles: Schematic view and boundary conditions.

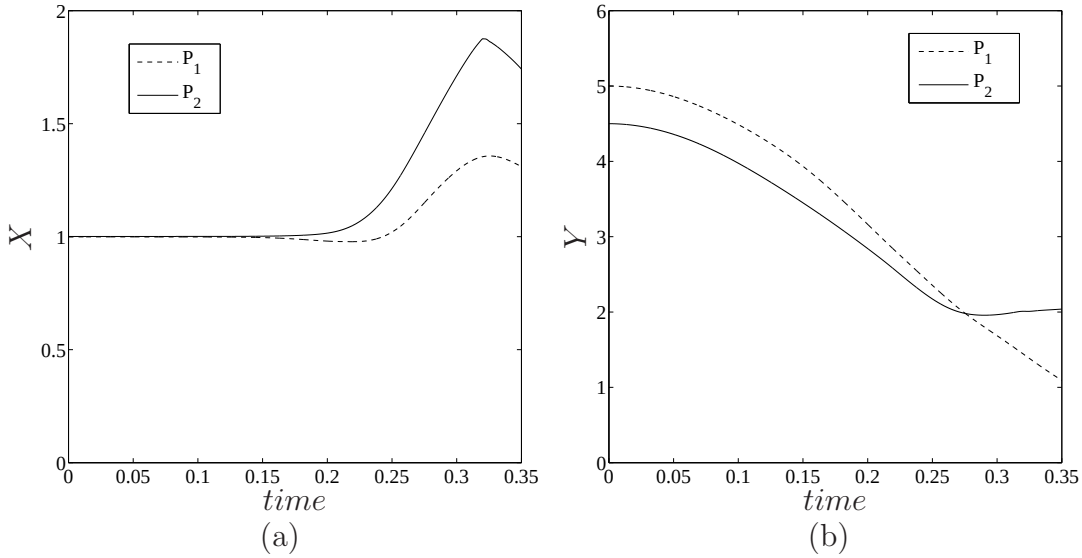


Figure 6.14: Drafting-kissing-tumbling of two settling particles, 71×211, $\Delta t = 6.25 \times 10^{-5}$: The evolution of the horizontal (a) and the vertical (b) positions of the centre of the two particles.

the legend of the color bar in Wang et al. (2008) is different with ours because they use a different scale of the domain for the post processing of their results.

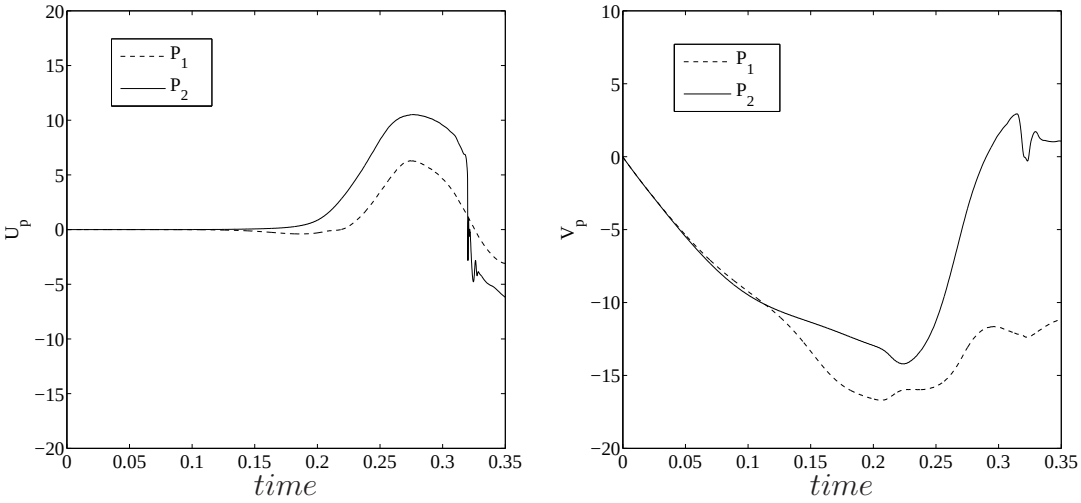


Figure 6.15: Drafting-kissing-tumbling of two settling particles, 71×211, $\Delta t = 6.25 \times 10^{-5}$: The evolution of the horizontal (a) and the vertical (b) velocities of the two particles.

Clearly, the DKT phenomenon is successfully reproduced. The explanation of this phenomenon can be found in Glowinski et al. (2001). When the lower particle falls, it creates a pressure drop in its wake. If two particles are close enough initially, this wake reduces the drag acted on the upper particle from the fluid while the lower particle encounters a bigger drag from the fluid. Therefore, the upper particle falls faster than the lower one. Falling faster, the upper particle touches the lower one at $t = 0.209$s. Once in contact, two particles act as an elongated body falling in an incompressible viscous fluid. As is well known, elongated bodies falling sufficiently fast in a Newtonian incompressible viscous fluid have a tendency to rotate so that their board sides become perpendicular to the flow direction. Indeed rotation takes place, as seen in Figure 6.17 at $t = 0.273$s, but the two-particles assemblage is unstable and the two particles separate at $t = 0.317$s and then they continue falling down at $t = 0.342$s. One can see that the results computed by the present method are similar to those observed in Wang et al. (2008), Wan and Turek (2007b), Glowinski et al. (2001).

6.5 Concluding remarks

In this chapter, we present a DFIB-CIRBF method for the discretisation of PDEs governing the problems of particulate flows in the Cartesian-grid point-collocation framework. The CIRBF approximation together with the extrapolation/interpolation by the regularised discrete delta function are employed in a DFIB method for fully-resolved direct numerical simulation of fluid-particle interaction. The capability of the present method is tested by its applications

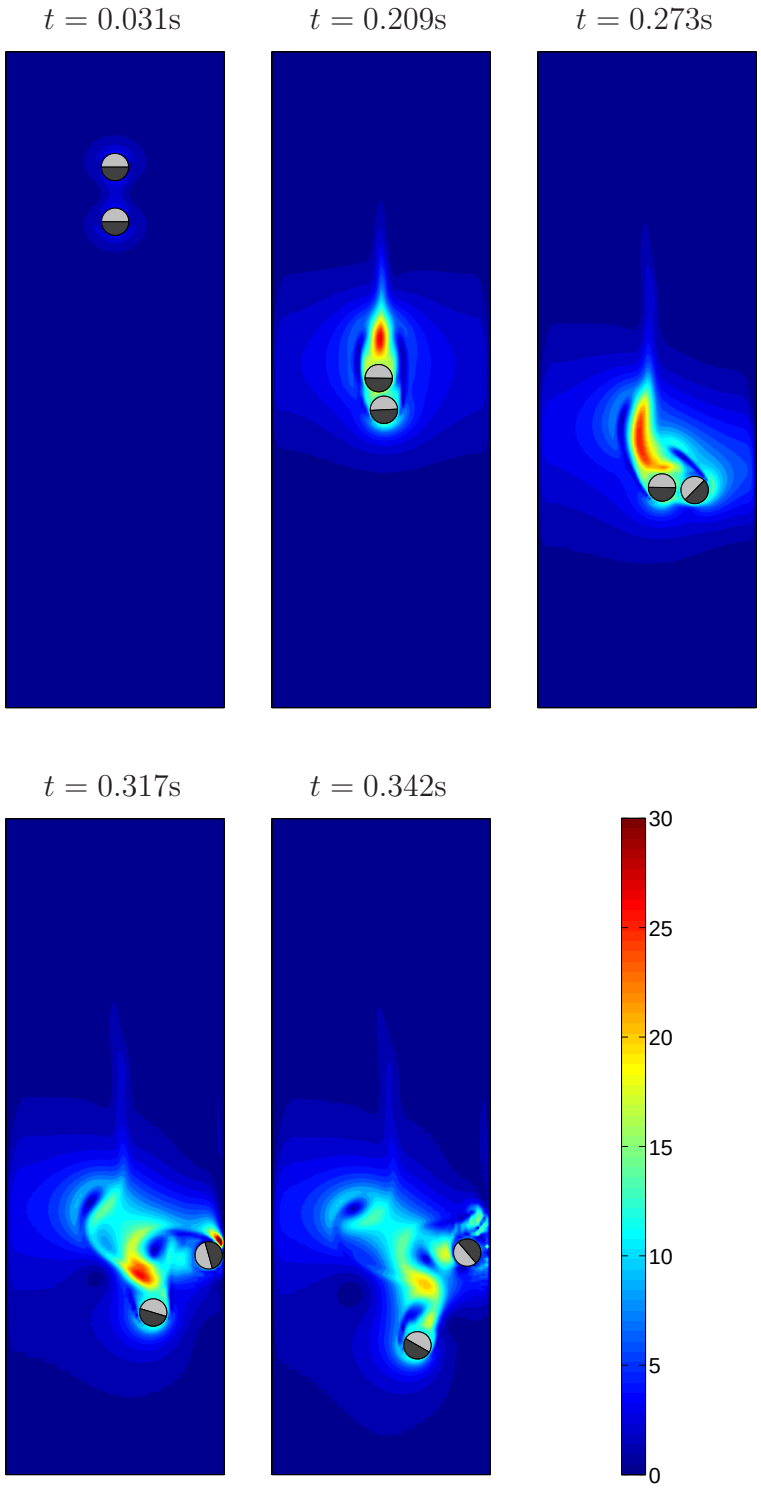


Figure 6.16: Drafting-kissing-tumbling of two settling particles, 71×211, $\Delta t = 6.25 \times 10^{-5}$: Contours of the velocity magnitude and the positions of particles at different times.

to simulate both stationary and moving boundary problems. Numerical results

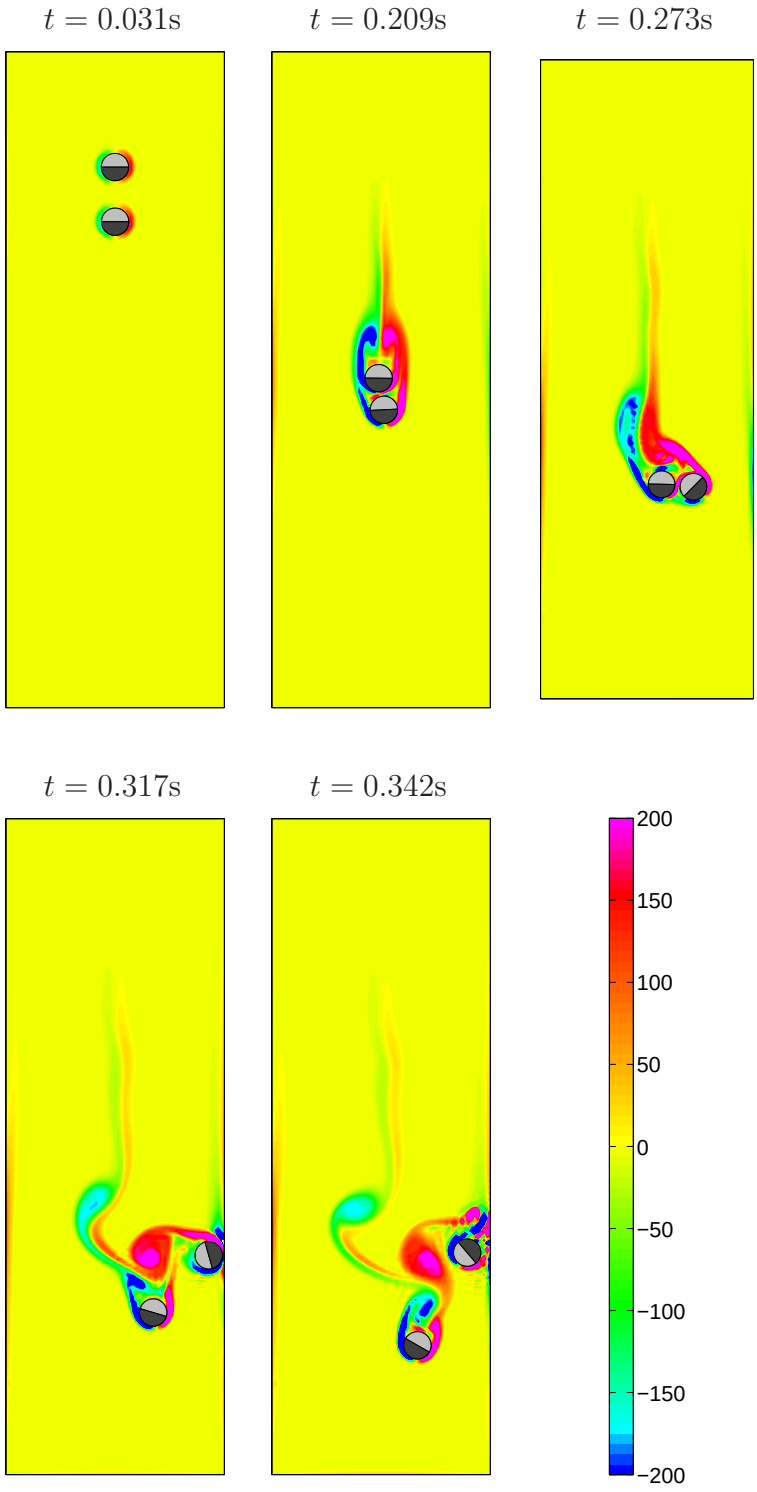


Figure 6.17: Drafting-kissing-tumbling of two settling particles, 71×211, $\Delta t = 6.25 \times 10^{-5}$: Contours of the vorticity and the positions of particles at different times.

show good agreement with available data in the literature. The complex flow

structures and physical phenomenon are successfully observed. In this chapter, only 2D problems are considered and the shape of the moving boundary in considered problems is circular. However, the present method can also be extended to 3D problems and the arbitrarily curved boundary can be handled as well.

Chapter 7

A DFIB-CIRBF approach for the rheology of particulate suspensions

In this chapter, a direct forcing immersed boundary-compact integrated radial basis function (DFIB-CIRBF) simulation method reported in chapters 5 and 6 is further developed for the investigation of the rheology of particulate suspensions in two dimensions. Suspensions of rigid particles in a Newtonian liquid are investigated under homogeneous shear flows that are modelled by the Lees-Edwards sliding bi-periodic frame. The equations of motion for the fluid and suspended particles are solved in a decoupled manner, where methods of computing the unknown rigid body motion are derived. Focus is also given to the implementation of shear bi-periodic boundary conditions. The proposed method is verified with bi-periodic shear flows of one, two and many particles. Results obtained, including relative viscosity and flow index, are in good agreement with those predicted by analytic theories and some other direct simulation methods.

7.1 Introduction

Particulate suspensions occur in many industrial applications such as food production, particles coating and plastics processing. The prediction of rheological properties of such complex materials is of great academic and industrial interest. The rheology of suspensions depends on many factors such as particle-volume fraction, particle size, particle geometry and flow rate. Due to the evolution of microstructure (i.e. spatial arrangement of rigid particles) in suspensions, the use of experiments to determine their macroscopic rheological properties is seen

to have its limitations and, in some cases, impractical (Phan-Thien and Kim, 1994). Analytical studies have traditionally been based on continuum models, but these models may involve some microscopic parameters which are difficult to determine (Raiskinmki et al., 2000). Numerical methods have emerged as a powerful predictive tool, complementing experiments and analytic models. If the fundamental equations for particles (Newton-Euler equations) and a fluid (Navier-Stokes equations) are solved in a direct manner, one speaks of direct numerical simulations (DNS). Advantages of DNSs lie in (i) easily handling particles of different shapes and sizes as well as any type of fluid; and (ii) having the ability to directly calculate the hydrodynamic forces and torques from the fluid flow. Based on the fluid-phase solver employed, DNS methods can be classified into two categories. In the first category, a mesh follows the movement of the particles, i.e. moving mesh. Methods in this category are usually based on Arbitrary Lagrangian-Eulerian (ALE) approach, e.g. Hu et al. (1992), Hu (1995) and Hu et al. (2001). In the second category, a mesh covers the whole domain and is independent of the particle movement, i.e. a fixed mesh. Methods in this category include the fictitious domain method (FDM) (Glowinski et al., 1994), immersed boundary method (IBM) (Peskin, 1972), virtual boundary method (VBM) (Saiki and Biringen, 1996), immersed interface method (IIM) (Leveque and Li, 1994), sharp interface method (SIM) (also called ghost-fluid method) (Liu et al., 2000) and fictitious boundary method (FBM) (Turek et al., 2003). The second group is more efficient and thus can deal with the particulate system with larger number of particles.

In the direct simulation of particulate flows under simple shear conditions, Hwang et al. (2004) proposed to consider the flow in a sliding frame with boundary conditions imposed periodically in all directions - the Lees-Edwards conditions (Lees and Edwards, 1972). The computational sliding frame was employed in the context of fictitious-domain finite element methods (Hwang et al., 2004) and radial basis function (RBF) methods (Le-Cao et al., 2010).

RBF networks (RBFNs) have emerged as a powerful tool for scattered data approximations. The application of RBFNs for the solution of ordinary (ODEs) and partial (PDEs) differential equations was first presented by Kansa (1990a). Since then, it has received a great deal of attention from both sciences and engineering communities. RBF methods have been used with success to solve various differential problems: heat transfer problems (Divo and Kassab, 2007; Šarler et al., 2004), fluid flows (Sanyasiraju and Chandhini, 2008; Demirkaya et al., 2008), fluid-solid interactions (de Boer et al., 2007; Beckert and Wendland, 2001), etc. To avoid the reduction of convergence rate caused by differentiation, Mai-Duy and Tran-Cong (2001a) proposed the use of integration to construct RBFN expressions (IRBFNs). The integration approach was implemented in a global form (Mai-Duy and Tran-Cong, 2001a), 1D global form (1D-IRBF) for 2D problems (Mai-Duy and Tanner, 2007; Mai-Duy and Tran-Cong, 2005), local form (Mai-Duy and Tran-Cong, 2009) and compact local form for second- and fourth-order elliptic PDEs (Mai-Duy et al., 2014; Thai-Quang, Le-Cao, Mai-

Duy, Tran and Tran-Cong, 2013; Mai-Duy and Tran-Cong, 2013; Thai-Quang, Le-Cao, Mai-Duy and Tran-Cong, 2012; Mai-Duy and Tran-Cong, 2011). For compact local form, sparse system matrices and high levels of accuracy are achieved altogether. The latter comes from the inclusion of the information about the governing equation or derivatives of the field variable in local approximations.

In this chapter, we apply a DFIB-CIRBF method (Thai-Quang, Mai-Duy, Tran and Tran-Cong, 2013) for the direct simulation of particulate Newtonian suspensions, with a focus on the computation of the rigid body motion and the implementation of shear bi-periodic boundary conditions. The method is consisted of three main components: (i) the direct forcing immersed boundary (DFIB) method for the handling of multiply-connected domains that vary with time; (ii) compact integrated radial basis function (CIRBF) approximations for the spatial discretisation, and (iii) second-order Adams-Bashforth/Crank-Nicolson algorithms for the temporal discretisation. An interactive force, representing the effects of rigid particles on the fluid region, is added directly to the governing Navier-Stokes equations (i.e. direct forcing) to enforce the non-slip boundary conditions on the fluid-particle interfaces. The interactive force is evaluated explicitly from the pressure gradient, convection and diffusion terms from the previous time level. Because the Eulerian grid nodes do not generally coincide with Lagrangian nodes representing the interfaces, a smoothed version of discrete delta function is employed to transfer the quantities between the two frames of nodes. The CIRBF approximations are constructed over 3-point stencils with extra information being nodal first- and second-order derivative values (Thai-Quang, Mai-Duy, Tran and Tran-Cong, 2012). The present computational procedure is verified with suspensions of identical particles of circular cylindrical shape. The remainder of the chapter is organised as follows. Section 7.2 outlines the mathematical equations for the particulate suspension and for the direct forcing (DF) approach in IBM. Section 7.3 provides some detailed descriptions of the numerical formulation including the temporal and spatial discretisations. In Section 7.4, numerical results are presented and compared with some data in the literature. Finally, Section 7.5 concludes the paper.

7.2 Mathematical formulation

In this section, we describe the equations used for the simulation of Newtonian suspensions of inertialess rigid particles in two dimensions, including the Navier-Stokes equation, sliding bi-periodic boundary conditions and the immersed boundary (IB) formulation. Let us consider the entire domain Ω including the fluid region Ω_f and particle region Ω_p occupied by the N_p particles P_k $\left(\Omega_p = \bigcup_{k=1}^{N_p} P_k\right)$ as shown in Figure 7.1. The outer boundary of the entire domain is $\Gamma = \bigcup_{q=1}^{4} \Gamma_q$ and the boundary surface of kth particle P_k is ∂P_k.

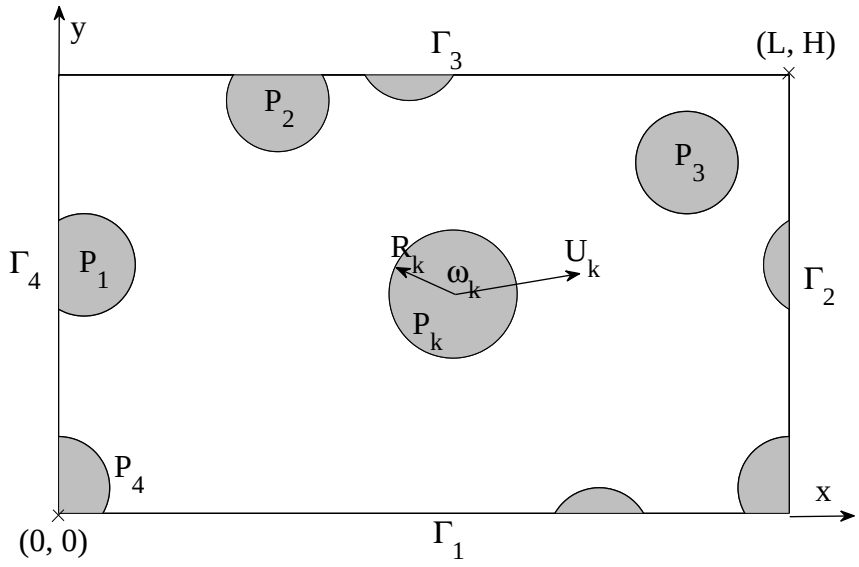


Figure 7.1: A sliding bi-periodic frame with crossing and non-crossing suspended particles.

7.2.1 Fluid motion

The creeping motion of a Newtonian fluid in a sliding bi-periodic domain Ω at a shear time t is governed by the Stokes equation (Hwang et al., 2004).

$$\nabla.\mathbf{u} = 0 \qquad \text{in} \quad \Omega,\ t \geq 0, \tag{7.1}$$

$$-\nabla p + \eta_s \nabla^2 \mathbf{u} = 0 \qquad \text{in} \quad \Omega,\ t \geq 0, \tag{7.2}$$

where η_s is the kinematic viscosity of the solvent (fluid).

In IBM, an interactive forcing term ($\mathbf{f}_I$) is added to (7.2) to represent the effect of rigid particles on the fluid. We will employ a pseudo-time marching technique to obtain the solution to equation (7.2) which is thus rewritten in the unsteady form as

$$\frac{\partial \mathbf{u}}{\partial t_p} = -\nabla p + \eta_s \nabla^2 \mathbf{u} + \mathbf{f}_I, \tag{7.3}$$

where t_p is the pseudo-time. Once the particles are replaced by the equivalent body force field $\mathbf{f}_I$, the entire volume now contains only the fluid whose motion is governed by (7.3). Equation (7.3) will be integrated with respect to the pseudo time. When the difference of the field variables between two successive time levels is less than a small tolerance, the iterative process is said to be converged and the obtained solution to (7.3) is also a solution to (7.2).

7.2.2 Sliding bi-periodic frame concept

A large domain of the flow under simple shear conditions in the x-direction can be divided into a set of identical small frames of width L and height H (Figure 7.1). Each frame translates along the shear direction at its own average velocity. A row of frames slides relatively to the adjacent one by an amount $\Delta = \dot{\gamma} H t$ (Hwang et al., 2004), where $\dot{\gamma}$ is the given shear rate. Because frames have similar solutions, we consider only one frame. About the properties of a sliding bi-periodic frame, the readers are referred to Hwang et al. (2004) for a detailed discussion. In a frame without particles, it is seen that the velocity profile is linear

$$u = u_0 + \dot{\gamma} y, \tag{7.4}$$

$$v = 0, \tag{7.5}$$

where $\mathbf{u}_0 = (u_0, 0)^T$ is the translational velocity vector of the frame; and y the local coordinate within a frame ($0 \leq y \leq H$). With the presence of particles, one has

$$u = u' + u_0 + \dot{\gamma} y, \tag{7.6}$$

$$v = v', \tag{7.7}$$

where u' and v' are the velocity perturbations by the particles from the linear profile. Since the solution is continuous across frames, the following shear bi-periodic boundary conditions for the velocity vector $\mathbf{u} = (u, v)$ and the traction vector $\boldsymbol{\tau} = (\tau_x, \tau_y)$ can be applied to each frame (Hwang et al., 2004).

$$\mathbf{u}(0, y, t) = \mathbf{u}(L, y, t), \quad 0 \leq y \leq H, \tag{7.8}$$

$$\boldsymbol{\tau}(0, y, t) = \boldsymbol{\tau}(L, y, t), \quad 0 \leq y \leq H, \tag{7.9}$$

for two vertical boundaries, and

$$\mathbf{u}(x, H, t) = \mathbf{u}(\{x - \dot{\gamma} H t\}^*, 0, t) + (\dot{\gamma} H, 0)^T, \quad 0 \leq x \leq L, \tag{7.10}$$

$$\boldsymbol{\tau}(x, H, t) = \boldsymbol{\tau}(\{x - \dot{\gamma} H t\}^*, 0, t), \quad 0 \leq x \leq L, \tag{7.11}$$

for two horizontal boundaries, where $\{.\}^*$ denotes a modular function of L, e.g. $\{1.7L\}^* = 0.7L$ and $\{-1.7L\}^* = 0.3L$. It is seen that the horizontal sliding periodicity between Γ_2 and Γ_4 is time-independent but the vertical sliding periodicity between Γ_1 and Γ_3 is time-dependent (Hwang et al., 2004). In terms of velocity and pressure, (7.9) and (7.11) take the forms

$$-p(0, y, t) + 2\eta_s \frac{\partial u}{\partial x}(0, y, t) = -p(L, y, t) + 2\eta_s \frac{\partial u}{\partial x}(L, y, t), \tag{7.12}$$

$$\eta_s \left[\frac{\partial u}{\partial y}(0, y, t) + \frac{\partial v}{\partial x}(0, y, t) \right] = \eta_s \left[\frac{\partial u}{\partial y}(L, y, t) + \frac{\partial v}{\partial x}(L, y, t) \right], \tag{7.13}$$

and

$$\eta_s \left[\frac{\partial v}{\partial x}(x,H,t) + \frac{\partial u}{\partial y}(x,H,t) \right] = \eta_s \left[\frac{\partial v}{\partial x}(\{x-\dot{\gamma}Ht\}^*,0,t) + \frac{\partial u}{\partial y}(\{x-\dot{\gamma}Ht\}^*,0,t) \right], \quad (7.14)$$

$$-p(x,H,t) + 2\eta_s \frac{\partial v}{\partial y}(x,H,t) = -p(\{x-\dot{\gamma}Ht\}^*,0,t) + 2\eta_s \frac{\partial v}{\partial y}(\{x-\dot{\gamma}Ht\}^*,0,t), \quad (7.15)$$

respectively.

The handling of pressure values on the interfaces was not discussed in the paper of Hwang et al. (2004); here, they are considered as unknowns, and, in this type of flow, we impose

$$p(0,y,t) = p(L,y,t), \quad 0 \le y \le H, \quad (7.16)$$

$$\frac{\partial p}{\partial x}(0,y,t) = \frac{\partial p}{\partial x}(L,y,t), \quad 0 \le y \le H, \quad (7.17)$$

for two vertical boundaries, and

$$p(x,H,t) = p(\{x-\dot{\gamma}Ht\}^*,0,t), \quad 0 \le x \le L, \quad (7.18)$$

$$\frac{\partial p}{\partial y}(x,H,t) = \frac{\partial p}{\partial y}(\{x-\dot{\gamma}Ht\}^*,0,t), \quad 0 \le x \le L, \quad (7.19)$$

for two horizontal boundaries.

Equations (7.8)-(7.11) together with (7.16)-(7.19) and (7.1)-(7.3) complete a set of governing equations for fluid motion.

7.2.3 Direct forcing method

The derivations of the interactive body force field for the problems in this chapter are similar to those in section 5.3.1 as follows.

At the present shear time n, the temporal discretisation of the momentum equation (7.3) at the current pseudo-time level $\widetilde{n}$ (iterative step) can be written in the form

$$\frac{\mathbf{u}^{n,\widetilde{n}} - \mathbf{u}^{n,\widetilde{n}-1}}{\Delta t_p} = \mathbf{rhs}^{n,\widetilde{n}-1/2} + \mathbf{f}_I^{n,\widetilde{n}-1/2}, \quad (7.20)$$

where Δt_p is the pseudo-time step. In (7.20), the pressure and diffusion terms at the shear time level t^n and pseudo-time level $t_p^{\widetilde{n}-1/2}$ are lumped together in $\mathbf{rhs}^{n,\widetilde{n}-1/2}$. It is noted that the current iterative step $\widetilde{n}$ is different from the present shear time level n. The former is a pseudo-time introduced for the purpose of facilitating an iterative calculation, while the latter is the physical time. For brevity, the superscript n is omitted since the field variables below are all associated with the present shear time n. Equation (7.20) thus reduces to

$$\frac{\mathbf{u}^{\widetilde{n}} - \mathbf{u}^{\widetilde{n}-1}}{\Delta t_p} = \mathbf{rhs}^{\widetilde{n}-1/2} + \mathbf{f}_I^{\widetilde{n}-1/2}. \tag{7.21}$$

The interactive force term yielding the desired velocity $\mathbf{u}^{(d)}$ can thus be defined as (Fadlun et al., 2000)

$$\mathbf{f}_I^{\widetilde{n}-1/2} = \frac{\mathbf{u}^{(d),\widetilde{n}} - \mathbf{u}^{\widetilde{n}-1}}{\Delta t_p} - \mathbf{rhs}^{\widetilde{n}-1/2}, \tag{7.22}$$

at some selected nodes (and zero elsewhere). The corresponding interactive force at the Lagrangian nodes will be

$$\mathbf{F}_I^{\widetilde{n}-1/2} = \frac{\mathbf{U}^{(d),\widetilde{n}} - \mathbf{U}^{\widetilde{n}-1}}{\Delta t_p} - \mathbf{RHS}^{\widetilde{n}-1/2}. \tag{7.23}$$

The desired velocity at a node on the fluid-particle interface in (7.23) is simply computed from the rigid-body motion of the particles

$$\mathbf{U}^{(d),\widetilde{n}}(\mathbf{X}) = \mathbf{U}_p^{\widetilde{n}} + \boldsymbol{\omega}_p^{\widetilde{n}} \times \left(\mathbf{X}^{\widetilde{n}} - \mathbf{X}_p^{\widetilde{n}}\right). \tag{7.24}$$

When the interactive force is absent, the preliminary velocity $\widetilde{\mathbf{U}}^{\widetilde{n}}$ is obtained from equation (7.23) as

$$\widetilde{\mathbf{U}}^{\widetilde{n}} = \mathbf{U}^{\widetilde{n}-1} + \mathbf{RHS}^{\widetilde{n}-1/2}\Delta t_p. \tag{7.25}$$

Its Eulerian counterpart is

$$\widetilde{\mathbf{u}}^{\widetilde{n}} = \mathbf{u}^{\widetilde{n}-1} + \mathbf{rhs}^{\widetilde{n}-1/2}\Delta t_p. \tag{7.26}$$

The term $\mathbf{rhs}^{\widetilde{n}-1/2}$ in (7.26) is computed explicitly as

$$\mathbf{rhs}^{\widetilde{n}-1/2} = -\left[\frac{3}{2}\nabla p^{\widetilde{n}-1} - \frac{1}{2}\nabla p^{\widetilde{n}-2}\right] + \eta_s\left[\frac{3}{2}\nabla^2\mathbf{u}^{\widetilde{n}-1} - \frac{1}{2}\nabla^2\mathbf{u}^{\widetilde{n}-2}\right]. \tag{7.27}$$

Now, one can compute the interactive force at Lagrangian nodes in the form of

$$\mathbf{F}_I^{\widetilde{n}-1/2} = \frac{\mathbf{U}^{(d),\widetilde{n}} - \widetilde{\mathbf{U}}^{\widetilde{n}}}{\Delta t_p}. \tag{7.28}$$

In order to complete the evaluation of the interactive force term in (7.21), a mechanism for transferring the preliminary velocities $(\widetilde{\mathbf{u}}^{\widetilde{n}}, \widetilde{\mathbf{U}}^{\widetilde{n}})$ and the forces $(\mathbf{F}_I^{\widetilde{n}-1/2}, \mathbf{f}_I^{\widetilde{n}-1/2})$ between the two Eulerian and Lagrangian node systems is required. The readers are referred to section 6.2.2 for detailed descriptions of this transfer process.

7.2.4 Particle motion

Let $\mathbf{F}^H$ and $\mathbf{T}^H$ be the total force and torque acting on a suspended particle, respectively. We now derive expressions for computing the linear and angular velocities of a suspended particle for the following two cases.

Non-boundary-crossing particles

Consider a suspended particle that lies within the computational domain. There are hydrodynamic forces/torques acting on it. At the current shear time level n and iterative step $\widetilde{n}$, one has

$$\mathbf{F}^{H,n,\widetilde{n}} = -\sum_{l=1}^{N_L} \mathbf{F}_{I,l}^{n,\widetilde{n}-1/2}, \tag{7.29}$$

$$\mathbf{T}^{H,n,\widetilde{n}} = -\sum_{l=1}^{N_L} \left(\mathbf{X}_l^{n,\widetilde{n}} - \mathbf{X}_p^{n,\widetilde{n}}\right) \times \mathbf{F}_{I,l}^{n,\widetilde{n}-1/2}. \tag{7.30}$$

Substituting (7.28) into (7.29)-(7.30) and making use of (7.24) yield

$$\begin{aligned}\mathbf{F}^{H,n,\widetilde{n}} &= -\sum_{l=1}^{N_L} \frac{\mathbf{U}_p^{n,\widetilde{n}} + \boldsymbol{\omega}_p^{n,\widetilde{n}} \times \left(\mathbf{X}_l^{n,\widetilde{n}} - \mathbf{X}_p^{n,\widetilde{n}}\right) - \widetilde{\mathbf{U}}_l^{n,\widetilde{n}}}{\Delta t_p} \\ &= -\frac{N_L}{\Delta t_p}\mathbf{U}_p^{n,\widetilde{n}} - \frac{1}{\Delta t_p}\boldsymbol{\omega}_p^{n,\widetilde{n}} \times \sum_{l=1}^{N_L}\left(\mathbf{X}_l^{n,\widetilde{n}} - \mathbf{X}_p^{n,\widetilde{n}}\right) + \frac{1}{\Delta t_p}\sum_{l=1}^{N_L}\widetilde{\mathbf{U}}_l^{n,\widetilde{n}},\end{aligned} \tag{7.31}$$

$$\mathbf{T}^{H,n,\widetilde{n}} = -\sum_{l=1}^{N_L}\left(\mathbf{X}_l^{n,\widetilde{n}} - \mathbf{X}_p^{n,\widetilde{n}}\right) \times \frac{\mathbf{U}_p^{n,\widetilde{n}} + \boldsymbol{\omega}_p^{n,\widetilde{n}} \times \left(\mathbf{X}_l^{n,\widetilde{n}} - \mathbf{X}_p^{n,\widetilde{n}}\right) - \widetilde{\mathbf{U}}_l^{n,\widetilde{n}}}{\Delta t_p}$$

$$= -\frac{1}{\Delta t_p}\sum_{l=1}^{N_L}\left(\mathbf{X}_l^{n,\widetilde{n}} - \mathbf{X}_p^{n,\widetilde{n}}\right) \times \mathbf{U}_p^{n,\widetilde{n}}$$

$$-\frac{1}{\Delta t_p}\sum_{l=1}^{N_L}\left(\mathbf{X}_l^{n,\widetilde{n}} - \mathbf{X}_p^{n,\widetilde{n}}\right) \times \left[\boldsymbol{\omega}_p^{n,\widetilde{n}} \times \left(\mathbf{X}_l^{n,\widetilde{n}} - \mathbf{X}_p^{n,\widetilde{n}}\right)\right]$$

$$+\frac{1}{\Delta t_p}\sum_{l=1}^{N_L}\left(\mathbf{X}_l^{n,\widetilde{n}} - \mathbf{X}_p^{n,\widetilde{n}}\right) \times \widetilde{\mathbf{U}}_l^{n,\widetilde{n}}. \quad (7.32)$$

Since the iteration implemented here is just a false transient process (t_p is a pseudo-time), we will keep the configuration of the particles unchanged (the initial configuration is retained) during the solving process for t^n. In this study, the particle is considered massless and therefore both external forces and torques are equal to zero. As a result, the left sides of (7.31)-(7.32) vanish and the two corresponding equations, in the case of two dimensions, can be solved for the motion of the particle as

$$\begin{pmatrix} U_p^{n,\widetilde{n}} \\ V_p^{n,\widetilde{n}} \\ \omega_p^{n,\widetilde{n}} \end{pmatrix} =$$

$$\begin{bmatrix} N_L & 0 & -\sum_{l=1}^{N_L}\left(Y_l^n - Y_p^n\right) \\ 0 & N_L & \sum_{l=1}^{N_L}\left(X_l^n - X_p^n\right) \\ \sum_{l=1}^{N_L}\left(Y_l^n - Y_p^n\right) & -\sum_{l=1}^{N_L}\left(X_l^n - X_p^n\right) & -\sum_{l=1}^{N_L}\left(\left(X_l^n - X_p^n\right)^2 + \left(Y_l^n - Y_p^n\right)^2\right) \end{bmatrix}^{-1}$$

$$\begin{bmatrix} \sum_{l=1}^{N_L}\widetilde{U}_l^{n,\widetilde{n}} \\ \sum_{l=1}^{N_L}\widetilde{V}_l^{n,\widetilde{n}} \\ \sum_{l=1}^{N_L}\left(\left(Y_l^n - Y_p^n\right)\widetilde{U}_l^{n,\widetilde{n}} - \left(X_l^n - X_p^n\right)\widetilde{V}_l^{n,\widetilde{n}}\right) \end{bmatrix}. \quad (7.33)$$

Boundary-crossing particles

When a suspended particle crosses the boundary (leaving the domain), another particle must enter the domain simultaneously by virtue of the periodic boundary conditions. In terms of numerical implementation, its outside part can be brought back to the computational domain with a modified position and velocity.

Let $\mathbf{x}' = (x', y')^T$ be the relocated position. At a given $(\dot{\gamma}, H, t)$, the relocated position $\mathbf{x}'$ is determined from the original position $\mathbf{x}$ as

$$(x', y') = \left(\{x - \dot{\gamma}Ht\}^*, y - H\right), \quad y > H, \quad (7.34)$$

$$(x', y') = (\{x + \dot{\gamma} H t\}^*, y + H), \quad y < 0, \tag{7.35}$$

$$(x', y') = (\{x\}^*, y), \quad x < 0 \text{ or } x > L. \tag{7.36}$$

It is noted that (7.34)-(7.36) are also applicable to Lagrangian nodes.

If the particle crosses the upper/lower boundary, one also needs to modify its velocities to account for the sliding effects. The x-velocity U' at the relocated position is determined from the original x-velocity U as

$$U' = U - \dot{\gamma} H, \quad y > H, \tag{7.37}$$

$$U' = U + \dot{\gamma} H, \quad y < 0. \tag{7.38}$$

The total force and torque exerted on the particle can be calculated as

$$\mathbf{F}^{H,n,\widetilde{n}} = -\sum_{l=1}^{N_{L_i}} \mathbf{F}_{I,l}^{n,\widetilde{n}-1/2} - \sum_{l'=1}^{N_{L_o}} \mathbf{F}_{I,l'}^{n,\widetilde{n}-1/2}, \tag{7.39}$$

$$\mathbf{T}^{H,n,\widetilde{n}} = -\sum_{l=1}^{N_{L_i}} \left(\mathbf{X}_l^{n,\widetilde{n}} - \mathbf{X}_p^{n,\widetilde{n}} \right) \times \mathbf{F}_{I,l}^{n,\widetilde{n}-1/2} - \sum_{l'=1}^{N_{L_o}} \left(\mathbf{X}_{l'}^{n,\widetilde{n}} - \mathbf{X}_{p'}^{n,\widetilde{n}} \right) \times \mathbf{F}_{I,l'}^{n,\widetilde{n}-1/2}. \tag{7.40}$$

where N_{L_i} and N_{L_o} are the total number of Lagrangian nodes representing the inside and outside parts of the particle, respectively ($N_{L_i} + N_{L_o} = N_L$). Subscript l indicates the original Lagrangian nodes while l' indicates the relocated Lagrangian nodes. Similarly, subscripts p and p' indicate the original and relocated particle centres, respectively. Since $\mathbf{X}_{l'} - \mathbf{X}_{p'} = \mathbf{X'}_l - \mathbf{X'}_p = \mathbf{X}_l - \mathbf{X}_p$, equation (7.40) is rewritten as

$$\mathbf{T}^{H,n,\widetilde{n}} = -\sum_{l=1}^{N_{L_i}} \left(\mathbf{X}_l^{n,\widetilde{n}} - \mathbf{X}_p^{n,\widetilde{n}} \right) \times \mathbf{F}_{I,l}^{n,\widetilde{n}-1/2} - \sum_{l'=1, l=l'}^{N_{L_o}} \left(\mathbf{X}_l^{n,\widetilde{n}} - \mathbf{X}_p^{n,\widetilde{n}} \right) \times \mathbf{F}_{I,l'}^{n,\widetilde{n}-1/2}. \tag{7.41}$$

The solving process here is similar to that for the case of non-crossing particles. Forces on the right sides of (7.39) and (7.41) are first expressed in terms of the

linear and angular velocities of the particle, leading to

$$\begin{aligned}\mathbf{F}^{H,n,\widetilde{n}} &= -\sum_{l=1}^{N_{L_i}} \frac{\mathbf{U}_p^{n,\widetilde{n}} + \boldsymbol{\omega}_p^{n,\widetilde{n}} \times \left(\mathbf{X}_l^{n,\widetilde{n}} - \mathbf{X}_p^{n,\widetilde{n}}\right) - \widetilde{\mathbf{U}}_l^{n,\widetilde{n}}}{\Delta t_p} \\ &- \sum_{l'=1,l=l'}^{N_{Lo}} \frac{\left(\mathbf{U}_p^{n,\widetilde{n}} + (sign\dot{\gamma}H,0)^T\right) + \boldsymbol{\omega}_p^{n,\widetilde{n}} \times \left(\mathbf{X}_l^{n,\widetilde{n}} - \mathbf{X}_p^{n,\widetilde{n}}\right) - \widetilde{\mathbf{U}}_{l'}^{n,\widetilde{n}}}{\Delta t_p} \\ &= -\frac{N_L}{\Delta t_p}\mathbf{U}_p^{n,\widetilde{n}} - \frac{1}{\Delta t_p}\boldsymbol{\omega}_p^{n,\widetilde{n}} \times \sum_{l=1}^{N_L}\left(\mathbf{X}_l^{n,\widetilde{n}} - \mathbf{X}_p^{n,\widetilde{n}}\right) + \frac{1}{\Delta t_p}\sum_{l=1}^{N_{L_i}}\widetilde{\mathbf{U}}_l^{n,\widetilde{n}} + \frac{1}{\Delta t_p}\sum_{l'=1}^{N_{Lo}}\widetilde{\mathbf{U}}_{l'}^{n,\widetilde{n}} \\ &\quad - \frac{N_{L_o}}{\Delta t_p}(sign\dot{\gamma}H,0)^T, \quad (7.42)\end{aligned}$$

$$\begin{aligned}\mathbf{T}^{H,n,\widetilde{n}} &= -\sum_{l=1}^{N_{L_i}}\left(\mathbf{X}_l^{n,\widetilde{n}} - \mathbf{X}_p^{n,\widetilde{n}}\right) \times \frac{\mathbf{U}_p^{n,\widetilde{n}} + \boldsymbol{\omega}_p^{n,\widetilde{n}} \times \left(\mathbf{X}_l^{n,\widetilde{n}} - \mathbf{X}_p^{n,\widetilde{n}}\right) - \widetilde{\mathbf{U}}_l^{n,\widetilde{n}}}{\Delta t_p} \\ &- \sum_{l'=1,l=l'}^{N_{Lo}}\left(\mathbf{X}_l^{n,\widetilde{n}} - \mathbf{X}_p^{n,\widetilde{n}}\right) \times \frac{\left(\mathbf{U}_p^{n,\widetilde{n}} + (sign\dot{\gamma}H,0)^T\right) + \boldsymbol{\omega}_p^{n,\widetilde{n}} \times \left(\mathbf{X}_l^{n,\widetilde{n}} - \mathbf{X}_p^{n,\widetilde{n}}\right) - \widetilde{\mathbf{U}}_{l'}^{n,\widetilde{n}}}{\Delta t_p} \\ &= -\frac{1}{\Delta t_p}\sum_{l=1}^{N_L}\left(\mathbf{X}_l^{n,\widetilde{n}} - \mathbf{X}_p^{n,\widetilde{n}}\right) \times \mathbf{U}_p^{n,\widetilde{n}} \\ &- \frac{1}{\Delta t_p}\sum_{l=1}^{N_L}\left(\mathbf{X}_l^{n,\widetilde{n}} - \mathbf{X}_p^{n,\widetilde{n}}\right) \times \left[\boldsymbol{\omega}_p^{n,\widetilde{n}} \times \left(\mathbf{X}_l^{n,\widetilde{n}} - \mathbf{X}_p^{n,\widetilde{n}}\right)\right] \\ &+ \frac{1}{\Delta t_p}\sum_{l=1}^{N_{L_i}}\left(\mathbf{X}_l^{n,\widetilde{n}} - \mathbf{X}_p^{n,\widetilde{n}}\right) \times \widetilde{\mathbf{U}}_l^{n,\widetilde{n}} + \frac{1}{\Delta t_p}\sum_{l'=1,l=l'}^{N_{Lo}}\left(\mathbf{X}_l^{n,\widetilde{n}} - \mathbf{X}_p^{n,\widetilde{n}}\right) \times \widetilde{\mathbf{U}}_{l'}^{n,\widetilde{n}} \\ &\quad - \frac{1}{\Delta t_p}\sum_{l=1}^{N_{Lo}}\left(\mathbf{X}_l^{n,\widetilde{n}} - \mathbf{X}_p^{n,\widetilde{n}}\right) \times (sign\dot{\gamma}H,0)^T, \quad (7.43)\end{aligned}$$

where $sign$ is the sign function defined as

$$sign = \begin{cases} -1, & Y_l > H, \\ 1, & Y_l < 0, \\ 0, & \text{others.} \end{cases} \quad (7.44)$$

Then, one solves (7.42) and (7.43) with their zero left sides, resulting in

$$\begin{pmatrix} U_p^{n,\widetilde{n}} \\ V_p^{n,\widetilde{n}} \\ \omega_p^{n,\widetilde{n}} \end{pmatrix} = \begin{bmatrix} N_L & 0 & -\sum_{l=1}^{N_L}\left(Y_l^n - Y_p^n\right) \\ 0 & N_L & \sum_{l=1}^{N_L}\left(X_l^n - X_p^n\right) \\ \sum_{l=1}^{N_L}\left(Y_l^n - Y_p^n\right) & -\sum_{l=1}^{N_L}\left(X_l^n - X_p^n\right) & -\sum_{l=1}^{N_L}\left(\left(X_l^n - X_p^n\right)^2 + \left(Y_l^n - Y_p^n\right)^2\right) \end{bmatrix}^{-1} \begin{bmatrix} \sum_{l=1}^{N_{L_i}} \widetilde{U}_l^{n,\widetilde{n}} + \sum_{l'=1}^{N_{L_o}} \widetilde{U}_{l'}^{n,\widetilde{n}} - N_{L_o} sign\dot{\gamma} H \\ \sum_{l=1}^{N_{L_i}} \widetilde{V}_l^{n,\widetilde{n}} + \sum_{l'=1}^{N_{L_o}} \widetilde{V}_{l'}^{n,\widetilde{n}} \\ \sum_{l=1}^{N_{L_i}}\left(\left(Y_l^n - Y_p^n\right)\widetilde{U}_l^{n,\widetilde{n}} - \left(X_l^n - X_p^n\right)\widetilde{V}_l^{n,\widetilde{n}}\right) + \\ \sum_{l'=1,l=l'}^{N_{L_o}}\left(\left(Y_l^n - Y_p^n\right)\widetilde{U}_{l'}^{n,\widetilde{n}} - \left(X_l^n - X_p^n\right)\widetilde{V}_{l'}^{n,\widetilde{n}}\right) - \\ \sum_{l=1}^{N_{L_o}}\left(Y_l^n - Y_p^n\right) sign\dot{\gamma} H \end{bmatrix}. \quad (7.45)$$

The motions of the liquid and rigid particles at the shear time level t^n (i.e. $\mathbf{u}^n, p^n, \mathbf{U}_p^n$ and $\boldsymbol{\omega}_p^n$) are obtained by iteratively solving equations (7.1), (7.3), (7.8)-(7.11), (7.16)-(7.19) and (7.33)/(7.45) (for non-boundary-crossing/boundary-crossing particle). The particle configuration at the next shear time level $(n+1)$ is derived the following kinematic equations

$$\frac{d\mathbf{X}_p}{dt} = \mathbf{U}_p, \quad (7.46)$$

$$\frac{d\boldsymbol{\theta}_p}{dt} = \boldsymbol{\omega}_p, \quad (7.47)$$

where $\boldsymbol{\theta}_p$ is the angular orientation of particles. Using the second-order Adams-Bashforth scheme for the temporal discretisations, equations (7.46)-(7.47) reduce to

$$\mathbf{X}_p^{n+1} = \mathbf{X}_p^n + \Delta t\left(\frac{3}{2}\mathbf{U}_p^n - \frac{1}{2}\mathbf{U}_p^{n-1}\right), \quad (7.48)$$

$$\boldsymbol{\theta}_p^{n+1} = \boldsymbol{\theta}_p^n + \Delta t\left(\frac{3}{2}\boldsymbol{\omega}_p^n - \frac{1}{2}\boldsymbol{\omega}_p^{n-1}\right). \quad (7.49)$$

7.2.5 Rheological properties

Bulk properties such as reduced/relative viscosities, and first and second normal stress differences are important results to characterise particulate suspensions (the non-Newtonian behaviour). For the present problem, the bulk stress is

calculated by (Hwang et al., 2004)

$$\langle \boldsymbol{\sigma} \rangle = \frac{1}{A} \int_{\Gamma} \mathbf{x} \boldsymbol{\tau} ds, \tag{7.50}$$

where A is the area of the domain, or in the detailed form

$$\begin{aligned} \langle \sigma_{xx} \rangle = \frac{1}{A} \int_{\Gamma} x \tau_x ds = -\frac{1}{A} \int_{\Gamma_1} x \eta_s \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) dx \\ + \frac{1}{A} \int_{\Gamma_3} x \eta_s \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) dx + \frac{1}{H} \int_{\Gamma_2} \left(-p + 2\eta_s \frac{\partial u}{\partial x} \right) dy, \end{aligned} \tag{7.51}$$

$$\langle \sigma_{yy} \rangle = \frac{1}{A} \int_{\Gamma} y \tau_y ds = \frac{1}{L} \int_{\Gamma_3} \left(-p + 2\eta_s \frac{\partial v}{\partial y} \right) dx, \tag{7.52}$$

$$\begin{aligned} \langle \sigma_{xy} \rangle = \frac{1}{A} \int_{\Gamma} x \tau_y ds = -\frac{1}{A} \int_{\Gamma_1} x \left(-p + 2\eta_s \frac{\partial v}{\partial y} \right) dx \\ + \frac{1}{A} \int_{\Gamma_3} x \left(-p + 2\eta_s \frac{\partial v}{\partial y} \right) dx + \frac{1}{H} \int_{\Gamma_2} \eta_s \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) dy, \end{aligned} \tag{7.53}$$

$$\langle \sigma_{yx} \rangle = \frac{1}{A} \int_{\Gamma} y \tau_x ds = \frac{1}{L} \int_{\Gamma_3} \eta_s \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) dx. \tag{7.54}$$

In the case of boundary-crossing particles (Figure 7.2), the formula needs be modified as

$$\langle \boldsymbol{\sigma}_c \rangle = \langle \boldsymbol{\sigma} \rangle - \frac{1}{A} \int_{\Gamma_c} \mathbf{x} \boldsymbol{\tau} ds, \tag{7.55}$$

where Γ_c is parts of boundaries crossed by particles ($\Gamma_c = \Gamma_{1,c} \cup \Gamma_{2,c} \cup \Gamma_{3,c} \cup \Gamma_{4,c}$).

Equation (7.50) can be written in a symmetric form as (Phan-Thien, 2013)

$$\langle \boldsymbol{\sigma} \rangle = \frac{1}{2A} \int_{\Gamma} \left(\mathbf{x} \boldsymbol{\tau} + \boldsymbol{\tau} \mathbf{x} \right) ds. \tag{7.56}$$

We employ Simpson's rule to evaluate integrals in (7.51)-(7.54). It is noted that $\langle \sigma_{xy} \rangle = \langle \sigma_{yx} \rangle$. The viscosity η and first normal-stress difference N_1 are obtained by further taking the averaging process in time

$$\eta = \frac{1}{K\dot{\gamma}} \int_0^K \langle \sigma_{xy} \rangle \, dt, \tag{7.57}$$

$$N_1 = \frac{1}{K} \int_0^K \langle \sigma_{xx} - \sigma_{yy} \rangle \, dt. \tag{7.58}$$

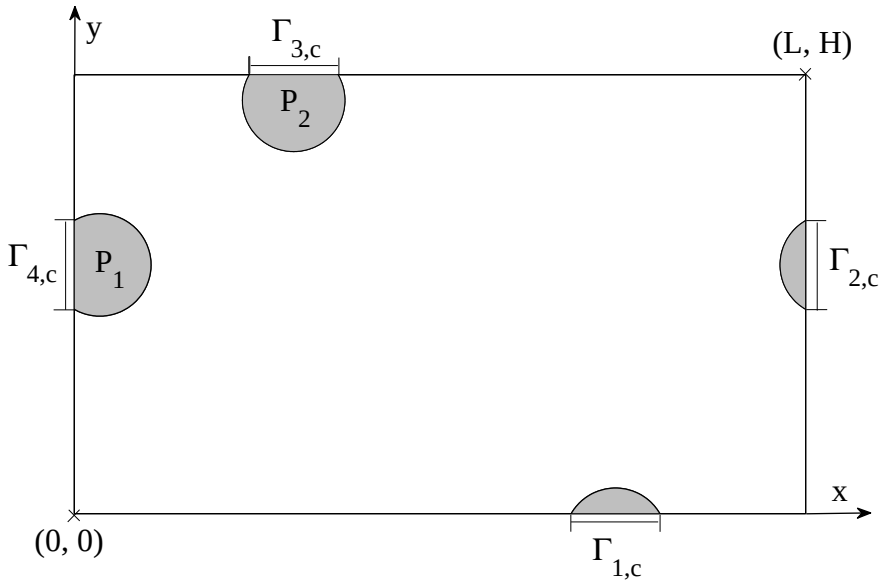


Figure 7.2: Illustration of parts of boundaries crossed by particles.

7.3 Numerical formulation

7.3.1 Spatial discretisation

The flow domain Ω is discretised using a Cartesian grid $n_x \times n_y$. We employ compact 3-nodes integrated RBF stencils (Thai-Quang, Mai-Duy, Tran and Tran-Cong, 2013) to approximate the field variables and their derivatives. It is noted that 1D global IRBF schemes (Mai-Duy and Tran-Cong, 2008) are used on the boundary Γ_1 to obtain boundary values at the sliding nodes.

7.3.2 Temporal discretisation

We discretise the time derivative term in (7.3) with a Crank-Nicolson scheme (Crank and Nicolson, 1996)

$$\frac{\mathbf{u}^{\widetilde{n}} - \mathbf{u}^{\widetilde{n}-1}}{\Delta t_p} = -\nabla p^{\widetilde{n}-1/2} + \frac{\eta_s}{2}\left(\nabla^2 \mathbf{u}^{\widetilde{n}} + \nabla^2 \mathbf{u}^{\widetilde{n}-1}\right) + \mathbf{f}_I^{\widetilde{n}-1/2}, \tag{7.59}$$

or

$$\left(I - \frac{\eta_s \Delta t_p}{2}\nabla^2\right)\mathbf{u}^{\widetilde{n}} + \Delta t_p \nabla p^{\widetilde{n}-1/2} = \left(I + \frac{\eta_s \Delta t_p}{2}\nabla^2\right)\mathbf{u}^{\widetilde{n}-1} + \Delta t_p \mathbf{f}_I^{\widetilde{n}-1/2}. \tag{7.60}$$

where I is the $N \times N$ identity matrix. For the present problem, the boundary conditions for the velocities are not given explicitly, but in the form of shear bi-periodic conditions. We will solve the set of governing equations (7.60) and (7.1) simultaneously. In the matrix-vector form, it is written as

$$\begin{pmatrix} \mathcal{A}_u & \mathcal{O} & \mathcal{G}_x \\ \mathcal{O} & \mathcal{A}_v & \mathcal{G}_y \\ \mathcal{D}_x & \mathcal{D}_y & \mathcal{O} \end{pmatrix} \begin{pmatrix} \widehat{u}^{\widetilde{n}} \\ \widehat{v}^{\widetilde{n}} \\ \widehat{p}^{\widetilde{n}-1/2} \end{pmatrix} = \begin{pmatrix} \widehat{ru}^{\widetilde{n}} \\ \widehat{rv}^{\widetilde{n}} \\ \mathrm{O} \end{pmatrix}, \tag{7.61}$$

where $\mathcal{A}_u$, $\mathcal{A}_v$, $\mathcal{G}_x$, $\mathcal{G}_y$, $\mathcal{D}_x$, $\mathcal{D}_y$ are $N \times N$ matrices defined as

$$\mathcal{A}_u = \mathcal{A}_v = I - \frac{\eta_s \Delta t_p}{2} \nabla^2, \tag{7.62}$$

$$\mathcal{G}_x = \Delta t_p \nabla_x, \quad \mathcal{G}_y = \Delta t_p \nabla_y, \tag{7.63}$$

$$\mathcal{D}_x = \nabla_x, \quad \mathcal{D}_y = \nabla_y, \tag{7.64}$$

$\widehat{ru}^{\widetilde{n}}$, $\widehat{rv}^{\widetilde{n}}$, $\widehat{u}^{\widetilde{n}}$, $\widehat{v}^{\widetilde{n}}$, $\widehat{p}^{\widetilde{n}-1/2}$ are vectors of length N defined as

$$ru^{\widetilde{n}} = \left(I + \frac{\eta_s \Delta t_p}{2} \nabla^2 \right) u^{\widetilde{n}-1} + \Delta t_p f_{I_x}^{\widetilde{n}-1/2}, \tag{7.65}$$

$$rv^{\widetilde{n}} = \left(I + \frac{\eta_s \Delta t_p}{2} \nabla^2 \right) v^{\widetilde{n}-1} + \Delta t_p f_{I_y}^{\widetilde{n}-1/2}, \tag{7.66}$$

$$\widehat{ru}^{\widetilde{n}} = \Big(ru_{1,1}^{\widetilde{n}}, ru_{1,2}^{\widetilde{n}}, \ldots, ru_{1,n_y}^{\widetilde{n}}, ru_{2,1}^{\widetilde{n}}, ru_{2,2}^{\widetilde{n}}, \ldots, ru_{2,n_y}^{\widetilde{n}}, \ldots, ru_{n_x,1}^{\widetilde{n}}, ru_{n_x,2}^{\widetilde{n}}, \ldots, ru_{n_x,n_y}^{\widetilde{n}} \Big)^T, \tag{7.67}$$

$$\widehat{rv}^{\widetilde{n}} = \Big(rv_{1,1}^{\widetilde{n}}, rv_{1,2}^{\widetilde{n}}, \ldots, rv_{1,n_y}^{\widetilde{n}}, rv_{2,1}^{\widetilde{n}}, rv_{2,2}^{\widetilde{n}}, \ldots, rv_{2,n_y}^{\widetilde{n}}, \ldots, rv_{n_x,1}^{\widetilde{n}}, rv_{n_x,2}^{\widetilde{n}}, \ldots, rv_{n_x,n_y}^{\widetilde{n}} \Big)^T, \tag{7.68}$$

$$\widehat{u}^{\widetilde{n}} = \Big(u_{1,1}^{\widetilde{n}}, u_{1,2}^{\widetilde{n}}, \ldots, u_{1,n_y}^{\widetilde{n}}, u_{2,1}^{\widetilde{n}}, u_{2,2}^{\widetilde{n}}, \ldots, u_{2,n_y}^{\widetilde{n}}, \ldots, u_{n_x,1}^{\widetilde{n}}, u_{n_x,2}^{\widetilde{n}}, \ldots, u_{n_x,n_y}^{\widetilde{n}} \Big)^T, \tag{7.69}$$

$$\widehat{v}^{\widetilde{n}} = \Big(v_{1,1}^{\widetilde{n}}, v_{1,2}^{\widetilde{n}}, \ldots, v_{1,n_y}^{\widetilde{n}}, v_{2,1}^{\widetilde{n}}, v_{2,2}^{\widetilde{n}}, \ldots, v_{2,n_y}^{\widetilde{n}}, \ldots, v_{n_x,1}^{\widetilde{n}}, v_{n_x,2}^{\widetilde{n}}, \ldots, v_{n_x,n_y}^{\widetilde{n}} \Big)^T, \quad (7.70)$$

$$\widehat{p}^{\widetilde{n}-1/2} = \Big(p_{1,1}^{\widetilde{n}-1/2}, p_{1,2}^{\widetilde{n}-1/2}, \ldots, p_{1,n_y}^{\widetilde{n}-1/2}, p_{2,1}^{\widetilde{n}-1/2}, p_{2,2}^{\widetilde{n}-1/2}, \ldots, p_{2,n_y}^{\widetilde{n}-1/2}, \ldots, p_{n_x,1}^{\widetilde{n}-1/2}, p_{n_x,2}^{\widetilde{n}-1/2}, \ldots, p_{n_x,n_y}^{\widetilde{n}-1/2} \Big)^T, \quad (7.71)$$

$\mathcal{O}$ is $N \times N$ zero matrix and O is zero vector of length N.

7.3.3 Solution procedure

The flow is simulated with respect to the shear time. At each shear time level t^n, the system of equations are integrated with respect to the pseudo-time t_p discretised by a series of $t^{\widetilde{n}}$. The solution procedure at a shear time level t^n can be summarised as follows.

- Step 1: Compute $\widetilde{\mathbf{u}}^{n,\widetilde{n}}$ from (7.26), and then $\widetilde{\mathbf{U}}^{n,\widetilde{n}}$ via (6.16) (at the begining, one has $\widetilde{n} = 1$).
- Step 2: Compute $\mathbf{U}_p^{n,\widetilde{n}}$ and $\boldsymbol{\omega}_p^{n,\widetilde{n}}$ (the particle motion) via (7.33) (non-boundary-crossing particle) and via (7.45) (boundary-crossing particles).
- Step 3: Compute $\mathbf{U}^{(d),n,\widetilde{n}}$ via (7.24) (and $U'^{(d),n,\widetilde{n}}$ via (7.37)-(7.38) if the particle crossing occurs). Compute $\mathbf{F}_I^{n,\widetilde{n}-1/2}$ from (7.28), and then $\mathbf{f}_I^{n,\widetilde{n}-1/2}$ via (6.17).
- Step 4: Solve the system formed by (7.61), (7.8)-(7.11) and (7.16)-(7.19) to obtain $\mathbf{u}^{n,\widetilde{n}}$ and $p^{n,\widetilde{n}-1/2}$.
- Step 5: Check the difference of the field variables between the current and previous iterations.

  If it is less than a specified tolerance (10^{-9}), update the position of particles, correct the position and velocity of crossing particles, and then proceed to next shear time level.

  If not, go back to Step 1.

7.4 Numerical results

A range of the β values, namely $\{20, 25, \ldots, 50\}$ is picked for the calculations in this chapter.

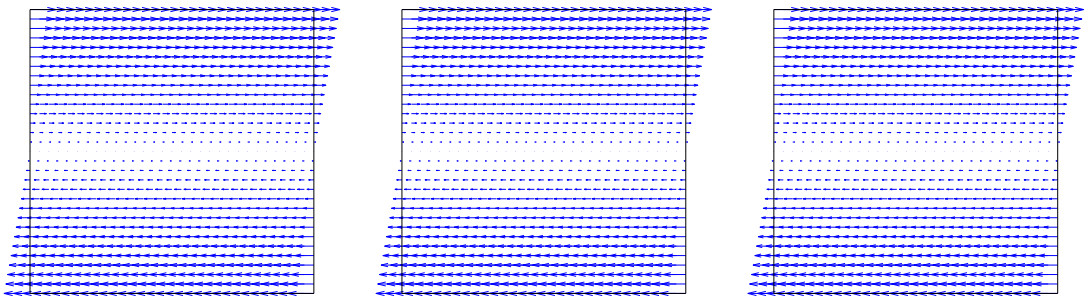
7.4.1 Analysis of periodic boundary conditions

Special care is needed in solving problems with periodic boundary conditions as they may have multiple solutions (each solution is different from another by a constant). For example, consider the following simple equation

$$\frac{\partial^2 u}{\partial x^2} = -\sin x, \quad 0 \leq x \leq 2\pi, \tag{7.72}$$

subject to periodic boundary conditions: $u(0) = u(2\pi)$ and $\partial u(0)/\partial x = \partial u(2\pi)/\partial x$. A solution to this problem can be verified to be $\overline{u} = \sin(x) + c$, where c is a constant. If one tries to solve this problem by a numerical discretisation scheme (e.g. a RBF method), the obtained system of algebraic equations will be ill-conditioned. To make the solution unique, an appropriate extra condition is required, e.g. the value of u at a point in the domain of interest. Taking the extra condition into account, the ill-conditioning problem of a numerical scheme is eliminated.

For the present problem, we simulate a flow in a rectangular frame, which is subject to periodic boundary conditions in both directions. To avoid the problems just mentioned above, apart from periodic boundary conditions, we also impose zero velocities at a point on the centreline. Consider a frame of dimensions 1×1, $\eta_s = 1$ and $\dot{\gamma} = 1$. Numerical results obtained using a uniform grid of 31×31 show that, with a reference point, similar velocity profiles are produced at different shear times (Figure 7.3), and the matrix condition number is in the range of 10^7-10^8 (Figure 7.4).



$t = 0.1$ $t = 0.5$ $t = 0.9$

Figure 7.3: Couette flow, 31×31: Velocity vector field at different shear times.

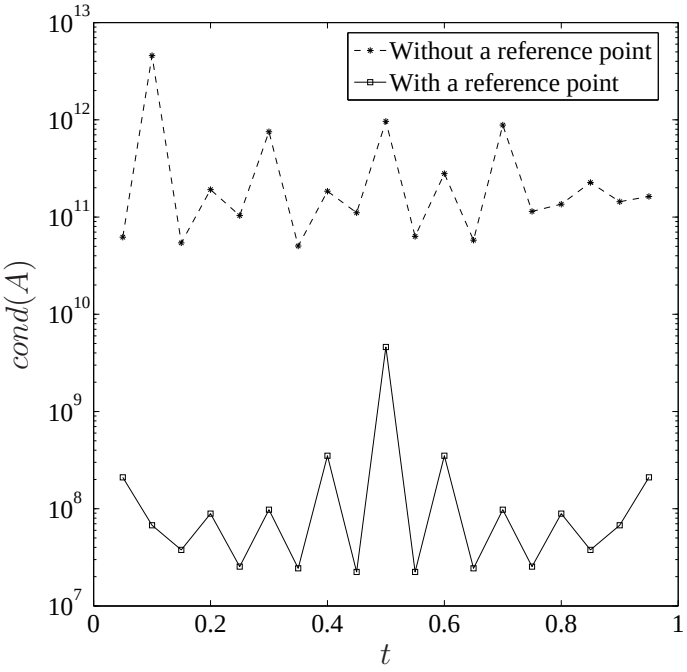


Figure 7.4: Couette flow, 31×31: Condition number of the system matrix A.

7.4.2 Particulate suspensions

We verify the present method with bi-periodic shear cells containing one, two and many particles.

One-particle

Consider a circular cylinder of radius R suspended freely at the centre of a frame. The frame is a basic part of the flow shown in Figure 7.5. The simulation is carried out with the geometric parameters: $L \times H = 1 \times 1$ and several values of R, namely $\{0.05,\ 0.15,\ 0.25,\ 0.35,\ 0.4,\ 0.45,\ 0.475,\ 0.485\}$, the imposed shear rate: $\dot{\gamma} = 1$, and the discretisation parameters: $\Delta t = 0.1$ (shear time step), $\Delta t_p = \{10^{-4},\ 5 \times 10^{-5},\ 1 \times 10^{-5}\}$ (pseudo-time steps) and Cartesian grids of $\{51 \times 51,\ 71 \times 71,\ 111 \times 111\}$. Higher grid densities and smaller pseudo-time steps are used for larger particle radii. The circular cylinder is expected to have a rotational motion only relative to the sliding frame. It can be seen that the initial configuration is reproduced after the time period $K = L/\dot{\gamma}H = 1$.

Figure 7.6 displays variations of the angular velocity of the particle with shear time for various particle radii. They graphically agree well with those produced

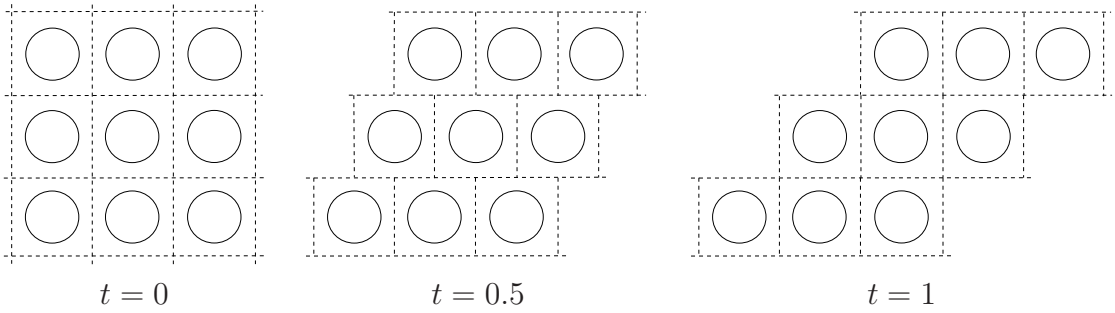


Figure 7.5: One-particle problem: A periodic configuration of particles can be modelled by a frame with one single particle for the analysis of the flow.

by the fictitious-domain finite element method (FEM) reported in Hwang et al. (2004) (Figure 13). We observe that the centre of the particle has nearly zero velocities in all cases, as expected. In Figure 7.7, the evolution of the flow for the case of $R = 0.25$ is demonstrated through the development of the vorticity field in one period $0 \leq t \leq K = 1$. It can be seen that (i) the contour plot at the shear time $t = 0.5$ is symmetric about the vertical centreline of the frame; and (ii) contour plots at other shear times are symmetric about the time $t = 0.5$ (half periodic). It also shows that the cylinder continuously rotates clockwise as the shear time increases.

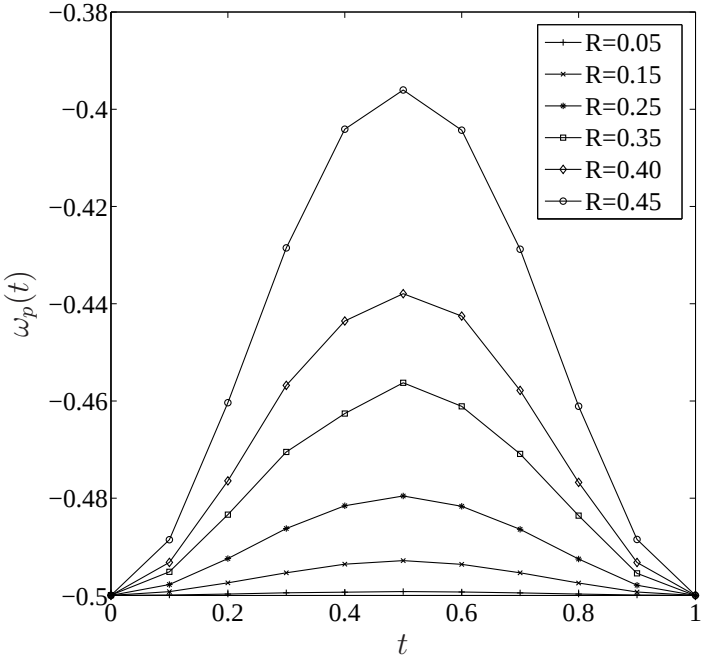


Figure 7.6: One-particle problem: The angular velocity against the shear time for different particle radii.

In a dilute regime, it has been found that the relative viscosity of the suspension

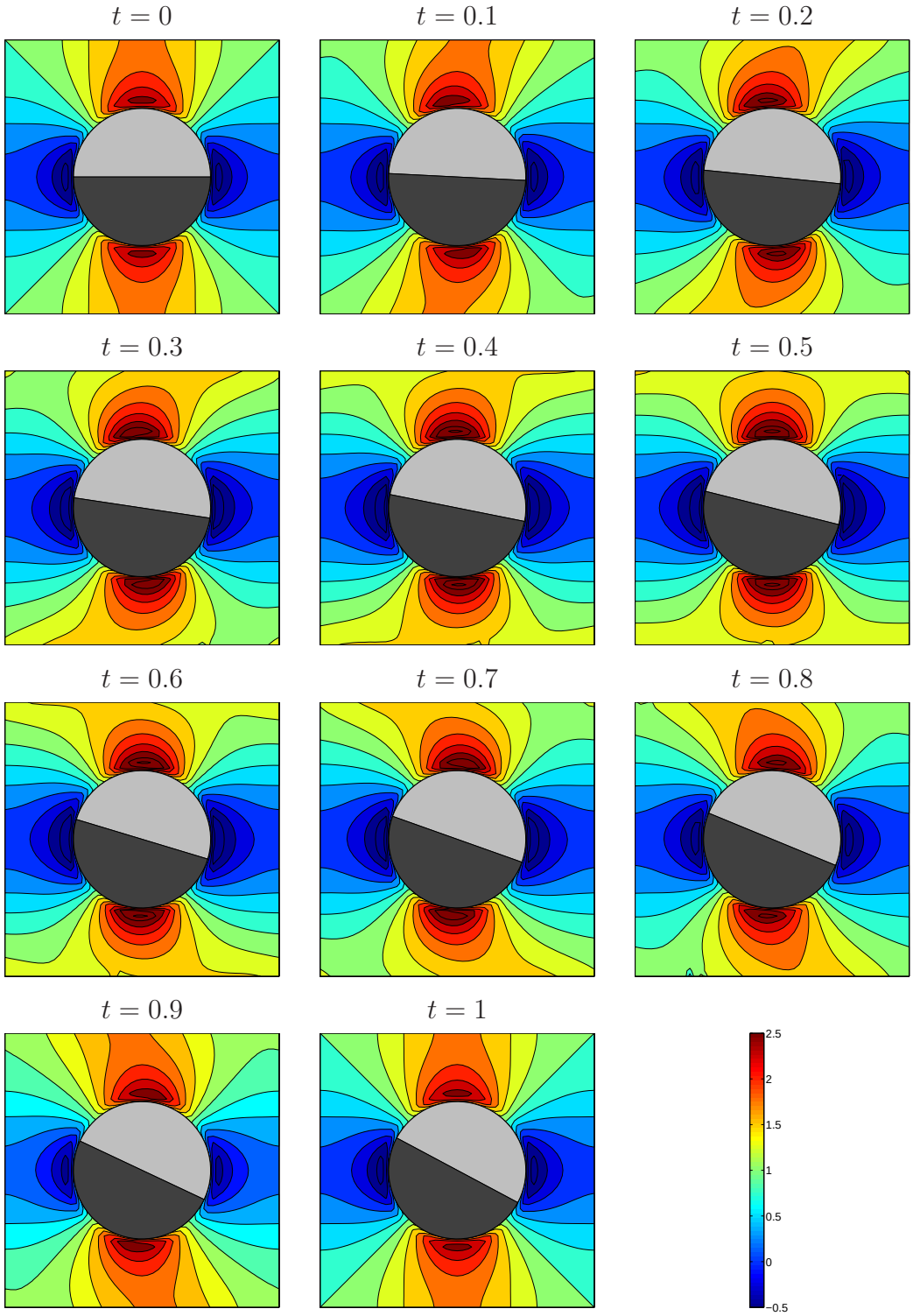


Figure 7.7: One-particle problem, $R = 0.25$, 51×51, $\Delta t_p = 10^{-4}$: Contours of vorticity of the flow. Values of isolines are -0.5:0.25:2.5.

depends linearly on the volume fraction $\eta_r = \eta/\eta_s = 1 + \eta_0\phi$, where η is the viscosity of the suspension; η_s the viscosity of the solvent (suspending fluid);

η_0 the intrinsic viscosity; ϕ the solid-volume fraction and Brady (1983) showed that $\eta_0 = 2$ for suspensions of circular cylinders. Consider the case $R = 0.05$ which corresponds to $\phi = 7.854 \times 10^{-3}$ (dilute suspension). We compute the bulk shear stress using the data over the whole time period $0 \leq t \leq K = 1$, from which the relative viscosity is predicted as $\eta_r = 1.0162$ and the corresponding intrinsic viscosity is thus 2.06 that is close to the theoretical value of 2.

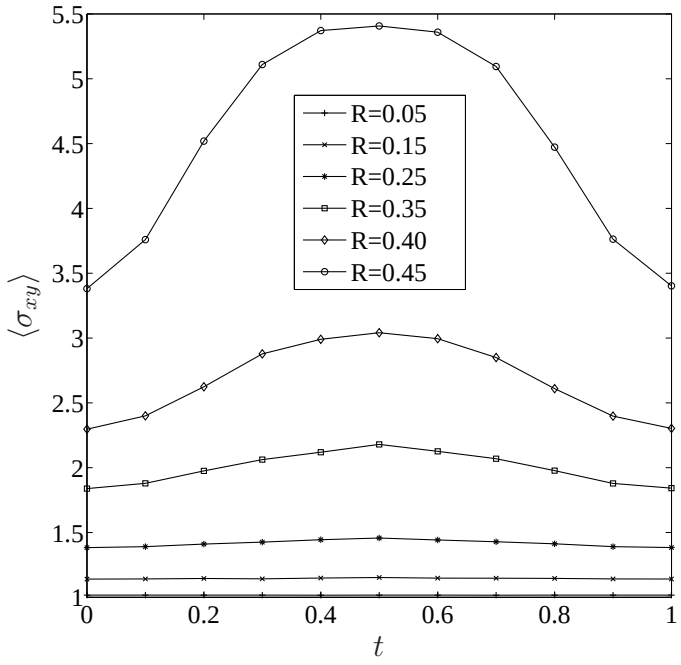
Results concerning shear stress and first normal stress difference are shown in Figure 7.8, where their time variations in the period are observed to be similar to those predicted by the fictitious-domain FEM (Hwang et al., 2004). Figure 7.9 shows a high dependence of the relative viscosity on the volume fraction. In this figure, a wide range of the volume fraction is employed, $\phi = \{0.0079,\ 0.0707,\ 0.1963,\ 0.3848,\ 0.5027,\ 0.6362,\ 0.7088,\ 0.7390\}$, which correspond to $R = \{0.05,\ 0.15,\ 0.25,\ 0.35,\ 0.4,\ 0.45,\ 0.475,\ 0.485\}$. The viscosity grows rapidly as the volume fraction increases. Our results again agree well with those of (Hwang et al., 2004).

Two-particle

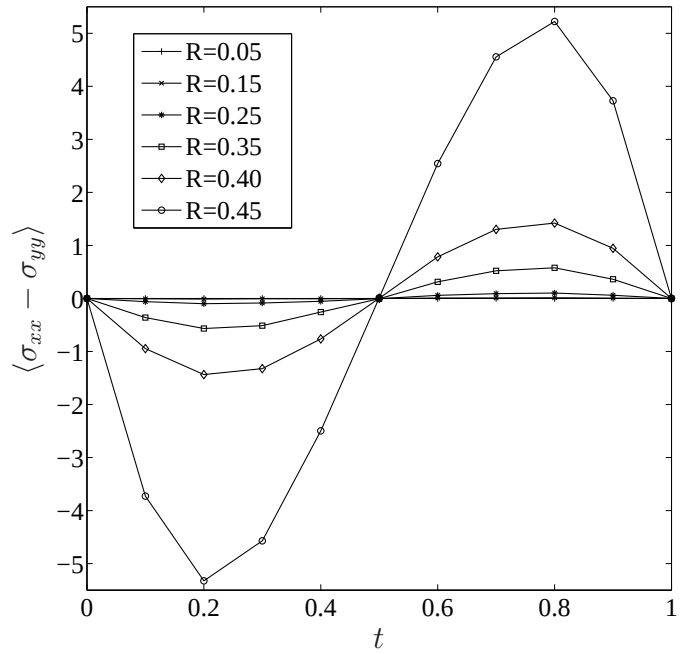
Consider two identical suspended circular cylindrical particles. Their initial locations are shown in Figure 7.10: the left upper particle denoted by P_1, the right lower one by P_2, $\mathbf{X}_{P_1} = (0.25, D)$ and $\mathbf{X}_{P_2} = (0.75, -D)$. One can expect that P_1 moves toward the right while P_2 moves toward the left. As shown in Hwang et al. (2004), the value of D strongly affects the flow field. In this work, we employ $R = 0.12$, $D = \{0.25,\ 0.025\}$, a uniform grid of 71×71, $\Delta t = 0.1$ and $\Delta t_p = 5 \times 10^{-5}$. Figures 7.11-7.12 show the flow behaviour in two cases $D = 0.25$ and $D = 0.025$, respectively. For $D = 0.25$, it is seen that the two particles simultaneously translate horizontally and rotate clockwise with respect to time (their translations in the vertical direction are insignificant). They start to cross the boundaries at about $t = 2.7$ and complete it at about $t = 3.7$. For $D = 0.025$, the two particles, apart from rotating, now have translations in both directions. They are close to each other at $t = 6.4$, tend to contact at $t = 7.8$ but prevented from doing so by lubrication forces, and then, separate at $t = 9.2$. Figure 7.13 displays the orbits of the particle centres - they look feasible and compare well with those in Hwang et al. (2004). Our results of time-dependent bulk shear stresses and those of Hwang et al. (2004) are shown in Figure 7.14. Good agreement is achieved.

7.4.3 Many particles

Periodic flows of many particles provide a good test for numerical methods as the randomness of suspended particles, occurring naturally in practice, is taken



(a)



(b)

Figure 7.8: One-particle problem: The bulk shear stress (a) and the bulk normal-stress difference (b) against the shear time for different particle radii.

in to account. Particles can now cross the boundary in any direction, and the symmetric behaviour of the flow in the case of one particle is no longer valid. Larger volume fractions are achieved by increasing the number of suspended

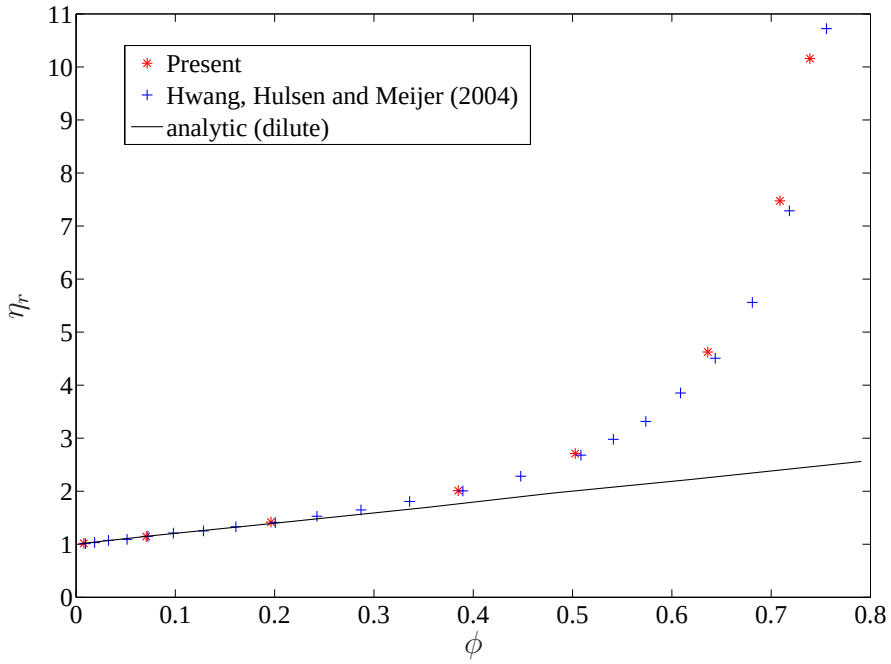


Figure 7.9: One-particle problem: Relative viscosity against solid-volume fraction.

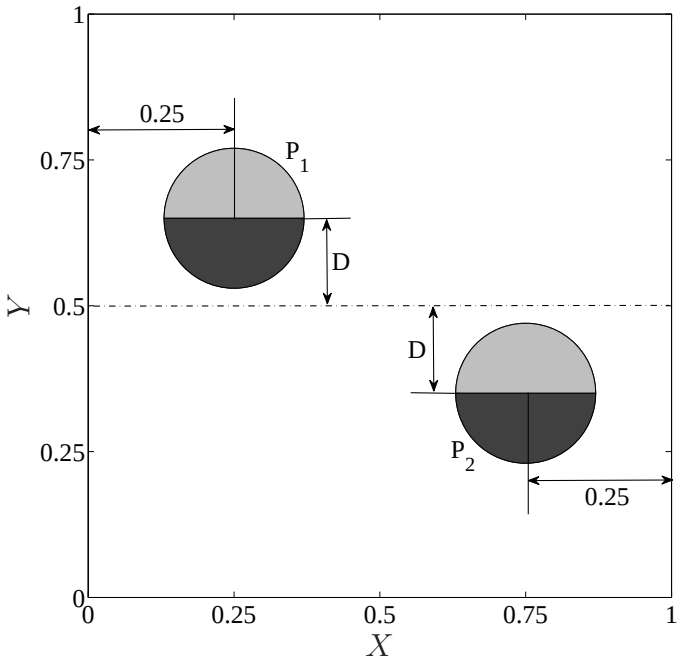


Figure 7.10: Two-particle problem: Initial configuration of two particles depending on D.

particles.

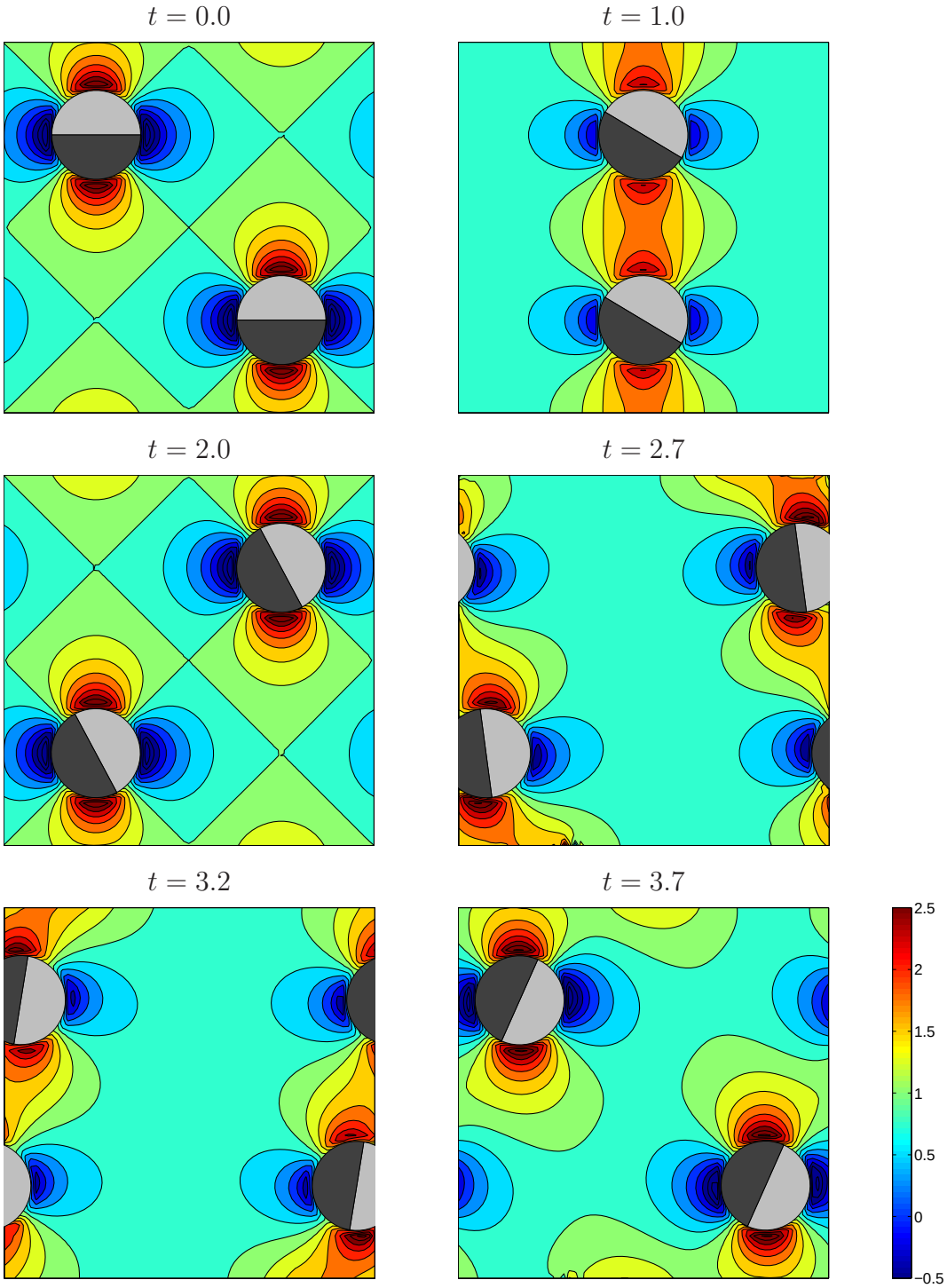


Figure 7.11: Two-particle problem, $R = 0.12$: Contours of vorticity of the flow for $D = 0.25$. Values of isolines are -0.5:0.25:2.5.

Viscosity against solid-area fraction

Our concern here is about the linear relation of the relative viscosity at low volume fractions since the result agreement among theoretical, experimental

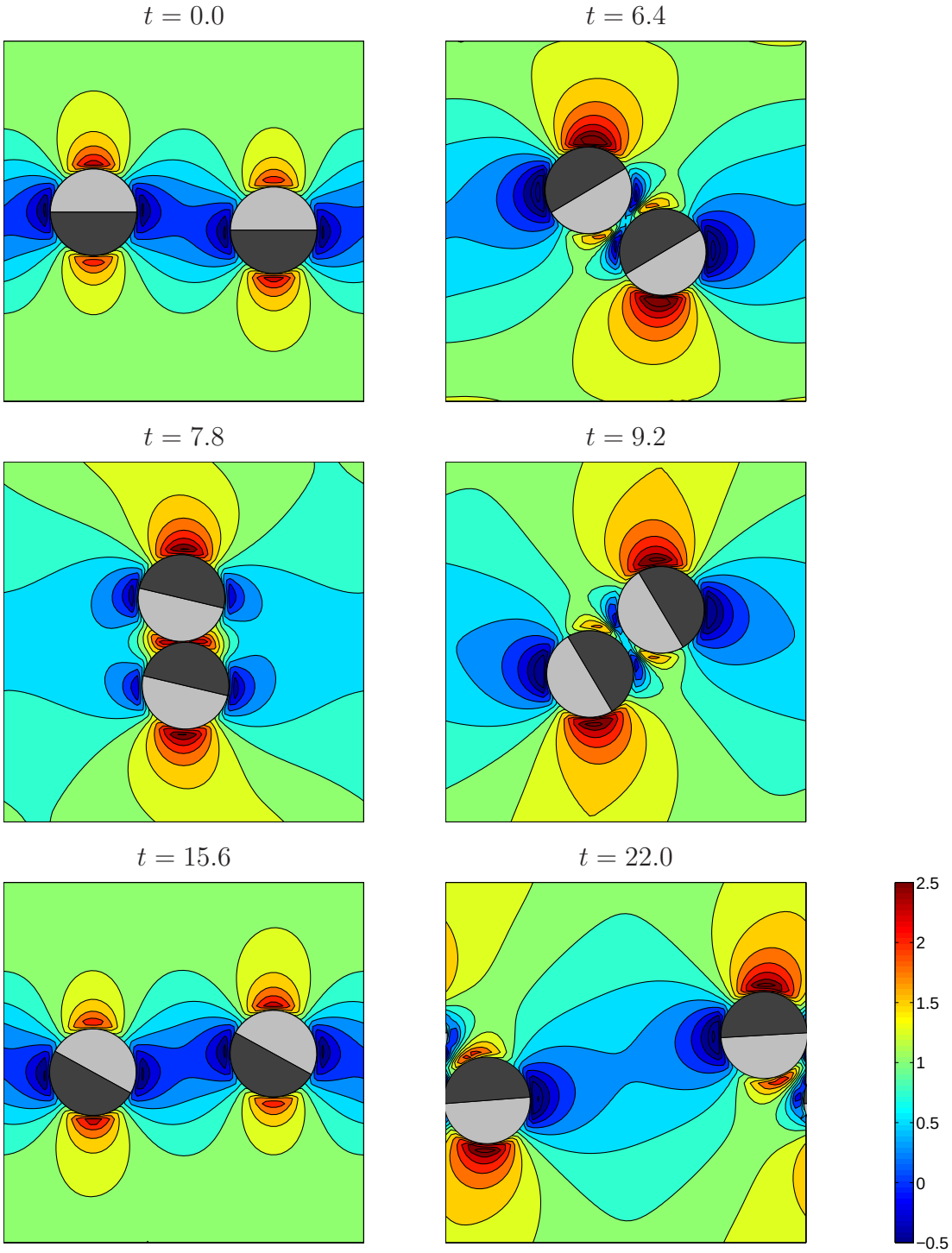


Figure 7.12: Two-particle problem, $R = 0.12$: Contours of vorticity of the flow for $D = 0.025$. Values of isolines are -0.5:0.25:2.5.

and numerical approaches is obtained only in dilute regime up to date. We employ particles of $R = 0.05$ and $R = 0.12$ (monodispersed suspensions), a uniform grid of 51×51, a shear time step $\Delta t = 0.01$ and a pseudo-time

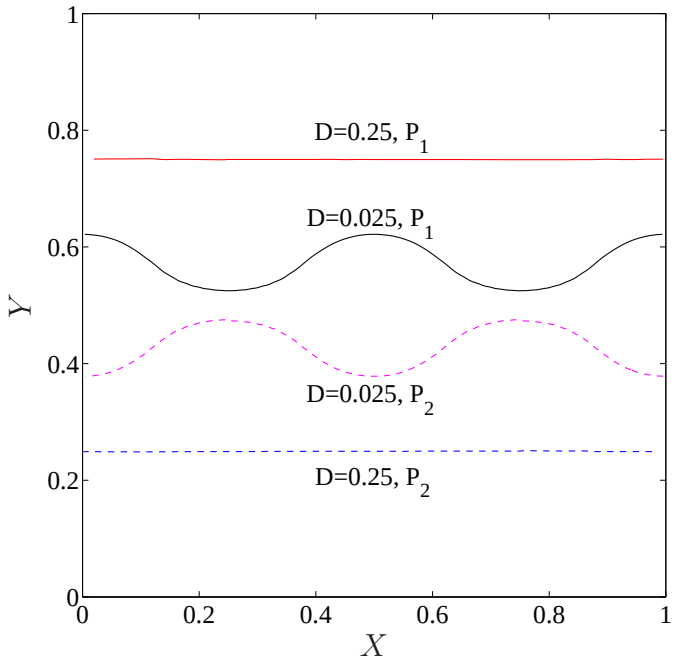


Figure 7.13: Two-particle problem, $R = 0.12$: The orbit of the two particle centres for $D = 0.025$ and $D = 0.25$.

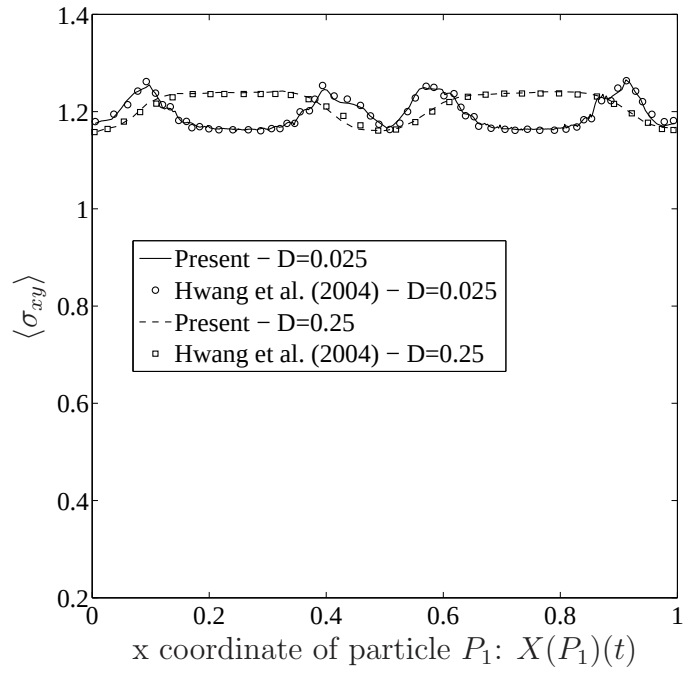


Figure 7.14: Two-particle problem, $R = 0.12$: Time-dependent bulk shear stress with respect to the x coordinate of the particle P_1 for $D = 0.025$ and $D = 0.25$.

step $\Delta t_p = 10^{-4}$. Low area fractions considered are $\phi = \{0.0079,\ 0.0157,$

0.0236, 0.0314, 0.0393, 0.0452, 0.0905, 0.1357, 0.1810, 0.2262} formed from the use of {1, 2, 3, 4, 5} suspended particles of $R = 0.05$ and $R = 0.12$. Long calculations are carried out here to minimize the effects of time on the rheological properties. Figure 7.15 shows that a relation $1 + 2\phi$ is obtained for the relative viscosity as the volume fraction approaches zero.

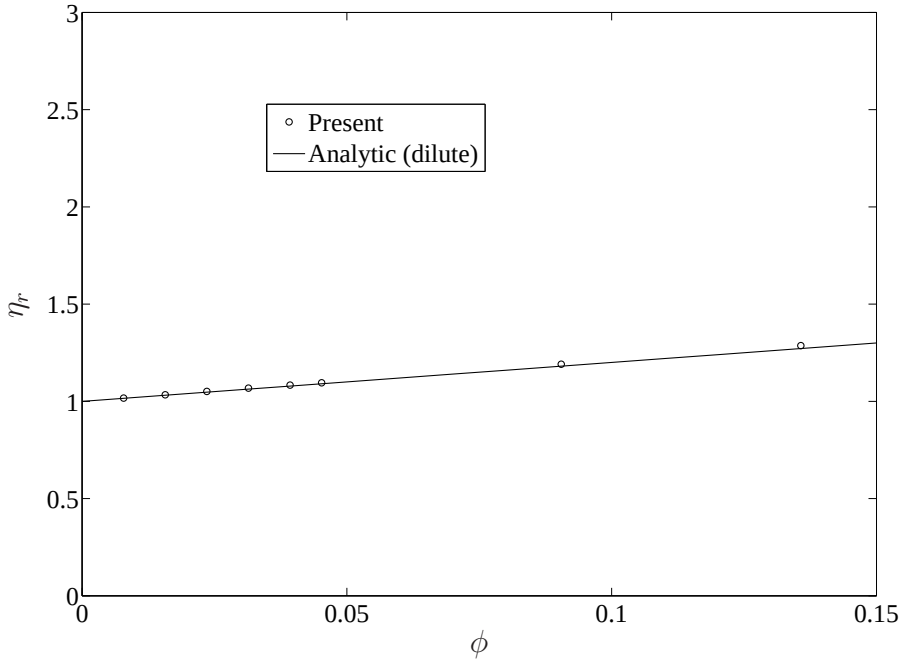


Figure 7.15: Many-particle problem, $N_p \in \{1, 2, 3, 4, 5\}$: Relative viscosity against solid-volume fraction in dilute suspensions. The first five points on the left correspond to $R = 0.05$ and the last three points on the right correspond to $R = 0.12$. The results show that the relative viscosity is independent of particle size in the dilute limit.

Figure 7.16 shows the initial locations for five-particle problems with $R = 0.05$ and $R = 0.12$ considered above. The vorticity isolines at a time $t = 4$ are displayed in Figure 7.17, showing that particles cross both x- and y-directions.

Flow index

In this section, we study the effects of shear rate on the flow behaviour. Several shear rates, namely $\dot{\gamma} = \{0.1, 0.5, 1, 5, 10\}$, are employed. According to Herschel-Bulkley model (Herschel and Bulkley, 1926), the shear stress σ_{xy} is expressed by

$$\sigma_{xy} = \sigma_{xy}^0 + C\dot{\gamma}^m, \tag{7.73}$$

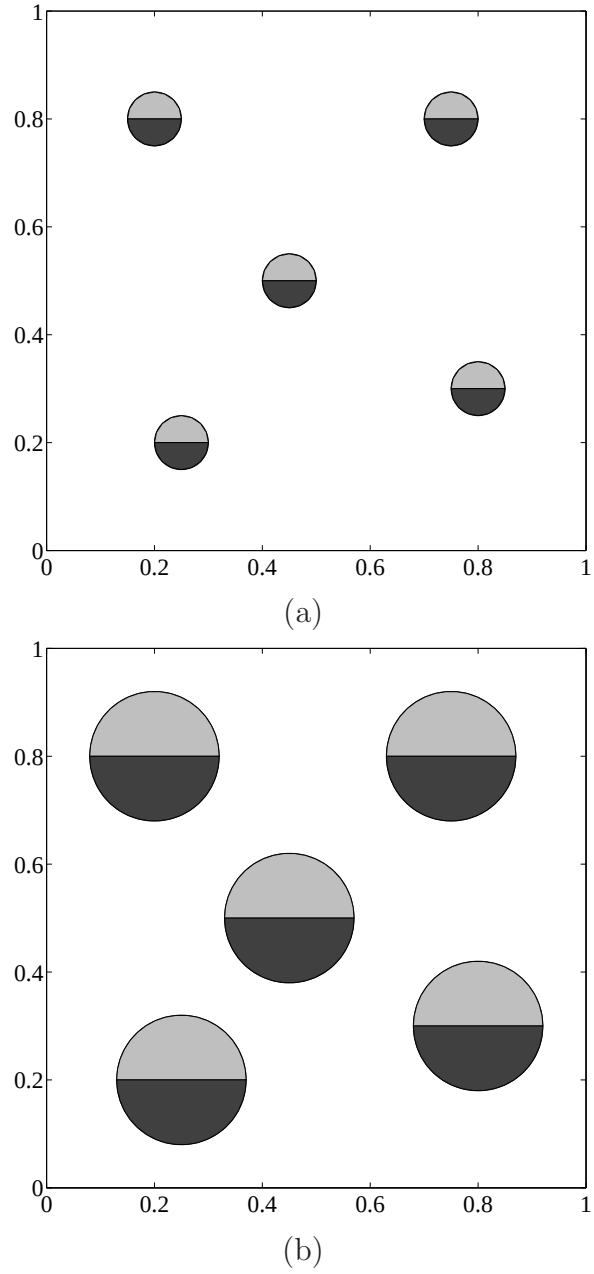
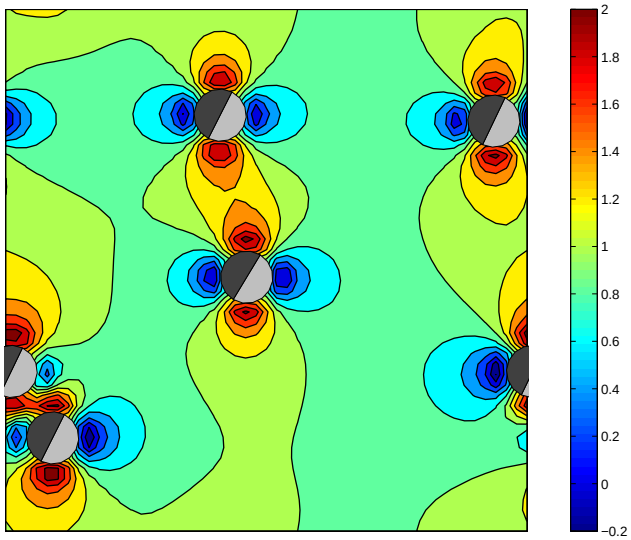


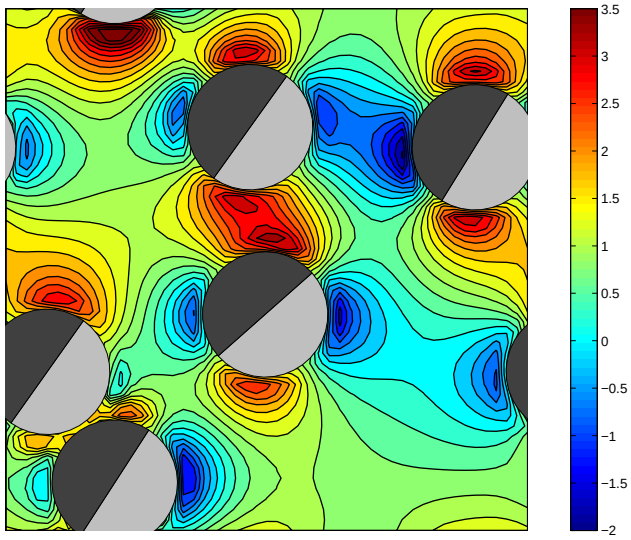
Figure 7.16: Many-particle problem: Initial configurations of particles for five-particle problems with $R = 0.05$ (a) and $R = 0.12$ (b).

where σ_{xy}^0 is the yield stress; C the consistency (same value as η at $\dot{\gamma} = 1$); and m the flow index which defines the degree of non-Newtonian behaviour (shear-thickening for $m > 1$ and shear-thinning for $m < 1$).

Figure 7.18 shows the dependence of shear stress on shear rate at different solid-volume fractions. It is observed that the profiles are almost linear. All curves pass through the origin, indicating $\sigma_{xy}^0 = 0$ as expected. For a given shear rate, shear stress is larger as solid-volume fraction increases. Figure 7.19 shows that



(a)



(b)

Figure 7.17: Many-particle problem: Vorticity isolines of the flow for five-particle problems with $R = 0.05$ (a) and $R = 0.12$ (b) at $t = 4$. Values of isolines are -0.2:0.2:2 for (a) and -2:0.25:3.5 for (b).

the flow index approaches 1 (Newtonian fluid) as the volume fraction is reduced.

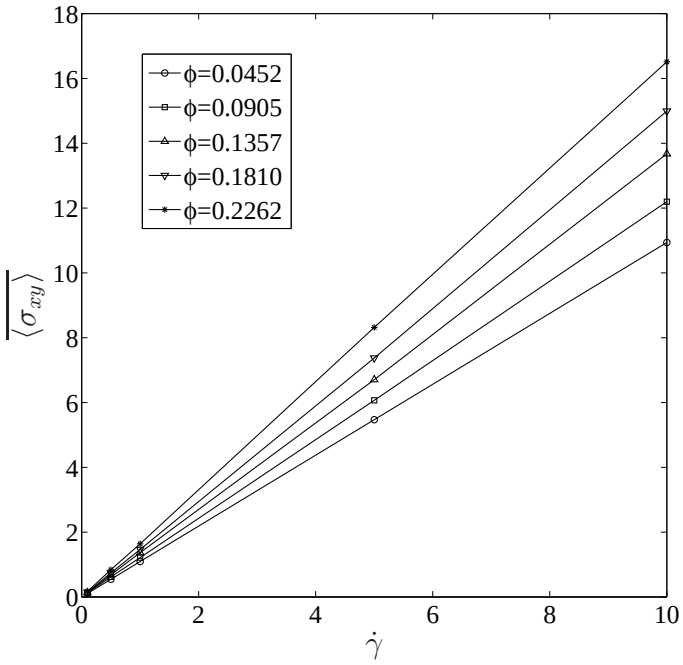


Figure 7.18: Many-particle problem, $R = 0.12$, $N_p \in \{1,\ 2,\ 3,\ 4,\ 5\}$: Shear stress against shear rate at different solid-volume fractions in dilute suspensions.

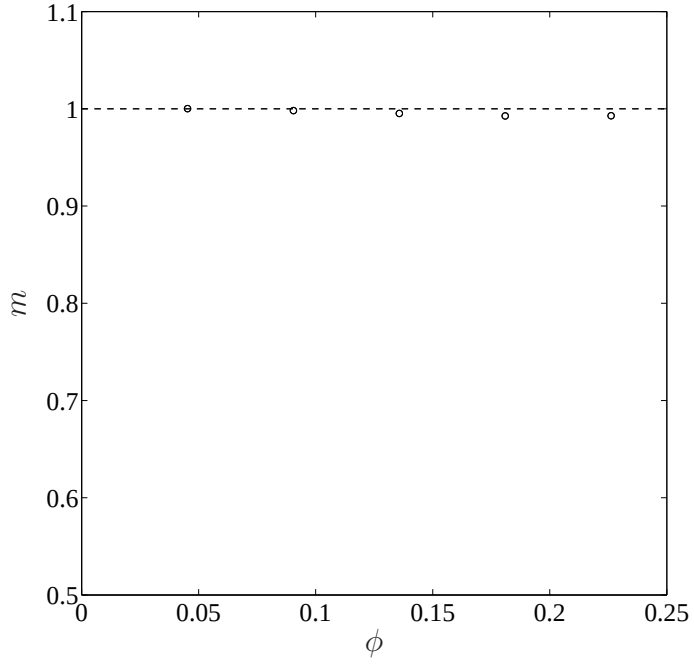


Figure 7.19: Many-particle problem, $R = 0.12$, $N_p \in \{1,\ 2,\ 3,\ 4,\ 5\}$: Flow index against solid-volume fraction in dilute suspensions.

7.5 Concluding remarks

In this study, a new computational procedure, based on the direct forcing immersed boundary method and compact integrated RBF approximations, is pro-

posed for the direct simulation of suspensions of rigid particles. Expressions of computing fluid and rigid boy motions are presented in detail. The multiply-connected domain whose geometry varies with time is simply discretised using a fixed Cartersion grid, while the Navier-Stokes equation is discretised efficiently with point collocation approach. Very encouraging results are obtained. Unlike particle-based simulation techniques, the present method does not suffer from the compressibility issues; however, the obtained system of equations for the solvent needs be solved simultaneously and thus requiring large computational efforts. Parallel calculations will be considered in our future work to throughout investigate concentrated suspensions with a particular focus on the handling of lubrication forces.

Chapter 8

Conclusions

This chapter concludes the thesis with a brief summary of the main contributions of the present research project and some suggestions for future developments and extensions.

8.1 Research contributions

The present research project results in new computation procedures for the simulation of flows of a Newtonian fluid in complex geometries, particularly multiply-connected domains, and their applications for the investigation of fluid-solid systems (i.e. fluid-particle interactions and particulate suspensions). The strengths of these procedures lie in the use of point collocation, Cartesian grids, integrated RBFs, compact local forms and direct forcing immersed boundary method. Key contributions/advances of the research are summarised below

A successful development of compact local IRBF approximations for the discretisation of the velocity-pressure formulation

This contribution has been presented in chapter 2. IRBF approximations are expressed in terms of values not only of the field variable but also of its derivatives. In addition, an effective boundary treatment for the pressure variable, where Neumann boundary conditions are transformed into Dirichlet ones, is proposed. These lead to a significant improvement in accuracy.

A successful development of compact IRBF approximations for the simulation of time-dependent problems

This contribution has been presented in chapter 3. We discretise temporal derivatives with the Adams-Bashforth/Crank-Nicolson algorithm. Compact

IRBF stencils presented in chapter 2 are employed in an implicit manner, resulting in stencils CIRBF-1. The proposed scheme is successfully applied to solve four types of time-dependent equations, namely the diffusion, Burgers, Stokes and Navier-Stokes equations. Numerical experiments show that the proposed scheme is stable and high-order accurate.

A successful incorporation of ADI algorithm for solving convection-diffusion equation

This contribution has been presented in chapter 4. We propose another compact IRBF stencil, referred to as CIRBF-2, to achieve tridiagonal matrices with the ADI algorithm (You, 2006) in solving 2D convection-diffusion equations. Such systems can be solved efficiently. Several steady and non-steady problems are considered to verify the present schemes and to compare their accuracy with some other ADI schemes. Numerical results show that highly accurate results are obtained with the proposed method.

A successful development of a direct forcing immersed boundary method employed with CIRBF approximations for heat transfer and fluid flow problems

This contribution has been presented in chapter 5. A direct forcing immersed boundary (DFIB) method employed with CIRBF approximations is successfully developed for the simulation of viscous flows and heat transfers in multiply-connected domains. The approximation scheme here is based on CIRBF-2, except that a compact 2-point stencil is employed at boundary nodes in order to produce tridiagonal coefficient matrices, and thus making the finally resultant system matrix sparser. This proposed scheme, namely CIRBF-3, is more efficient than CIRBF-2 and still keeps the accuracy as good as CIRBF-2. The present DFIB-CIRBF method is verified through several test problems and numerical results obtained using relatively coarse grids are in good agreement with available data in the literature.

Successful applications of DFIB-CIRBF method for fluid-solid interactions in particulate fluids

This contribution has been presented in chapter 6. The present DFIB-CIRBF method is successfully applied to investigate fluid-solid systems. Both prescribed and induced motions of the bodies/particles are considered. Particle-wall and particle-particle interactions are also taken into consideration. The capability of the present method is depicted by simulating several problems including Taylor-Green vortices, induced flow by an oscillating circular cylinder, single particle sedimentation and drafting-kissing-tumbling behaviour of two settling particles. Good agreement is achieved.

Successful application of DFIB-CIRBF method for numerical predic-

tion of rheological properties of particulate suspensions

This contribution has been presented in chapter 7. The present DFIB-CIRBF method is further developed to simulate particulate suspensions in a sliding bi-periodic frame and predict their rheological properties numerically. It is noted that there have been only a few continuum simulation works studying the rheology of particulate suspensions. Methods for computing fluid and rigid body motions are presented in detail. The implementation of shear bi-periodic boundary conditions is also discussed in detail. We verify the proposed method in bi-periodic shear flows of one, two and many particles. Results obtained, including relative viscosity and flow index, are in good agreement with those predicted by analytic theories and some other direct simulation methods.

8.2 Possible future work

In the present thesis, only 2D fluid flows and suspensions of monodispersed circular cylinders in a Newtonian liquid are considered. The present computational procedures can be further developed to investigate the size and shape distributions of rigid particles in suspensions. Other explorations include (i) employing a viscoelastic fluid as the suspending liquid; (ii) taking into account the energy equation to study the effects of the temperature; (iii) solution to other engineering applications (e.g. turbulent models); and (iv) incorporating domain decompositions and implementing the computer code in parallel to deal with large scale problems.

References

Abdallah, S. (1987). Numerical solutions for the incompressible Navier-Stokes equations in primitive variables using a non-staggered grid, II., *Journal of Computational Physics* **70**(1): 193–202.

Ali, A. H. A., Gardner, G. A. and Gardner, L. R. T. (2005). A collocation solution for Burgers' equation using cubic B-spline finite elements, *Applied Mathematics and Computation* **170**: 781–800.

Almgren, A. S. (1996). A numerical method for the incompressible Navier-Stokes equations based on an approximate projection, *SIAM J. Sci. Compt.* **17**.

Atluri, S. N., Han, Z. D. and Rajendran, A. M. (2004). A new implementation of the meshless finite volume method, through the MLPG "Mixed" approach, *CMES: Computer Modeling in Engineering & Sciences* **6**(6): 491–513.

Atluri, S. N. and Shen, S. (2002). The meshless local Petrov-Galerkin (MLPG) method: A simple & less-costly alternative to the finite element and the boundary element methods, *CMES: Computer Modeling in Engineering & Sciences* **3**: 11–51.

Atluri, S. N. and Zhu, T. (1998). A new meshless local Petrov-Galerkin (MLPG) approach in computational mechanics, *Computational Mechanics* **22**: 117–127.

Bateman, H. (1915). Some recent researches on the motion of fluids, *Monthly Weather Review* **43**: 163–170.

Beckert, A. and Wendland, H. (2001). Multivariate interpolation for fluid-structure-interaction problems using radial basis functions, *Aerospace Science and Technology* **5**(2): 125 – 134.

Bell, J., Colella, P. and Glaz, H. (1989). A second-order projection method for the incompressible Navier-Stokes equations, *Journal of Computational Physics* **85**(2): 257–283.

Belytschko, T., Lu, Y. Y. and Gu, L. (1994). Element-free Galerkin methods, *International Journal for Numerical Methods in Engineering* **37**: 229–256.

Blasco, J., Calzada, M. C. and Marn, M. (2009). A fictitious domain, parallel numerical method for rigid particulate flows, *Journal of Computational Physics* **228**(20): 7596–7613.

Botella, O. and Peyret, R. (1998). Benchmark spectral results on the lid-driven cavity flow, *Computers & Fluids* **27**(4): 421–433.

Brady, J. F. (1983). The Einstein viscosity correction in n dimensions, *International Journal of Multiphase Flow* **10**(1): 113–114.

Brown, D. L., Cortez, R. and Minion, M. L. (2001). Accurate projection methods for the incompressible Navier-Stokes equations, *Journal of Computational Physics* **168**(2): 464–499.

Bruneau, C. and Saad, M. (2006). The 2d lid-driven cavity problem revisited, *Computers and Fluids* **35**(3): 326–348.

Butcher, J. C. (2003). *Numerical Methods for Ordinary Differential Equations*, John Wiley.

Caldwell, J., Wanless, P. and Cook, A. E. (1987). Solution of Burgers' equation for large Reynolds number using finite elements with moving nodes, *Applied Mathematical Modelling* **11**(3): 211–214.

Canuto, C., Hussaini, M., Quarteroni, A. and Zang, T. (1988). *Spectral Methods in Fluid Dynamics*, Springer-Verlag, New York.

Carlson, R. and Foley, T. (1991). The parameter r^2 in multiquadric interpolation, *Computers and Mathematics with Applications* **21**(9): 29–42.

Chen, R. and Wu, Z. (2006). Applying multiquadric quasi-interpolation to solve Burgers' equation, *Applied Mathematics and Computation* **172**(1): 472–484.

Cheng, A. H. D., Golberg, M. A., Kansa, E. J. and Zammito, G. (2003). Exponential convergence and h-c multiquadric collocation method for partial differential equations, *Numerical Methods for Partial Differential Equations* **19**: 571–594.

Cheng, S. I. (1968). *Accuracy of difference formulation of Navier-Stokes equations*, A. M. S. Department, Princeton University, Princeton, New Jersey.

Chorin, A. (1968). Numerical solution of the Navier-Stokes equations, *Mathematics of Computation* **22**(104): 745–762.

Chung, H. L. and Denis, J. D. (1999). Velocity-vorticity formualtion with vortex particle-in-cell method for incompressible viscous flow simulation, part i: Formulation and validation, *Numerical Heat Transfer, Part B: Fundamentals* **35**(3): 251–275.

Crank, J. and Nicolson, P. (1996). A practical method for numerical evaluation of solutions of partial differential equations of the heat-conduction type, *Advances in Computational Mathematics* **6**(1): 207–226.

Cyrus, N. J. and Fulton, R. E. (1967). *Accuracy study of finite difference methods*, NASA TN D-4372, National Aeronautics and Space Administration, Langley Research Centre, Langley Station, Hampton, Virginia.

Dag, I., Irk, D. and Saka, B. (2005). A numerical solution of the Burgers' equation using cubic B-splines, *Applied Mathematics and Computation* **163**: 199–211.

Dag, I., Saka, B. and Boz, A. (2005). B-spline Galerkin methods for numerical solutions of the Burgers' equation, *Applied Mathematics and Computation* **166**(3): 506–522.

de Boer, A., van der Schoot, M. and Bijl, H. (2007). Mesh deformation based on radial basis function interpolation, *Computers & Structures* **85**(11-14): 784–795.

Dehghan, M. (2004). Weighted finite difference techniques for the one-dimensional advection-diffusion equation, *Applied Mathematics and Computation* **147**(2): 307–319.

Demirkaya, G., Soh, C. W. and Ilegbusi, O. (2008). Direct solution of Navier-Stokes equations by radial basis functions, *Applied Mathematical Modelling* **32**(9): 1848–1858.

Deng, G., Piqueta, J., Queuteya, P. and Visonneaua, M. (1994). Incompressible flow calculations with a consistent physical interpolation finite volume approach, *Computers & Fluids* **23**(8): 1029–1047.

Dilts, G. A. (1999). Moving-least-squares-particle hydrodynamics-i. consistency and stability, *International Journal for Numerical Methods in Engineering* **44**: 1115–1155.

Ding, H., Shu, C., Yeo, K. and Xu, D. (2006). Numerical computation of three-dimensional incompressible viscous flows in the primitive variable form by local multiquadric differential quadrature method, *Computer Methods in Applied Mechanics and Engineering* **195**(7-8): 516–533.

Divo, E. and Kassab, A. J. (2007). An efficient localized radial basis function meshless method for fluid flow and conjugate heat transfer, *Journal of Heat Transfer* **129**(2): 124–136.

Dogan, A. (2004). A Galerkin finite element approach to Burgers' equation, *Applied Mathematics and Computation* **157**: 331–346.

Douglas, J. and Rachford, H. (1956). On the numerical solution of heat conduction problems in two and three space variables, *Transactions of the American mathematical Society* **82**(2): 421–439.

Fadel, H. and Agouzoul, M. (2011). High-order finite difference schemes for incompressible flows, *International Journal for Numerical Methods in Fluids* **65**(9): 1050–1070.

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

Feng, Z.-G. and Michaelides, E. E. (2004). The immersed boundary-lattice Boltzmann method for solving fluid-particles interaction problems, *Journal of Computational Physics* **195**(2): 602–628.

Feng, Z.-G. and Michaelides, E. E. (2005). Proteus: a direct forcing method in the simulations of particulate flows, *Journal of Computational Physics* **202**(1): 20–51.

Fortes, A. F., Joseph, D. D. and Lundgren, T. S. (1987). Nonlinear mechanics of fluidization of beds of spherical particles, *Journal of Fluid Mechanics* **177**: 467–483.

Franke, R. (1982). Scattered data interpolation: Tests of some method, *Mathematics of Computation* **38**: 181–200.

Ghia, U., Ghia, K. N. and Shin, C. (1982). High-Re solutions for incompressible flow using the Navier-Stokes equations and a multigrid method, *Journal of Computational Physics* **48**(3): 387–411.

Gil, A., Carreo, A. A., Bonet, J. and Hassan, O. (2010). The immersed structural potential method for haemodynamic applications, *Journal of Computational Physics* **229**(22): 8613–8641.

Gingold, R. A. and Monaghan, J. J. (1977). Smoothed particle hydrodynamics-theory and application to non-spherical stars, *Monthly notices of the royal astronomical society* **181**: 375–389.

Girault, V. and Raviart, P. (1986). *Finite element methods for Navier-Stokes Equations. Theory and Algorithms*, Springer-Verlag, Berlin.

Glowinski, R., Ciarlet, P. G. and Lions, J. L. (2003). *Numerical methods for fluids*, Elsevier Science & Technology.

Glowinski, R., Pan, T. W., Hesla, T. I., Josephand, D. D. and Periaux, J. (2001). A fictitious domain approach to the direct numerical simulation of incompressible viscous flow past moving rigid bodies: Application to particulate flow, *Journal of Computational Physics* **169**(2): 363–426.

Glowinski, R., Pan, T.-W., Hesla, T. and Joseph, D. (1999). A distributed Lagrange multiplier/fictitious domain method for particulate flows, *International Journal of Multiphase Flow* **25**(5): 755–794.

Glowinski, R., Pan, T.-W. and Priaux, J. (1994). A fictitious domain method for external incompressible viscous flow modeled by Navier-Stokes equations., *Computer Methods in Applied Mechanics and Engineering* **112**: 133–148.

Goldstein, D., Handler, R. and Sirovich, L. (1993). Modeling a no-slip flow boundary with an external force field, *Journal of Computational Physics* **105**(2): 354–366.

Gupta, M. M., Manohar, R. P. and Stephenson, J. W. (2005). High-order difference schemes for two-dimensional elliptic equations, *Numerical Methods for Partial Differential Equations* **1**(1): 71–80.

Hashemi, M. R., Fatehi, R. and Manzari, M. T. (2012). A modified SPH method for simulating motion of rigid bodies in Newtonian fluid flows, *International Journal of Non-Linear Mechanics* **47**: 626–638.

Hassanien, I. A., Salama, A. A. and Hosham, H. A. (2005). Fourth-order finite difference method for solving Burgers' equation, *Applied Mathematics and Computation* **170**: 781–800.

Hayashi, K., Rojas, R., Seta, T. and Tomiyama, A. (2012). Immersed boundary-Lattice Boltzmann method using two relaxation times, *Journal of Computational Multiphase Flows* **4**(2): 193–210.

Herschel, W. and Bulkley, R. (1926). Konsistenzmessungen von Gummi-Benzollsungen, *Kolloid-Zeitschrift* **39**(4): 291–300.

Hosseini, B. and Hashemi, R. (2011). Solution of Burgers' equation using a local-RBF meshless method, *International Journal for Computational Methods in Engineering Science and Mechanics* **12**(1): 44–58.

Hu, H. H. (1995). Motion of a circular cylinder in a viscous liquid between parallel plates, *Theoretical and Computational Fluid Dynamics* **7**: 441–455.

Hu, H. H., Joseph, D. D. and Crochet, M. J. (1992). Direct simulation of fluid particle motions, *Theoretical and Computational Fluid Dynamics* **3**: 285–306.

Hu, H. H., Patankar, N. A. and Zhu, M. Y. (2001). Direct numerical simulations of fluid-solid systems using the arbitrary Lagrangian-Eulerian technique, *Journal of Computational Physics* **169**(2): 427–462.

Hwang, W. R., Hulsen, M. A. and Meijer, H. E. H. (2004). Direct simulation of particle suspensions in sliding bi-periodic frames, *Journal of Computational Physics* **194**: 742–772.

Isenberg, J. and Gutfinger, C. (1972). Heat transfer to a draining film, *International Journal of Heat and Mass Transfer* **16**(2): 505–512.

Iskander, L. and Mohsen, A. (1992). Some numerical experiments on the splitting of Burgers' equation, *Numerical Methods for Partial Differential Equations* **8**(3): 267–276.

Iwatsu, R., Hyun, J. M. and Kuwahara, K. (1992). Numerical simulation of flows driven by a torsionally oscillating lid in a square cavity, *Journal of Fluids Engineering* **114**(2): 143–151.

Ji, C., Munjiza, A. and Williams, J. (2012). A novel iterative direct-forcing immersed boundary method and its finite volume applications, *Journal of Computational Physics* **231**(4): 1797–1821.

Kalita, J. C. and Chhabra, P. (2006). An improved (9,5) higher order compact scheme for the transient two-dimensional convection-diffusion equation, *International Journal for Numerical Methods in Fluids* **51**(7): 703–717.

Kalita, J. C., Dalal, D. C. and Dass, A. K. (2002). A class of higher order compact schemes for the unsteady two-dimensional convection-diffusion equation with variable convection coefficients, *International Journal for Numerical Methods in Fluids* **38**(12): 1111–1131.

Kansa, E. J. (1990a). Multiquadrics- A scattered data approximation scheme with applications to computational fluid-dynamics-I. surface approximations and partial derivative estimates, *Computers and Mathematics with Applications* **19**(8/9): 127–145.

Kansa, E. J. (1990b). Multiquadrics- A scattered data approximation scheme with applications to computational fluid-dynamics-II. solutions to parabolic, hyperbolic and elliptic partial differential equations, *Computers and Mathematics with Applications* **19**(8/9): 147–161.

Karaa, S. and Zhang, J. (2004). High order ADI method for solving unsteady convection-diffusion problems, *Journal of Computational Physics* **198**(1): 1–9.

Karniadakis, G. and Sherwin, S. (1999). *Spectral /Hp Element Methods for CFD*, Oxford University Press, New York.

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

Kim, J. and Moin, P. (1985). Application of a fractional-step method to incompressible Navier-Stokes equations, *Journal of Computational Physics* **59**(2): 308–323.

Kouatchou, J. (2001). Parallel implementation of a high-order implicit collocation method for the heat equation, *Mathematics and Computers in Simulation* **54**(6): 509–519.

Lai, M.-C. and Peskin, C. S. (2000). An immersed boundary method with formal second-order accuracy and reduced numerical viscosity, *Journal of Computational Physics* **160**(2): 705–719.

Le-Cao, K., Mai-Duy, N., Tran, C.-D. and Tran-Cong, T. (2010). Towards the analysis of shear suspension flows using radial basis functions, *CMES: Computer Modeling in Engineering & Sciences* **67**(3): 265–294.

Le-Cao, K., Mai-Duy, N. and Tran-Cong, T. (2009). An effective integrated-RBFN Cartesian-grid discretization for the stream function-vorticity-temperature formulation in nonrectangular domains, *Numerical Heat Transfer, Part B: Fundamentals* **55**(6): 480–502.

Le-Cao, K., Mai-Duy, N. and Tran-Cong, T. (2011). Numerical study of stream-function formulation governing flows in multiply-connected domains by integrated RBFs and Cartesian grids, *Computers & Fluids* **44**(1): 32–42.

Le, D. V., Khoo, B. C. and Peraire, J. (2006). An immersed interface method for viscous incompressible flows involving rigid and flexible boundaries, *Journal of Computational Physics* **220**(1): 109–138.

Lees, A. W. and Edwards, S. F. (1972). The computer study of transport processes under extreme conditions, *Journal of Physics C: Solid State Physics* **5**(15): 1921.

Leveque, R. J. (2002). *Finite Volume Methods for Hyperbolic Problems*, Cambridge Texts in Applied Mathematics, Cambridge University Press, Cambridge.

Leveque, R. J. and Li, Z. (1994). The immersed interface method for elliptic equations with discontinuous coefficients and singular sources, *SIAM Journal on Numerical Analysis* **31**(4): 1019–1044.

Lewis, E. (1979). Steady flow between a rotating circular cylinder and fixed square cylinder, *Journal of Fluid Mechanics* **95**: 497–513.

Liao, C. C., Chang, Y. W., Lin, C. A. and McDonough, J. M. (2010). Simulating flows with moving rigid boundary using immersed-boundary method, *Computers & Fluids* **39**(1): 152–167.

Liu, W. K., Jun, S. and Zhang, Y. F. (1995). Reproducing kernel particle methods, *International Journal for Numerical Methods in Fluids* **20**: 1081–1106.

Liu, X.-D., Fedkiw, R. P. and Kang, M. (2000). A boundary condition capturing method for poisson's equation on irregular domains, *Journal of Computational Physics* **160**(1): 151–178.

Lucy, L. B. (1977). A numerical approach to the testing of the fission hypothesis, *The Astronomical Journal* **82**: 1013–1024.

Luo, K., Klein, M., Fan, J.-R. and Cen, K.-F. (2006). Effects on particle dispersion by turbulent transition in a jet, *Physics Letters A* **357**(4-5): 345–350.

Ma, Y., Sun, C.-P., Haake, D. A., Churchill, B. M. and Ho, C.-M. (2012). A high-order alternating direction implicit method for the unsteady convection-dominated diffusion problem, *International Journal for Numerical Methods in Fluids* **70**(6): 703–712.

Madych, W. R. (1992). Miscellaneous error bounds for multiquadric and related interpolators, *Computers and Mathematics with Applications* **24**(121-138).

Mai-Cao, L. and Tran-Cong, T. (2005). A meshless IRBFN-based method for transient problems, *CMES: Computer Modeling in Engineering & Sciences* **7**(2): 149–171.

Mai-Cao, L. and Tran-Cong, T. (2008). A meshless approach to capturing moving interfaces in passive transport problems, *CMES: Computer Modeling in Engineering & Sciences* **31**(3): 157–188.

Mai-Duy, N., Le-Cao, K. and Tran-Cong, T. (2008). A Cartesian grid technique based on one-dimensional integrated radial basis function networks for natural convection in conccentric annuli, *International Journal for Numerical Methods in fluids* **57**: 1709–1730.

Mai-Duy, N. and Tanner, R. I. (2007). A collocation method based on one-dimensional RBF interpolation scheme for solving PDEs, *International Journal of Numerical Methods for Heat & Fluid Flow* **17**(2): 165–186.

Mai-Duy, N., Thai-Quang, N., Hoang-Trieu, T.-T. and Tran-Cong, T. (2014). A compact 9 point stencil based on integrated RBFs for the convection-diffusion equation, *Applied Mathematical Modelling* **38**(4): 1495–1510.

Mai-Duy, N. and Tran-Cong, T. (2001a). Numerical solution of differential equations using multiquadric radial basic function networks, *Neural Networks* **14**: 185–199.

Mai-Duy, N. and Tran-Cong, T. (2001b). Numerical solution of Navier-Stokes equations using multiquadric radial basis function networks, *International Journal for Numerical Methods in Fluids* **37**(1): 65–86.

Mai-Duy, N. and Tran-Cong, T. (2003). Approximation of function and its derivatives using radial basis function networks, *Applied Mathematical Modeling* **27**: 197–220.

Mai-Duy, N. and Tran-Cong, T. (2005). An efficient indirect RBFN-based method for numerical solution of PDEs, *Numerical Methods for Partial Differetial Equations* **21**: 770–790.

Mai-Duy, N. and Tran-Cong, T. (2008). Integrated radial-basis-function networks for computing Newtonian and non-Newtonian fluid flows, *Computers & Structures* **87**(11-12): 642–650.

Mai-Duy, N. and Tran-Cong, T. (2009). A Cartesian-grid discretisation scheme based on local integrated RBFNs for two-dimensional elliptic problems, *CMES: Computer Modeling in Engineering & Sciences* **51**(3)): 213–238.

Mai-Duy, N. and Tran-Cong, T. (2011). Compact local integrated-RBF approximations for second-order elliptic differential problems, *Journal of Computational Physics* **230**(12): 4772–4794.

Mai-Duy, N. and Tran-Cong, T. (2013). A compact five-point stencil based on integrated RBFs for 2D second-order differential problems, *Journal of Computational Physics* **235**: 302–321.

Marchioli, C., Giusti, A., Salvetti, M. V. and Soldati, A. (2003). Direct numerical simulation of particle wall transfer and deposition in upward turbulent pipe flow, *International Journal of Multiphase Flow* **29**(6): 1017–1038.

Mittal, R. and Iaccarino, G. (2005). Immersed boundary methods, *Annual Review of Fluid Mechanics* **37**(1): 239–261.

Monaghan, J. J. (2005). Smoothed particle hydrodynamics, *Reports on progress in physics* **68**: 1703.

Morris, J. P., Fox, P. J. and Zhu, Y. (1997). Modeling low Reynolds number incompressible flows using SPH, *Journal of Computational Physics* **136**: 214–226.

Mukherjee, Y. X. and Mukherjee, S. (1997). The boundary node method for potential problems, *International Journal for Numerical Methods in Engineering* **40**: 797–815.

Niu, X. D., Shu, C., Chew, Y. T. and Peng, Y. (2006). A momentum exchange-based immersed boundary-Lattice Boltzmann method for simulating incompressible viscous flows, *Physics Letters A* **354**(3): 173–182.

Noye, B. J. and Tan, H. H. (1989). Finite difference methods for solving the two-dimensional advection-diffusion equation, *International Journal for Numerical Methods in Fluids* **9**(1): 75–98.

Parlange, J. Y. (1980). Water transport in soils, *Annual Review of Fluid Mechanics* **12**(1): 77–102.

Patankar, N. A. and Sharma, N. (2005). A fast projection scheme for the direct numerical simulation of rigid particulate flows, *Communications in Numerical Methods in Engineering* **21**(8): 419–432.

Patankar, N. A., Singh, P., Joseph, D. D., Glowinski, R. and Pan, T. W. (2000). A new formulation of the distributed Lagrange multiplier/fictitious domain method for particulate flows, *International Journal Multiphase Flow* **26**: 1509–1524.

Patankar, S. (1980). *Numerical Heat Transfer and Fluid Flow*, Hemisphere Publishihg Corporation, USA.

Peaceman, D. W. and H. H. Rachford, J. (1955). The numerical solution of parabolic and elliptic differential equations, *Journal of the Society for Industrial and Applied Mathematics* **3**(1): 28–41.

Perot, J. (1993). An analysis of the fractional step method, *Journal of Computational Physics* **108**(1): 51–58.

Peskin, C. S. (1972). Flow patterns around heart valves: A numerical method, *Journal of Computational Physis* **10**(2): 252–271.

Peskin, C. S. (1977). Numerical analysis of blood flow in the heart, *Journal of Computational Physis* **25**: 220–252.

Peskin, C. S. (2002). The immersed boundary method, *Acta Numerica* **11**: 479–517.

Peyret, R. (2002). *Spectral Methods for Incompressible Viscous Flow*, Vol. 148 of *Applied Mathematical Sciences*, New York Springer-Verlag.

Peyret, R. and Taylor, T. (1983). *Computational Methods for Fluid Flow*, Springer-Verlag, New York, USA.

Phan-Thien, N. (2013). *Understanding Viscoelasticity: An Introduction to Rheology*, Springer.

Phan-Thien, N. and Kim, S. (1994). *Microstructures in Elastic media principles and computational methods*, Oxford University Press.

Pozrikidis, C. (2002). *A Practical Guide to Boundary Element Methods with The Software Library BEMLIB*, Chapman & Hall/CRC Press.

Raiskinmki, P., Shakib-Manesh, A., Koponen, A., Jsberg, A., Kataja, M. and Timonen, J. (2000). Simulations of non-spherical particles suspended in a shear flow, *Computer Physics Communications* **129**(1-3): 185–195.

Ramadanand, M. A., El-Danaf, T. S. and Alaal, F. (2005). A numerical solution of the Burgers' equation using septic B-splines, *Chaos, Solitons & Fractals* **26**(3): 795–804.

Randles, P. and Libersky, L. (1996). Smoothed particle hydrodynamics: some recent improvements and applications, *Computer Methods in Applied Mechanics and Engineering* **139**: 375–408.

Rippa, S. (1999). An algorithm for selecting a good value for the parameter c in radial basis function interpolation, *Advances in Computational Mathematics* **11**: 193–210.

Roache, P. J. (1998). *Fundamentals of Computational Fluid Dynamics*, Hermosa Publishers, USA.

Roma, A. M., Peskin, C. S. and Berger, M. J. (1999). An adaptive version of the immersed boundary method, *Journal of Computational Physics* **153**(2): 509–534.

Sahin, M. and Owens, R. (2003). A novel fully implicit finite volume method applied to the lid-driven cavity problem - Part I: High Reynolds number flow calculations, *International Journal for Numerical Methods in Fluids* **42**(1): 57–77.

Saiki, E. M. and Biringen, S. (1996). Numerical simulation of a cylinder in uniform flow: Application of a virtual boundary method, *Journal of Computational Physics* **123**(2): 450–465.

Saka, B. and Dag, I. (2007). Quartic B-spline collocation methods to the numerical solutions of the Burgers' equation, *Chaos, Solitons & Fractals* **32**(3): 1125–1137.

Sanmiguel-Rojas, E., Ortega-Casanova, J., del Pino, C. and Fernandez-Feria, R. (2005). A Cartesian grid finite-difference method for 2D incompressible viscous flows in irregular geometries, *Journal of Computational Physics* **204**(1): 302–318.

Sanyasiraju, Y. V. S. S. and Chandhini, G. (2008). Local radial basis function based gridfree scheme for unsteady incompressible viscous flows, *Journal of Computational Physics* **227**(20): 8922–8948.

Šarler, B., Perko, J. and Chen, C. S. (2004). Radial basis function collocation method solution of natural convection in porous media, *International Journal of Numerical Methods for Heat and Fluid Flow* **14**(2): 187–212.

Sheng, Z., Thiriet, M. and Hecht, F. (2011). An efficient numerical method for the equations of steady and unsteady flows of homogeneous incompressible Newtonian fluid, *Journal of Computational Physics* **230**(3): 551–571.

Sheu, T. W. H., Kao, N. S. C., Chiu, P. H. and Lin, C. S. (2011). Development of an upwinding scheme through the minimization of modified wavenumber error for the incompressible Navier-Stokes equations, *Numerical Heat Transfer, Part B: Fundamentals* **60**(3): 179–202.

Shih, T. M. and Tan, C. H. (1989). Effects of grid staggering on numerical schemes, *International Journal for Numerical Methods in Fluids* **9**(2): 193–212.

Shu, C. and Wu, Y. L. (2002). Domain-free discretization method for doubly connected domain and its application to simulate natural convection in eccentric annuli, *Computer Methods in Applied Mechanics and Engineering* **191**(17-18): 1827–1841.

Shu, C., Yao, Q., Yeo, K. S. and Zhu, Y. D. (2002). Numerical analysis of flow and thermal fields in arbitrary eccentric annulus by differential quadrature method, *Journal of Heat and Mass Transfer* **38**: 597–608.

Soh, W. Y. and Goodrich, J. W. (1988). Unsteady solution of incompressible Navier-Stokes equations, *Journal of Computational Physics* **79**(1): 113–134.

Soltani, M., Ahmadi, G., Ounis, H. and McLaughlin, J. (1998). Direct simulation of charged particle deposition in a turbulent flow, *International Journal of Multiphase Flow* **24**(1): 77–92.

Squires, K. D. and Eaton, J. K. (1990). Particle response and turbulence modification in isotropic turbulence, *Physics of Fluids A: Fluid Dynamics* **2**(7): 1191–1203.

Su, S.-W. and Lai, M.-C. (2007). An immersed boundary technique for simulating complex flows with rigid boundary, *Computers & Fluids* **36**(2): 313–324.

Sun, H. and Zhang, J. (2003). A high-order compact boundary value method for solving one-dimensional heat equations, *Numerical Methods for Partial Differential Equations* **19**(6): 846–857.

Tannehill, J., Anderson, D. and Pletcher, R. (1997). *Computational Fluid Mechanics and Heat Transfer*, Taylor & Francis, USA.

Tanner, R. I. (1985). *Engineering Rheology, Oxford Engineering Science Series 14*, Oxford University Press, USA.

Thai-Quang, N., Le-Cao, K., Mai-Duy, N., Tran, C.-D. and Tran-Cong, T. (2013). A numerical scheme based on compact integrated-RBFs and Adams-Bashforth/Crank-Nicolson algorithms for diffusion and unsteady fluid flow problems, *Engineering Analysis with Boundary Elements* **37**(12): 1653–1667.

Thai-Quang, N., Le-Cao, K., Mai-Duy, N. and Tran-Cong, T. (2011). Discretisation of the velocity-pressure formulation with integrated radial-basis-function networks, *Structural Longevity* **6**(2): 77–92.

Thai-Quang, N., Le-Cao, K., Mai-Duy, N. and Tran-Cong, T. (2012). A high-order compact local integrated-RBF scheme for steady-state incompressible viscous flows in the primitive variables, *CMES: Computer Modeling in Engineering & Sciences* **84**(6): 528–558.

Thai-Quang, N., Mai-Duy, N., Tran, C.-D. and Tran-Cong, T. (2012). High-order alternating direction implicit method based on compact integrated-RBF approximations for unsteady/steady convection-diffusion equations, *CMES: Computer Modeling in Engineering & Sciences* **89**(3): 189–220.

Thai-Quang, N., Mai-Duy, N., Tran, C.-D. and Tran-Cong, T. (2013). A direct forcing immersed boundary method employed with compact integrated RBF approximations for heat transfer and fluid flow problems, *CMES: Computer Modeling in Engineering & Sciences* **96**(1): 49–90.

Thomas, J. W. (1995). *Numerical Partial Differential Equations: Finite Difference Methods*, Springer, New York.

Tian, Z. F. and Ge, Y. B. (2007). A fourth-order compact ADI method for solving two-dimensional unsteady convection-diffusion problems, *Journal of Computational and Applied Mathematics* **198**(1): 268–286.

Tian, Z., Liang, X. and Yu, P. (2011). A higher order compact finite difference algorithm for solving the incompressible Navier-Stokes equations, *International Journal for Numerical Methods in Fluids* **88**: 511–532.

Torrance, K., Davis, R., Eike, K., Gill, P., Gutman, D., Hsui, A., Lyons, S. and Zien, H. (1972). Cavity flows driven by bouyancy and shear, *Journal of Fluid Mechanics* **51**(part2): 221–231.

Tran-Cong, T. (1989). *Boundary element method for some three-dimensional problems in continuum mechanics*, PhD thesis, The University of Sydney, Sydney, Australia.

Turek, S., Wan, D. and Rivkind, L. S. (2003). The fictitious boundary method for the implicit treatment of dirichlet boundary conditions with applications to incompressible flow simulations, *Challenges in Scientific Computing, Lecture Notes in Computational Science and Engineering* **35**: 37–68.

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

Van Kan, J. (1986). A second-order accurate pressure-correction scheme for viscous incompressible flow, *SIAM Journal on Scientific and Statistical Computing* **7**: 870.

Veeramani, C., Minev, P. and Nandakumar, K. (2007). A fictitious domain formulation for flows with rigid particles: A non-Lagrange multiplier version, *Journal of Computational Physics* **224**(2): 867–879.

Wan, D. and Turek, S. (2006). Direct numerical simulation of particulate flow via multigrid FEM techniques and the fictitious boundary method, *International Journal of Numerical Method Fluids* **51**: 531–566.

Wan, D. and Turek, S. (2007a). An efficient multigrid-FEM method for the simulation of solid-liquid two phase flows, *Journal of Computational and Applied Mathematics* **203**(2): 561–580.

Wan, D. and Turek, S. (2007b). Fictitious boundary and moving mesh methods for the numerical simulation of rigid particulate flows, *Journal of Computational Physics* **222**(1): 28–56.

Wang, Z., Fan, J. and Luo, K. (2008). Combined multi-direct forcing and immersed boundary method for simulating flows with moving particles, *International Journal of Multiphase Flow* **34**(3): 283–302.

Wu, J. and Shu, C. (2010). Particulate flow simulation via a boundary condition-enforced immersed boundary-Lattice Boltzmann scheme, *Communications in Computational Physics* **7**(4): 793–812.

Yang, Q. N., Zheng, J. J., Miao, Y. and Sima, Y. Z. (2011). An improved hybrid boundary node method for solving steady fluid flow problems, *Engineering Analysis with Boundary Elements* **35**: 18–24.

You, D. (2006). A high-order Pad ADI method for unsteady convection-diffusion equations, *Journal of Computational Physics* **214**(1): 1–11.

Yu, Z., Phan-Thien, N., Fan, Y. and Tanner, R. I. (2002). Viscoelastic mobility problem of a system of particles, *Journal of Non-Newtonian Fluid Mechanics* **104**: 87–124.

Yu, Z., Phan-Thien, N. and Tanner, R. I. (2004). Dynamic simulation of sphere motion in a vertical tube, *Journal of Fluid Mechanics* **518**: 61–93.

Yu, Z., Phan-Thien, N. and Tanner, R. I. (2007). Rotation of a spheroid in a Couette flow at moderate Reynolds numbers, *Physical Review E* **76**(2).

Zhu, C. G. and Wang, R. H. (2009). Numerical solution of Burgers' equation by cubic B-spline quasi-interpolation, *Applied Mathematics and Computation* **208**(1): 260–272.

Zhu, T., Zhang, J. and Atluri, S. N. (1998). A meshless local boundary integral equation (lbie) method for solving nonlinear problems, *Computational Mechanics* **22**: 174–186.

Zienkiewicz, O. and Taylor, R. (2000). *The Finite Element Method - Volum 3: Fluid Dynamics*, Butterworth-Heinemann, Oxford, UK.