

**DEVELOPMENT OF GAS KINETIC FLUX SOLVERS
AND THEIR APPLICATIONS**

SUN YU

(B. Eng., Zhejiang University, China)

**A THESIS SUBMITTED
FOR THE DEGREE OF DOCTOR OF PHILOSOPHY
DEPARTMENT OF MECHANICAL ENGINEERING**

NATIONAL UNIVERSITY OF SINGAPORE

2016

DECLARATION

I hereby declare that this thesis is my original work and it has been written by me in its entirety. I have duly acknowledged all the sources of information which have been used in the thesis.

This thesis has also not been submitted for any degree in any university previously.

Sun Yu

Sun Yu

01 August 2016

ACKNOWLEDGEMENTS

First and foremost, I would like to express my sincere gratitude to my advisors, Professor Shu Chang and Associate Professor Teo Chiang Juay, for their immense knowledge and honorable tenacity in research, and for their generous support and beneficial discussions during the past four years. This thesis would not have been possible without their invaluable guidance.

In addition, I would like to express my appreciation to the National University of Singapore for supporting my research in the Department of Mechanical Engineering.

My special thanks also go to all my friends and research group members: Dr. Wang Yan, Mr. Yang Liming, Dr. Ren Weiwei, Dr. Shao Jiangyan, Dr. Wang Junhong, Ms. Ge Mengyi, Mr. Yao Yang, Mr. Yu Kok Hwa, Mr. Chen Zhen, Ms. Wu Xiaodi, Mr. Ashoke Raman Kuppa and many others. My apology that I could not mention personally one by one.

Last, but most importantly, I wish to express my heartfelt gratitude to my family, for their endless love, support and encouragement.

TABLE OF CONTENTS

DECLARATION	i
ACKNOWLEDGEMENTS	ii
TABLE OF CONTENTS	iii
SUMMARY	ix
LIST OF TABLES	xi
LIST OF FIGURES	xii
NOMENCLATURE	xviii
Chapter 1 Introduction.....	1
1.1 Background	1
1.2 Navier-Stokes solvers.....	3
1.3 Boltzmann-type solvers.....	4
1.3.1 Lattice Boltzmann method	5
1.3.2 Kinetic flux vector splitting method	6
1.3.3 Gas kinetic BGK scheme	8
1.4 Motivations and objectives of the thesis	13
1.5 Organization of the thesis.....	14
Chapter 2 Development of a Switch Function-based Gas Kinetic Scheme for Simulation of Inviscid and Viscous Compressible Flows	17
2.1 Kinetic Flux Vector Splitting (KFVS) Method.....	18
2.2 A Switch Function-based Gas Kinetic Scheme (SF-GKS)	20

2.2.1 Governing equations and finite volume discretization	20
2.2.2 Evaluation of inviscid flux by switch function-based gas kinetic scheme	23
2.2.3 Evaluation of viscous flux by smooth function approximation	30
2.2.4 Computational sequence	31
2.3 Numerical Results and Discussion	32
2.3.1 1D Euler shock tube	33
2.3.2 Regular shock reflection	34
2.3.3 Double Mach reflection	34
2.3.4 Compressible turbulent flow around a RAE2822 airfoil	35
2.3.5 Hypersonic flow around one half of a cylinder	36
2.4 Concluding Remarks	37
Chapter 3 Development of Gas Kinetic Flux Solver for Two-dimensional Incompressible and Compressible Viscous Flows.....	47
3.1 Relationships between Particle Distribution Function in Boltzmann Equations and Conservative Flow Variables and Fluxes in Navier-Stokes Equations	48
3.1.1 Boltzmann equations and 7 conservation forms of moments.	48
3.1.2 Relationships of conservative flow variables and fluxes in Navier-Stokes equations with distribution function in Boltzmann equation	50
3.2 Gas kinetic BGK scheme	51
3.3 Gas kinetic Flux Solver (GKFS)	55
3.3.1 Basic formulations for evaluation of conservative flow variables.....	57
3.3.2 Basic formulations for evaluation of flux at cell interface	59

3.3.3 Three schemes for evaluation of conservative flow variables and fluxes at cell interface.....	61
3.3.4 Collision time and Prandtl number fix.....	68
3.3.5 Computational sequence	69
3.4 Numerical Results and Discussion.....	71
3.4.1 Decaying vortex flow.....	72
3.4.2 2D lid-driven flow in a square cavity	73
3.4.3 Viscous flow past a circular cylinder.....	75
3.4.4 Couette flow with a temperature gradient.....	77
3.4.5 Shock wave-boundary layer interaction.....	79
3.4.6 Transonic laminar flows over a NACA0012 airfoil	80
3.4.7 Hypersonic flow around a circular cylinder.....	81
3.5 Concluding Remarks	83
Chapter 4 Development of a Three-dimensional Gas kinetic Flux Solver for Simulation of Viscous Flows with Explicit formulations of Conservative Flow Variables and Numerical Fluxes.....	99
4.1 Boltzmann Equation, Maxwellian Distribution Function and Navier-Stokes Equations.....	100
4.1.1 Boltzmann equation and conservative forms of moments for Maxwellian distribution function.....	100
4.1.2 Macroscopic governing equations discretized by finite volume method..	102
4.2 Three-dimensional Gas kinetic Flux Solver.....	105

4.2.1 Evaluation of conservative flow variables \mathbf{W}^* at cell interface	107
4.2.2 Evaluation of numerical fluxes .. at cell interface	108
4.2.3 Collision time and Prandtl number fix.....	113
4.2.4 Computational sequence	115
4.3 Numerical Results and Discussions	116
4.3.1 3D lid-driven cavity flow.....	117
4.3.2 Incompressible flow past a stationary sphere	118
4.3.3 Flow around an ONERA M6 wing.....	120
4.3.4 Turbulent flow over the DPW Wings W1	121
4.3.5 DLR-F6 wing-body configuration	121
4.4 Concluding Remarks	123
Chapter 5 Boundary condition-enforced Immersed Boundary-Gas Kinetic Flux Solver and its applications for Moving Boundary Flows	134
5.1 Conventional immersed boundary method (IBM)	135
5.2 Boundary condition-enforced immersed boundary-gas kinetic flux solver (IB-GKFS)	137
5.2.1 Gas kinetic flux solver for prediction of the flow field \mathbf{u}^*	138
5.2.2 Boundary condition-enforced IBM for velocity correction	141
5.2.3 Computational sequence and force calculation.....	143
5.3 Numerical example and discussion	145
5.3.1 Flow past a stationary circular cylinder	145
5.3.2 Flow over a NACA0012 airfoil	147

5.3.3 Flow past an in-line oscillating cylinder	148
5.3.4 Flow past a transverse oscillating cylinder	150
5.3.5 Laminar flow around a rapid pitching NACA0015 airfoil	151
5.3.6 One particle sedimentation in a rectangular domain.....	153
5.4 Concluding remarks	154
Chapter 6 A diffuse interface immersed boundary method for simulation of compressible viscous flows around stationary and moving boundaries	168
6.1 Governing equations and GKFS for prediction of intermediate flow field	169
6.1.1 Governing equations and fractional step method.....	169
6.1.2 GKFS for prediction of intermediate flow variables	171
6.2 Diffuse interface IBM for compressible flow around stationary boundaries..	172
6.2.1 For stationary boundary problems	174
6.2.2 For moving boundary problems.....	179
6.2.3 Force calculation and solution procedure	183
6.3 Numerical validation and discussion	185
6.3.1 Flow around a circular cylinder	186
6.3.2 Compressible flow over a NACA0012 airfoil	187
6.3.3 Flow over a high-frequency plunging SD7003 airfoil.....	189
6.3.4 Compressible flow over a rotating cylinder.....	190
6.3.5 Laminar flow over a harmonic oscillating NACA0012 airfoil.....	191
6.4 Concluding remarks	193
Chapter 7 Conclusions and Recommendations.....	208

7.1 Conclusions	208
7.2 Recommendations	213
Appendix.....	214
Appendix A: Moments of Maxwellian Distribution Function	214
Appendix B: Expressions of Conservative Flow Variables and Fluxes in y Direction.....	216
Appendix C: Expression for heat flux q	220
References.....	223

SUMMARY

In the simulation of fluid flows, various mathematical models and the corresponding numerical approaches have been developed based on different scales and regimes. For simulating continuum flows, the well-established and dominant approaches are the Navier-Stokes (N-S) solvers and the Boltzmann-type solvers, which are respectively based on the macroscopic conservation laws and mesoscopic kinetic theory. In fact, there are some intrinsic connections between these two types of solvers because the N-S equations can be derived from the Boltzmann equation using the Chapman-Enskog (C-E) expansion analysis. An N-S solver can be theoretically obtained by solving the Boltzmann equation with Bhatnagar-Gross-Krook (BGK) approximation. Moreover, different from N-S solvers, Boltzmann-type solvers can compute the inviscid and viscous fluxes simultaneously. Generally, there are mainly three categories of Boltzmann-type solvers, i.e., the lattice Boltzmann method (LBM), kinetic flux vector scheme (KFVS) and gas kinetic BGK scheme. Among them, the gas kinetic BGK scheme performs much more consistently in the simulation of both incompressible and compressible flows, inviscid and viscous flows. However, the complexity and inefficiency of gas kinetic BGK scheme put a brake on its development and practical applications. As a result, this thesis is devoted to developing a series of novel gas kinetic flux solvers (GKFSs) and their applications for a variety of flow problems.

Firstly, two types of GKFSs have been successfully proposed for both inviscid and viscous flow simulations. The GKFSs are finite volume based schemes which directly solve the conservative governing equations recovered by Boltzmann equations with C-E theory. The macroscopic variables, which are defined at cell centers, are directly

updated by marching in time with fluxes calculated at cell interfaces. The fluxes of the GKFSs are modeled at each interface by local reconstruction of Boltzmann solutions with the connection between the macroscopic fluxes and the mesoscopic particle distribution function. The developed solvers have been validated in a variety of 1D to 3D flow simulations. Numerical results demonstrate that the present GKFSs not only keep the intrinsic advantages of the gas kinetic scheme but also remove the drawbacks, such as the complexity and inefficiency.

Subsequently, the extensions of the GKFSs to study complex and moving boundary problems have also been built. A GKFS-based solver combined with the immersed boundary method (IBM) has been proposed for incompressible flows. In this solver, a fractional step technique is applied to simplify the solution process. Numerical experiments demonstrate that the present method can accurately satisfy both the governing equations and boundary conditions. In addition, a diffuse interface IBM is further developed for the simulation of compressible moving boundary flows. A simple and flexible way to correct all the flow variables is introduced. This is the first time that the diffuse interface IBM is successfully applied to simulate compressible moving boundary flows.

LIST OF TABLES

Table 3.1 Locations of primary vortex centers at different Reynolds numbers.....	85
Table 3.2 Comparison of computational time (seconds)	85
Table 3.3 Comparison of drag coefficient, recirculation lengths and separation angles for a steady flow past a circular cylinder at different Reynolds numbers	86
Table 3.4 Comparison of dynamic parameters for an unsteady flow past a circular cylinder at different Reynolds numbers	87
Table 4.1 Comparison of drag coefficient for flow past a stationary sphere.....	124
Table 4.2 Comparison of force coefficients for DLR-F6 wing-body configuration..	124
Table 5.1 Comparison of drag coefficients, recirculation lengths and separation angles for steady flow past a circular cylinder at $Re=20$ and 40	156
Table 5.2 Comparison of dynamic parameters for unsteady flow past a circular cylinder at $Re=100$ and 200	157
Table 6.1 Comparison of drag coefficient for supersonic flow around a circular cylinder	195
Table 6.2 Lift and drag coefficients for viscous flow over a NACA0012 airfoil	195

LIST OF FIGURES

Figure 2.1 Schematic of velocity transformation at cell interface	40
Figure 2.2 Comparison of density, velocity, pressure and internal energy with the exact solution at time $t = 0.25$ of Test Case 1	40
Figure 2.3 Comparison of density, velocity, pressure and internal energy with the exact solution at time $t = 0.035$ of Test Case 2	41
Figure 2.4 Pressure contour of the shock reflection	41
Figure 2.5 Comparison of density profile for the shock reflection along the line $y = 0.5$	42
Figure 2.6 Density contours of the double Mach reflection obtained from present scheme (Upper) and Woodward and Colella (1984) (Lower)	42
Figure 2.7 Pressure contours of the double Mach reflection obtained from present scheme (Upper) and Woodward and Colella (1984) (Lower)	43
Figure 2.8 Contours of Mach number (Left) and switch function (Right) around the RAE2822 airfoil.....	43
Figure 2.9 Comparisons of pressure coefficient (Left) and skin friction (Right) distributions on the RAE2822 airfoil surface	44
Figure 2.10 Contours of original switch function (Left) and modified switch function (Right) for hypersonic flow around one half of a circular cylinder.....	44
Figure 2.11 Pressure (Left) and temperature (Right) contours of hypersonic flow.....	45
Figure 2.12 Pressure (Upper) and heat flux (Lower) distribution along the cylindrical surface.....	46
Figure 3.1 Schematic diagram of initial distribution function at the interface	88
Figure 3.2 L_2 norm of relative error of u versus h for the decaying vortex flow.....	88

Figure 3.3 U and V velocity profiles along horizontal and vertical central lines for a lid-driven cavity flow at various Reynolds numbers	90
Figure 3.4 Streamlines for a steady flow past a circular cylinder of two steady cases	91
Figure 3.5 Pressure coefficient distribution along the cylinder surface at $Re=40$	91
Figure 3.6 Evolution of the lift and drag coefficients for a flow past a circular cylinder	92
Figure 3.7 Streamlines for flow past a circular cylinder of two unsteady cases.....	92
Figure 3.8 Vorticity contours for a flow past a circular cylinder of two unsteady cases	93
Figure 3.9 Schematic of Couette flow with a temperature gradient	93
Figure 3.10 Comparison of temperature profile for various Prandtl numbers when $T_1 = T_0$	93
Figure 3.11 Comparison of temperature profile for various Prandtl numbers when $T_1 \neq T_0$	94
Figure 3.12 Computational mesh for shock-boundary layer interaction	94
Figure 3.13 Pressure contours of shock-boundary layer interaction given from present scheme (Upper) and gas kinetic BGK scheme (Xu, 2001) (Lower).....	95
Figure 3.14 Pressure (Upper) and skin friction (Lower) distributions along the flat plate of shock-boundary layer interaction.....	96
Figure 3.15 Comparison of CPU time between GKFS and gas kinetic BGK scheme	96
Figure 3.16 Streamlines of transonic laminar flow over a NACA0012 airfoil.....	97
Figure 3.17 Comparison of pressure coefficient (Left) and skin friction (Right) distributions on the airfoil surface	97
Figure 3.18 Pressure (Left) and temperature (Right) contours of hypersonic flows over one half of a cylinder	98

Figure 3.19 Pressure (Upper) and heat flux (Lower) along the cylindrical surface for hypersonic flow around one half of a cylinder	98
Figure 4.1 u and v velocity profiles on the plane of z=0.5 of cubic cavity for 3D lid-driven cavity flow at Reynolds numbers of 100 and 400	125
Figure 4.2 Streamlines on three mid-planes for 3D lid-driven cavity flow at Reynolds numbers of 100 (left) 400 (right)	127
Figure 4.3 Streamlines at four different Reynolds numbers of 50, 100, 150 and 200 in the steady axisymmetric regime.....	127
Figure 4.4 Comparison of recirculation length L_s at different Reynolds numbers ..	127
Figure 4.5 Streamlines for flow past a stationary sphere at Re=250 in the steady non-axisymmetric regime.....	128
Figure 4.6 Pressure contour of flow around an ONERA M6 wing	128
Figure 4.7 Comparison of pressure coefficient distribution at selected positions for ONERA M6 Wing	129
Figure 4.8 The DPW-W1 geometry with pressure contours.....	130
Figure 4.9 The Pressure Coefficient distribution along the DPW-W1	131
Figure 4.10 Pressure contours of DLR-F6 wing/body.....	131
Figure 4.11 Separation bubble on the intersection of wing and body obtained from Vassberg et al. (2007) (Left) and present scheme (Right).....	132
Figure 4.12 Comparison of pressure coefficient distribution of DLR-F6 wing/body at different locations	133
Figure 5.1 Streamlines for the flow over a circular cylinder at Re=20 and 40.....	158
Figure 5.2 Evolution of drag and lift coefficients for flow over a cylinder at Re=100 and 200.....	158

Figure 5.3 Pressure contours and streamlines for flow over a NACA0012 airfoil at $Re=500$ and $AoA=0^\circ$	159
Figure 5.4 Velocity profiles at five cross sections of NACA0012 airfoil	160
Figure 5.5 Vorticity contours at four different phase-angles for in-line oscillating cylinder in a fluid at rest. (a) 0° ; (b) 96° ; (c) 192° ; (d) 288°	160
Figure 5.6 Comparison of velocity profiles (u-component in the left column, v-component in the right column) at four different x locations and three different phase angles. (Lines are the present results, filled symbols are the experimental results of DÜTsch et al. (1998), and empty symbols represent numerical results of Wang et al. (2009)).....	161
Figure 5.7 Comparison of time evolution of in-line force F_x in a period time T for flow past an in-line oscillating cylinder at $Re=100$ and $KC=5$	162
Figure 5.8 Lift and drag coefficients for flow past an oscillating cylinder at $Re=185$	163
Figure 5.9 Comparison of \overline{Cd} , Cl_{RMS} and Cd_{RMS} for flow past a transverse oscillating cylinder at $Re=185$	163
Figure 5.10 Streamlines and vorticity contours for flow past a transverse oscillating cylinder at $Re=185$	164
Figure 5.11 Lift (Upper curve) and drag (Lower curve) coefficients versus angle of attack in degrees.....	165
Figure 5.12 Evolution of vorticity field for flow around a rapid pitching NACA0015 airfoil.....	166
Figure 5.13 Comparison of time evolution of four representative quantities for freely falling particle in a rectangular domain. (a) Longitudinal coordinate. (b) Longitudinal velocity. (c) Particle Reynolds number. (d) Translational kinetic energy.	166

Figure 5.14 Instantaneous vorticity contours for the freely falling particle in a rectangular domain at different times	167
Figure 6.1 Comparison of contours for flow over cylinder at $M_\infty = 1.2$, $Re = 300$ obtained from present scheme (Solid lines) and DVM (Flood and dashed lines)	196
Figure 6.2 Comparison of contours for flow over cylinder at $M_\infty = 2.0$, $Re = 300$ obtained from present scheme (Solid lines) and DVM (Flood and dashed lines)	197
Figure 6.3 Pressure coefficient distribution on the cylinder surface	198
Figure 6.4 Schematic diagram for flow over a NACA0012 airfoil	198
Figure 6.5 Comparison of contours for NACA0012 airfoil at $M_\infty = 0.5$, $\alpha = 0^\circ$, $Re = 5000$ obtained from present scheme (Solid lines) and body-fitted grids (Flood and dashed lines).....	199
Figure 6.6 Streamlines for flow over a NACA0012 airfoil at $M_\infty = 0.5$, $\alpha = 0^\circ$, $Re = 5000$	199
Figure 6.7 Pressure coefficient distribution for NACA0012 airfoil at $M_\infty = 0.5$, $\alpha = 0^\circ$, $Re = 5000$	200
Figure 6.8 Comparison of contours for NACA0012 airfoil at $M_\infty = 2.0$, $\alpha = 10^\circ$, $Re = 1000$ obtained from present scheme (Solid lines) and body-fitted grids (Flood and dashed lines).....	201
Figure 6.9 Pressure coefficient distribution for NACA0012 airfoil at $M_\infty = 2.0$, $\alpha = 10^\circ$, $Re = 1000$	201
Figure 6.10 Lift and drag coefficients versus normalized time t/T	202
Figure 6.11 Comparison of the vorticities of present results (Left) and Visbal et al. (2009) (Right)	203

Figure 6.12 The streamlines near the cylinder with pressure contours obtained from present scheme (Left) and DVM (Right)	203
Figure 6.13 Comparison of contours for flow past a rotational cylinder obtained from present scheme (Solid lines) and DVM (Flood and dashed lines).....	204
Figure 6.14 Pressure coefficient distributions along the y-direction	204
Figure 6.15 Schematic diagram for flow over a harmonic oscillating cylinder.....	205
Figure 6.16 Instantaneous density contours for the flow around an oscillating NACA0012 airfoil at $M_\infty = 0.4$, $Re = 5000$, $K_s = 0.5$ of Guo et al. (1994) (Left) and present results (Right)	207

NOMENCLATURE

Roman Letters

A	Velocity correction matrix
B, B_T	Velocity correction matrix
b	Total degree of freedom of molecules
c	Chord length of airfoil
$C_d, C_{d,p}, C_{d,f}$	Total, pressure, friction drag coefficients
C_l, C_p, C_f, C_M	Lift, pressure, skin friction and moment coefficients
D	Diameter of the cylinder
dt	Time step
E	Total energy
Ec	Eckert number
$F = (F_x, F_y, F_z)$	Flux vector
F_D, F_L	Drag and Lift forces
f	Particle distribution function
f_q	Vortex shedding frequency
f^{eq}, g	Equilibrium distribution function
f^{neq}	Non-equilibrium distribution function
h	Mesh Spacing
$H(x)$	Heaviside function
K	Internal degree of freedom
KC	Keulegan-Carpenter number

K_s	Reduced frequency of oscillation
L_s	Recirculation length of the vortex
M_∞	Free stream Mach number
$\mathbf{n} = (n_x, n_y, n_z)$	The unit outer normal vector
N	Number of interfaces
p, p_∞	Pressure
Pr	Prandtl number
Q	Collision operator
Q_s	Source term
R	Gas constant
Re	Reynolds number
S_i	Length of interface
St	Strouhal number
T	Temperature
δt	Streaming time
U_∞	Free stream velocity
$\mathbf{U}, \mathbf{U}_B, \mathbf{u}, \mathbf{u}^*, \Delta \mathbf{u}, \Delta \mathbf{u}_B^k$	Velocity vector
ΔV	Volume of the control cell
\mathbf{W}, \mathbf{W}_i	Vector of Conservative flow properties
u, v, w	Particle velocity
$\mathbf{X}, \mathbf{X}_B, \mathbf{X}_T$	Coordinate vector
x, y, z	Coordinates

Greek Letters

$d\Xi$	Volume element in the phase space
α	Angle of attack
δ	Dirac delta function
κ	Coefficient in free energy equation
μ	Dynamic viscosity\
τ	Collision time
τ_w	Wall shear stress
η, η^+	Switch function
ν	Kinetic viscosity
ρ	Density
θ_s	Separation angle
Ω	Computational domain
ξ	Internal energy
ω	Angular velocity
φ_α	Vector of moments
λ	Function of temperature
γ	Specific heat ratio

Abbreviations

2D	Two-dimensional
3D	Three-dimensional
ALE	Arbitrary Lagrangian-Eulerian

AP	Asymptotic preserving
BGK	Bhatnagar-Gross-Krook
C-E	Chapman-Enskog
CFD	Computational fluid dynamics
DG	Discontinuous Galerkin
DPD	Dissipative particle dynamics
DSMC	Direct simulation Monte Carlo
DUGKS	Discrete unified gas kinetic scheme
DVM	Discrete velocity method
EFM	Equilibrium-flux method
ENO	Essential non-oscillatory
FDM	Finite difference method
FDS	Flux Difference Splitting
FEM	Finite element method
FVM	Finite volume method
FVS	Flux vector splitting
GKFS	Gas kinetic flux solver
GKS	Gas kinetic scheme
IBM	Immersed boundary method
KFVS	Kinetic flux vector scheme
LBFS	Lattice Boltzmann flux solver
LBM	Lattice Boltzmann method
LGA	Lattice gas automation
MD	Molecular dynamics
N-S	Navier-Stokes

PDEs	Partial differential equations
SD	Spectral difference
SF-GKS	Switch function-based gas-kinetic scheme
SPH	Smoothed particle hydrodynamics
SV	Spectral volume
TVD	Total variation diminishing
UGKS	Unified gas kinetic scheme
WENO	Weighted essential non-oscillatory

Chapter 1

Introduction

1.1 Background

As a branch of fluid dynamics, Computational fluid dynamics (CFD) is developed to investigate the mechanism and behavior of fluid flows. With the great strides made in computers, CFD has become an efficient and powerful tool in both academic research and industrial applications, such as aerodynamics, industrial manufacturing, civil and environmental engineering.

In the analysis of fluid flows, different mathematical models have been developed based on different scales and regimes, i.e., macroscopic level, mesoscopic level and microscopic level. At the macroscopic level, the fundamental physical principles are the conservation of mass, momentum and energy. Based on these principles, the mathematical statement can be made with a set of partial differential equations (PDEs), namely, the Navier-Stokes (N-S) equations. In the evolution of N-S equations, three important assumptions are made, which are (i) continuum hypothesis, (ii) linear constitutive relationship, and (iii) Fourier's law for heat conduction. When the flow is inviscid, in which the dissipation, mass diffusion, thermal conductivity and transport phenomena of viscosity can be neglected, the PDEs can be simplified to the Euler equations. The numerical approaches to solve the macroscopic governing equations can be termed as the N-S solvers.

Different from the macroscopic view, the other ways to describe the fluid behavior are in the framework of mesoscopic or microscopic levels. If a relative large framework is chosen, the fluid can be viewed as a set of particles. The particle distribution function is introduced to describe fluid behavior and the governing equations become the Boltzmann equation or the Newton's equation of motion. The representative numerical approaches in this category are the gas kinetic scheme (GKS), lattice Boltzmann method (LBM), dissipative particle dynamics (DPD) and smoothed particle hydrodynamics (SPH). On the other hand, in the relatively small framework of microscopic level, the behavior of fluid molecules is investigated by using the Newtonian law of the classical mechanics. The numerical methods in this category solve the Boltzmann equations directly with the discretization of velocity space of molecules. Among the microscopic methods, the molecular dynamics (MD) method and direct simulation Monte Carlo (DSMC) method are the ones which are the most well-known. These methods can be applied to solve fluid problems in all flow regimes, ranging from continuum regime to highly-rarefied regime. However, it should be noted that these methods are seldom implemented in continuum flow simulations due to enormous computational resources required.

In the remaining parts of this chapter, a brief introduction and literature review of several popular methods on both macroscopic and mesoscopic levels will be presented.

1.2 Navier-Stokes solvers

The N-S and Euler equations are the basic macroscopic governing equations for viscous and inviscid fluid flows, respectively. Owing to the constraint of nonlinearity of the governing equations, they cannot be solved directly except for very few cases. Instead, the approximate numerical solutions of the governing equations can be obtained by applying discretization methods. By using discretization methods, the governing equations can be written as a set of algebraic equations or difference equations, which can be solved on a computer. Traditionally, the finite difference method (FDM), the finite volume method (FVM) and the finite element method (FEM) are the most popular numerical discretization methods. Detailed information about these numerical methods for macroscopic governing equations has been accumulated in the literature (Roache, 1972; LeVeque, 1992; Anderson and Wendt, 1995; Versteeg and Malalasekera, 2007; Donea and Huerta, 2003).

In the past several decades, a large number of prominent numerical algorithms for hyperbolic conservation laws, e.g. the Euler equations, have been developed. The pioneer work in this category can be referred to the Godunov method (1959), which laid a foundation for the development of modern upwind schemes. After that, a number of upwind schemes have been proposed, including Roe's method (Roe, 1981), Osher's method (Engquist and Osher, 1981), flux vector splitting (FVS) method (Steger and Warming, 1981) and TVD (Total Variation Diminishing) (Harten, 1983). In recent years, high-order numerical algorithms for hyperbolic conservation laws have attracted great attention in the simulation of complex flow structures, such as Essential Non-Oscillatory (ENO) (Harten et al., 1987; Shu and Osher, 1988), Weighted Essential Non-Oscillatory (WENO) (Liu et al. 1994; Jiang and Shu, 1996),

discontinuous Galerkin (DG) method (Cockburn and Shu, 1989; Bassi and Rebay, 1997), spectral volume (SV) (Wang, 2002; Liu et al., 2006b), spectral difference (SD) (Liu et al., 2006a) methods and so on.

Although the above solvers can well simulate inviscid flows, it is another story for viscous flows. As the Riemann solvers for N-S equations are not available, they cannot be directly used for viscous flows. One of the popular ways is to treat the convective terms and dissipative terms separately. Take the FVM as an example, the inviscid flux of the N-S equations is evaluated by the approximate Riemann solvers and the viscous flux is calculated by the smooth function approximation due to the elliptical characteristic of the viscous flux. Although this treatment works well in many applications, it is not a consistent way and sometimes creates ambiguity from both physical and mathematical views (Liu, 2007).

1.3 Boltzmann-type solvers

In the past few years, the Boltzmann-type solvers for the simulation of both incompressible and compressible fluid flows have attracted much attention. In general, this type of solvers is constructed based on the Boltzmann equation, as opposed to N-S solvers based on discretization of the macroscopic governing equations. In fact, there are some intrinsic connections between the Boltzmann-type solvers and N-S solvers. To be specific, by applying the Chapman-Enskog (C-E) expansion analysis, the N-S and Euler equations can be recovered from the Boltzmann equation with the zeroth and first order of collision time, respectively.

Among the Boltzmann-type solvers, the lattice Boltzmann method (LBM), kinetic flux vector scheme (KFVS) and gas kinetic Bhatnagar-Gross-Krook (BGK) scheme have been extensively studied and significant progress has been made in the development.

1.3.1 Lattice Boltzmann method

In recent decades, the LBM has emerged as a promising and competent numerical algorithm for simulating incompressible fluid flows and modeling physics of fluids (Chen and Doolen, 1998). This method is based on mesoscopic kinetic equation (the lattice Boltzmann equation) and microscopic particle model. The macroscopic dynamics of a fluid are evaluated by the collective behavior of microscopic particle distributions in the simplified particle-velocity space.

The LBM is originated from the lattice gas automation (LGA) (Hardy et al., 1973), which is constructed as a simplified, fictitious molecular dynamic model. The LGA suffers from some essential drawbacks, such as non-Galilean invariance, dependence of pressure on velocity and large numerical dissipation. These drawbacks hampered its development in the practical application. To overcome these drawbacks, LBM was first developed by McNamara and Zanetti (1988). In their work, the single-particle distribution function was introduced to replace the Boolean variables and thus the statistical noise was eliminated. In 1989, Higuera and Jimenez (1989) further simplified this method by introducing a linearized collision operator. The computational efficiency was effectively enhanced as compared with the previous work. By introducing the BGK relaxation approximation into the collision operator, the LBM simplified by Qian et al. (1992) completely eliminates the drawbacks of

LGA. Since then, the LBM has been widely employed for the simulation of complex fluid flows in a variety of areas. Recently, a lattice Boltzmann flux solver (LBFS) (Shu et al., 2014; Wang et al., 2014) was proposed based on the local reconstruction of lattice Boltzmann equation. The LBFS removes the major drawbacks of the conventional LBM, such as uniform mesh, tie-up of mesh spacing and time interval. However, one fundamental flaw for both LBM and LBFS is that they are only valid for near incompressible flows.

1.3.2 Kinetic flux vector splitting method

The KFVS method, which is also referred as equilibrium-flux method (EFM), is based on the collisionless Boltzmann equation. There are two stages in KFVS method: free transport and collision. Firstly, in the free transport stage, the collisionless Boltzmann equation is solved for the flux evaluation. Then, in the collision stage, the artificial collision is implicitly implemented in the calculation of initial Maxwellian distribution at the beginning of the next time step.

The KFVS method was first developed for the solution of Euler equations governing inviscid compressible flows. The early work of KFVS method can be referred to the EFM (Pullin, 1980), which was the first to split the Maxwellian distribution into two parts. Elizarova and Chetverushkin (1985) later presented a kinetic-consistent finite scheme, which was quite similar in principle with EFM. Macrossan (1989) proved that the EFM was a natural upwind scheme and could be applied to calculate a chemically reacting gas mixture problem. Deshpande (1986) and Mandal and Deshpande (1994) successfully proposed the KFVS method based on finite volume discretization with the Maxwell-Boltzmann distribution function on each side of a cell

interface. By using the half Maxwellian distribution, the interface flux could be computed by taking the appropriate moments with the complete error function. Perthame (1992) further developed this scheme by replacing the real Maxwellian function with the characteristic function. The entropy and positivity properties of the developed scheme were also proven. Moschetta and Pullin (1997) combined the robustness of KFVS method for strong shocks with the accuracy of flux difference splitting (FDS) schemes for contact discontinuities. Recently, Yang et al. (2013) proposed a circular function-based KFVS method, where the Maxwellian function was simplified to a circular function so that the error and exponential functions were avoided.

There are also some attempts to apply the KFVS scheme for the viscous flows. Chou and Baganoff (1997) applied the KFVS method to solve the N-S equations by using the C-E approximation to the Boltzmann equation with the BGK model. It was claimed that the positivity property of the first-order KFVS was rigorously kept for the simulation of compressible viscous flows. However, it was demonstrated that the KFVS scheme usually gave more diffusive results than the Godunov or FDS scheme (Xu, 2001). This is because the numerical dissipation of the KFVS scheme is proportional to the mesh size (Xu, 1998). Owing to the large dissipation introduced, the KFVS scheme is not able to give accurate N-S solutions except for some cases in which the physical viscosity is much larger than the numerical viscosity. As a result, the KFVS scheme is usually limited to approximate the Euler equations.

1.3.3 Gas kinetic BGK scheme

The gas kinetic BGK scheme was firstly proposed by Prendergast and Xu (1993) and then developed gradually afterwards (Xu et al., 1995; Chae et al., 2000; Xu, 2001). In this method, the BGK collision model is adopted in the gas evaluation stage to obtain the numerical fluxes across the interface. As a consequence, the dissipation in the transport can be controlled by a real collision time, which is a function of dynamic viscosity and pressure. In contrast to conventional upwind schemes, the gas kinetic BGK scheme computes the inviscid and viscous fluxes simultaneously from the solution of the Boltzmann equation with BGK approximation. In the work of Xu (2001), it has been shown that this scheme is able to generate a stable and crisp shock transition in the discontinuous region with a delicate dissipative mechanism. At the same time, an accurate N-S solution can be obtained in the smooth region. Moreover, it is demonstrated that the entropy condition is always fulfilled in the gas kinetic BGK scheme and the “carbuncle phenomenon” is avoided for hypersonic flow simulations (Xu et al., 2010). Owing to its distinctive features, the gas kinetic BGK scheme has attracted more and more attention and has been applied to various flow problems, including low speed flow, all Knudsen number flow, three-dimensional (3D) flow, multi-material flow. A brief review of these applications will be given next.

Low speed flow

The gas kinetic BGK scheme is originally targeted to simulate compressible flows. The extension of the gas kinetic BGK scheme to low speed flow was first carried out by Su et al. (1999). They showed that the gas kinetic BGK scheme could faithfully model low Mach number flows. In contrast to N-S solvers, the Poisson equation is not involved in the gas kinetic BGK scheme. Later, Xu and He (2003) compared the LBM

and gas kinetic BGK scheme by simulating isothermal low speed flow. The similarities between the LBM and gas kinetic BGK scheme were introduced and the accuracy of both methods was examined. Based on the work of Xu and He (2003), Guo et al. (2008) further compared the numerical stability and computational efficiency of the LBM and gas kinetic BGK scheme. They demonstrated that the gas kinetic BGK scheme was far more robust than the LBM, while on the other hand, the LBM had a dominant computational efficiency in the computation of low speed flows. Chen et al. (2011) adopted discontinuous derivatives for the initial reconstruction of flow variables around a cell interface. They claimed that the accuracy and robustness of the gas kinetic BGK scheme were improved with this weak discontinuity. Yuan et al. (2015) combined the gas kinetic BGK scheme and the immersed boundary method (IBM) for the simulation of incompressible viscous flows. In their method, the external force involved in the gas distribution function was calculated by an iterative procedure to guarantee the no-slip boundary condition. The results showed that the flow penetration was eliminated in the simulation of incompressible flows with complex and moving objects.

All Knudsen number flows

The work of exploring the capability of the gas kinetic scheme for all Knudsen number flows has attracted a lot of attention in recent years. Xu and Huang (2010) proposed a unified gas kinetic scheme (UGKS) which could perform simulations in both continuum and rarefied flow regimes with discretized particle velocity space. As an aggressive extension of the gas kinetic BGK scheme, the UGKS couples the effects of the particle transport and collision effects when updating the distribution function. Chen et al. (2012) extended the UGKS to an adaptive quadtree version in the particle

velocity space in order to improve the efficiency of the UGKS. Moreover, in their work, a moving mesh technique was implemented to enable the UGKS to compute moving solid-gas interactions. Guo et al. (2013, 2015) recently proposed an alternative unified kinetic method for low-speed isothermal and compressible thermal flows, which is called discrete unified gas kinetic scheme (DUGKS). The DUGKS has the same modeling mechanism and shares some common features with the UGKS method, such as asymptotic preserving (AP) properties. The main difference between the UGKS and DUGKS lies in the evolution of numerical flux for the discrete distribution function. Different from the UGKS, a modified distribution function is introduced in DUGKS to remove the implicit treatment of the collision term. Furthermore, the updating rule is much simpler for the DUGKS as the evolution of macroscopic variables is avoided. Yang et al. (2016) adopted the idea of streaming and collision processes of UGKS and proposed a discrete velocity method (DVM) for simulation of flows from the free molecular regime to the continuum regime. With the integration of kinetic equation for each discrete velocity, the algebraic formulation can be simply obtained.

Three-dimensional flow

The simulation of 3D high-speed viscous flows is one of the most popular research topics in aerodynamics. It is still a challenging issue today to accurately capture the complex flow physics under 3D high Mach number conditions. Many numerical methods suffer from the instability and inconsistency of the physical viscosity and the artificial one. As an alternative approach, the gas kinetic BGK scheme was first applied to simulate 3D compressible flow by Ruan and Jameson (2002). They showed that with the help of the convergence acceleration methods such as local time stepping

and multi-grid technique, the gas kinetic BGK scheme could calculate the 3D subsonic and supersonic flows accurately. May et al. (2007) proposed a modification to the data reconstruction procedure and extended the modified scheme to 3D complex geometries with unstructured meshes. Tian et al. (2007) then broadened the application of gas kinetic BGK scheme to multi-dimensional gas dynamics equations under gravitational fields. Later, the hypersonic flow around a blunt body was investigated by Li and Fu (2011). The consistency in the wall heat flux and flow pattern with the experimental data showed the accuracy and stability of gas kinetic BGK scheme to the application of 3D hypersonic flows. Based on WENO reconstruction, Pan and Xu (2015) proposed a high-order gas kinetic scheme for 3D Euler and N-S solutions.

Multi-material flow

The compressible multi-material flows associated with discontinuities and shock waves arise in academic research and technological applications. As an alternative approach, the gas kinetic BGK scheme can be extended to solve multi-material flows with the incorporation of appropriate multifluid flow models. Jiang and Ni (2004, 2007) proposed a conservative γ -model scheme in the gas kinetic BGK scheme. In this model, the specific heat ratio γ is considered as a contact discontinuity for two different materials. Li et al. (2005) proposed an easy and efficient scheme by directly coupling the scalar function with the gas distribution function. As a result, the scalar function, more or less like a color function, was involved and updated together with other conservative flow variables. Ni et al. (2012) presented an arbitrary Lagrangian-Eulerian (ALE)-based adaptive moving mesh technique to keep the sharpness of material interfaces. Pan et al. (2012) developed a new gas kinetic scheme combined

with the Baer-Nunziato two-phase flow model containing non-conservative terms. In their method, the non-conservative terms are explicitly calculated during the construction of numerical fluxes by the gas kinetic theory.

From the above review, it is clear that the gas kinetic BGK scheme is developed to be a powerful numerical algorithm and has been successfully applied to simulate a variety of flow problems. The distinct features of the gas kinetic BGK scheme can be summarized as: First, based on the kinetic theory, the gas kinetic BGK scheme has an intrinsic kinetic nature and can include extended hydrodynamics or aerodynamics beyond the regime of N-S equations. For this reason, the gas kinetic BGK scheme is particularly appealing in the modeling of non-linear physics of complex fluids. Second, the introduction of a real collision time by the BGK model provides the gas kinetic BGK scheme with a delicate dissipative mechanism. Therefore, the gas kinetic BGK scheme is able to generate a stable and crisp shock transition in the discontinuous region. At the same time, accurate N-S solutions can be obtained in the smooth region.

In spite of these advantages, the gas kinetic BGK scheme also suffers from some drawbacks. Firstly, it is usually more complicated and inefficient than conventional N-S solvers. This is because at each interface and each time step, numerous terms and coefficients associated with non-equilibrium distribution functions have to be calculated and stored to yield the numerical fluxes. Guo et al. (2008) claimed that for two-dimensional (2D) problems, the gas kinetic BGK scheme is about 10 and 3 times slower than LBM for steady and unsteady flow calculations, respectively. Secondly, it is an arduous task to obtain explicit formulations for the numerical flux. Tang (2012)

pointed out that the explicit expressions for the numerical flux in terms of the macroscopic variables in gas kinetic BGK scheme took more than eight pages after weeks of tedious derivation. The work of Tang (2012) was only limited to 2D cases and it can be imagined that it will be much more difficult to obtain the explicit expressions for the numerical flux in 3D cases. These drawbacks motivate the work on developing a new solver.

1.4 Motivations and objectives of the thesis

From the above review, we can see that, the gas kinetic BGK scheme has some remarkable advantages. On the other hand, the complexity and inefficiency of this method put a brake on its development and practical applications. In the literature, several works have been done to simplify this method. Chae et al. (2000) abandoned a time evolution term in the integral solution of BGK model and they claimed that the computational efficiency and convergence were improved. May et al. (2007) proposed two modifications to the conventional gas kinetic BGK scheme. First, they proposed a new formulation to the calculation of the initial non-equilibrium terms in the consideration of the relaxation state. A new time derivative was also introduced to reduce the CPU time. In the work of Tang (2012), the spatial derivative of the equilibrium distribution function across the interface was assumed to be continuous rather than piecewise linear in the conventional scheme. All the time-related terms were not considered in the calculation of flux. Most of the above modifications are to improve the original gas kinetic BGK scheme, where the non-equilibrium distribution function is approximated by a low order polynomial in terms of time, physical space and phase velocity space. In this way, many terms associated with phase velocity,

space coordinate and time should still be considered. The simplification in these terms may add more uncertainty to the derivation of the gas kinetic BGK scheme.

In view of these features, it is natural to ask whether we can develop a simplified gas kinetic scheme which keeps the intrinsic advantages of the original gas kinetic BGK scheme and removes the drawbacks at the same time. Motivated by this, we aim to develop a series of new solvers in a completely new framework for a variety of flow problems. Hence, the main objectives of this study are:

- To develop a switch function-based gas kinetic scheme (SF-GKS) for inviscid and viscous compressible flows;
- To develop a gas kinetic flux solver (GKFS) for 2D incompressible and compressible viscous flows;
- To develop a truly 3D GKFS for simulation of incompressible and compressible viscous flows;
- To develop an immersed boundary-gas kinetic flux solver (IB-GKFS) for incompressible stationary and moving boundary problems,
- To develop a novel diffuse interface IBM in the fixed Eulerian coordinates for solving compressible viscous flow problems.

1.5 Organization of the thesis

The rest of this thesis is organized as follows:

In Chapter 2, the SF-GKS will be proposed for inviscid and viscous compressible flows. Firstly, the KFVS method, which is based on the collisionless Boltzmann equation, is introduced. Thereafter, in order to remove the drawback of the KFVS that

numerous numerical dissipation is introduced in the scheme, the SF-GKS, in which a switch function is brought in to control the numerical dissipation, will be developed. The accuracy and capability of the SF-GKS for both inviscid and viscous flows will be validated by simulating several 1D and 2D problems.

In Chapter 3, development of the GKFS for 2D incompressible and compressible viscous flows will be presented. First, the conventional gas kinetic BGK scheme will be described. After that, the GKFS, which is a finite volume solver that reconstructs the viscous and inviscid fluxes from the continuous Boltzmann equation, will be developed. The accuracy and high efficiency of the GKFS will be validated by simulating the decaying vortex flows and shock wave-boundary layer interaction. The capability of the GKFS for 2D incompressible and compressible flows including hypersonic flow problems will be demonstrated by simulating a few benchmark test cases.

In Chapter 4, to extend the GKFS to solve 3D flows, a truly 3D GKFS for simulation of incompressible and compressible viscous flows will be proposed. In the development, a local coordinate transformation will be introduced to transform the velocities in the Cartesian coordinate system to the local normal and tangential directions at each cell interface. For the first time, the explicit formulations for evaluating the conservative flow variables and numerical fluxes for the 3D viscous flow problems will be given. The proposed solver will be validated through various 3D numerical examples, including 3D lid-driven cavity flow, incompressible flow past a stationary sphere, flow around an ONERA M6 wing, turbulent flow over the DPW III wing and DLR-F6 wing-body configuration.

In Chapter 5, to extend the GKFS for complex moving boundary flows, the IB-GKFS will be proposed using fixed Eulerian coordinates. To begin with, the basic idea of conventional IBM and two different ways to compute the force density will be introduced. After that, the development of IB-GKFS, which introduces the boundary condition-enforced IBM to the GKFS by applying the fractional-step approach, will be presented. The proposed IB-GKFS will be validated through numerical simulation of a variety of 2D and 3D stationary and moving boundary flow problems. The numerical examples include: flows past a stationary cylinder and a NACA0012 airfoil, flows past a moving cylinder and a NACA0015 airfoil with fixed body trajectory and one particle sedimentation in a rectangular box.

In Chapter 6, a novel diffuse interface IBM in the fixed Eulerian coordinates will be proposed for solving compressible viscous flows. The mechanism to correct the density, velocity, pressure and temperature fields for stationary and moving boundary problems will be presented, respectively. In the validation, the proposed solver will be first tested by simulating several stationary boundary problems. Thereafter, numerical experiments of three moving boundary problems will be carried out to validate the flexibility and capability of the present method for solving moving boundary problems.

In the last chapter, a conclusion of the thesis will be summarized and recommendations for future research will also be addressed.

Chapter 2

Development of a Switch Function-based Gas Kinetic Scheme for Simulation of Inviscid and Viscous Compressible Flows

This chapter first introduces the basic theory of kinetic flux vector splitting (KFVS) method. This method is based on the Boltzmann equation with vanishing collision term. In this situation, the distribution function f is equal to the Maxwellian distribution function g . The Euler solutions can be derived from the collisionless Boltzmann equation with the numerical dissipation added in the projection stage. However, the KFVS usually produces a large numerical dissipation and heat conduction because the numerical dissipation of KFVS is proportional to the mesh size. Thus, the KFVS is not able to give accurate N-S solutions except for cases in which the physical viscosity is much larger than the numerical viscosity. To remove this drawback, a switch function-based gas kinetic scheme (SF-GKS) is proposed in this chapter. A switch function is brought in to control the numerical dissipation. In this way, the present SF-GKS can well capture strong shock waves and thin boundary layers simultaneously. Another advantage of SF-GKS is that the value of the switch function can be easily determined, which makes this method efficient and easy to be implemented. To validate the present scheme, some inviscid flows such as 1-D Euler shock tube, regular shock reflection and double Mach reflection are first examined. Then, the SF-GKS is further extended to simulate viscous flows, including

compressible turbulent flows around a RAE2822 airfoil and hypersonic flow over one half of a cylinder.

2.1 Kinetic Flux Vector Splitting (KFVS) Method

Without the external forcing term, the standard 2D Boltzmann equation can be written as

$$\frac{\partial f}{\partial t} + u \frac{\partial f}{\partial x} + v \frac{\partial f}{\partial y} = Q(f, f) \quad (2.1)$$

where f is the particle distribution function, (x, y) and (u, v) are the particle space and velocity, and $Q(f, f)$ is the collision operator. With the physical constraints of conservation of mass, momentum and energy during collisions, the compatibility condition should be satisfied:

$$\int Q(f, f) \varphi_\alpha d\Xi = 0 \quad (2.2)$$

Here $d\Xi = dudvd\xi$ is the volume element in the phase space with $d\xi = d\xi_1 d\xi_2 \dots d\xi_K$,

where K is the internal degree of freedom. φ_α are the moments given as

$$\varphi_\alpha = \left(1, u, v, \frac{1}{2}(u^2 + v^2 + \xi^2) \right)^T \quad (2.3)$$

where ξ is the internal energy with the notation $\xi^2 = \xi_1^2 + \xi_2^2 + \dots + \xi_K^2$. From the Boltzmann equation, the N-S and Euler equations can be obtained from Eq. (2.1) by using the C-E expansions analysis (Chapman and Cowling, 1970).

In KFVS, the collision term $Q(f, f)$ goes to zero, which means that the distribution function f is equal to the Maxwellian distribution g for an equilibrium state. As a result, the Boltzmann equation becomes

$$\frac{\partial f}{\partial t} + u \frac{\partial f}{\partial x} + v \frac{\partial f}{\partial y} = 0 \quad (2.4)$$

which is called the collisionless Boltzmann equation. With the initial condition of the gas distribution function $f_0(x, y, 0)$ at time $t=0$, the exact solution of the collisionless Boltzmann equation is

$$f(x, y, t) = f_0(x-ut, y-vt, 0) \quad (2.5)$$

For simulation of inviscid flows, f_0 can be taken as a piecewise constant function around the cell interface $x=0$, i.e.,

$$f_0 = \begin{cases} g_l, & x < 0 \\ g_r, & x \geq 0 \end{cases} \quad (2.6)$$

$$= g_l(1-H(x)) + g_r H(x)$$

where $H(x)$ is the Heaviside function, $H(x)=1$ for $x \geq 0$ and $H(x)=0$ for $x < 0$.

g_l and g_r stand for the equilibrium distribution function at the left and right sides of cell interface with

$$g = \rho \left(\frac{\lambda}{\pi} \right)^{\frac{K+2}{2}} e^{-\lambda((u-U)^2 + (v-V)^2 + \xi^2)} \quad (2.7)$$

Here $\lambda = \frac{1}{2RT}$, where R is the gas constant and T is the temperature.

From the distribution function in Eq. (2.6), the numerical fluxes for the mass, momentum and energy across the cell interface can be constructed by

$$\mathbf{F} = \int u \varphi_\alpha f d\Xi = \int_{u \geq 0} u \varphi_{\alpha,l} g_l d\Xi + \int_{u < 0} u \varphi_{\alpha,r} g_r d\Xi \quad (2.8)$$

In the above Eq. (2.8), “ \bullet_l ” and “ \bullet_r ” (“ \bullet ” stands for any variables) can be constructed from conservative flow variables at the left and right sides of cell interface, respectively. As soon as the fluxes at each cell interface are calculated via

Eq. (2.8), the conservative flow variables on each cell center can be updated by solving the governing equations with finite volume discretization.

The KFVS has been demonstrated with good positivity property for simulation of flows with strong shock waves (Yang et al., 2014b). However, the KFVS suffers from its intrinsic drawback. The KFVS usually gives more diffusive results than the Godunov or Flux Difference Splitting (FDS) scheme because the numerical dissipation in KFVS is proportional to the mesh size (Xu, 1998). As a result, the results might be contaminated in smooth regions, such as laminar boundary layer. Thus, the KFVS is not able to give accurate N-S solutions except for cases in which the physical viscosity is much larger than the numerical viscosity. To eliminate this drawback, the following novel method is proposed, which can well capture both strong shock waves and thin boundary layers simultaneously.

2.2 A Switch Function-based Gas Kinetic Scheme (SF-GKS)

In this section, SF-GKS is proposed for the simulation of inviscid and viscous compressible flows by solving the governing equations with finite volume discretization. The inviscid flux is calculated by the SF-GKS and the viscous flux is obtained by smooth function approximation.

2.2.1 Governing equations and finite volume discretization

For an inviscid flow, Euler equations are commonly used in conventional computational fluid dynamics. On the other hand, N-S equations are usually used to simulate viscous flows. Generally, with the finite volume discretization, the discretized form of 2D N-S equations can be written as

$$\frac{d\mathbf{W}}{dt} + \frac{1}{\Omega} \sum_{i=1}^N (\mathbf{F}_c - \mathbf{F}_v)_i S_i = 0 \quad (2.9)$$

where, Ω and N are the volume and number of interfaces of the control volume and S_i is the length of interface i . The conservative flow variables \mathbf{W} , convective flux \mathbf{F}_c and viscous flux \mathbf{F}_v are given by

$$\mathbf{W} = \begin{bmatrix} \rho \\ \rho U \\ \rho V \\ \rho E \end{bmatrix}, \quad \mathbf{F}_c = \begin{bmatrix} \rho U_n \\ \rho U U_n + n_x p \\ \rho V U_n + n_y p \\ (\rho E + p) U_n \end{bmatrix}, \quad \mathbf{F}_v = \begin{bmatrix} 0 \\ n_x \tau_{xx} + n_y \tau_{xy} \\ n_x \tau_{yx} + n_y \tau_{yy} \\ n_x \Theta_x + n_y \Theta_y \end{bmatrix} \quad (2.10)$$

In the above equation, ρ , U , V and p are the density, velocities and pressure of the mean flow. E is the total energy defined as

$$E = \frac{p}{(\gamma - 1)\rho} + \frac{1}{2}(U^2 + V^2) \quad (2.11)$$

where γ is the specific heat ratio (n_x, n_y) is the unit normal vector of the control surface and U_n is the normal velocity of the cell interface. In the viscous flux \mathbf{F}_v , τ_{ij} denotes the components of viscous stress tensor and Θ_i is energy flux contributed by the viscous stress and the heat transfer.

As shown in Eq. (2.9), with the conservative flow variables at the cell center, we need to evaluate numerical fluxes at cell interfaces. For inviscid flow problems, the viscous flux \mathbf{F}_v is abandoned and only the inviscid flux \mathbf{F}_c is to be evaluated. Therefore, the governing equations become the Euler equations

$$\frac{d\mathbf{W}}{dt} + \frac{1}{\Omega} \sum_{i=1}^N (\mathbf{F}_c)_i S_i = 0 \quad (2.12)$$

For viscous flow problems, the viscous flux F_v is usually approximated by a smooth function, such as polynomial approximation. Therefore, for both inviscid and viscous flows, the remaining problem is the evaluation of inviscid flux F_c at cell interface, which is computed by the gas kinetic scheme in this work. As shown in Figure 2.1, the relationship between velocities in the normal and tangential directions (U_n, U_τ) and velocities in the Cartesian coordinate system (U, V) are

$$\begin{aligned} U_n &= U \cdot n_x + V \cdot n_y \\ U_\tau &= V \cdot n_x - U \cdot n_y \end{aligned} \quad (2.13)$$

In the application of gas kinetic scheme, the flux F_c at cell interface is calculated by the local application of the Boltzmann equation, which can be expressed as

$$F_c = \int u_n \varphi_\alpha f d\Xi \quad (2.14)$$

where u_n is the phase velocity in the normal direction of the interface.

Similar to Eq. (2.13), the relationship between the phase velocities in the normal and tangential directions (u_n, u_τ) and the phase velocities in the Cartesian coordinate system (u, v) are

$$u_n = u \cdot n_x + v \cdot n_y, \quad u_\tau = v \cdot n_x - u \cdot n_y \quad (2.15)$$

and

$$u = u_n \cdot n_x - u_\tau \cdot n_y, \quad v = u_\tau \cdot n_x + u_n \cdot n_y \quad (2.16)$$

Substituting Eq. (2.16) into Eq. (2.3), we have

$$\varphi_\alpha = \left(1, u_n \cdot n_x - u_\tau \cdot n_y, u_\tau \cdot n_x + u_n \cdot n_y, \frac{1}{2} (u_n^2 + u_\tau^2 + \xi^2) \right)^T \quad (2.17)$$

With the definition of a new moment

$$\varphi_\alpha^* = \left(1, u_n, u_\tau, \frac{1}{2}(u_n^2 + u_\tau^2 + \xi^2) \right)^T \quad (2.18)$$

and its corresponding convective flux vector,

$$\mathbf{F}_c^* = \int u_n \varphi_\alpha^* f d\Xi = (F_1^*, F_2^*, F_3^*, F_4^*)^T \quad (2.19)$$

the real convective flux vector \mathbf{F}_c can be obtained by substituting Eq. (2.17) into Eq. (2.14) and using Eq. (2.19)

$$\mathbf{F}_c = \int u_n \varphi_\alpha f d\Xi = (F_1^*, F_2^* n_x - F_3^* n_y, F_3^* n_x + F_2^* n_y, F_4^*)^T \quad (2.20)$$

From Eq. (2.20), it is clear that the calculation of \mathbf{F}_c is equivalent to the evaluation of \mathbf{F}_c^* and the key issue is to obtain the gas distribution function f . In the next part, the evolution of the gas distribution function f by the SF-GKS will be introduced.

2.2.2 Evaluation of inviscid flux by switch function-based gas kinetic scheme

With the Maxwellian distribution function in Eq. (2.7), to recover Euler equations by Eq. (2.1) through C-E expansion analysis, the following 5 conservation forms of moments should be satisfied,

$$\int g d\Xi = \rho \quad (2.21)$$

$$\int g u_\alpha d\Xi = \rho U_\alpha \quad (2.22)$$

$$\int g \left(u_\alpha u_\alpha + \sum_{j=1}^K \xi_j^2 \right) d\Xi = \rho (U_\alpha U_\alpha + bRT) \quad (2.23)$$

$$\int g u_\alpha u_\beta d\Xi = \rho U_\alpha U_\beta + p \delta_{\alpha\beta} \quad (2.24)$$

$$\int g \left(u_\alpha u_\alpha + \sum_{j=1}^K \xi_j^2 \right) u_\beta d\Xi = \rho [U_\alpha U_\alpha + (b+2)RT] U_\beta \quad (2.25)$$

where u_α , u_β and U_α , U_β are the phase velocity and macroscopic flow velocity in α and β directions. $b = K + D$ represents the total degree of freedom of molecules, where D is the abbreviation of the dimension. The integral domains in all the above equations are from $-\infty$ to $+\infty$. In the above 5 equations, Eqs. (2.21)-(2.23) are applied to recover fluid density, momentum and energy, respectively. In addition, Eqs. (2.24) and (2.25) are to recover the convective flux of momentum equations and energy equation.

As the convective flux is evaluated at the local interface, a local coordinate system is used in the following derivation of the distribution function f . Based on the C-E expansion analysis (Ohwada and Xu, 2004), the non-equilibrium distribution function can be approximated as

$$f^{neq} = -\tau \left(\frac{\partial}{\partial t} + \mathbf{u} \cdot \nabla \right) f^{eq} \quad (2.26)$$

where the equilibrium distribution function is

$$f^{eq} = g \quad (2.27)$$

Therefore, the real distribution function in the local Boltzmann equation becomes

$$f = f^{eq} + f^{neq} = g - \tau \left(\frac{\partial g}{\partial t} + u_n \frac{\partial g}{\partial n} + u_\tau \frac{\partial g}{\partial \tau} \right) \quad (2.28)$$

By applying the Taylor series expansion in time and physical space, the above equation can be simplified to

$$f(0,0,t+\delta t) = g(0,0,t+\delta t) - \frac{\tau}{\delta t} \left[g(0,0,t+\delta t) - g(-u_n \delta t, -u_\tau \delta t, t) \right] + O(\delta t^2) \quad (2.29)$$

where $f(0,0,t+\delta t)$ is the gas distribution function at the interface, and $g(0,0,t+\delta t)$ and $g(-u_n \delta t, -u_\tau \delta t, t)$ are the equilibrium distribution functions at cell interface and

its surrounding points, respectively. δt is the streaming time step. Here, $\frac{\tau}{\delta t}$ can be viewed as the dimensionless collision time. Similar to Eq. (2.6), $g(-u_n \delta t, -u_\tau \delta t, t)$ can be taken as a piecewise constant function around the cell interface

$$g(-u_n \delta t, -u_\tau \delta t, t) = \begin{cases} g_l, & u_n \geq 0 \\ g_r, & u_n < 0 \end{cases} \quad (2.30)$$

According to the C-E analysis, the equilibrium part f^{eq} of the distribution function in Eq. (2.29) contributes to the inviscid flux. On the other hand, the non-equilibrium part f^{neq} contributes to the viscous flux. As the SF-GKS is merely used to evaluate the inviscid flux, only the equilibrium part is needed:

$$f(0,0,t + \delta t) = g(0,0,t + \delta t) \quad (2.31)$$

In Eq. (2.31), no numerical dissipation is introduced. Therefore, this scheme can capture the thin boundary layer accurately. However, in the region of discontinuity, such as around strong shock waves, this scheme will be unstable and sometimes diverges because of the lack of numerical dissipation.

Another approach is to include the non-equilibrium part f^{neq} in the distribution function to introduce the numerical dissipation. By setting $\frac{\tau}{\delta t} = 1$, Eq. (2.29) becomes

$$f(0,0,t + \delta t) = g(-u_n \delta t, -u_\tau \delta t, t) \quad (2.32)$$

In fact, the above method is the KFVS as Eq. (2.32) is equivalent to Eq. (2.5). It has been discussed in Section 2.1 that a large numerical dissipation is introduced in this scheme because the numerical dissipation is proportional to the mesh size. Therefore,

the results will easily get smeared in the smooth region, such as in thin boundary layers.

In the conventional gas kinetic scheme (Xu, 1998), the distribution function is

$$\begin{aligned} f(0,0,t+\delta t) &= g(0,0,t+\delta t) - \eta \left[g(0,0,t+\delta t) - g(-u_n\delta t, -u_\tau\delta t, t) \right] \\ &= (1-\eta) \cdot g(0,0,t+\delta t) + \eta \cdot g(-u_n\delta t, -u_\tau\delta t, t) \end{aligned} \quad (2.33)$$

where η is a constant with $\eta \in [0, 1]$. Obviously, Eq. (2.33) is exactly the same as Eq.

(2.29) if we denote $\eta = \frac{\tau}{\delta t}$. In Eq. (2.33), the constant η controls the numerical

dissipation. To be more specific, when $\eta = 0$, Eq. (2.33) is the same as Eq. (2.31), in

which there is no numerical dissipation. When $\eta = 1$, the distribution function

becomes Eq. (2.32), where a large numerical dissipation is introduced. In the practical

application, η is usually taken as a constant in the whole domain, such as $\eta = 0.5$.

However, the introduction of constant η in Eq. (2.33) means that the numerical

dissipation is added uniformly in the regions of strong shock waves and thin boundary

layers, which is not reasonable. The above problem motivates the current work by

introducing a switch function to control the numerical dissipation. That is, in the

region of thin boundary layers, the numerical dissipation should not be introduced and

η should be close to 0, while in the region of a discontinuity, large dissipation should

be introduced to ensure the stable solution with η as 1. In order to distinguish it from

the constant η , the switch function is denoted as η^+ with the definition of

$$\eta^+ = \tanh\left(C \frac{|p_L - p_R|}{p_L + p_R}\right) \quad (2.34)$$

where $\tanh(x)$ is the hyperbolic tangent function; C is the amplification factor and $C = 10$ is used in the present work; p_L and p_R are the pressure at the left and right sides of cell interface. Therefore, the distribution function becomes

$$f(0,0,t+\delta t) = (1-\eta^+) \cdot g(0,0,t+\delta t) + \eta^+ \cdot g(-u_n\delta t, -u_\tau\delta t, t) \quad (2.35)$$

From the characteristics of the hyperbolic tangent function, it can be seen that η^+ ranges from 0 to 1. In the smooth region, the switch function η^+ is close to 0 due to the small pressure difference at two sides of cell interface and Eq. (2.35) approaches Eq. (2.31). In the region of a discontinuity, there will be a large difference of pressure at the two sides of the interface and Eq. (2.35) approaches Eq. (2.32).

From the above distribution function in Eq. (2.35), the numerical fluxes for the mass, momentum and energy across the cell interface can be calculated by

$$\begin{aligned} F_c^* &= \int u_n \varphi_\alpha^* f d\Xi \\ &= (1-\eta^+) \cdot \int u_n \varphi_\alpha^* g(0,0,t+\delta t) d\Xi + \eta^+ \cdot \int u_n \varphi_\alpha^* g(-u_n\delta t, -u_\tau\delta t, t) d\Xi \end{aligned} \quad (2.36)$$

In the above equation, the convective flux F_c^* can be separated into two parts: the flux contributed by $g(0,0,t+\delta t)$ which is denoted as $F_{c,0}^*$ and the flux contributed by $g(-u_n\delta t, -u_\tau\delta t, t)$ which is denoted as $F_{c,1}^*$. As a result, the total inviscid flux across the cell interface can be written as

$$F_c^* = (1-\eta^+) \cdot F_{c,0}^* + \eta^+ \cdot F_{c,1}^* \quad (2.37)$$

In Eq. (2.12), the governing equations are solved by the finite volume method, and conservative flow variables are defined at cell centers. With the conservative flow variables at the cell center, the conservative flow variables at the left and right sides of the cell interface can be obtained by interpolation. Then, the velocities in the

Cartesian coordinate system should be transferred to the velocities in the normal and tangential directions of the interface by Eq. (2.13). In order to evaluate the flux $\mathbf{F}_{c,0}^*$ contributed by $g(0,0,t+\delta t)$, the conservative flow variables at the cell interface should be first computed. By applying the compatibility condition $\int (g-f)\varphi_\alpha^* d\Xi = 0$, the conservative flow variables at cell interface can be computed as

$$\mathbf{W}_0^* = \int \varphi_\alpha^* f(0,0,t+\delta t) d\Xi = \int \varphi_\alpha^* g(0,0,t+\delta t) d\Xi \quad (2.38)$$

where \mathbf{W}_0^* is the vector of conservative flow variables at cell interface. By substituting Eq. (2.35) into Eq. (2.38), we have

$$\begin{aligned} \mathbf{W}_0^* &= \int \varphi_\alpha^* f(0,0,t+\delta t) d\Xi \\ &= \int \varphi_\alpha^* g(0,0,t+\delta t) d\Xi \\ &\quad - \eta^+ \left[\int \varphi_\alpha^* g(0,0,t+\delta t) d\Xi - \int \varphi_\alpha^* g(-u_n \delta t, -u_\tau \delta t, t) d\Xi \right] \end{aligned} \quad (2.39)$$

Using Eq. (2.38), we can find the relation in Eq. (2.39) that

$$\int \varphi_\alpha^* g(0,0,t+\delta t) d\Xi = \int \varphi_\alpha^* g(-u_n \delta t, -u_\tau \delta t, t) d\Xi \quad (2.40)$$

By substituting Eq. (2.30) and Eq. (2.40) into Eq. (2.38), we have

$$\mathbf{W}_0^* = \int \varphi_\alpha^* g(-u_n \delta t, -u_\tau \delta t, t) d\Xi = \int \int_{u_n \geq 0} \varphi_\alpha^* g_l d\Xi + \int \int_{u_n < 0} \varphi_\alpha^* g_r d\Xi \quad (2.41)$$

Eq. (2.41) shows that the conservative flow variables on the interface can be constructed by the flow variables at its left and right sides. In the above equation, the integral domains for normal velocity u_n and tangential velocity u_τ are different with consideration of a discontinuity around the interface. In the tangential direction, the integral domain is always from $-\infty$ to $+\infty$. However, in the normal direction, the integral domain is from $-\infty$ to 0 on the right side and 0 to $+\infty$ on the left side. The explicit expressions of \mathbf{W}_0^* can be given as

$$\rho = \rho_l \cdot a_l + \rho_r \cdot a_r \quad (2.42)$$

$$\rho U_n = \rho_l \cdot b_l + \rho_r \cdot b_r \quad (2.43)$$

$$\rho U_\tau = \rho_l U_{\tau,l} \cdot a_l + \rho_r U_{\tau,r} \cdot a_r \quad (2.44)$$

$$\rho E = \frac{1}{2} \rho_l \left[c_l + (U_{\tau,l}^2 + (b-1)RT_l) \cdot a_l \right] + \frac{1}{2} \rho_r \left[c_r + (U_{\tau,r}^2 + (b-1)RT_r) \cdot a_r \right] \quad (2.45)$$

where

$$a_l = \frac{1}{2} \operatorname{erfc}(-\sqrt{\lambda_l} U_{n,l}), \quad a_r = \frac{1}{2} \operatorname{erfc}(\sqrt{\lambda_r} U_{n,r}) \quad (2.46)$$

$$b_l = U_{n,l} \cdot a_l + \frac{1}{2} \frac{e^{-\lambda_l U_{n,l}^2}}{\sqrt{\pi \lambda_l}}, \quad b_r = U_{n,r} \cdot a_r - \frac{1}{2} \frac{e^{-\lambda_r U_{n,r}^2}}{\sqrt{\pi \lambda_r}} \quad (2.47)$$

$$c_l = U_{n,l} \cdot b_l + \frac{1}{2\lambda_l} \cdot a_l, \quad c_r = U_{n,r} \cdot b_r + \frac{1}{2\lambda_r} \cdot a_r \quad (2.48)$$

Once W_0^* is obtained, the equilibrium distribution function $g(0,0,t+\delta t)$ can be constructed by Eq. (2.7) and correspondingly, the flux $F_{c,0}^*$ can be evaluated as

$$F_{c,0}^* = \int u_n \varphi_\alpha^* g(0,0,t+\delta t) d\Xi = \begin{bmatrix} \rho U_n \\ \rho U_n^2 + p \\ \rho U_n U_\tau \\ (\rho E + p) U_n \end{bmatrix} \quad (2.49)$$

After the evaluation of flux $F_{c,0}^*$, the flux $F_{c,1}^*$ contributed by $g(-u_n \delta t, -u_\tau \delta t, t)$ is calculated as

$$F_{c,1}^* = \int u_n \varphi_\alpha^* g(-u_n \delta t, -u_\tau \delta t, t) d\Xi = \int \int_{u_n \geq 0} u_n \varphi_\alpha^* g_l d\Xi + \int \int_{u_n < 0} u_n \varphi_\alpha^* g_r d\Xi \quad (2.50)$$

Similarly, the explicit formulations of $F_{c,1}^*$ can also be given as

$$F_{c,1}^*(1) = \rho_l \cdot b_l + \rho_r \cdot b_r \quad (2.51)$$

$$F_{c,1}^*(2) = \rho_l \cdot c_l + \rho_r \cdot c_r \quad (2.52)$$

$$F_{c,1}^*(3) = \rho_l U_{\tau,l} \cdot b_l + \rho_r U_{\tau,r} \cdot b_r \quad (2.53)$$

$$F_{c,1}^*(4) = \rho_l \left[d_l + (U_{\tau,l}^2 + (b-1)RT_l) \cdot b_l \right] + \rho_r \left[d_r + (U_{\tau,r}^2 + (b-1)RT_r) \cdot b_r \right] \quad (2.54)$$

where $a_l, a_r, b_l, b_r, c_l, c_r$ have been defined in Eqs. (2.46)-(2.48) and d_l, d_r are

$$d_l = U_{n,l} \cdot c_l + \frac{1}{\lambda_l} \cdot b_l, \quad d_r = U_{n,r} \cdot c_r + \frac{1}{\lambda_r} \cdot b_r \quad (2.55)$$

Note that in the expressions (2.51) - (2.54), $F_{c,1}^*(m)$ means the m th component of the vector $F_{c,1}^*$.

2.2.3 Evaluation of viscous flux by smooth function approximation

The convective flux F_c in Eq. (2.9) has been obtained by the above SF-GKS and the Euler equations can be solved correspondingly. Nevertheless, when solving viscous flow problems, the evaluation of the viscous flux is also necessary. As has been presented in Eq. (2.10), the viscous flux can be written as

$$F_v = \begin{bmatrix} 0 \\ n_x \tau_{xx} + n_y \tau_{xy} \\ n_x \tau_{yx} + n_y \tau_{yy} \\ n_x \left(u \tau_{xx} + v \tau_{xy} + \kappa \frac{\partial T}{\partial x} \right) + n_y \left(u \tau_{yx} + v \tau_{yy} + \kappa \frac{\partial T}{\partial y} \right) \end{bmatrix} \quad (2.56)$$

Here, τ_{ij} is the stress $\tau_{ij} = \mu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3} \frac{\partial u_k}{\partial x_k} \delta_{ij} \right)$ and μ is the dynamic viscosity.

$\kappa = \frac{\mu c_p}{Pr Re}$ is the thermal conductivity, where c_p is the heat capacity, Pr and Re are

the Prandtl and Reynolds numbers.

Owing to the elliptic nature of the viscous flux, the viscous flux F_v can be approximated by a smooth function. In FVM, all the flow variables are defined at the

cell center. At the cell interface, the values of the velocity components (u, v) , the dynamic viscosity μ and the thermal conductivity κ can be calculated by simple averaging technique, i.e.,

$$\phi_{i+1/2,j} = \frac{1}{2}(\phi_{i,j} + \phi_{i+1,j}) \quad (2.57)$$

where, ϕ stands for any of the above flow variables. Next, we need to compute the derivatives of the velocity and temperature in Eq. (2.56). In FVM, the derivatives $\nabla\phi$ at the cell interface $(i+1/2, j)$ can be approximated by the finite volume method with application of Gauss theorem,

$$\nabla\phi_{i+1/2,j} \approx \frac{1}{\Omega_{i+1/2,j}} \oint_{\partial\Omega_{i+1/2,j}} \phi \mathbf{n} ds \quad (2.58)$$

2.2.4 Computational sequence

Overall, the basic solution procedure of SF-GKS can be summarized below:

- (1) With the flow variables at cell centers, the first order derivatives of conservative flow variables are calculated and the initial reconstructions are conducted at two sides of cell interface.
- (2) Convert the velocity components in the Cartesian coordinate system (U, V) to the normal and tangential directions (n_x, n_y) of the cell interface via Eq. (2.13).
- (3) Calculate the conservative flow variables at cell interface \mathbf{W}_0^* by using Eqs. (2.42)-(2.45).
- (4) Use Eq. (2.49) to compute the flux $\mathbf{F}_{c,0}^*$.
- (5) Calculate the flux $\mathbf{F}_{c,1}^*$ via Eqs. (2.51)-(2.54).

- (6) Compute the value of switch function η^+ via Eq. (2.34) and then the inviscid flux across cell interface \mathbf{F}_c^* can be obtained by using Eq. (2.37).
- (7) Convert the inviscid flux \mathbf{F}_c^* to \mathbf{F}_c via Eq. (2.20).
- (8) When solving viscous flow problems, the viscous flux \mathbf{F}_v should also be computed. By approximating the flow variables and their first order derivatives via Eq. (2.57) and Eq. (2.58), the viscous flux \mathbf{F}_v can be computed by Eq. (2.56).
- (9) Once fluxes at all cell interfaces are obtained, solve ordinary differential equation (2.9) by using 4-stage Runge-Kutta scheme unless otherwise stated.

2.3 Numerical Results and Discussion

In this section, the developed SF-GKS is validated by applying it to simulate some test problems. First, the standard case of 1D Euler shock tube is simulated to test the ability of SF-GKS for 1D problems with shock waves. After that, the regular shock reflection is simulated to compare the performance of KFVS and SF-GKS in inviscid flow problems. Another inviscid flow case is the double Mach reflection. We will demonstrate that SF-GKS has the ability to capture the complex flow structures. Furthermore, SF-GKS is extended to solve viscous compressible flow problems. The turbulent flow around a RAE2822 airfoil is simulated. On one hand, we will use this example to test the ability of SF-GKS for problems with curved boundaries. On the other hand, through this example, we will demonstrate that SF-GKS has the capability to accurately simulate viscous flows. The last problem is hypersonic flow around one half of a cylinder. This problem is studied to examine the capability of SF-GKS in solving hypersonic flow problems.

2.3.1 1D Euler shock tube

The 1D Euler shock tube is a standard Riemann problem. In the tube, there are two different initial constant states in the left and right parts. Three waves emerge from the initial discontinuity, which are rarefaction wave, contact discontinuity and shock wave. Two different Riemann problems are selected to test the performance of the current SF-GKS. For both two cases, the mesh size is chosen as $\Delta x = 1/250$ and the time step size is taken as $\Delta t = 0.001$. The reference density and reference length are set as $\rho_0 = 1$ and $L_0 = 1$, respectively. The initial conditions of first test case are

$$\rho_L = 1.0, u_L = 0.0, p_L = 1.0$$

$$\rho_R = 0.125, u_R = 0.0, p_R = 0.1$$

This is a mild test and the solution consists of a left rarefaction, a contact discontinuity and a right shock wave (Toro, 2009). Figure 2.2 shows the solution profiles at time $t = 0.25$, in which the results of density, velocity, pressure and internal energy are presented and well compared with exact solutions.

The other test is

$$\rho_L = 5.99924, u_L = 19.5975, p_L = 460.894$$

$$\rho_R = 5.99242, u_R = -6.19633, p_R = 46.095$$

The solution from the above initial condition consists of a left facing shock, a right travelling contact discontinuity and a right travelling shock wave (Toro, 2009). This test case is made up of two emerging shocks from the right and left sides. Similar to the first test case, the results of density, velocity, pressure and internal energy at time $t = 0.035$ are shown in Figure 2.3. The exact solution is also included for comparison.

The above two tests demonstrate that the results of SF-GKS match well with the exact solutions and show stable performance for the 1D shock tube problem.

2.3.2 Regular shock reflection

To further validate the present scheme, the 2D regular shock reflection is tested. The computational domain is a rectangle of length 4 and height 1. A uniform mesh of 120×40 is used. The Dirichlet conditions are imposed on the left and top boundaries, which are

$$\rho_L = 1.0, u_L = 2.9, v_L = 0.0, p_L = 1/1.4$$

$$\rho_T = 1.69997, u_T = 2.61934, v_T = -0.50633, p_T = 1.52819$$

In this case, reference density and reference length are set as $\rho_0 = 1$ and $L_0 = 1$, respectively. The right boundary condition is supersonic outflow where extrapolation is applied. A reflecting condition is applied along the bottom boundary. Figure 2.4 shows the pressure contour in which the shock wave is well captured. Figure 2.5 presents the density profile at the position of $y = 0.5$. The exact solution (Mittal and Tezduyar, 1998) and the result obtained by KFVS are also included for comparison and good agreement can be found in these results. Furthermore, it can be seen that the present result is slightly better than that of KFVS because of less numerical dissipation introduced.

2.3.3 Double Mach reflection

The test case of double Mach reflection is a hypersonic flow problem with a pressure ratio of 116.5. Initially, a strong normal shock wave with Mach number 10 passes through a 30° wedge. The computational domain is chosen to be a rectangle of 4×1

with a uniform mesh of 480×120 . The detailed information about the boundary conditions of this test can be found in Woodward and Colella (1984). Figure 2.6 and Figure 2.7 show the density and pressure contours at $t = 0.2$ using the present scheme. The complex flow structures such as the main shock, the oblique wave and the three-shock intersection are all well captured. As seen from the figures, the present results are in good agreement with those of Woodward and Colella (1984).

2.3.4 Compressible turbulent flow around a RAE2822 airfoil

The above three test cases are all inviscid flow problems and only the convective flux is evaluated. This test case involves the transonic flow around a RAE2822 airfoil and the viscous flux should also be calculated via the smooth function approximation. For this case, the free-stream conditions are set according to the experimental Case 9 in Cook et al. (1979), with Mach number $M_\infty = 0.73$, Reynolds number $Re = 6.5 \times 10^6$ and angle of attack $\alpha = 2.79^\circ$. A 369×65 C-type grid is adopted in this test case. To take turbulent effects (Lee and Wang, 1995; Lee et al., 2000; Lee and Wu, 2008) into consideration, the Spalart-Allmaras turbulence model is applied. Figure 2.8 shows the Mach number contour obtained using the present scheme and the switch function contour around the airfoil. It can be seen from the figure that the switch function supplies sufficient dissipation around the strong shock waves and approaches 0 in the smooth region, which matches well with the theory. The pressure coefficient and skin friction distributions along the airfoil surface obtained using the present scheme is compared with the experimental data (Cook et al., 1979) and numerical results (Kim et al., 2002, Turkel et al., 1997) in Figure 2.9. These numerical results are close to each other and both of them match well with the experimental data (Cook et al., 1979).

2.3.5 Hypersonic flow around one half of a cylinder

To investigate the capability of the present scheme for the simulation of hypersonic flows with strong shock waves, the hypersonic flow around one half of a cylinder is simulated. The inflow condition for this case is (Wieting and Holden, 1988)

$$M_\infty = 8.03, \quad Re = 1.835 \times 10^5, \quad T_\infty = 124.94K, \quad T_w = 294.44K$$

In the current simulation, the computational mesh is 160×160 , in which the cell Reynolds number is $Re_{cell} = \rho_\infty U_\infty \Delta r / \mu_\infty = 1.835$, where Δr is the mesh spacing of the first cell in the normal direction adjacent to the cylinder surface. To accelerate the convergence, the LU-SGS scheme (Yoon and Jameson, 1988) is applied to solve the governing equations.

Firstly, when the switch function is chosen following the approach in Eq. (2.34), the heat flux exhibits oscillations in the vicinity of the stagnation point. This is probably because the numerical dissipation introduced by the switch function in Eq. (2.34) is not sufficient in hypersonic flows. It can be verified with Figure 2.10 that the switch function values did not approach 1 in the region around strong shock waves. In fact, the maximum value of the switch function is around 0.84. To overcome this deficiency, a minor revision is made to Eq. (2.34) for this hypersonic case. The new values of the switch function are defined as

$$\eta^{++} = \max\{\eta_L^+, \eta_R^+\} \quad (2.59)$$

where η_L^+ and η_R^+ are the maximum values of the switch function in the left and right control cells, respectively. Accordingly, the inviscid flux in Eq. (2.34) should be changed to

$$\mathbf{F}_c^* = (1 - \eta^{++}) \cdot \mathbf{F}_{c,0}^* + \eta^{++} \cdot \mathbf{F}_{c,1}^* \quad (2.60)$$

The contour of modified switch function is also displayed in Figure 2.10. It is clear that the modified switch function values are larger than the original ones in the vicinity of strong shock waves.

The current results of pressure and temperature contours around the cylinder are presented in Figure 2.11. It is clear that the “carbuncle phenomenon” and post-shock oscillations do not appear in the present results. Figure 2.12 presents the normalized pressure and heat flux distribution along half of the cylinder surface, where the pressure p_0 has the value 0.9209 and heat flux q_0 is equal to 0.003655 at the stagnation point. The experimental results (Wieting and Holden, 1988) and numerical results of Xu et al. (2005) are also displayed in the figure for comparison. Good agreements can be found in Figure 2.12, which verifies the capability of the current SF-GKS in solving hypersonic flow problems.

2.4 Concluding Remarks

In this chapter, the basic theory of KFVS was first introduced. Based on the collisionless Boltzmann equation, KFVS lacks particle collisions in the gas evolution stage. The artificial particle collisions are added in the calculation of initial Maxwellian distribution at the beginning of the next time step. As a result, a large numerical dissipation is introduced in KFVS. On one hand, KFVS has the good positivity property for simulation of flows with strong shock waves with the numerical dissipation. On the other hand, KFVS usually gives more diffusive results than the Godunov or FDS scheme because the numerical dissipation in KFVS is proportional to the mesh size. Therefore, KFVS is not able to yield accurate N-S

solutions except for cases in which the physical viscosity is much larger than the numerical viscosity.

In order to overcome the drawback of KFVS and present an effective numerical algorithm, the SF-GKS was proposed based on the C-E analysis. SF-GKS is a finite volume solver for evaluating the convective flux across the cell interface. By introducing a simple switch function, the numerical dissipation could be well controlled. Thus, SF-GKS could capture strong shock waves and thin boundary layers simultaneously. For viscous flow problems, the viscous flux was evaluated by the smooth function approximation. Compared with KFVS, SF-GKS not only inherits the advantages of KFVS, but also provides accurate N-S solutions. Furthermore, the value of the switch function can be easily determined, which makes this method efficient and easy to implement. The proposed SF-GKS was validated by simulating 1D Euler shock tube, regular shock reflection, double Mach reflection, compressible turbulent flow around a RAE2822 airfoil and hypersonic flow around one half of a cylinder. Numerical results showed that SF-GKS could be applied to both compressible inviscid and viscous flow problems. The results were stable and accurate, even in the simulation of hypersonic flows. Numerical results also showed that SF-GKS performed slightly better than KFVS for inviscid flow problems.

However, it should be noted that the different treatments of inviscid and viscous parts in the current SF-GKS might introduce numerical errors in the computation with strong coupling of the inviscid and viscous flow interaction. In the next chapter, a gas kinetic flux solver (GKFS) will be proposed, in which both inviscid and viscous parts are recovered in a single gas distribution function f . In GKFS, real particle collisions

will be introduced in the gas evolution stage. As a consequence, the dissipation in the transport can be controlled by a real collision time. A simple way to evaluate the non-equilibrium distribution function will be presented.

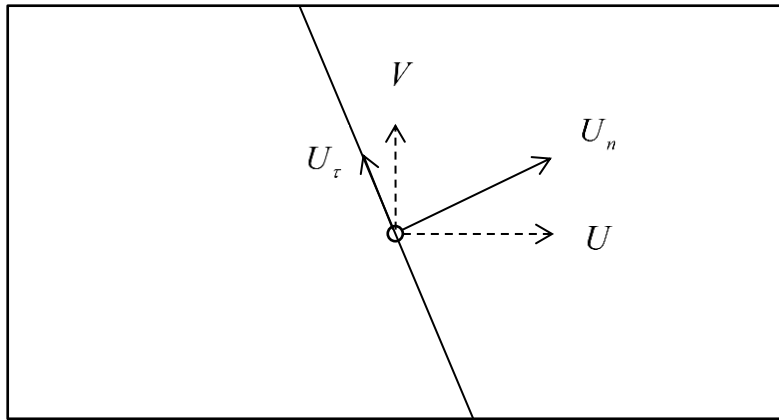


Figure 2.1 Schematic of velocity transformation at cell interface

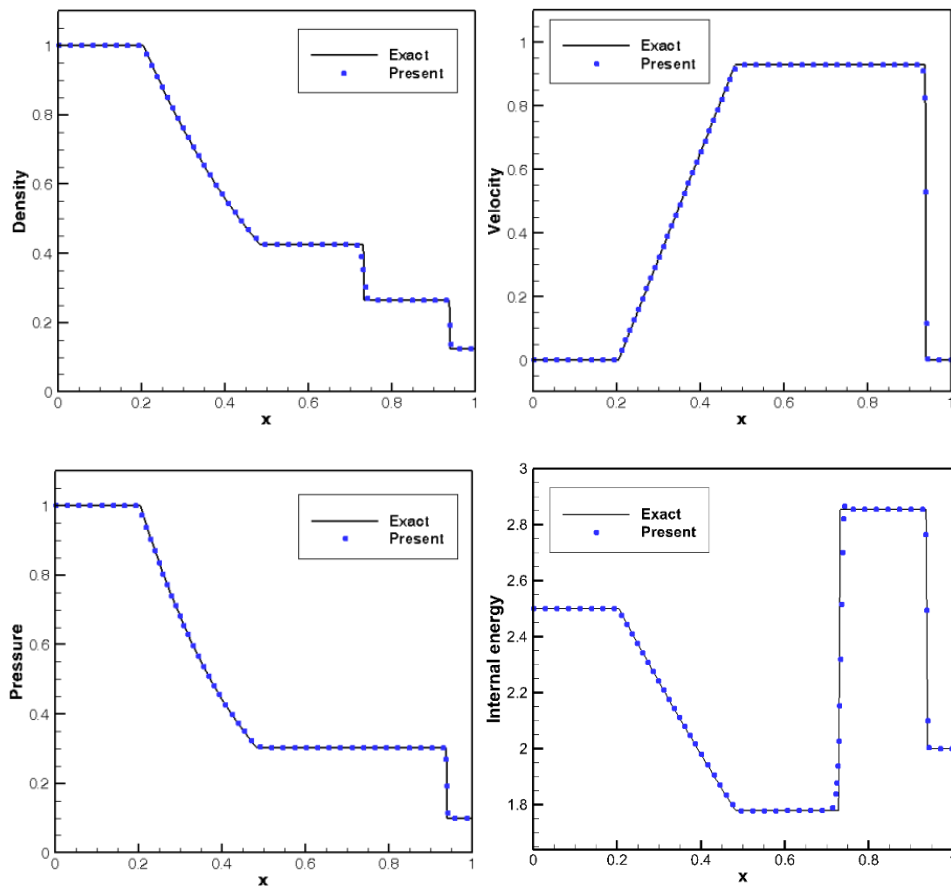


Figure 2.2 Comparison of density, velocity, pressure and internal energy with the exact solution at time $t = 0.25$ of Test Case 1

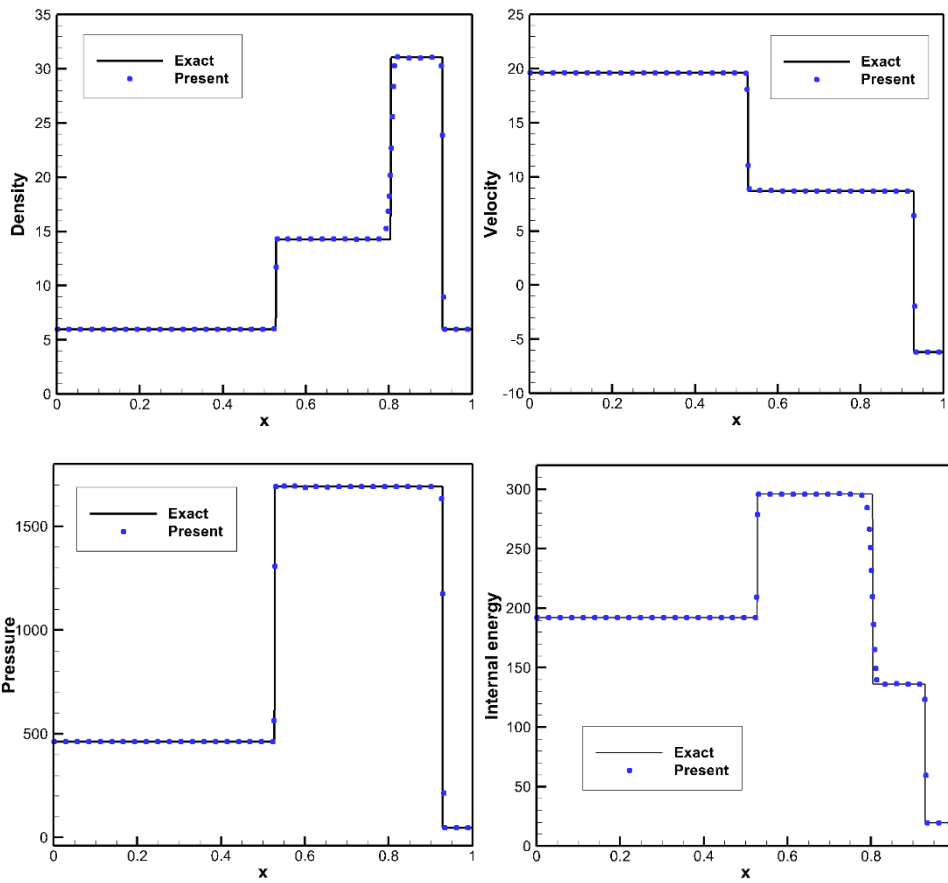


Figure 2.3 Comparison of density, velocity, pressure and internal energy with the exact solution at time $t = 0.035$ of Test Case 2

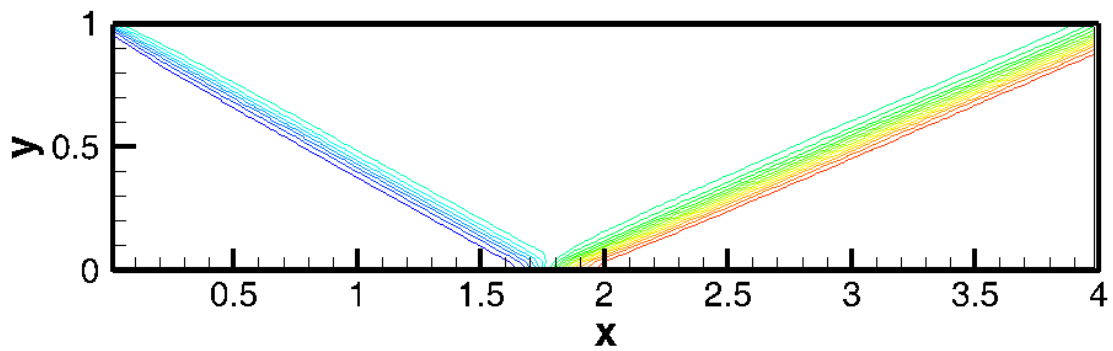


Figure 2.4 Pressure contour of the shock reflection

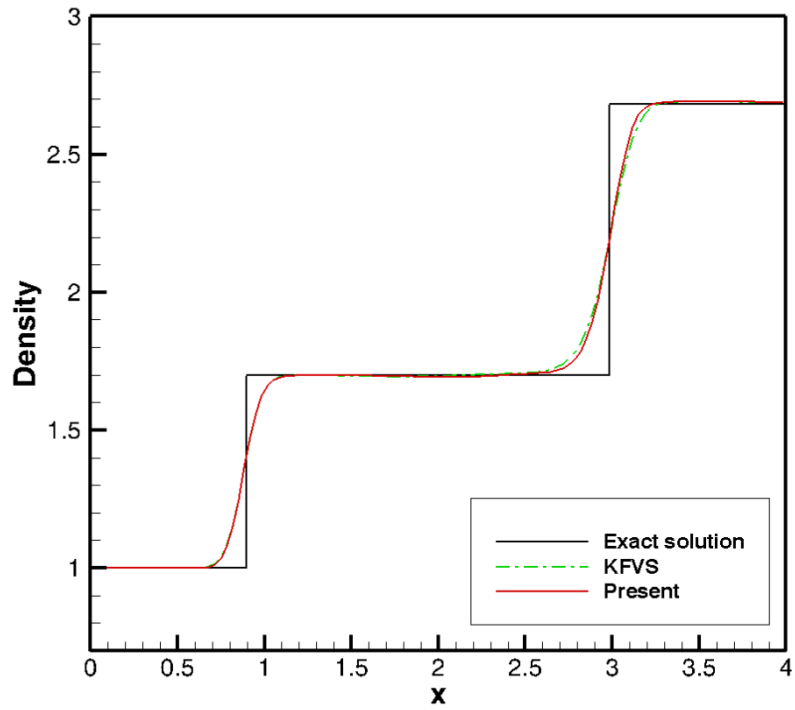


Figure 2.5 Comparison of density profile for the shock reflection along the line $y = 0.5$

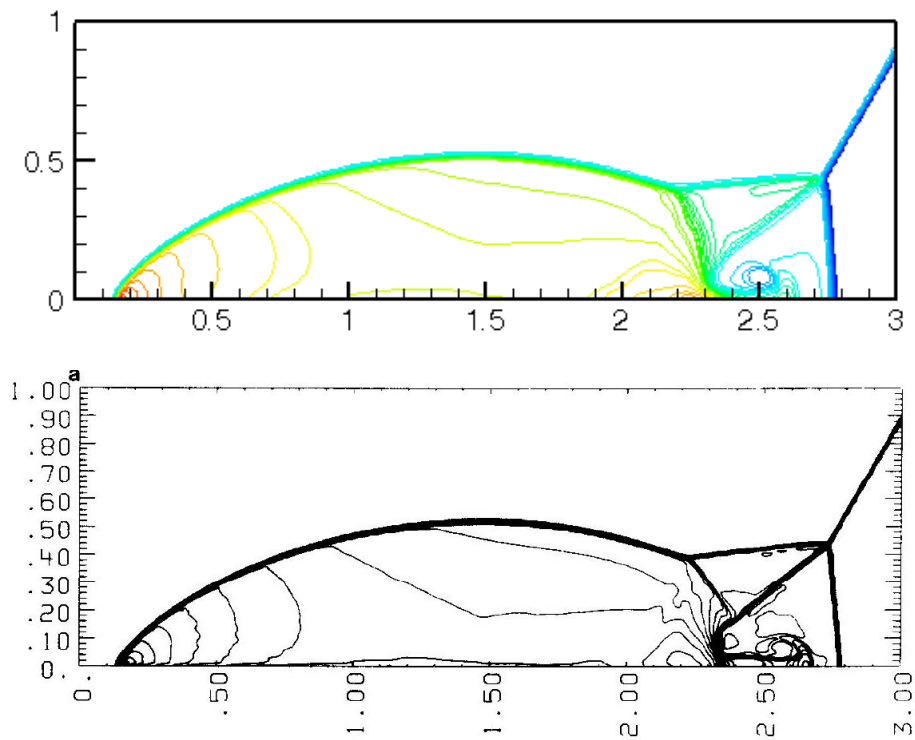


Figure 2.6 Density contours of the double Mach reflection obtained from present scheme (Upper) and Woodward and Colella (1984) (Lower)

(upper: Present; lower: Woodward and Colella (1984))

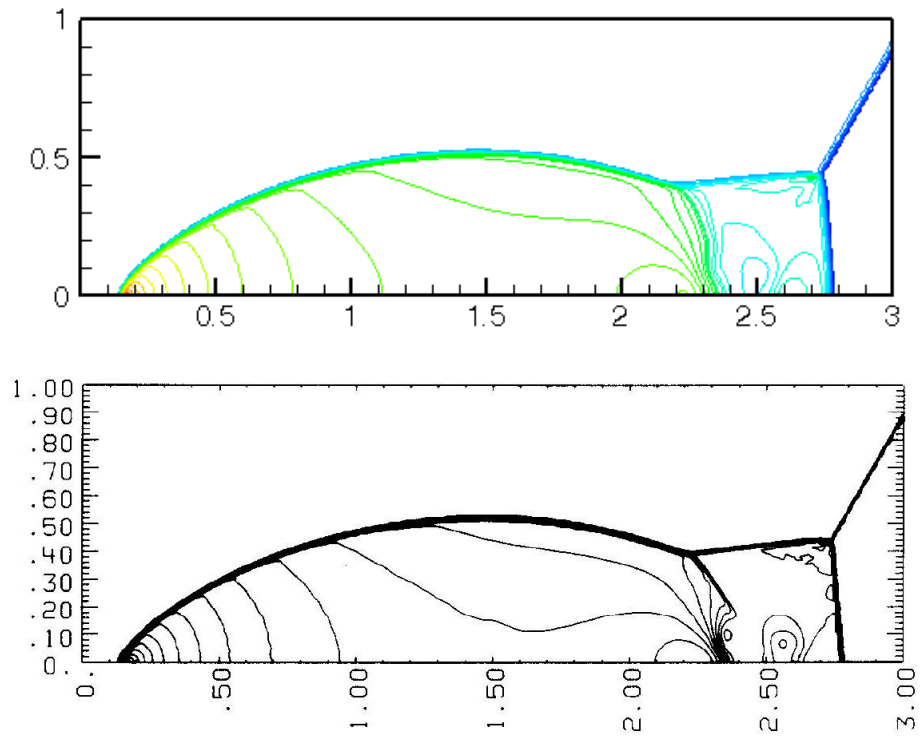


Figure 2.7 Pressure contours of the double Mach reflection obtained from present scheme (Upper) and Woodward and Colella (1984) (Lower)

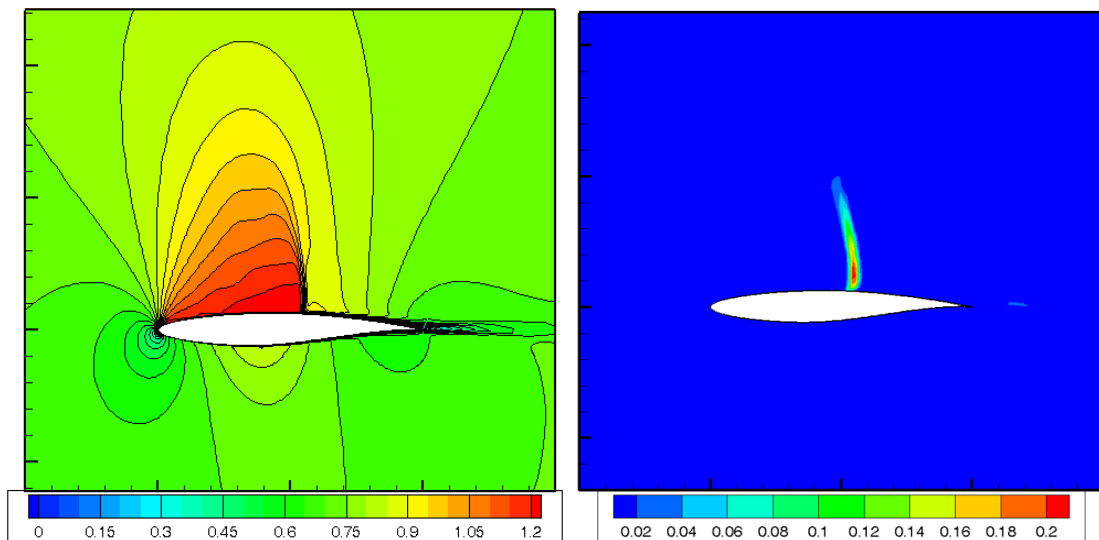


Figure 2.8 Contours of Mach number (Left) and switch function (Right) around the RAE2822 airfoil

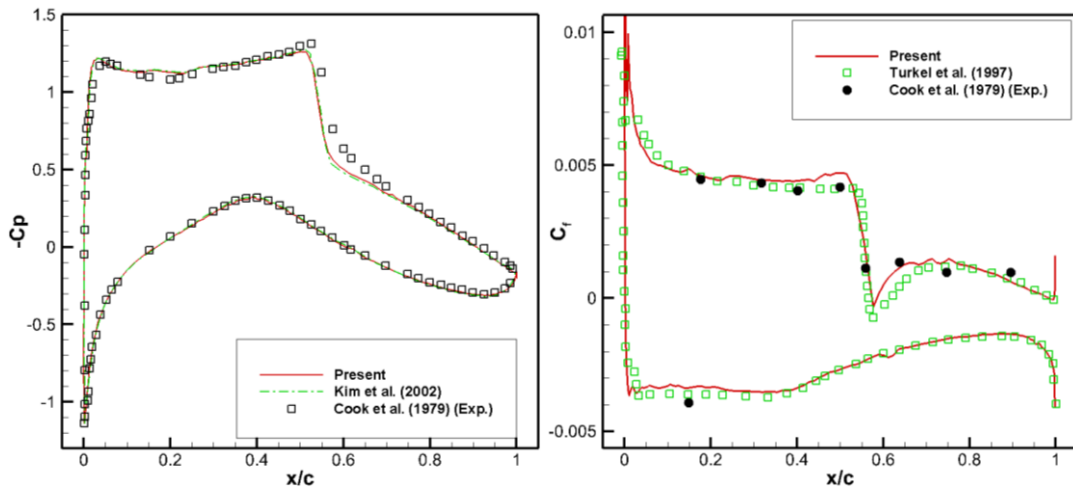


Figure 2.9 Comparisons of pressure coefficient (Left) and skin friction (Right) distributions on the RAE2822 airfoil surface

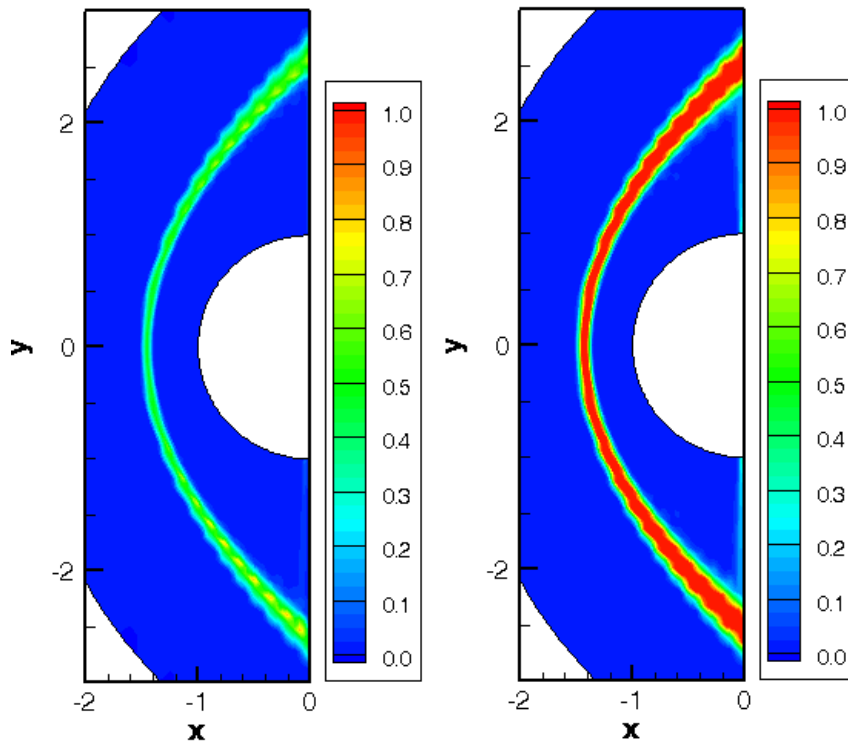


Figure 2.10 Contours of original switch function (Left) and modified switch function (Right) for hypersonic flow around one half of a circular cylinder

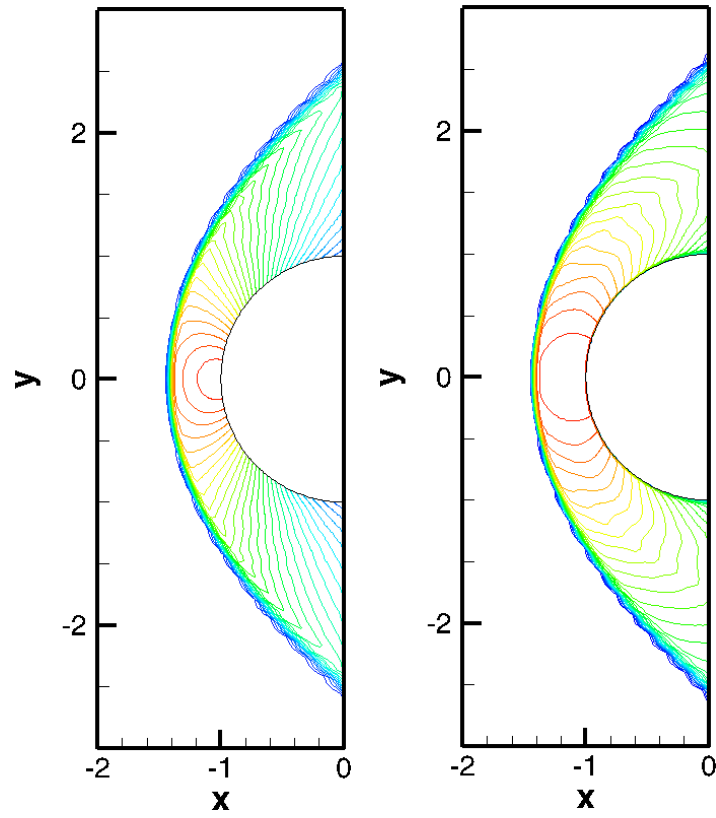
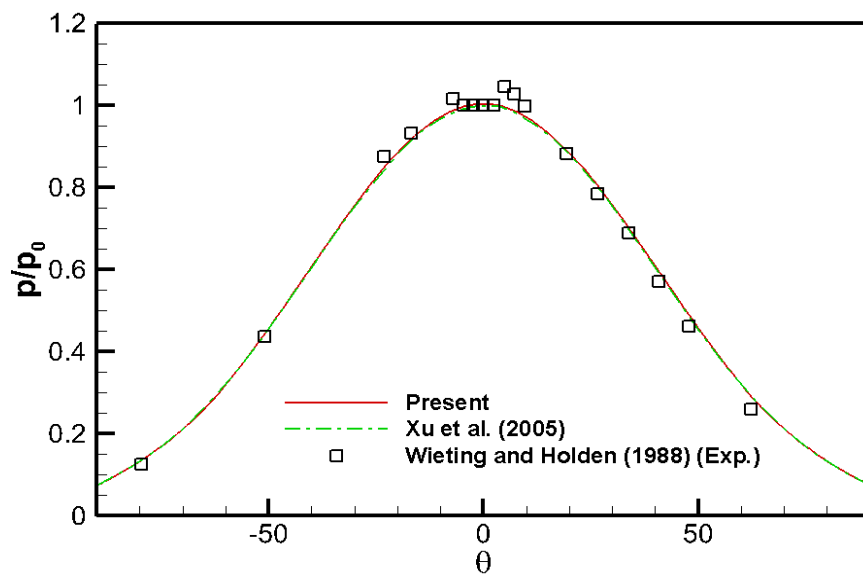


Figure 2.11 Pressure (Left) and temperature (Right) contours of hypersonic flow



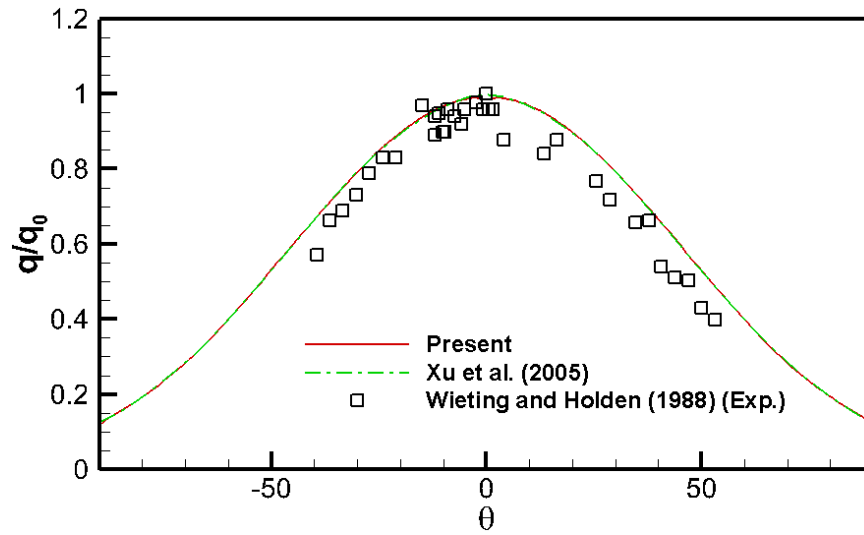


Figure 2.12 Pressure (Upper) and heat flux (Lower) distribution along the cylindrical surface

Chapter 3

Development of Gas Kinetic Flux Solver for Two-dimensional Incompressible and Compressible Viscous Flows

The switch function-based gas kinetic scheme (SF-GKS) has been developed in the previous chapter and successfully applied to simulate inviscid and viscous compressible flows. In this chapter, a brief introduction of the gas kinetic BGK scheme for N-S equations is given first. In this method, the BGK collision model is adopted in the gas evolution stage to obtain the numerical fluxes at the cell interface. As a consequence, the dissipation in the transport can be controlled by a real collision time. According to the C-E expansion analysis of the BGK model, the gas kinetic BGK scheme provides accurate N-S solutions in the continuum regime. At the same time, a stable and crisp shock transition can be generated with a delicate dissipative mechanism. However, the gas kinetic BGK scheme suffers from some drawbacks such as complexity and low computational efficiency compared with conventional CFD schemes. In order to simplify the original gas kinetic BGK scheme while keeping its intrinsic advantages, a gas kinetic flux solver (GKFS) is proposed in this chapter. In this solver, the finite volume method is applied to discretize the governing equations. The fluxes at the cell interface are evaluated by locally reconstructing the solutions for the continuous Boltzmann equation. Different from the conventional gas kinetic BGK scheme, a simple way is presented to evaluate the non-equilibrium distribution function, which is calculated by the difference of equilibrium distribution functions at the cell interface and its surrounding points. In particular, three specific

schemes are proposed. The accuracy and efficiency of the proposed GKFS are examined through numerical simulations of several incompressible and compressible flows, such as decaying vortex flows, 2D lid-driven cavity flows, viscous flows past a circular cylinder, Couette flow with a temperature gradient, Shock wave-boundary layer interaction, transonic laminar flows over a NACA0012 airfoil and hypersonic flow around a circular cylinder.

3.1 Relationships between Particle Distribution Function in Boltzmann Equations and Conservative Flow Variables and Fluxes in Navier-Stokes Equations

3.1.1 Boltzmann equations and 7 conservation forms of moments.

In the gas kinetic scheme, the BGK model (Bhatnagar et al., 1954) is widely used as approximation of the complicated collision term in the Boltzmann equation. Without the external forcing term, the Boltzmann equation with BGK model (or BGK equation) in two dimensions can be written as

$$\frac{\partial f}{\partial t} + u \frac{\partial f}{\partial x} + v \frac{\partial f}{\partial y} = -\frac{f - g}{\tau} \quad (3.1)$$

where f is the real particle distribution function and g is the equilibrium particle distribution function. τ is the collision time, which is determined by dynamic viscosity and pressure. The right side of the equation is the collision term which alters the distribution function from f to g within a collision time scale τ . Both f and g are functions of space (x, y) , time (t) , particle velocity (u, v) and internal energy ξ . The internal degree of freedom K in ξ is determined by the space dimension and the

ratio of specific heats with the relation $K + D = 2/(\gamma - 1)$ (Chae et al., 2000), where D is the abbreviation of the dimension. The equilibrium state g of Maxwellian distribution is

$$g = \rho \left(\frac{\lambda}{\pi} \right)^{\frac{K+2}{2}} e^{-\lambda((u-U)^2 + (v-V)^2 + \xi^2)} \quad (3.2)$$

Here, ρ is the density, $\mathbf{U} = (U, V)$ are the macroscopic velocity in the x and y directions. $\lambda = \frac{m}{2kT} = \frac{1}{2RT}$, where m is the molecular mass, k is the Boltzmann constant, R is the gas constant and T is the temperature. In the equilibrium state, the internal energy is $\xi^2 = \xi_1^2 + \xi_2^2 + \dots + \xi_K^2$.

With the above Maxwellian distribution function, to recover N-S equations by Eq. (3.1) through C-E expansion analysis, the following 7 conservation forms of moments should be satisfied,

$$\int g d\Xi = \rho \quad (3.3)$$

$$\int g u_\alpha d\Xi = \rho U_\alpha \quad (3.4)$$

$$\int g \left(u_\alpha u_\alpha + \sum_{j=1}^K \xi_j^2 \right) d\Xi = \rho (U_\alpha U_\alpha + bRT) \quad (3.5)$$

$$\int g u_\alpha u_\beta d\Xi = \rho U_\alpha U_\beta + p \delta_{\alpha\beta} \quad (3.6)$$

$$\int g \left(u_\alpha u_\alpha + \sum_{j=1}^K \xi_j^2 \right) u_\beta d\Xi = \rho [U_\alpha U_\alpha + (b+2)RT] U_\beta \quad (3.7)$$

$$\int g u_\alpha u_\beta u_\chi d\Xi = \rho (U_\alpha \delta_{\beta\chi} + U_\beta \delta_{\alpha\chi} + U_\chi \delta_{\alpha\beta}) + \rho U_\alpha U_\beta U_\chi \quad (3.8)$$

$$\int g \left(u_\alpha u_\alpha + \sum_{j=1}^K \xi_j^2 \right) u_\beta u_\chi d\Xi \quad (3.9)$$

$$= \rho \left\{ U_\alpha U_\alpha U_\beta U_\chi + [(b+4)U_\beta U_\chi + U_\alpha U_\alpha \delta_{\beta\chi}] RT + (b+2)R^2 T^2 \delta_{\beta\chi} \right\}$$

where u_α , u_β , u_χ and U_α , U_β , U_χ are the phase velocity and macroscopic flow velocity in the α , β and γ direction. p is the pressure and $b = K + D$ represents the total degree of freedoms of molecules. $d\Xi = du_\alpha du_\beta du_\chi d\xi_1 d\xi_2 \cdots d\xi_K$ is the volume element in the phase space. The integral domains in all the above equations are from $-\infty$ to $+\infty$. Eqs. (3.3)-(3.5) are applied to recover the fluid density, momentum and energy, respectively. Eqs. (3.6) and (3.7) are used to recover convective fluxes of momentum equations and energy equation. Eqs. (3.8) and (3.9) are used to recover diffusive fluxes of momentum equations and energy equation.

3.1.2 Relationships of conservative flow variables and fluxes in Navier-Stokes equations with distribution function in Boltzmann equation

For the gas kinetic scheme, the inviscid and viscous terms are computed simultaneously (Xu, 2001). With the finite volume discretization, the discretized form of 2D N-S equations are written as

$$\frac{d\mathbf{W}}{dt} + \frac{1}{\Omega} \sum_{i=1}^N \mathbf{F}_i S_i = 0 \quad (3.10)$$

where \mathbf{W} is the vector of conservative flow variables, Ω and N are the volume and number of interfaces of the control volume respectively, \mathbf{F}_i and S_i are the flux vector and length of interface i . In the gas kinetic scheme, the connection between distribution function f and conservative flow variables is

$$\mathbf{W} = (\rho, \rho U, \rho V, \rho E)^T = \int f \varphi_\alpha d\Xi \quad (3.11)$$

where $E = \frac{1}{2}(U^2 + V^2 + bRT)$ and φ_α is the moment given by

$$\varphi_\alpha = \left(1, u, v, \frac{1}{2}(u^2 + v^2 + \xi^2) \right) \quad (3.12)$$

The volume element is $d\Xi = dudvd\xi$ for two dimensional problems. With the compatibility condition

$$\int \frac{g-f}{\tau} \varphi_\alpha d\Xi = 0 \quad (3.13)$$

Eq. (3.11) is equivalent to

$$\mathbf{W} = (\rho, \rho U, \rho V, \rho E)^T = \int g \varphi_\alpha d\Xi \quad (3.14)$$

The above equation shows that the conservative flow variables at the interface can be obtained directly from the equilibrium distribution function.

As has been discussed above, the non-equilibrium distribution function has no contribution to the conservative flow variables, but it affects the fluxes across the interface. The flux vector in the x direction is

$$\mathbf{F}^x = (F_1^x, F_2^x, F_3^x, F_4^x)^T = \int f u \varphi_\alpha d\Xi \quad (3.15)$$

and similarly, the flux vector in the y direction is

$$\mathbf{F}^y = (F_1^y, F_2^y, F_3^y, F_4^y)^T = \int f v \varphi_\alpha d\Xi \quad (3.16)$$

3.2 Gas kinetic BGK scheme

The general solution f of the BGK model in Eq. (3.1) at a cell interface $(x_{i+1/2}, y_j)$ and time t is

$$f(x_{i+1/2}, y_j, t, u, v, \xi) = \frac{1}{\tau} \int_0^t g(x', y', t', u, v, \xi) e^{-(t-t')/\tau} dt' + e^{-t/\tau} f_0(x_{i+1/2} - ut, y_j - vt) \quad (3.17)$$

where $x' = x_{i+1/2} - u(t-t')$, $y' = y_j - v(t-t')$ are the trajectory of a particle motion and f_0 is the initial gas distribution function f at the beginning of each time step ($t=0$). In order to obtain the solution f , the distribution functions g and f_0 have to be specified. To simplify the notation, $(x_{i+1/2} = 0, y_j = 0)$ will be used in the following text.

In the gas kinetic scheme, with the initial discontinuous macroscopic variables at the left and right sides of a cell interface, the initial gas distribution function f_0 has the form

$$f_0 = \begin{cases} g^l (1 + a^l x + b^l y - \tau(a^l u + b^l v + A^l)), & x < 0, \\ g^r (1 + a^r x + b^r y - \tau(a^r u + b^r v + A^r)), & x \geq 0, \end{cases} \quad (3.18)$$

and the equilibrium state g around cell interface is constructed as

$$g = g_0 (1 + (1 - H(x)) \bar{a}^l x + H(x) \bar{a}^r x + \bar{b} y + \bar{A} t) \quad (3.19)$$

where $H(x)$ is the Heaviside function defined as

$$H(x) = \begin{cases} 0, & x < 0 \\ 1, & x \geq 0 \end{cases} \quad (3.20)$$

Here, g_l , g_r , g_0 are local Maxwellian distribution functions located to the left, to the right, and at the cell interface, respectively. In both f_0 and g , a^l , b^l , A^l , a^r , b^r , A^r , \bar{a}^l , \bar{a}^r , \bar{b} and \bar{A} are related to the derivatives of a Maxwellian distribution function in space and time. The dependence of a^l, b^l, \dots, \bar{A} on the particle velocities can be obtained by a Taylor series expansion of a Maxwellian distribution function,

$$\begin{aligned}
a^l &= a_1^l + a_2^l u + a_3^l v + a_4^l \frac{1}{2}(u^2 + v^2 + \xi^2) = a_\alpha^l \varphi_\alpha \\
b^l &= b_1^l + b_2^l u + b_3^l v + b_4^l \frac{1}{2}(u^2 + v^2 + \xi^2) = b_\alpha^l \varphi_\alpha
\end{aligned} \tag{3.21}$$

$$\bar{A} = \bar{A}_1 + \bar{A}_2 u + \bar{A}_3 v + \bar{A}_4 \frac{1}{2}(u^2 + v^2 + \xi^2) = \bar{A}_\alpha \varphi_\alpha$$

where $a_1^l, a_2^l, \dots, \bar{A}_4$ are local constants to be determined. Substituting Eq. (3.18) and Eq. (3.19) into Eq. (3.17), the gas distribution function f at a cell interface can be expressed as

$$\begin{aligned}
&f(x_{i+1/2}, y_j, t, u, v, \xi) \\
&= (1 - e^{-t/\tau}) g_0 + (\tau(-1 + e^{-t/\tau}) + t e^{-t/\tau}) (\bar{a}^l H(u) + \bar{a}^r (1 - H(u)) + \bar{b}v) u g_0 \\
&\quad + \tau(t/\tau - 1 + e^{-t/\tau}) \bar{A} g_0 + e^{-t/\tau} \left[(1 - (t + \tau)(ua^l + vb^l)) H(u) g^l \right. \\
&\quad \left. + (1 - (t + \tau)(ua^r + vb^r)) (1 - H(u)) g^r + (-\tau A^l H(u) g^l - \tau A^r (1 - H(u)) g^r) \right]
\end{aligned} \tag{3.22}$$

The above equation (3.22) is the gas distribution function in the gas kinetic BGK scheme. It is shown that a number of coefficients related to the physical space and phase space, such as $a_\alpha^l, b_\alpha^l, \dots, \bar{A}_\alpha$, have to be determined first.

To calculate these constants, the conservative flow variables at cell centers and their first-order derivatives could be used. With the initial reconstruction, the macroscopic status at the left and right side of the interface $\mathbf{W}_i(x_{i+1/2}, y_j)$ and $\mathbf{W}_{i+1}(x_{i+1/2}, y_j)$ can be obtained by interpolation. By using the relation between the gas distribution function and the macroscopic variables, we get

$$\int \varphi_\alpha g^l d\Xi = \mathbf{W}_i(x_{i+1/2}, y_j); \quad \int \varphi_\alpha g^l a^l d\Xi = \mathbf{n} \cdot \nabla \mathbf{W}^l \tag{3.23}$$

$$\int \varphi_\alpha g^r d\Xi = \mathbf{W}_{i+1}(x_{i+1/2}, y_j); \quad \int \varphi_\alpha g^r a^r d\Xi = \mathbf{n} \cdot \nabla \mathbf{W}^r \tag{3.24}$$

where $\nabla \mathbf{W}^l$ and $\nabla \mathbf{W}^r$ are the gradients of macroscopic variables on the left and right sides of a cell interface, and \mathbf{n} is the unit normal direction. From Eqs. (3.23) and (3.24), the parameters g^l , g^r , a^l and a^r can be uniquely determined. Similarly, in the tangential direction b^l and b^r can be obtained from

$$\int \varphi_\alpha g^l b^l d\Xi = \mathbf{t} \cdot \nabla \mathbf{W}^l; \quad \int \varphi_\alpha g^r b^r d\Xi = \mathbf{t} \cdot \nabla \mathbf{W}^r \quad (3.25)$$

where \mathbf{t} is the unit vector in the tangential direction along the cell interface. After determining the terms a^l, b^l, a^r and b^r , A^l and A^r in f_0 can be calculated by

$$\begin{aligned} M_{\alpha\beta}^l A_\beta^l &= -\int (a^l u + b^l v) \varphi_\alpha g^l d\Xi; \\ M_{\alpha\beta}^r A_\beta^r &= -\int (a^r u + b^r v) \varphi_\alpha g^r d\Xi \end{aligned} \quad (3.26)$$

where $M_{\alpha\beta}^l = \int g^l \varphi_\alpha \varphi_\beta d\Xi$ and $M_{\alpha\beta}^r = \int g^r \varphi_\alpha \varphi_\beta d\Xi$.

For the equilibrium state g in Eq. (3.19), the conservative flow variables at the cell interface \mathbf{W}_0 should be determined first. Taking the limit $t \rightarrow 0$ in Eq. (3.17) and substituting its solution into Eq. (3.13), the conservation constraint at the cell interface gives

$$\mathbf{W}_0 = \int g_0 \varphi_\alpha d\Xi = \int_{u \geq 0} \int g^l \varphi_\alpha d\Xi + \int_{u < 0} \int g^r \varphi_\alpha d\Xi \quad (3.27)$$

Since g^l and g^r have been obtained earlier, the above moments can be evaluated explicitly. Therefore, g_0 can be uniquely determined. Then, \bar{a}^l and \bar{a}^r of g in Eq. (3.19) can be obtained through the relation of

$$\frac{\mathbf{W}_0 - \mathbf{W}_i(x_i, y_j)}{\Delta x^-} = \bar{M}_{\alpha\beta}^0 \bar{a}_\beta^l; \quad \frac{\mathbf{W}_{i+1}(x_{i+1}, y_j) - \mathbf{W}_0}{\Delta x^+} = \bar{M}_{\alpha\beta}^0 \bar{a}_\beta^r \quad (3.28)$$

where Δx^- and Δx^+ are the distances from the cell interface to cell centers. Since the matrix $\bar{M}_{\alpha\beta}^0 = \int g_0 \varphi_\alpha \varphi_\beta d\Xi$ is known, \bar{a}^l and \bar{a}^r can be evaluated accordingly. The term \bar{b} is evaluated from

$$\int \bar{b} \varphi_\alpha d\Xi = \int_{u \geq 0} b^l \varphi_\alpha d\Xi + \int_{u < 0} b^r \varphi_\alpha d\Xi \quad (3.29)$$

Up to this point, all the parameters in the initial gas distribution function f_0 and the equilibrium state g at the beginning of each time step $t=0$ have been determined.

The only unknown left in Eq. (3.22) is \bar{A} . Since f and g contain \bar{A} , the integration of the conservation constraint at the cell interface over the whole time step Δt gives

$$\int_0^{\Delta t} \int (g - f) \varphi_\alpha dt d\Xi = 0 \quad (3.30)$$

which can be used to get \bar{A} uniquely.

The above derivation process clearly shows that the evaluation of numerical fluxes by the gas kinetic BGK scheme is relatively complex and expansive because numerous coefficients related to the physical space and phase space have to be calculated at every cell interface and each time step. Moreover, it is an arduous task to derive explicit expressions for the numerical fluxes, which are lacking in literature so far. To eliminate these drawbacks, the following solver is proposed.

3.3 Gas kinetic Flux Solver (GKFS)

In this section, GKFS is proposed for simulation of incompressible and compressible viscous flows. Different from the above gas kinetic BGK scheme, a simple way is introduced to evaluate the non-equilibrium distribution function. To the order of N-S equations, the distribution function can be approximated as

$$f = g - \tau(g_t + ug_x + vg_y) \quad (3.31)$$

By applying the Taylor series expansion in time and physical space, the above equation can be simplified to

$$f(0,0,t + \delta t) = g(0,0,t + \delta t) - \frac{\tau}{\delta t} [g(0,0,t + \delta t) - g(-u\delta t, -v\delta t, t)] \quad (3.32)$$

where δt is the streaming time step. $f(0,0,t + \delta t)$ is the gas distribution function at the interface ($x=0, y=0$), while $g(0,0,t + \delta t)$ and $g(-u\delta t, -v\delta t, t)$ are the equilibrium distribution functions at the interface and its surrounding points, respectively. From Eq. (3.32), it can be seen that the distribution function at the interface $f(0,0,t + \delta t)$ can be separated into two parts. The first one is the equilibrium distribution function at the interface, $g(0,0,t + \delta t)$. The other one is the difference of equilibrium distribution functions at the interface and its surrounding points, $-\frac{\tau}{\delta t} [g(0,0,t + \delta t) - g(-u\delta t, -v\delta t, t)]$, which is the non-equilibrium part.

Suppose that we know the conservative flow variables and their first order derivatives at the left and right sides of the interface (they can be easily given by interpolation from those at cell centers). With Eq. (3.2), the corresponding equilibrium distribution function and its first order derivatives at left and right sides of the interface can also be given. Then the second order approximation of $g(-u\delta t, -v\delta t, t)$ at a time level t can be written as

$$g(-u\delta t, -v\delta t, t) = \begin{cases} g_l - \frac{\partial g_l}{\partial x} u\delta t - \frac{\partial g_l}{\partial y} v\delta t, & u \geq 0 \\ g_r - \frac{\partial g_r}{\partial x} u\delta t - \frac{\partial g_r}{\partial y} v\delta t, & u < 0 \end{cases} \quad (3.33)$$

where g_l and g_r are the equilibrium distribution function at left and right side of the interface, respectively. By substituting Eq. (3.33) into Eq. (3.32), we have

$$\begin{aligned}
& f(0,0,t+\delta t) \\
&= g(0,0,t+\delta t) - \frac{\tau}{\delta t} \left[g(0,0,t+\delta t) - \left(g_l - \frac{\partial g_l}{\partial x} u \delta t - \frac{\partial g_l}{\partial y} v \delta t \right) \cdot H(u) \right. \\
&\quad \left. - \left(g_r - \frac{\partial g_r}{\partial x} u \delta t - \frac{\partial g_r}{\partial y} v \delta t \right) \cdot (1-H(u)) \right] \tag{3.34} \\
&= g(0,0,t+\delta t) - \frac{\tau}{\delta t} \left[g(0,0,t+\delta t) - g_l \cdot H(u) - g_r \cdot (1-H(u)) \right] \\
&\quad - \tau \left[\left(\frac{\partial g_l}{\partial x} u + \frac{\partial g_l}{\partial y} v \right) \cdot H(u) + \left(\frac{\partial g_r}{\partial x} u + \frac{\partial g_r}{\partial y} v \right) \cdot (1-H(u)) \right]
\end{aligned}$$

where $H(u)$ is the Heaviside function defined in Eq. (3.20). The above equation shows that once we have the equilibrium distribution functions at the cell interface and its surrounding points, the full information of distribution function at the interface can be obtained.

3.3.1 Basic formulations for evaluation of conservative flow variables

It is known that the non-equilibrium part has no effect on the calculation of conservative flow variables. As a result, the conservative flow variables at the cell interface can be computed by Eq. (3.14)

$$W = \int \varphi_\alpha g(0,0,t+\delta t) d\Xi$$

Substituting Eqs. (3.32) and (3.33) into Eq. (3.14), we have

$$\begin{aligned}
W &= \int \varphi_\alpha g(-u\delta t, -v\delta t, t) d\Xi \\
&= \iint_{u \geq 0} \varphi_\alpha \left(g_l - \frac{\partial g_l}{\partial x} u \delta t - \frac{\partial g_l}{\partial y} v \delta t \right) d\Xi \tag{3.35} \\
&\quad + \iint_{u < 0} \varphi_\alpha \left(g_r - \frac{\partial g_r}{\partial x} u \delta t - \frac{\partial g_r}{\partial y} v \delta t \right) d\Xi
\end{aligned}$$

The above Eq. (3.35) shows that conservative flow variables at the cell interface can be calculated by equilibrium distribution function of the surrounding points.

Having considered that the interface may have a discontinuity in the x -direction (u direction), the integral domains for u and v in Eq. (3.35) are different. In the v direction, the integral domain is always from $-\infty$ to $+\infty$, while in the u direction, the integral domain is from $-\infty$ to 0 on the right side and 0 to $+\infty$ on the left side. This is clearly shown in Eq. (3.35). With parameters defined in Appendix A, the conservative flow variables at the interface \mathbf{W} are given by

$$\rho = (\rho_l \cdot a_l + \rho_r \cdot a_r) - \left[\frac{\partial(\rho_l \cdot b_l + \rho_r \cdot b_r)}{\partial x} + \frac{\partial(\rho_l V_l \cdot a_l + \rho_r V_r \cdot a_r)}{\partial y} \right] \delta t \quad (3.36)$$

$$\rho U = (\rho_l \cdot b_l + \rho_r \cdot b_r) - \left[\frac{\partial(\rho_l \cdot c_l + \rho_r \cdot c_r)}{\partial x} + \frac{\partial(\rho_l V_l \cdot b_l + \rho_r V_r \cdot b_r)}{\partial y} \right] \delta t \quad (3.37)$$

$$\rho V = (\rho_l V_l \cdot a_l + \rho_r V_r \cdot a_r) - \left[\frac{\partial(\rho_l V_l \cdot b_l + \rho_r V_r \cdot b_r)}{\partial x} + \frac{\partial[(\rho_l V_l^2 + p_l) \cdot a_l + (\rho_r V_r^2 + p_r) \cdot a_r]}{\partial y} \right] \delta t \quad (3.38)$$

$$\begin{aligned} \rho E = & \frac{1}{2} \rho_l [c_l + (V_l^2 + (b-1)RT_l) \cdot a_l] + \frac{1}{2} \rho_r [c_r + (V_r^2 + (b-1)RT_r) \cdot a_r] \\ & - \frac{1}{2} \left\{ \frac{\partial}{\partial x} \left[\rho_l [d_l + (V_l^2 + (b-1)RT_l) \cdot b_l] + \rho_r [d_r + (V_r^2 + (b-1)RT_r) \cdot b_r] \right] \right. \\ & \left. + \frac{\partial}{\partial y} \left[\rho_l V_l [c_l + (V_l^2 + (b+1)RT_l) \cdot a_l] + \rho_r V_r [c_r + (V_r^2 + (b+1)RT_r) \cdot a_r] \right] \right\} \delta t \end{aligned} \quad (3.39)$$

where “ \cdot_l ” and “ \cdot_r ” (“ \cdot ” stands for any variable) denote the variable at the left and right side of interface, respectively.

3.3.2 Basic formulations for evaluation of flux at cell interface

Once the conservative flow variables at the cell interface \mathbf{W} are obtained, the equilibrium distribution function $g(0,0,t+\delta t)$ can be calculated via Eq. (3.2). Then the flux across the interface can be calculated via Eq. (3.34). Take \mathbf{F}^x as an example to illustrate the process.

$$\begin{aligned}
\mathbf{F}^x &= \int u\varphi_\alpha f(0,0,t+\delta t)d\Xi \\
&= \int u\varphi_\alpha g(0,0,t+\delta t)d\Xi - \frac{\tau}{\delta t} \left[\int u\varphi_\alpha g(0,0,t+\delta t)d\Xi \right. \\
&\quad \left. - \int \int_{u \geq 0} u\varphi_\alpha g_l d\Xi - \int \int_{u < 0} u\varphi_\alpha g_r d\Xi \right] \\
&\quad - \tau \left[\frac{\partial}{\partial x} \int \left(\int_{u \geq 0} u^2 \varphi_\alpha g_l + \int_{u < 0} u^2 \varphi_\alpha g_r \right) d\Xi + \frac{\partial}{\partial y} \int \left(\int_{u \geq 0} uv\varphi_\alpha g_l + \int_{u < 0} uv\varphi_\alpha g_r \right) d\Xi \right]
\end{aligned} \tag{3.40}$$

In the above Eq. (3.40), the flux across the interface can be categorized into two parts. The first part is the flux contributed by equilibrium distribution function at the interface, which is denoted as $\mathbf{F}^{x(0)}$:

$$\mathbf{F}^{x(0)} = \int u\varphi_\alpha g(0,0,t+\delta t)d\Xi = \begin{bmatrix} \rho U \\ \rho U^2 + p \\ \rho UV \\ (\rho E + p)U \end{bmatrix} \tag{3.41}$$

where ρ , U , V and p are the density, velocities and pressure at the cell interface given above.

The other part is the flux contributed by non-equilibrium distribution function, which is calculated by the difference of equilibrium distribution functions at the interface and its surrounding points, which is denoted as $\mathbf{F}^{x(1)}$:

$$\begin{aligned}
\mathbf{F}^{x(1)} &= -\frac{\tau}{\delta t} \left[\int u \varphi_\alpha g(0, 0, t + \delta t) d\Xi - \int u \varphi_\alpha g(-u\delta t, -v\delta t, t) d\Xi \right] \\
&= -\frac{\tau}{\delta t} \left[\int u \varphi_\alpha g(0, 0, t + \delta t) d\Xi - \int \int_{u \geq 0} u \varphi_\alpha g_l d\Xi - \int \int_{u < 0} u \varphi_\alpha g_r d\Xi \right] \\
&\quad - \tau \left[\frac{\partial}{\partial x} \int \left(\int_{u \geq 0} u^2 \varphi_\alpha g_l + \int_{u < 0} u^2 \varphi_\alpha g_r \right) d\Xi + \frac{\partial}{\partial y} \int \left(\int_{u \geq 0} uv \varphi_\alpha g_l + \int_{u < 0} uv \varphi_\alpha g_r \right) d\Xi \right]
\end{aligned} \tag{3.42}$$

Refer to Eqs. (A.14) – (A.23) in Appendix A, the explicit expressions of $\mathbf{F}^{x(1)}$ are

$$\begin{aligned}
F_1^{x(1)} &= -\frac{\tau}{\delta t} (\rho U - \rho_l \cdot b_l - \rho_r \cdot b_r) \\
&\quad - \tau \left[\frac{\partial (\rho_l \cdot c_l + \rho_r \cdot c_r)}{\partial x} + \frac{\partial (\rho_l V_l \cdot b_l + \rho_r V_r \cdot b_r)}{\partial y} \right]
\end{aligned} \tag{3.43}$$

$$\begin{aligned}
F_2^{x(1)} &= -\frac{\tau}{\delta t} \left[(\rho U^2 + p) - \rho_l \cdot c_l - \rho_r \cdot c_r \right] \\
&\quad - \tau \left[\frac{\partial (\rho_l \cdot d_l + \rho_r \cdot d_r)}{\partial x} + \frac{\partial (\rho_l V_l \cdot c_l + \rho_r V_r \cdot c_r)}{\partial y} \right]
\end{aligned} \tag{3.44}$$

$$\begin{aligned}
F_3^{x(1)} &= -\frac{\tau}{\delta t} (\rho UV - \rho_l V_l \cdot b_l - \rho_r V_r \cdot b_r) \\
&\quad - \tau \left[\frac{\partial (\rho_l V_l \cdot c_l + \rho_r V_r \cdot c_r)}{\partial x} + \frac{\partial [(\rho_l V_l^2 + p_l) \cdot b_l + (\rho_r V_r^2 + p_r) \cdot b_r]}{\partial y} \right]
\end{aligned} \tag{3.45}$$

$$\begin{aligned}
F_4^{x(1)} &= -\frac{1}{2} \frac{\tau}{\delta t} \left\{ \rho \left[(U^2 + V^2) + (b+2)RT \right] U \right. \\
&\quad \left. - \rho_l \left[d_l + (V_l^2 + (b-1)RT_l) \cdot b_l \right] - \rho_r \left[d_r + (V_r^2 + (b-1)RT_r) \cdot b_r \right] \right\} \\
&\quad - \frac{1}{2} \tau \left\{ \frac{\partial}{\partial x} \left\{ \rho_l \left[e_l + (V_l^2 + (b-1)RT_l) \cdot c_l \right] + \rho_r \left[e_r + (V_r^2 + (b-1)RT_r) \cdot c_r \right] \right\} \right. \\
&\quad \left. + \frac{\partial}{\partial y} \left\{ \rho_l V_l \left[d_l + (V_l^2 + (b+1)RT_l) \cdot b_l \right] + \rho_r V_r \left[d_r + (V_r^2 + (b+1)RT_r) \cdot b_r \right] \right\} \right\}
\end{aligned} \tag{3.46}$$

Therefore, the final flux \mathbf{F}^x can be calculated by,

$$\mathbf{F}^x = \mathbf{F}^{x(0)} + \mathbf{F}^{x(1)} \tag{3.47}$$

3.3.3 Three schemes for evaluation of conservative flow variables and fluxes at cell interface

As shown in the above section, the calculation of conservative flow variables and numerical fluxes at cell interface involves the streaming time step δt . The principle for evaluation of δt is that the location of the equilibrium distribution function around the interface $g(-u\delta t, -v\delta t, t)$ in Eq. (3.32) must be within either the left cell or the right cell of the interface. In other words, the transport distance of particle should be smaller than half of the cell size, i.e.,

$$\begin{aligned} u\delta t &\leq \frac{1}{2}\Delta x \\ v\delta t &\leq \frac{1}{2}\Delta y \end{aligned} \tag{3.48}$$

However, in the current gas kinetic schemes, the velocities of particles in the phase space range from $-\infty$ to $+\infty$. Therefore, the streaming time step δt cannot be simply determined. In order to resolve this problem, three schemes will be introduced and their performance will be tested in the numerical examples.

Scheme I

Figure 3.1 shows the particle distribution function at the interface (Xu, 1998). As shown, the distribution function decreases to zero when u is increased to infinity. Therefore, a finite domain in the velocity space is adopted, $u \in [-U_{\max}, U_{\max}]$, where U_{\max} is the maximum velocity of particles in the streaming process. The particles beyond this region are neglected. This idea has been applied in the unified gas kinetic scheme (Xu and Huang, 2010). Thus, the streaming time step can be calculated via

Eq.(3.48) when u and v take the extreme case of U_{\max} . Mathematically, δt can be chosen by

$$\delta t \leq \frac{\text{Minimum}(\Delta x, \Delta y)}{2U_{\max}}$$

Note that as local solution of the BGK equation is reconstructed at each cell interface, δt is locally determined at different interfaces. After streaming time step δt is determined, the conservative flow variables at the interface are calculated via Eqs. (3.36)-(3.39), and the numerical fluxes across the interface are computed via Eqs. (3.41), (3.43)-(3.46), (3.47).

Scheme II

In the gas kinetic BGK scheme (Xu, 2001), the conservative flow variables at the interface are calculated by Eq. (3.27)

$$W = \int g_0 \varphi_\alpha d\Xi = \int_{u \geq 0} g^l \varphi_\alpha d\Xi + \int_{u < 0} g^r \varphi_\alpha d\Xi$$

The above equation is equivalent to Eq. (3.35) when streaming time step δt is set to zero. In Scheme II, Eq. (3.27) is adopted to calculate the conservative flow variables at the interface. In this way, the conservative flow variables at the interface can be simply computed by the reconstructed variables of left and right sides without involving their derivatives, that is,

$$\rho = (\rho_l \cdot a_l + \rho_r \cdot a_r) \quad (3.49)$$

$$\rho U = (\rho_l \cdot b_l + \rho_r \cdot b_r) \quad (3.50)$$

$$\rho V = (\rho_l V_l \cdot a_l + \rho_r V_r \cdot a_r) \quad (3.51)$$

$$\rho E = \frac{1}{2} \rho_l [c_l + (V_l^2 + (b-1)RT_l) \cdot a_l] + \frac{1}{2} \rho_r [c_r + (V_r^2 + (b-1)RT_r) \cdot a_r] \quad (3.52)$$

Once the conservative flow variables are computed by the above equations, the flux $F^{x(0)}$ can be calculated by Eq. (3.41) without influence of streaming time step. However, it was found that this treatment will introduce inconsistency for evaluation of fluxes ρU in the mass conservation equation. As shown above, ρU can be given from Eq. (3.50), which has nothing to do with the collision time τ . On the other hand, by taking $\varphi_\alpha = 1$ in Eq. (3.40), the flux of mass conservation equation becomes

$$F_1^x = \rho U - \tau \left[\frac{\partial}{\partial x} \int \left(\int_{u \geq 0} u^2 g_l + \int_{u < 0} u^2 g_r \right) d\Xi + \frac{\partial}{\partial y} \int \left(\int_{u \geq 0} uv g_l + \int_{u < 0} uv g_r \right) d\Xi \right] \quad (3.53)$$

In the above equation, the term proportional to τ is the viscous flux as τ is proportional to the dynamic viscosity. It is unphysical because there is no viscous term in the mass conservation equation. To remove this contradiction, the flux F_1^x in the mass conservation equation is computed by Eq. (3.50) rather than by Eq. (3.53) in this work, that is,

$$F_1^x = \rho U \quad (3.54)$$

Other fluxes in the momentum equations and energy equation are computed by using the same formulations as in Scheme I. Note that the streaming time step has effect on the flux vector $F^{x(1)}$, which is determined by the same equations (3.44)-(3.46) as in Scheme I.

Scheme III

In the above two schemes, the determination of streaming time step is necessary to evaluate the fluxes. However, in the present work, the streaming time step cannot be determined precisely and it is approximated by the finite domain in the phase velocity space. This will probably bring numerical errors and instability into the schemes,

especially in the case of compressible flows. This motivates the development of Scheme III, in which the effect of streaming time step is eliminated.

In Scheme III, the way to calculate the conservative flow variables at the interface are the same as in Scheme II, which has been given in Eqs. (3.49)-(3.52). For the calculation of flux, it is obvious that only the middle term on the right side in Eq. (3.40) involves the streaming time step δt . Note that $g(0,0,t+\delta t)$ is the equilibrium distribution function at the interface and time level $t+\delta t$, and g_l, g_r are the distribution functions at the left and right sides of the interface and the time level t . In fact, the middle term of Eq. (3.40) can be approximated by

$$\begin{aligned} & -\frac{\tau}{\delta t} \left[\int u \varphi_\alpha g(0,0,t+\delta t) d\Xi - \int_{u \geq 0} u \varphi_\alpha g_l d\Xi - \int_{u < 0} u \varphi_\alpha g_r d\Xi \right] \\ & = -\tau \int u \varphi_\alpha \frac{\partial g(0,0,t+\delta t)}{\partial t} d\Xi \end{aligned} \quad (3.55)$$

According to the work of Xu (2001), $\partial g / \partial t$ can be approximated by

$$\frac{\partial g(0,0,t+\delta t)}{\partial t} = g(0,0,t+\delta t) (A_1 + A_2 u + A_3 v + A_4 \varepsilon) \quad (3.56)$$

where A_1, A_2, A_3 and A_4 are the derivatives of macroscopic variables with respect to time, which will be determined from the compatibility condition, $\varepsilon = \frac{1}{2}(u^2 + v^2 + \xi^2)$.

Thus, the flux expression in Eq. (3.40) can be written as

$$\begin{aligned} \mathbf{F}^x & = \int u \varphi_\alpha g(0,0,t+\delta t) d\Xi - \tau \int u \varphi_\alpha g(0,0,t+\delta t) (A_1 + A_2 u + A_3 v + A_4 \varepsilon) d\Xi \\ & \quad - \tau \left[\frac{\partial}{\partial x} \int \left(\int_{u \geq 0} u^2 \varphi_\alpha g_l + \int_{u < 0} u^2 \varphi_\alpha g_r \right) d\Xi + \frac{\partial}{\partial y} \int \left(\int_{u \geq 0} uv \varphi_\alpha g_l + \int_{u < 0} uv \varphi_\alpha g_r \right) d\Xi \right] \end{aligned} \quad (3.57)$$

In the above equation, the effect of streaming time step is eliminated. The only undetermined variables in this scheme are the coefficients A_1, A_2, A_3 and A_4 .

From Eq. (3.35), we have

$$\begin{aligned} W &= \int \varphi_\alpha g(0,0,t+\delta t) d\Xi \\ &= \iint_{u \geq 0} \varphi_\alpha \left(g_l - \frac{\partial g_l}{\partial x} u \delta t - \frac{\partial g_l}{\partial y} v \delta t \right) d\Xi + \iint_{u < 0} \varphi_\alpha \left(g_r - \frac{\partial g_r}{\partial x} u \delta t - \frac{\partial g_r}{\partial y} v \delta t \right) d\Xi \end{aligned} \quad (3.58)$$

The above equation can be rewritten as

$$\begin{aligned} &\int \varphi_\alpha \left[\int g(0,0,t+\delta t) - \int_{u \geq 0} g_l - \int_{u < 0} g_r \right] d\Xi \\ &= - \left[\iint_{u \geq 0} \varphi_\alpha \left(\frac{\partial g_l}{\partial x} u + \frac{\partial g_l}{\partial y} v \right) d\Xi + \iint_{u < 0} \varphi_\alpha \left(\frac{\partial g_r}{\partial x} u + \frac{\partial g_r}{\partial y} v \right) d\Xi \right] \delta t \end{aligned} \quad (3.59)$$

Using Eqs. (3.55) and (3.56), the above equation can be rewritten as

$$\begin{aligned} &\int \varphi_\alpha g(0,0,t+\delta t) \cdot (A_1 + A_2 u + A_3 v + A_4 \varepsilon) d\Xi \\ &= - \left[\frac{\partial}{\partial x} \int \left(\int_{u \geq 0} u \varphi_\alpha g_l + \int_{u < 0} u \varphi_\alpha g_r \right) d\Xi + \frac{\partial}{\partial y} \int \left(\int_{u \geq 0} v \varphi_\alpha g_l + \int_{u < 0} v \varphi_\alpha g_r \right) d\Xi \right] \end{aligned} \quad (3.60)$$

Defining

$$\frac{\partial}{\partial x} \int \left(\int_{u \geq 0} u \varphi_\alpha g_l + \int_{u < 0} u \varphi_\alpha g_r \right) d\Xi + \frac{\partial}{\partial y} \int \left(\int_{u \geq 0} v \varphi_\alpha g_l + \int_{u < 0} v \varphi_\alpha g_r \right) d\Xi = \begin{pmatrix} G_1 \\ G_2 \\ G_3 \\ G_4 \end{pmatrix} \quad (3.61)$$

Then Eq. (3.60) can be written as

$$\rho \begin{pmatrix} 1 & U & V & \mathfrak{R}_1 \\ U & U^2 + \frac{1}{2\lambda} & UV & \mathfrak{R}_2 \\ V & UV & V^2 + \frac{1}{2\lambda} & \mathfrak{R}_3 \\ \mathfrak{R}_1 & \mathfrak{R}_2 & \mathfrak{R}_3 & \mathfrak{R}_4 \end{pmatrix} \begin{pmatrix} A_1 \\ A_2 \\ A_3 \\ A_4 \end{pmatrix} + \begin{pmatrix} G_1 \\ G_2 \\ G_3 \\ G_4 \end{pmatrix} = 0 \quad (3.62)$$

where ρ , U , V and λ are macroscopic flow variables at the interface and

$$\mathfrak{R}_1 = \frac{1}{2} \left(U^2 + V^2 + \frac{K+2}{2\lambda} \right) \quad (3.63)$$

$$\mathfrak{R}_2 = \frac{1}{2}U \left(U^2 + V^2 + \frac{K+4}{2\lambda} \right) \quad (3.64)$$

$$\mathfrak{R}_3 = \frac{1}{2}V \left(U^2 + V^2 + \frac{K+4}{2\lambda} \right) \quad (3.65)$$

$$\mathfrak{R}_4 = \frac{1}{4} \left[(U^2 + V^2)^2 + (U^2 + V^2) \frac{K+4}{\lambda} + \frac{K^2 + 6K + 8}{4\lambda^2} \right] \quad (3.66)$$

From Eq. (3.62), the coefficients A_1 , A_2 , A_3 and A_4 are determined as

$$\begin{aligned} A_4 &= -\frac{8\lambda^2}{(K+2)\rho} \left[G_4 - UG_2 - VG_3 - (\mathfrak{R}_1 - U^2 - V^2)G_1 \right] \\ &= -\frac{8\lambda^2}{(K+2)\rho} \left[G_4 - UG_2 - VG_3 + \frac{1}{2} \left(U^2 + V^2 - \frac{K+2}{2\lambda} \right) G_1 \right] \end{aligned} \quad (3.67)$$

$$A_3 = -\frac{2\lambda}{\rho} (G_3 - VG_1) - VA_4 \quad (3.68)$$

$$A_2 = -\frac{2\lambda}{\rho} (G_2 - UG_1) - UA_4 \quad (3.69)$$

$$A_1 = -\frac{1}{\rho} G_1 - UA_2 - VA_3 - \mathfrak{R}_1 A_4 \quad (3.70)$$

Once the coefficients are obtained, the fluxes across the interface can be calculated via Eq. (3.57).

Similar to Schemes I and II, the explicit expressions for conservative flow variables and fluxes in Scheme III can also be given. As has been introduced above, the conservative flow variables at the interface are calculated via Eq. (3.27) and the explicit expressions are given in Eqs. (3.49)-(3.52). The explicit expression for $\mathbf{F}^{x(0)}$ is the same as Eq. (3.41), that is,

$$\mathbf{F}^{x(0)} = \int u \varphi_\alpha g(0, 0, t + \delta t) d\Xi = \begin{bmatrix} \rho U \\ \rho U^2 + p \\ \rho UV \\ (\rho E + p)U \end{bmatrix}$$

For $\mathbf{F}^{x(1)}$, the explicit expressions are different from those in Scheme I and Scheme II, which are given in Eqs. (3.43)-(3.46). Using the definition in Appendix A and coefficients A_1 , A_2 , A_3 and A_4 , the explicit expressions for $\mathbf{F}^{x(1)}$ in Scheme III become

$$F_1^{x(1)} = -\tau \rho \left[A_1 \langle u^1 \rangle + A_2 \langle u^2 \rangle + A_3 \langle u^1 \rangle \langle v^1 \rangle + \frac{1}{2} A_4 \left(\langle u^3 \rangle + \langle u^1 \rangle \langle v^2 \rangle + \langle u^1 \rangle \langle \xi^2 \rangle \right) \right] - \tau \left[\frac{\partial(\rho_l \cdot c_l + \rho_r \cdot c_r)}{\partial x} + \frac{\partial(\rho_l V_l \cdot b_l + \rho_r V_r \cdot b_r)}{\partial y} \right] \quad (3.71)$$

$$F_2^{x(1)} = -\tau \rho \left[A_1 \langle u^2 \rangle + A_2 \langle u^3 \rangle + A_3 \langle u^2 \rangle \langle v^1 \rangle + \frac{1}{2} A_4 \left(\langle u^4 \rangle + \langle u^2 \rangle \langle v^2 \rangle + \langle u^2 \rangle \langle \xi^2 \rangle \right) \right] - \tau \left[\frac{\partial(\rho_l \cdot d_l + \rho_r \cdot d_r)}{\partial x} + \frac{\partial(\rho_l V_l \cdot c_l + \rho_r V_r \cdot c_r)}{\partial y} \right] \quad (3.72)$$

$$F_3^{x(1)} = -\tau \rho \left[A_1 \langle u^1 \rangle \langle v^1 \rangle + A_2 \langle u^2 \rangle \langle v^1 \rangle + A_3 \langle u^1 \rangle \langle v^2 \rangle + \frac{1}{2} A_4 \left(\langle u^3 \rangle \langle v^1 \rangle + \langle u^1 \rangle \langle v^3 \rangle + \langle u^1 \rangle \langle v^1 \rangle \langle \xi^2 \rangle \right) \right] - \tau \left[\frac{\partial(\rho_l V_l \cdot c_l + \rho_r V_r \cdot c_r)}{\partial x} + \frac{\partial[(\rho_l V_l^2 + p_l) \cdot b_l + (\rho_r V_r^2 + p_r) \cdot b_r]}{\partial y} \right] \quad (3.73)$$

$$F_4^{x(1)} = -\frac{1}{2} \tau \rho \left\{ A_1 \left[\langle u^3 \rangle + \langle u^1 \rangle \langle v^2 \rangle + \langle u^1 \rangle \langle \xi^2 \rangle \right] + A_2 \left[\langle u^4 \rangle + \langle u^2 \rangle \langle v^2 \rangle + \langle u^2 \rangle \langle \xi^2 \rangle \right] + A_3 \left[\langle u^3 \rangle \langle v^1 \rangle + \langle u^1 \rangle \langle v^3 \rangle + \langle u^1 \rangle \langle v^1 \rangle \langle \xi^2 \rangle \right] + \frac{1}{2} A_4 \left[\langle u^5 \rangle + \langle u^1 \rangle \langle v^4 \rangle + \langle u^1 \rangle \langle \xi^4 \rangle + 2 \langle u^3 \rangle \langle v^2 \rangle + 2 \langle u^3 \rangle \langle \xi^2 \rangle + 2 \langle u^1 \rangle \langle v^2 \rangle \langle \xi^2 \rangle \right] \right\} - \frac{1}{2} \tau \left\{ \frac{\partial}{\partial x} \left[\rho_l \left[e_l + (V_l^2 + (b-1)RT_l) \cdot c_l \right] + \rho_r \left[e_r + (V_r^2 + (b-1)RT_r) \cdot c_r \right] \right] + \frac{\partial}{\partial y} \left[\rho_l V_l \left[d_l + (V_l^2 + (b+1)RT_l) \cdot b_l \right] + \rho_r V_r \left[d_r + (V_r^2 + (b+1)RT_r) \cdot b_r \right] \right] \right\} \quad (3.74)$$

Correspondingly, the total flux F^x can be calculated by $F^x = F^{x(0)} + F^{x(1)}$. The explicit expressions for evaluation of conservative flow variables and fluxes in the y direction are given in Appendix B.

3.3.4 Collision time and Prandtl number fix

Theoretically, the collision time in the gas kinetic scheme is determined by the relationship of

$$\tau = \mu / p \quad (3.75)$$

where μ is the dynamical viscosity and p is the pressure. For an incompressible flow, μ is usually treated as a constant. For a compressible flow or an incompressible flow with heat transfer, such as hypersonic flow around a circular cylinder, the Sutherland's law is adopted

$$\mu = \mu_\infty \left(\frac{T}{T_\infty} \right)^{3/2} \frac{T_\infty + S}{T + S} \quad (3.76)$$

where $T_\infty = 285K$ is the reference temperature and S is the Sutherland temperature chosen as $110.4K$. μ_∞ is the viscosity at the reference temperature.

When there is a shock wave with a thickness in the order of the cell size, the numerical viscosity should be combined with the physical viscosity to take the pressure jump into account. Xu (2001) presented a simple and effective treatment to introduce the numerical dissipation into their gas kinetic BGK scheme, which is also adopted in the present work:

$$\tau = \frac{\mu}{p} + \frac{|p_L - p_R|}{p_L + p_R} \Delta t \quad (3.77)$$

where Δt is the time step in the solution of N-S equations, p_L and p_R are the pressure at the left and right sides of the cell interface, respectively. The second part of the above equation corresponds to numerical viscosity, which is applied to capture the shock wave and increase the robustness of the scheme.

As shown in the work of Xu (1998), the Prandtl number in the gas kinetic scheme corresponds to unity. In order to adjust the Prandtl number for the present scheme to any realistic value, one of the convenient approaches is to modify the heat flux with a variable Prandtl number (Xu, 2001)

$$F_E^{new} = F_E + \left(\frac{1}{Pr} - 1 \right) q \quad (3.78)$$

where F_E is the energy flux. q is the heat flux evaluated from

$$q = \frac{1}{2} \int (u-U) \left((u-U)^2 + (v-V)^2 + \xi^2 \right) f d\Xi \quad (3.79)$$

The explicit expression of q is given in Appendix C.

3.3.5 Computational sequence

In the proposed GKFS, the computational procedure of the first two schemes is similar to each other, which is summarized below:

- (1) Firstly, the maximum velocity of particles in the phase velocity space, U_{\max} , is chosen properly. Then, we need to specify the streaming time step δt at each interface. The principle for the choice of δt is that the location of the equilibrium distribution function around the interface $g(-u\delta t, -v\delta t, t)$ in Eq. (3.32) should be within either the left or right cell of the interface.

- (2) The first order derivatives of conservative flow variables are calculated and the initial reconstructions are conducted at two sides of the cell interface.
- (3) Calculate the conservative flow variables at the cell interface W by using Eqs. (3.36)-(3.39) for Scheme I or Eqs. (3.49)-(3.52) for Scheme II. After that, flux $F^{x(0)}$ can be calculated by Eq. (3.41).
- (4) Compute the flux $F^{x(1)}$ by using Eqs. (3.43)-(3.46).
- (5) Calculate the total flux across the cell interface F^x by using Eq. (3.47). For Scheme II, the flux of mass conservation equation is directly obtained by conservative flow variables by using Eq. (3.54).
- (6) Calculate the heat flux q via Eq. (3.79), and make correction for energy flux by using Eq. (3.78).
- (7) The total flux across the cell interface in the y direction F^y can be calculated in the similar way. The flux expressions are given in Appendix B. Once the fluxes at all cell interfaces are obtained, solve ordinary differential equation (Eq. (3.10)) by using 4-stage Runge-Kutta scheme.
- (8) Repeat steps (1) - (7) until convergence criterion is satisfied.

For Scheme III, the solution procedure is a little different from the one above because the calculation of streaming time step is not necessary and some additional coefficients should be computed. The basic solution procedure is:

- (1) Firstly, the first order derivatives of conservative flow variables are calculated and the initial reconstructions are conducted at two sides of cell interface.
- (2) Calculate the conservative flow variables at the cell interface W by using Eqs. (3.49)-(3.52) and then the flux of $F^{x(0)}$ can be obtained by Eq. (3.41).

- (3) Calculate the vector $(G_1, G_2, G_3, G_4)^T$ by using Eq. (3.61) and further compute coefficients A_1, A_2, A_3, A_4 by Eqs. (3.67)-(3.70).
- (4) Calculate the flux of $F^{x(1)}$ by Eqs. (3.71)-(3.74).
- (5) Calculate the total flux across the cell interface F^x by Eq. (3.47).
- (6) Calculate the heat flux q via Eq. (3.79), and make correction for energy flux by using Eq. (3.78).
- (7) The total flux across the cell interface in the y direction F^y can be calculated in the similar way. The flux expressions are given in Appendix B. Once the fluxes at all cell interfaces are obtained, solve ordinary differential equation (Eq. (3.10)) by using 4-stage Runge-Kutta scheme.
- (8) Repeat steps (1) - (7) until convergence criterion is satisfied.

3.4 Numerical Results and Discussion

In this section, the developed GKFS is validated by applying it to simulate test problems of both incompressible and compressible viscous flows. First, the decaying vortex problem is solved on a uniform mesh to study the order of solution accuracy. Subsequently, the 2D driven cavity flow is simulated by using non-uniform grids. The comparison of solution accuracy and computational time required by three schemes is shown. Viscous flow past a circular cylinder is also simulated to examine the capability of the present GKFS for problems with curved boundaries. The simulations of Couette flow with a temperature gradient are carried out to further investigate the validity of the present solver for variable Prandtl number. To determine the streaming time step δt , the finite domain in the phase velocity space is set as $[-5|U_{ref}|, 5|U_{ref}|]$ for Scheme I and Scheme II, where U_{ref} is usually selected as the free stream velocity.

Apart from incompressible flows, GKFS can also be applied to solve compressible flow problems. However, when compressible flows with shock waves are solved, on one hand, the mesh spacing should be chosen to be very small to capture the thin boundary layers and thus the streaming time step will be very small. On the other hand, the conservative flow variables on the left and right sides of the interface may differ a lot from the value at the interface. As a consequence, the term divided by δt in Eq. (3.42) might be very large, which brings numerical instability into the computation when Scheme I and Scheme II are applied. In contrast, the performance of Scheme III is much more consistent for simulation of both incompressible and compressible flows. Therefore, in the following discussion, only the results of Scheme III will be presented for compressible flows. Numerical simulations of shock wave-boundary layer interaction, flow over airfoils and hypersonic flow past a circular cylinder are carried out by using Scheme III. Except for the case of hypersonic flow around a circular cylinder, the Venkatakrishnan's limiter (Venkatakrishnan, 1995) is used to calculate the conservative flow variables at the two sides of cell interface in the reconstruction stage. For temporal discretization, four-stage Runge-Kutta method is used and the CFL number is set as 1 unless otherwise stated.

3.4.1 Decaying vortex flow

The numerical accuracy of three schemes of GKFS is examined by simulating the decaying vortex flow, which has an analytical solution

$$\rho(x, y, t) = \rho_0 - \frac{\rho_0 U^2}{4c_s^2} [\cos(2\pi x/L) + \sin(2\pi y/L)] e^{-4\pi^2 U t / (\text{Re} L)} \quad (3.80)$$

$$u(x, y, t) = -U \cos(\pi x/L) \sin(\pi y/L) e^{-2\pi^2 U t / (\text{Re} L)} \quad (3.81)$$

$$v(x, y, t) = U \sin(\pi x/L) \cos(\pi y/L) e^{-2\pi^2 U t / (Re L)} \quad (3.82)$$

where U is the characteristic velocity. In the present test, the Reynolds number is selected as $Re = \rho_0 U L / \mu = 10$. The computational domain is $[-L, L] \times [-L, L]$ and five different uniform grids ($N \times N$, $N = 41, 61, 81, 101, 161$) are used. The solution at time level $t = 1$ is selected and the relative error of velocity u is quantified by L_2 norm which is defined as

$$L_2 = \sqrt{\frac{\sum_{N \times N} \left(\frac{u^{numerical} - u^{exact}}{U} \right)^2}{N \times N}} \quad (3.83)$$

where $u^{numerical}$ and u^{exact} are the numerical and exact results, respectively. The numerical error versus mesh spacing in log scale is presented in Figure 3.2. As shown in the figure, the overall accuracy of the numerical results for the three schemes is slightly less than second order as the slopes of the lines are about 1.90 to 1.91.

3.4.2 2D lid-driven flow in a square cavity

A non-uniform mesh is used to simulate the lid-driven cavity flow at various Reynolds numbers. The mesh points are generated according to the following equations:

$$x_i = \frac{1 - \cos(k_i \pi)}{2}, \quad k_i = \frac{i-1}{i_{\max} - 1}, \quad i = 1, \dots, i_{\max}$$

$$y_j = \frac{1 - \cos(k_j \pi)}{2}, \quad k_j = \frac{j-1}{j_{\max} - 1}, \quad j = 1, \dots, j_{\max}$$

where i, j are the mesh point indices, x_i, y_j are the coordinates in the x and y directions, respectively; i_{\max} and j_{\max} are the maximum numbers of mesh points in the x and y directions. In this kind of mesh, the mesh spacing near the wall will be

very small to capture the boundary layers and in the middle region, the mesh spacing is relatively large. This characteristic can be used to reduce the total number of mesh points, especially for high Reynolds number cases. Non-uniform grids of 61×61 for $Re = 100$, 81×81 for $Re = 400$, 101×101 for $Re = 1000$, 121×121 for $Re = 5000$ and 10000 are applied. As this is a steady flow problem, the local time-stepping method is applied to accelerate the convergence, in which the largest time step is chosen for each control volume. In the present simulations, the density ρ and the lid velocity U are set as 1.0 and 0.1, respectively. Initially, the density is constant and the velocity is zero all over the flow domain. The boundary conditions are implemented by introducing ghost cells.

Table 3.1 compares the locations of the primary vortex centers at different Reynolds numbers obtained using the present three schemes of gas kinetic flux solver with those obtained by Ghia et al. (1982). As shown in the table, the vortex center moves towards the cavity center when the Reynolds number is increased. The maximum relative error between the present solutions and those of Ghia et al. (1982) is less than 1.0%. Figure 3.3 presents velocity u along the horizontal central line and v along the vertical central line for a lid-driven cavity flow at various Reynolds numbers. The results given by the three schemes are almost the same, and they are in good agreement with those of Ghia et al. (1982), which show the feasibility and accuracy of the current schemes. Table 3.2 compares the computational times for the three schemes. It can be seen from the table that, the computational efficiency of Scheme I and Scheme III is similar. This is because all terms which need to be calculated and stored are almost the same except for the coefficients A_1, A_2, A_3, A_4 used in Scheme

III. The computational time for Scheme II is about 55% and 65% of that in Scheme I and Scheme III.

The pure stability of GKFS is examined without considering the accuracy. Uniform grids are applied for simulating lid-driven cavity flow at $Re = 10000$. The numerical results show that all three schemes can obtain converged results even when the mesh is reduced to 5 points in one direction. This test demonstrates the good stability performance of GKFS as it requires a small number of grid points to yield stable solution.

3.4.3 Viscous flow past a circular cylinder

Although the complex lid-driven cavity flows have been tested to validate the present schemes, the geometry of the cavity which only involves straight boundaries is nevertheless simple. To further illustrate the capability of the present GKFS for problems with a curved boundary, the flow past a circular cylinder is simulated. This is an attractive test problem, which has been investigated extensively.

Different kinds of flow behaviors for this flow are characterized by the Reynolds number which can be defined as

$$Re = \frac{\rho_{\infty} U_{\infty} D}{\mu} \quad (3.84)$$

where ρ_{∞} is the free stream density, U_{∞} is the free stream velocity, D is the diameter of the cylinder and μ is the dynamic viscosity of the fluid. The drag coefficient C_d and lift coefficient C_L are commonly used to verify the accuracy of the numerical results, which are defined as

$$C_d = \frac{F_D}{1/2 \rho_\infty U_\infty^2 D} \quad (3.85)$$

$$C_L = \frac{F_L}{1/2 \rho_\infty U_\infty^2 D} \quad (3.86)$$

where F_D and F_L are drag and lift forces, respectively. For unsteady flows, another important parameter to examine the vortex shedding frequency is the Strouhal number:

$$St = \frac{fD}{U_\infty} \quad (3.87)$$

where f is the vortex shedding frequency.

The typical O-type structured grid, where the outer boundary is taken as a circle, is used in the present study. For the steady flow, the outer boundary is 25.5 diameters away from the center of the circular cylinder, and a non-uniform mesh of 240×160 is applied. For the unsteady case, the outer boundary is 55.5 diameters away from the center of cylinder, and a non-uniform mesh of 480×320 is used. The no-slip boundary condition is imposed on the circular cylinder and far field boundary condition is used on the outer boundary.

When $Re = 20$ and 40 , the flow past a circular cylinder finally reaches a steady state. To examine the accuracy of the present three schemes, a detailed comparison of three parameters, including recirculation length L_s , separation angle θ_s and drag coefficient C_d , is made with previous studies in Table 3.3. Obviously, the results of all the three schemes agree well with the results in the literature. Figure 3.4 shows the streamlines for the two steady cases. For simplicity, only the results of Scheme I are illustrated since all the three schemes give almost the same results. As shown in the

figure, a pair of stationary recirculating vortices forms behind the cylinder and the length of the vortices increases when the Reynolds number increases. To further demonstrate the performance of the present solver, the pressure coefficient distribution around the cylinder surface at $Re = 40$ is depicted in Figure 3.5. The experimental data from Park et al. (1998) and the numerical solutions obtained by Shu et al. (2014) are also included in this figure. The orientation angle θ is measured in degree from the leading stagnation point to the trailing stagnation point. Since the flow is symmetric about x -axis in this steady case, only the pressure coefficient distribution on the upper surface of the cylinder is presented.

For $Re = 100$ and 200 , the flow is unsteady and eventually reaches a periodic state. Table 3.4 shows the quantitative comparison of the lift and drag coefficients and the Strouhal numbers for the three schemes. It can be seen that the results of all three schemes compare well with those in literature (Braza et al., 1986; Liu et al., 1998; Ding et al., 2004; Shu et al., 2014). Figure 3.6 shows the temporal evolution of the lift and drag coefficients on the cylinder, in which the periodic behavior of flow pattern is shown clearly and the period of the lift coefficient is twice of the drag coefficient for the two cases. Figure 3.7 and Figure 3.8 show the streamlines and vorticity contours for the two unsteady cases, from which the famous Karman vortex street can be clearly seen.

3.4.4 Couette flow with a temperature gradient

Couette flow with a temperature gradient is a standard heat-transfer case which can provide a good test to show the viscous heat conduction. The schematic of this problem is shown in Figure 3.9. The bottom wall is fixed at a temperature T_0 . The top

plate is moving at a speed u and the temperature of the top wall is T_1 . The distance between these two infinite parallel flat plates is H . When the flow reaches a steady state, the temperature distribution can be obtained under the assumption of constant viscosity and heat conduction coefficients, which can be written as

$$T_0 = T_1, \quad T - T_0 = Pr \frac{u^2}{2c_p} \frac{y}{H} \left(1 - \frac{y}{H}\right) \quad (3.88)$$

$$T_0 \neq T_1, \quad \frac{T - T_0}{T_1 - T_0} = \frac{y}{H} + \frac{Pr \cdot Ec}{2} \frac{y}{H} \left(1 - \frac{y}{H}\right) \quad (3.89)$$

where c_p is the specific heat capacity at constant pressure, Pr is the Prandtl number, $Ec = u^2/c_p(T_1 - T_0)$ is the Eckert number. The distance between the two plates H is set as 1 in this case and a mesh size of 20×40 is used in the simulation. The Mach number is chosen as 0.2 for all the test cases. At the inlet and outlet, a periodic boundary condition is applied. Along the top and bottom walls, the isothermal no-slip boundary condition is imposed. This test problem can be viewed as a 1D problem in the y direction.

Firstly, the condition of $T_1 = T_0$ is considered. The test cases with different Prandtl numbers of $Pr = 0.7, 1.0$ and 2.0 are solved. Figure 3.10 shows the temperature profiles along the vertical central line obtained by three schemes and analytical solutions under different Prandtl numbers. It can be observed that all the present results match well with analytical solutions. In addition, the condition of $T_1 \neq T_0$ is also considered. Figure 3.11 presents the solutions of Eckert number $Ec = 40.0$ with different Prandtl numbers, in which the results also agree well with analytical solutions. It can be concluded that the technique of Prandtl number fix used in the present work can correctly consider the heat conduction term.

3.4.5 Shock wave-boundary layer interaction

We will use this case and the following two cases to test the performance of Scheme III. As mentioned above, Schemes I and II encounter numerical instability for the simulation of compressible flows. So, in this work, we will only use Scheme III to solve compressible flows. In this test case, an oblique shock wave at an angle of 32.6° interacts with a laminar boundary layer. The incident Mach number ahead of the shock wave is equal to 2 and the Reynolds number for the upstream flow is 2.96×10^5 . The dynamic viscosity is computed via Sutherland's law with $\gamma = 1.4$ and $Pr = 0.72$. Thus, the Prandtl number fix process should be used which has been described in the previous section. A non-uniform mesh of 100×100 is used and it is shown in Figure 3.12. The mesh spacing varies from 3.2×10^{-4} around $y = 0$ to 1.8×10^{-2} at the upper boundary in the y direction and it is uniform in the x direction. The oblique shock hits the boundary layer on the wall at $x = 1$. The pressure contours obtained from the current scheme and gas kinetic BGK scheme (Xu, 2001) in the whole computational domain are presented in Figure 3.13. The complex features are well captured in the separation zone. Figure 3.14 shows the pressure and skin friction coefficient distributions along the plate surface, where the experimental data (Hakkinen et al., 1959) and results of Xu (2001) are also included for comparison. For comparison of pressure distribution, the averaged value of difference between the results of present solver and those of Xu (2001) is 0.52% and the maximum difference is around 3%. On the other hand, for the comparison of skin friction, the present results match well with reference data (Xu, 2001) in the region of expansion waves and reattachment shock ($x \geq 1$). However, in the region of compression waves ($x < 1$), there are some small deviations with those of Xu (2001).

To compare the computational efficiency of the current GKFS with gas kinetic BGK scheme (Xu, 2001), the above non-uniform grid is used for both two schemes. Numerical simulations are carried out on a Dell Desktop (Intel Core i5 with 3.10GHz and 8 GB RAM). The CPU time versus iterative steps are depicted in Figure 3.15. In this figure, it shows that the current GKFS takes about 91.75 seconds while the gas kinetic BGK scheme needs about 160.63 seconds for 10000 time steps. The present scheme only takes about 57.1% of the CPU time of gas kinetic BGK scheme for this case, which shows the high efficiency of the present scheme.

3.4.6 Transonic laminar flows over a NACA0012 airfoil

In this case, transonic laminar flows over a NACA0012 airfoil are simulated. For the problem, the chord length of the airfoil is taken as 1, and the distance between the leading edge and the free stream boundary is chosen as 15 chord lengths. The outer boundary is located 21 chord lengths away from the leading edge. The far field boundary condition is imposed at the outer boundary, except for the cut-line, where a periodic boundary condition is applied. On the airfoil surface, the adiabatic no-slip wall condition is imposed. A 396×80 C-type grid is used for all simulations. There are 239 grid points on the airfoil surface and 80 grid points on the cut-line. The minimum mesh spacing normal to the wall boundary is chosen as 1.0×10^{-3} in units of chord length.

The pressure coefficient C_p and skin friction coefficient C_f along the wall will be computed to compare with the reference data. These two coefficients are defined as

$$C_p = \frac{p - p_\infty}{1/2 \rho_\infty U_\infty^2} \quad (3.90)$$

$$C_f = \frac{\tau_w}{1/2 \rho_\infty U_\infty^2} \quad (3.91)$$

Here, ρ_∞ is the fluid density, U_∞ is the free-stream velocity and τ_w is the local wall shear stress defined as

$$\tau_w = \mu \left. \frac{\partial u_\tau}{\partial n} \right|_{wall} \quad (3.92)$$

where u_τ is the tangential velocity and $\frac{\partial}{\partial n}$ is the derivative in the normal direction on the wall surface.

The case of Mach number $M_\infty = 0.8$ and attack angle of $\alpha = 10^\circ$ is considered. The Reynolds number based on the chord length and the free-stream conditions is $Re = 500$. This is the test case A3 of GAMM workshop in 1985 (Bristeau et al., 1987). Figure 3.16 shows the streamlines around the airfoil. There is a large separation region on the upper surface of the airfoil and the present scheme accurately resolves this complex flow feature. The comparisons of pressure coefficient and skin friction coefficient distributions on the airfoil surface are shown in Figure 3.17. Also included in the figure are the data given in literature (Jawahar and Kamath, 2000; Katz, 2009). Both the pressure coefficient C_p and the skin friction C_f agree well with the reference data.

3.4.7 Hypersonic flow around a circular cylinder

Numerical simulation of hypersonic flow, which is a big concern in the design of aerospace vehicles, is still a challenging problem in computational fluid dynamics. To

investigate the capability of the current flux solver for simulation of hypersonic flows with strong shock waves, the hypersonic flow around one half of a cylinder is simulated. The inflow condition for the air is taken from the experiment done by Wieting and Holden (1988), in which the free-stream Mach number is $M_\infty = 8.03$, the Reynolds number is $Re = 1.835 \times 10^5$, the free-stream temperature is $T_\infty = 124.94K$ and wall temperature is $T_w = 294.44K$. A computational mesh of 160×160 is used in the current simulations. The cell Reynolds number is chosen as $Re_{cell} = \rho_\infty U_\infty \Delta r / \mu_\infty = 1.835$, where Δr is the mesh spacing of the first cell in the normal direction next to the cylinder surface. The LU-SGS scheme (Yoon and Jameson, 1988) is applied to solve the resultant algebraic equations.

Figure 3.18 shows the pressure and temperature contours around the cylinder. It can be seen that there are no oscillations or “carbuncle phenomenon” in the present simulation. Figure 3.19 presents the pressure and heat flux distributions along the cylinder surface, which have been normalized by the pressure $p_0 = 0.9209$ and heat flux $q_0 = 0.003655$ at the stagnation point (Xu et al., 2005). The experimental data (Wieting and Holden, 1988) and results given by the multi-dimensional gas kinetic BGK scheme (Xu et al., 2005) are also included for comparison. It can be found that the pressure and heat flux distributions match well with those obtained from Xu et al. (2005). There is a slight deviation in the heat flux distribution at the stagnation point and this may be caused by different mesh sizes used in the respective simulations.

3.5 Concluding Remarks

In this chapter, the gas kinetic BGK scheme for N-S equations was first introduced. In this method, the BGK collision model is adopted in the gas evolution stage to obtain the numerical fluxes at the cell interface. As a consequence, the dissipation in the transport can be controlled by a real collision time. In contrast to conventional upwind schemes, the inviscid and viscous fluxes can be computed simultaneously from the solution of the Boltzmann equation with BGK approximation in the gas kinetic BGK scheme. According to the C-E expansion analysis of the BGK collision model, the gas kinetic BGK scheme provides an accurate N-S solution in the smooth region. At the same time, a stable and crisp shock transition can be generated with a delicate dissipative mechanism. However, the gas kinetic BGK scheme also suffers from several drawbacks such as complexity and low computational efficiency. A number of coefficients related to the physical space and phase space have to be calculated at every cell interface and each time step. This causes the evaluation of numerical flux using the gas kinetic BGK scheme be expansive. Moreover, it is an arduous task to derive the explicit expressions for the numerical fluxes.

In order to overcome these drawbacks of the gas kinetic BGK scheme and present an effective numerical algorithm, GKFS was proposed based on the C-E expansion analysis. GKFS is a finite volume solver and directly solves the governing equations reproduced by C-E theory. The conservative flow variables at the cell centers are updated by marching in time with fluxes at the cell interfaces. The key issue in GKFS is to evaluate the flux at the cell interface by local reconstruction of the Boltzmann solutions. A simple method was proposed to evaluate the gas distribution function at the cell interface and thus the numerical fluxes can be derived easily. GKFS removes

the drawbacks of the gas kinetic BGK scheme, such as complexity and inefficiency. In addition, the formulations of conservative flow variables and numerical fluxes at cell interfaces were explicitly given, which were the first published in the literature as far as we know. Three specific schemes of GKFS were proposed and validated by simulating both incompressible and compressible viscous flows. Firstly, the three schemes were applied to simulate incompressible flows, including decaying vortex flow, lid-driven cavity flow, flow over a cylinder and Couette flow with a temperature gradient. Numerical results showed that all the three schemes roughly have second order of accuracy in space. Among them, Scheme II was more attractive due to its high efficiency. For simulation of compressible flows such as shock-boundary layer interaction, laminar flow over a NACA0012 airfoil and hypersonic flow over a cylinder, Schemes I and II encountered numerical instability. In contrast, Scheme III performed equally well for simulation of both incompressible and compressible flows. The reason could be that the streaming time step δt is not involved in Scheme III. Numerical results also showed that GKFS only takes about 57.1% of the CPU time of gas kinetic BGK scheme for shock boundary layer interaction on the same non-uniform grids. It can be seen from these simulations that GKFS can be effectively applied for simulation of both incompressible and compressible viscous flows.

In the next chapter, we will further extend the GKFS to simulate three-dimensional (3D) viscous flows. A truly 3D gas kinetic flux solver will be proposed for simulation of incompressible and compressible flows.

Table 3.1 Locations of primary vortex centers at different Reynolds numbers

<i>Re</i>	Ghia et al. (1982)	Scheme I	Scheme II	Scheme III
100	(0.6172, 0.7344)	(0.6182, 0.7425)	(0.6160, 0.7393)	(0.6199, 0.7480)
400	(0.5547, 0.6055)	(0.5568, 0.6066)	(0.5573, 0.6072)	(0.5568, 0.6066)
1000	(0.5313, 0.5625)	(0.5349, 0.5675)	(0.5336, 0.5677)	(0.5336, 0.5675)
5000	(0.5117, 0.5352)	(0.5176, 0.5361)	(0.5168, 0.5365)	(0.5168, 0.5365)
10000	(0.5117, 0.5333)	(0.5150, 0.5371)	(0.5145, 0.5321)	(0.5145, 0.5321)

Table 3.2 Comparison of computational time (seconds)

<i>Re</i>	Scheme I	Scheme II	Scheme III
100	33.48	21.42	34.97
400	83.55	54.02	89.92
1000	397.24	229.13	436.15
5000	4651.73	2570.69	4732.84
10000	7720.24	3913.15	7040.23

Table 3.3 Comparison of drag coefficient, recirculation lengths and separation angles for a steady flow past a circular cylinder at different Reynolds numbers

<i>Re</i>	References	<i>C_d</i>	<i>L/D</i>	<i>θ_s</i>
20	Dennis and Chang (1970)	2.05	0.94	43.7
	Shukla et al. (2007)	2.07	0.92	43.3
	Ding et al. (2007)	2.14	0.94	43.8
	Shu et al. (2014)	2.062	0.935	42.94
	Scheme I	2.064	0.927	42.94
	Scheme II	2.063	0.927	43.09
	Scheme III	2.065	0.933	43.33
40	Dennis and Chang (1970)	1.52	2.35	53.8
	Shukla et al. (2007)	1.55	2.34	52.7
	Ding et al. (2007)	1.58	2.32	52.8
	Shu et al. (2014)	1.53	2.240	52.69
	Scheme I	1.544	2.225	52.77
	Scheme II	1.544	2.235	52.95
	Scheme III	1.546	2.252	53.33

Table 3.4 Comparison of dynamic parameters for an unsteady flow past a circular cylinder at different Reynolds numbers

<i>Re</i>	References	C_l	C_d	St
100	Braza et al. (1986)	± 0.30	1.28 ± 0.02	0.16
	Liu et al. (1998)	± 0.339	1.350 ± 0.012	0.164
	Ding et al. (2004)	± 0.28	1.325 ± 0.008	0.164
	Shu et al. (2014)	± 0.33	1.334 ± 0.009	0.164
	Scheme I	± 0.328	1.326 ± 0.009	0.164
	Scheme II	± 0.330	1.329 ± 0.009	0.164
	Scheme III	± 0.333	1.333 ± 0.0093	0.166
200	Braza et al. (1986)	± 0.78	1.38 ± 0.07	0.19
	Liu et al. (1998)	± 0.69	1.31 ± 0.049	0.192
	Ding et al. (2004)	± 0.60	1.327 ± 0.045	0.196
	Shu et al. (2014)	± 0.69	1.338 ± 0.045	0.197
	Scheme I	± 0.682	1.323 ± 0.044	0.194
	Scheme II	± 0.687	1.331 ± 0.044	0.196
	Scheme III	± 0.693	1.335 ± 0.045	0.196

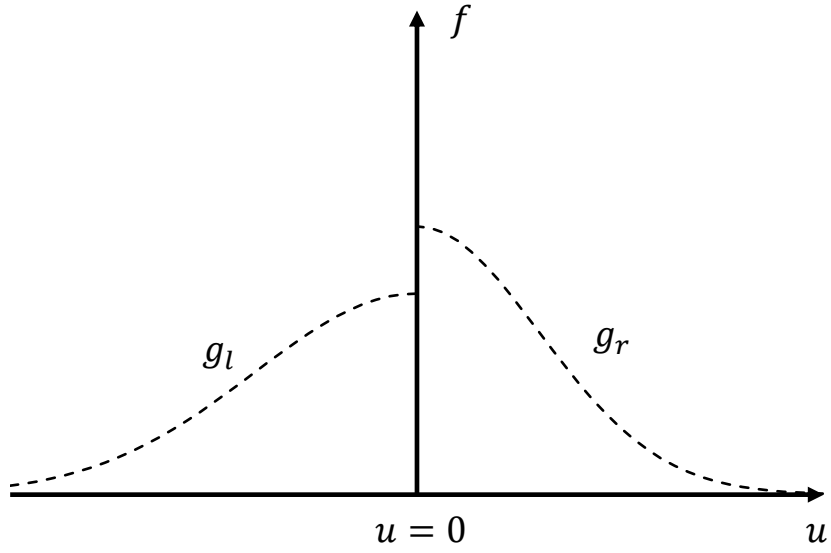


Figure 3.1 Schematic diagram of initial distribution function at the interface

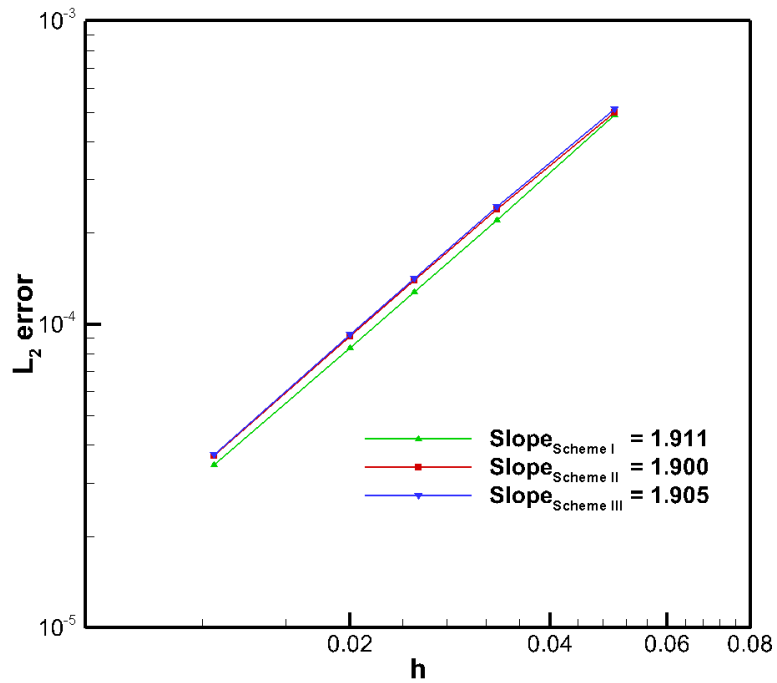
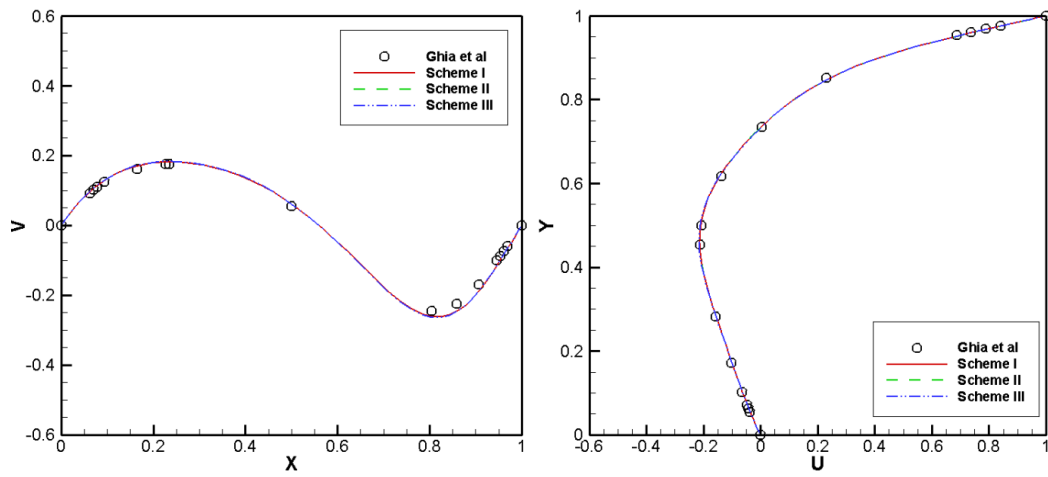
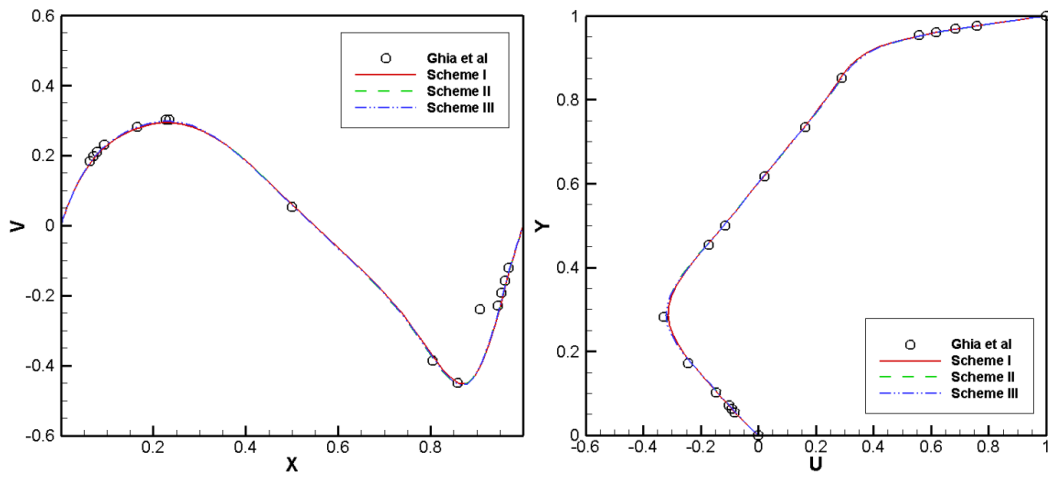


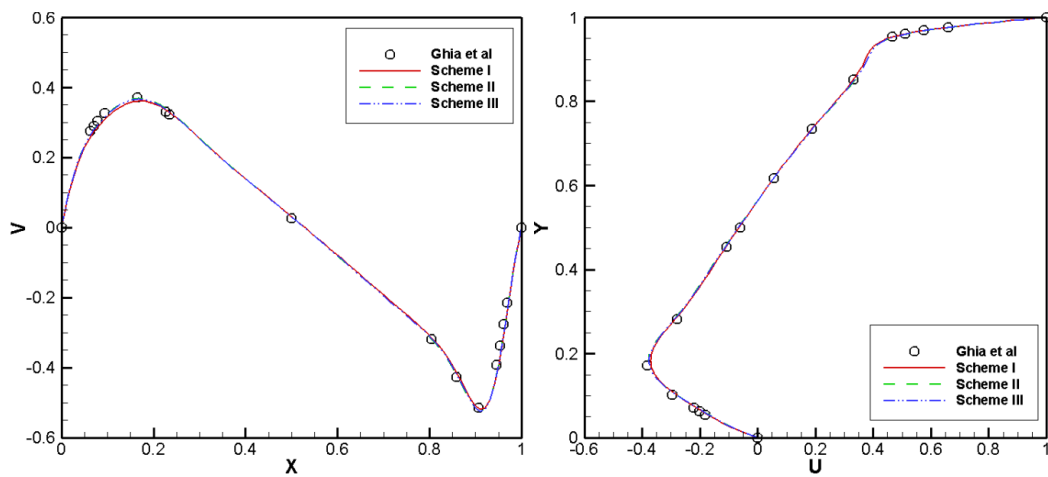
Figure 3.2 L_2 norm of relative error of u versus h for the decaying vortex flow



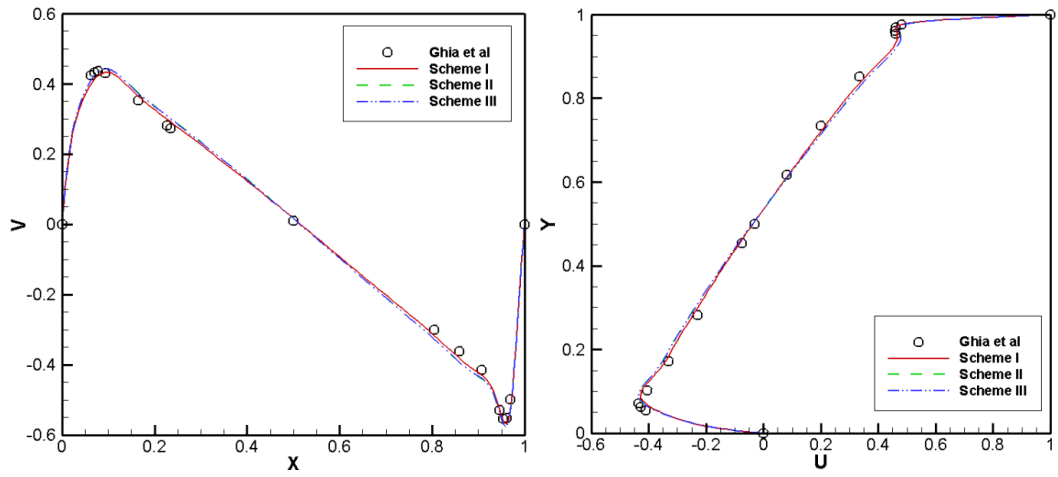
$Re=100, 61 \times 61$



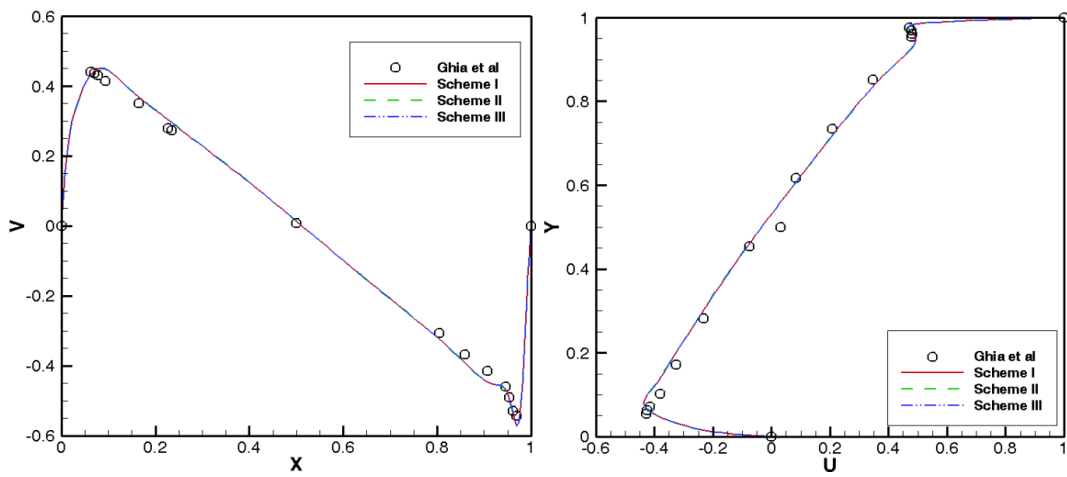
$Re=400, 61 \times 61$



$Re=1000, 81 \times 81$



$Re = 5000, 121 \times 121$



$Re = 10000, 121 \times 121$

Figure 3.3 U and V velocity profiles along horizontal and vertical central lines for a lid-driven cavity flow at various Reynolds numbers

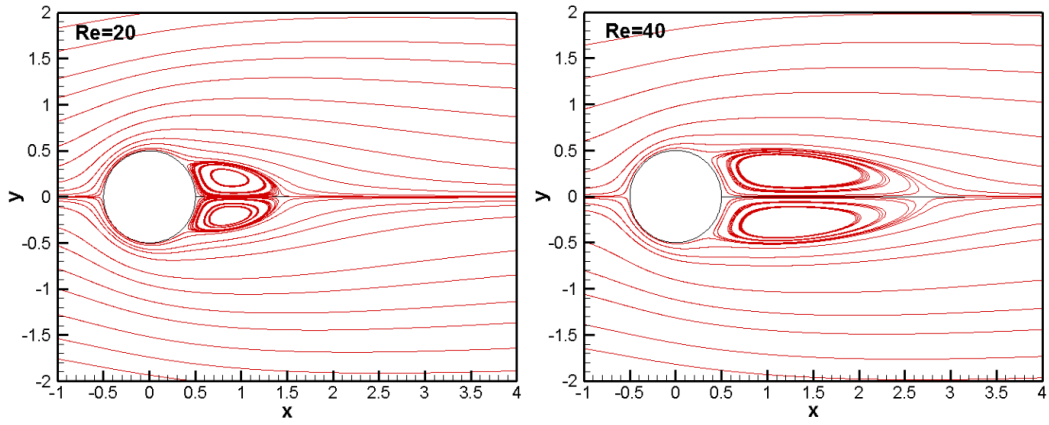


Figure 3.4 Streamlines for a steady flow past a circular cylinder of two steady cases

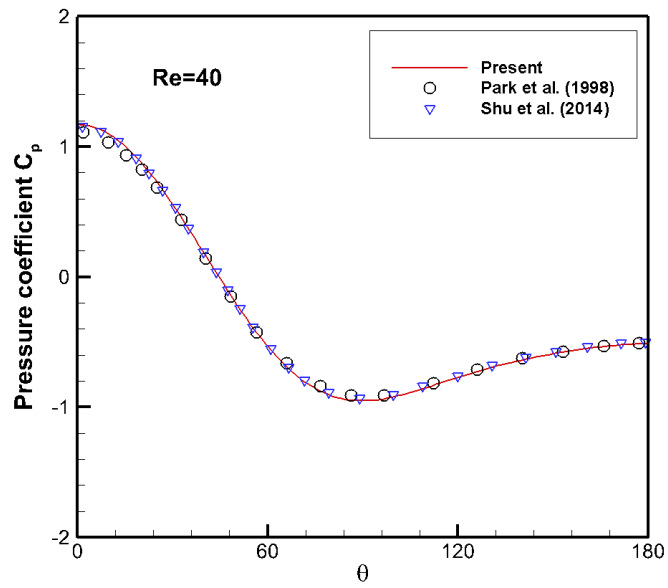


Figure 3.5 Pressure coefficient distribution along the cylinder surface at $Re=40$

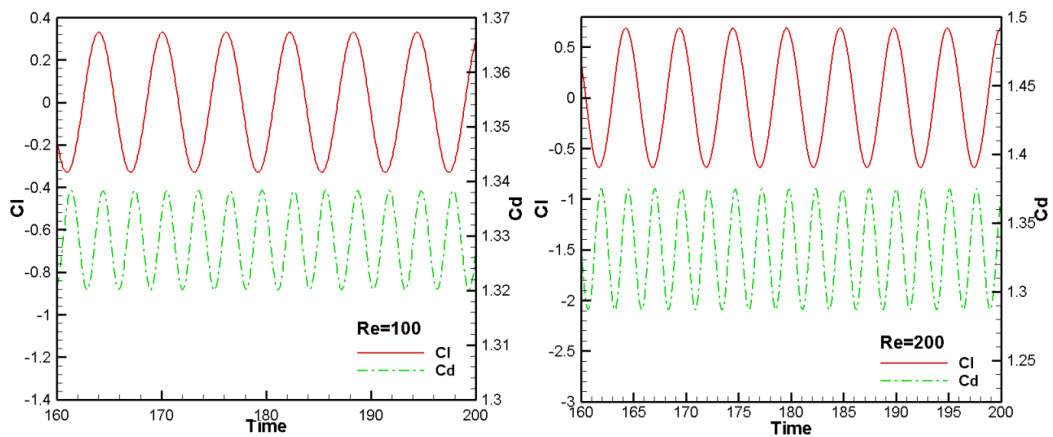


Figure 3.6 Evolution of the lift and drag coefficients for a flow past a circular cylinder

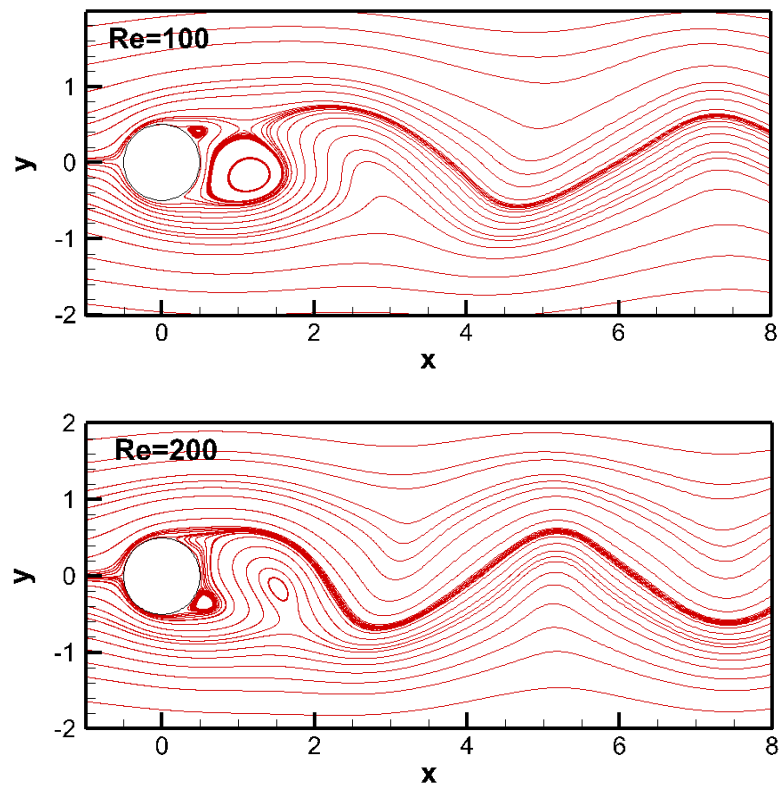
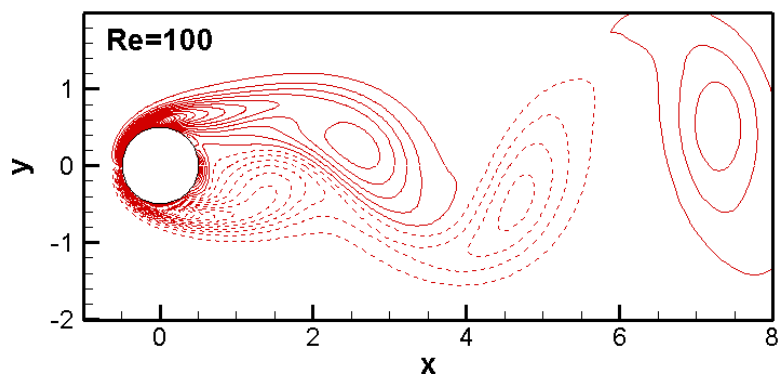


Figure 3.7 Streamlines for flow past a circular cylinder of two unsteady cases



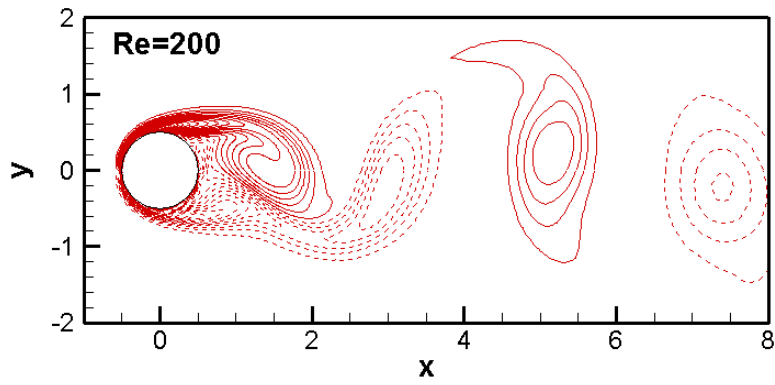


Figure 3.8 Vorticity contours for a flow past a circular cylinder of two unsteady cases

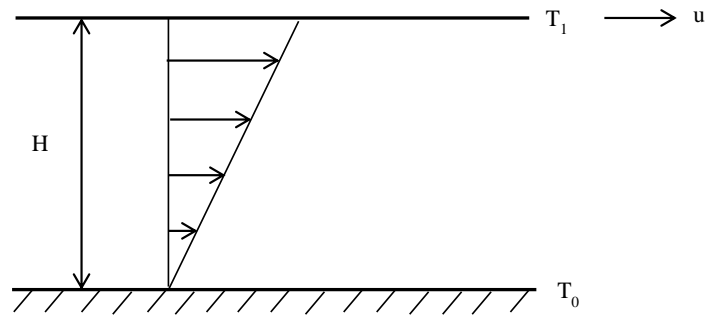


Figure 3.9 Schematic of Couette flow with a temperature gradient

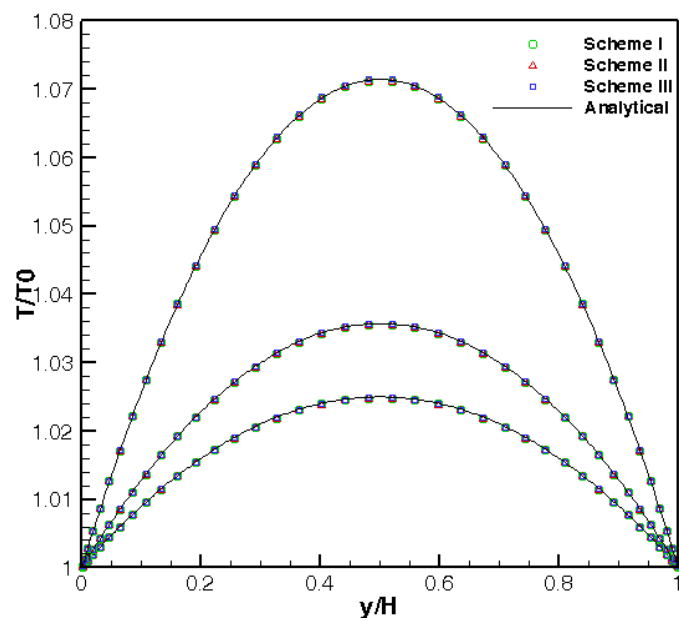


Figure 3.10 Comparison of temperature profile for various Prandtl numbers when

$$T_1 = T_0$$

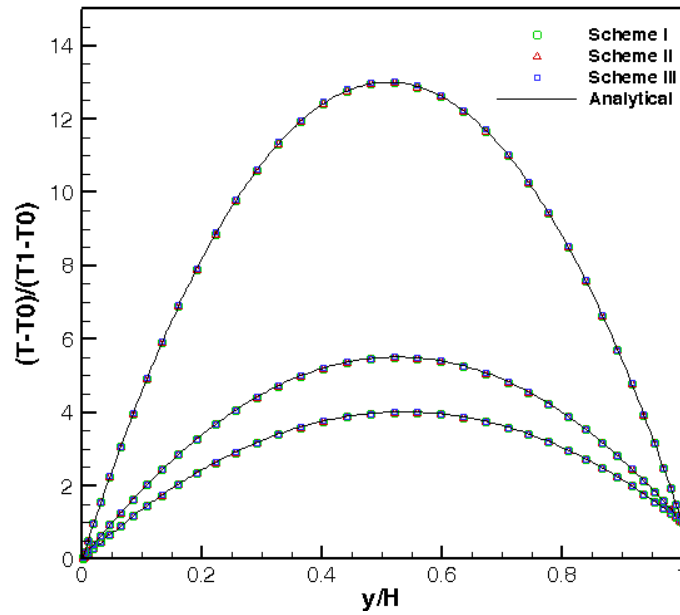


Figure 3.11 Comparison of temperature profile for various Prandtl numbers when $T_1 \neq T_0$

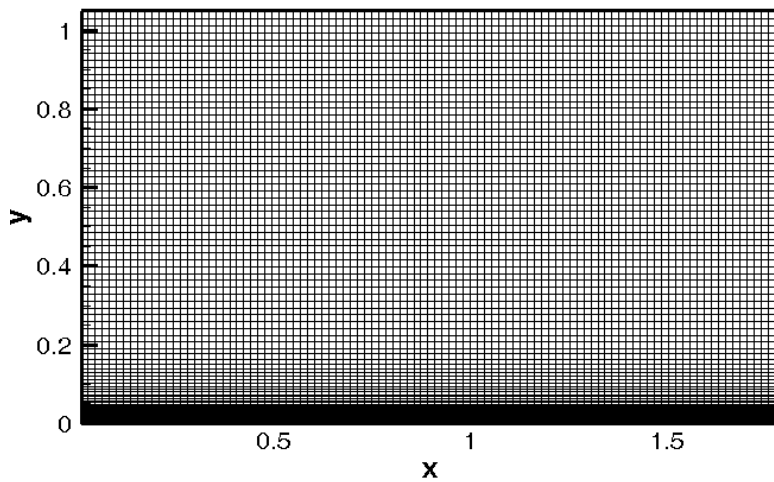


Figure 3.12 Computational mesh for shock-boundary layer interaction

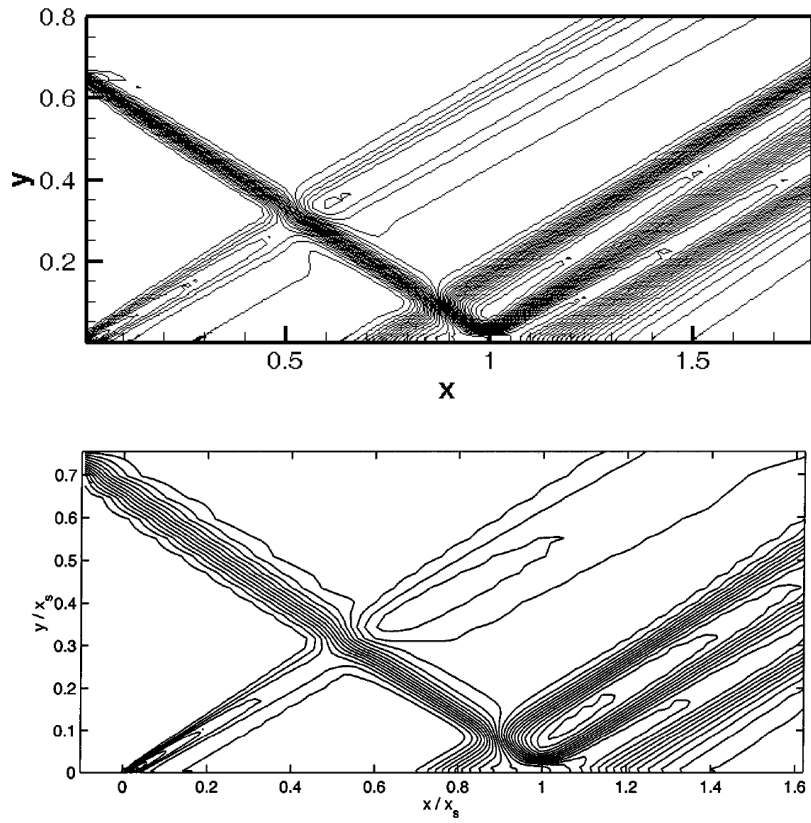
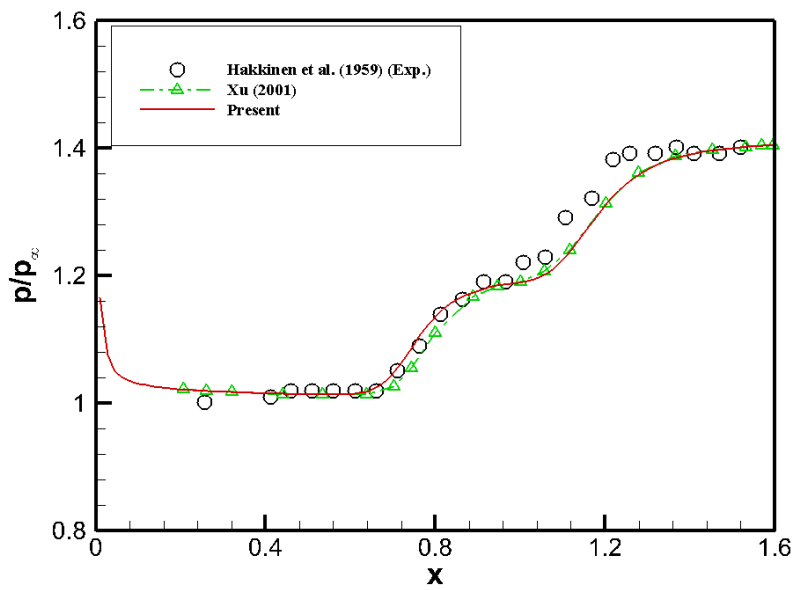


Figure 3.13 Pressure contours of shock-boundary layer interaction given from present scheme (Upper) and gas kinetic BGK scheme (Xu, 2001) (Lower)



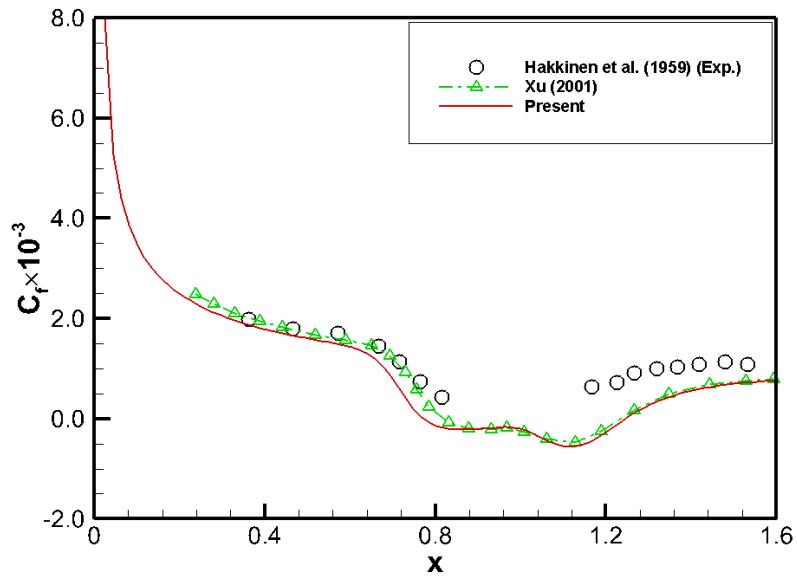


Figure 3.14 Pressure (Upper) and skin friction (Lower) distributions along the flat plate of shock-boundary layer interaction

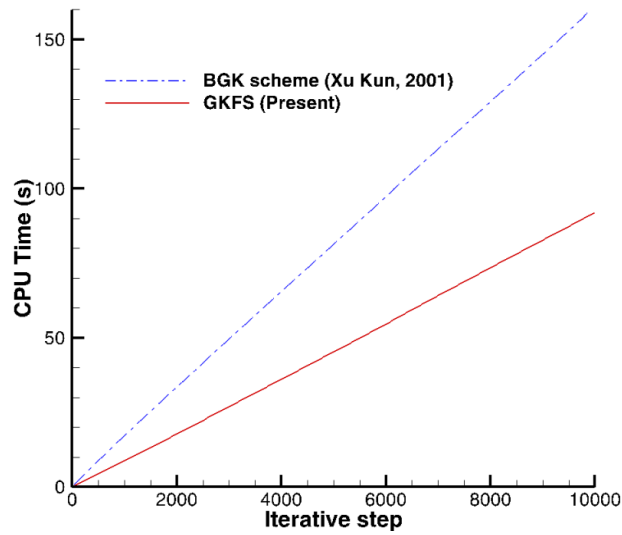


Figure 3.15 Comparison of CPU time between GKFS and gas kinetic BGK scheme

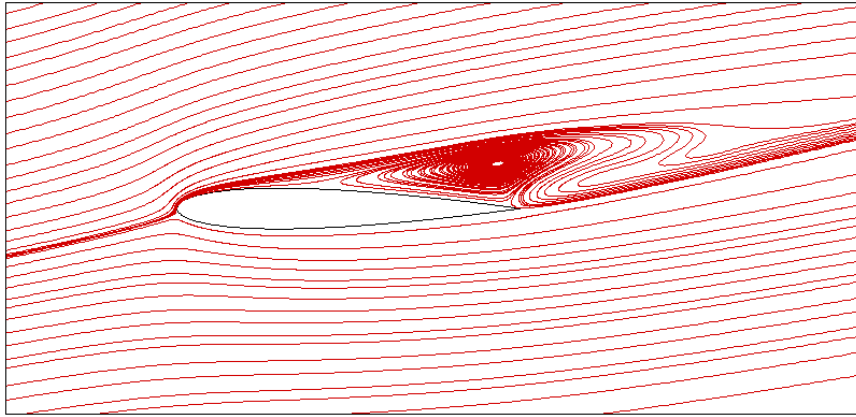


Figure 3.16 Streamlines of transonic laminar flow over a NACA0012 airfoil

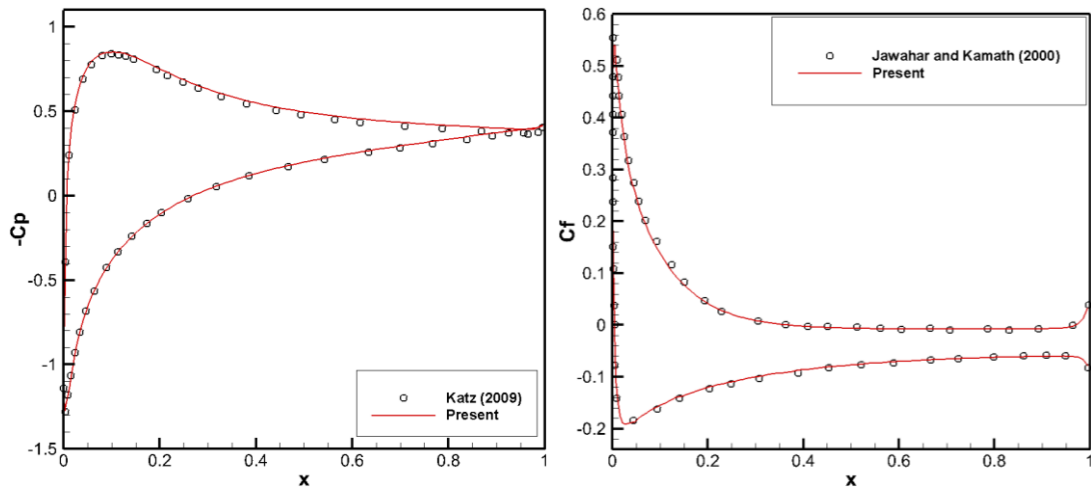


Figure 3.17 Comparison of pressure coefficient (Left) and skin friction (Right) distributions on the airfoil surface

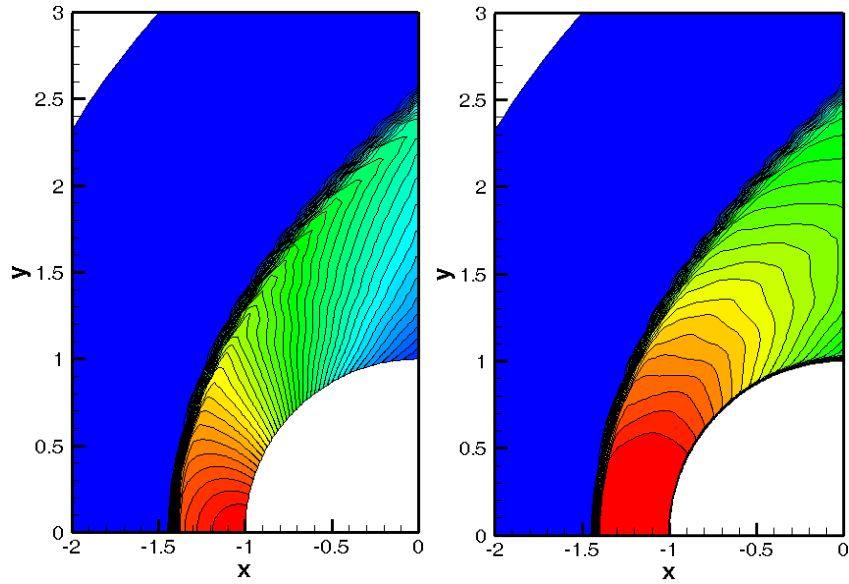


Figure 3.18 Pressure (Left) and temperature (Right) contours of hypersonic flows over one half of a cylinder

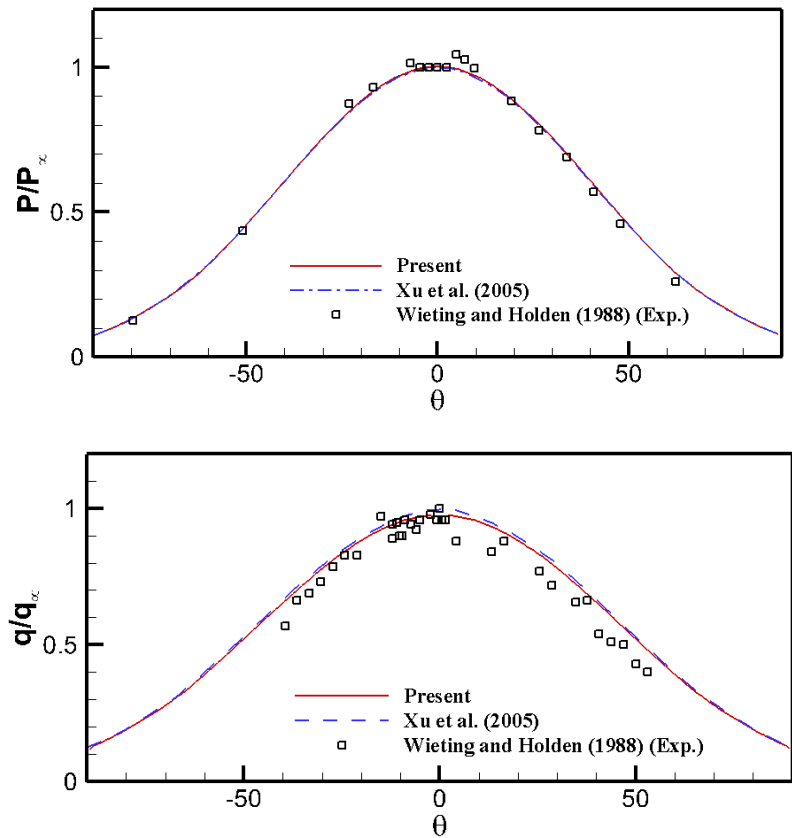


Figure 3.19 Pressure (Upper) and heat flux (Lower) along the cylindrical surface for hypersonic flow around one half of a cylinder

Chapter 4

Development of a Three-dimensional Gas kinetic Flux Solver for Simulation of Viscous Flows with Explicit formulations of Conservative Flow Variables and Numerical Fluxes

In the previous chapter, the gas kinetic flux solver (GKFS) has been successfully developed and applied to simulate 2D incompressible and compressible flows. The novel solver not only keeps the intrinsic advantages of gas kinetic scheme but also effectively eliminates its drawbacks. In this chapter, we will extend the GKFS to solve three-dimensional (3D) flows. A truly 3D GKFS for simulation of incompressible and compressible viscous flows will be presented. The 3D GKFS is a finite volume solver for the direct update of the macroscopic variables at cell centers. The fluxes of the 3D GKFS are evaluated at each cell interface by local reconstruction of the 3D Boltzmann equation with BGK approximation. In the development, a local coordinate transformation is introduced to transform the velocities in the Cartesian coordinate system to the local normal and tangential directions at each cell interface. In this way, all the interfaces can be treated in the same manner. Different from the conventional gas kinetic scheme, the non-equilibrium distribution function is calculated by the difference of equilibrium distribution functions between the cell interface and its surrounding points. As a result, the distribution function at the cell interface can be computed in a simple way. It is indicated that the present work is the first time to give explicit formulations for evaluating the conservative flow variables

and numerical fluxes for 3D viscous flow problems. To validate the developed scheme, both incompressible and compressible viscous test cases will be solved, including 3D driven cavity flow, incompressible flow past a stationary sphere, flow around an ONERA M6 wing, turbulent flow over the DPW III wing and DLR-F6 wing-body configuration.

4.1 Boltzmann Equation, Maxwellian Distribution Function and Navier-Stokes Equations

4.1.1 Boltzmann equation and conservative forms of moments for Maxwellian distribution function

With Bhatnagar-Gross-Krook (BGK) collision model (Bhatnagar et al., 1954), the continuum Boltzmann equation in the 3D Cartesian coordinate system can be written as

$$\frac{\partial f}{\partial t} + u \frac{\partial f}{\partial x} + v \frac{\partial f}{\partial y} + w \frac{\partial f}{\partial z} = \frac{g - f}{\tau} \quad (4.1)$$

where f is the real particle distribution function and g is the equilibrium particle distribution function. τ is the collision time, which is determined by dynamic viscosity and pressure. The right side of the equation is the collision term which alters the distribution function from f to g within a collision time scale τ . Both f and g are functions of space (x, y, z) , time (t) , particle velocity (u, v, w) and internal energy ξ . The internal degree of freedom K in ξ is determined by the space dimension and the ratio of specific heats with the relation $K + D = 2/(\gamma - 1)$, where D is the abbreviation of the dimension ($D = 3$ in three dimension) and γ is the specific heat ratio. The equilibrium state g of Maxwellian distribution is

$$g = \rho \left(\frac{\lambda}{\pi} \right)^{\frac{K+3}{2}} e^{-\lambda((u-U)^2 + (v-V)^2 + (w-W)^2 + \xi^2)} \quad (4.2)$$

where ρ is the density of mean flow; $\mathbf{U} = (U, V, W)$ are the macroscopic velocity in the x -, y - and z -directions, respectively; $\lambda = m/(2kT) = 1/(2RT)$, where m is the molecular mass, k is the Boltzmann constant, R is the gas constant and T is the temperature. In the equilibrium state, the internal energy ξ^2 is the abbreviation of $\xi^2 = \xi_1^2 + \xi_2^2 + \dots + \xi_K^2$.

With the Maxwellian distribution function in Eq. (4.2), the following 7 conservation forms of moments should be satisfied, which are used to recover N-S equations by Eq. (4.1) through C-E expansion analysis:

$$\int g d\Xi = \rho \quad (4.3)$$

$$\int g u_\alpha d\Xi = \rho U_\alpha \quad (4.4)$$

$$\int g \left(u_\alpha u_\alpha + \sum_{j=1}^K \xi_j^2 \right) d\Xi = \rho (U_\alpha U_\alpha + bRT) \quad (4.5)$$

$$\int g u_\alpha u_\beta d\Xi = \rho U_\alpha U_\beta + p \delta_{\alpha\beta} \quad (4.6)$$

$$\int g \left(u_\alpha u_\alpha + \sum_{j=1}^K \xi_j^2 \right) u_\beta d\Xi = \rho [U_\alpha U_\alpha + (b+2)RT] U_\beta \quad (4.7)$$

$$\int g u_\alpha u_\beta u_\chi d\Xi = p (U_\alpha \delta_{\beta\chi} + U_\beta \delta_{\alpha\chi} + U_\chi \delta_{\alpha\beta}) + \rho U_\alpha U_\beta U_\chi \quad (4.8)$$

$$\begin{aligned} & \int g \left(u_\alpha u_\alpha + \sum_{j=1}^K \xi_j^2 \right) u_\beta u_\chi d\Xi \\ &= \rho \left\{ U_\alpha U_\alpha U_\beta U_\chi + [(b+4)U_\beta U_\chi + U_\alpha U_\alpha \delta_{\beta\chi}] RT + (b+2)R^2 T^2 \delta_{\beta\chi} \right\} \end{aligned} \quad (4.9)$$

where $u_\alpha, u_\beta, u_\gamma$ and $U_\alpha, U_\beta, U_\gamma$ are the phase velocity and macroscopic flow velocity in the α -, β - and γ - direction. p is the pressure and $b = K + D$ represents the total degree of freedoms of molecules. $d\Xi = du_\alpha du_\beta du_\gamma d\xi_1 d\xi_2 \cdots d\xi_K$ is the volume element in the phase space. The integral domain for $u_\alpha, u_\beta, u_\gamma, \xi_1, \xi_2, \dots, \xi_K$ is from $-\infty$ to $+\infty$. Eqs. (4.3)-(4.5) are applied to recover the fluid density, momentum and energy, respectively. Eqs. (4.6) and (4.7) are used to recover convective fluxes of momentum equations and energy equation. Eqs. (4.8) and (4.9) are to recover diffusive fluxes of momentum equations and energy equation.

4.1.2 Macroscopic governing equations discretized by finite volume method

In this chapter, the 3D N-S equations are solved using the finite volume discretization with the conservative flow variables defined at cell centers, which can be written as

$$\frac{d\mathbf{W}}{dt} + \frac{1}{\Omega} \sum_{i=1}^N \mathbf{F}_i S_i = 0 \quad (4.10)$$

where \mathbf{W} is the vector of conservative flow variables, Ω and N are the volume and number of interfaces of the control volume, respectively, \mathbf{F}_i and S_i are the flux vector and length of interface i . It should be noted that the numerical flux \mathbf{F}_i are reconstructed locally at cell interfaces from the conservative flow variables \mathbf{W} at the cell centers. In the gas kinetic scheme, the connection between the distribution function f and the conservative flow variables is

$$\mathbf{W} = (\rho, \rho U, \rho V, \rho W, \rho E)^T = \int f \varphi_\alpha d\Xi \quad (4.11)$$

where $E = \frac{1}{2}(U^2 + V^2 + W^2 + bRT)$ and $d\Xi = dudvdwd\xi$ is the volume element for 3D computations. φ_α is the moment given by

$$\varphi_\alpha = \left(1, u, v, w, \frac{1}{2}(u^2 + v^2 + w^2 + \xi^2) \right)^T \quad (4.12)$$

With the compatibility condition,

$$\int \frac{g-f}{\tau} \varphi_\alpha d\Xi = 0 \quad (4.13)$$

Eq. (4.11) is equivalent to

$$\mathbf{W} = (\rho, \rho U, \rho V, \rho W, \rho E)^T = \int g \varphi_\alpha d\Xi \quad (4.14)$$

The above equation shows that the non-equilibrium distribution function has no contribution to the calculation of conservative flow variables.

After evaluation of conservative flow variables, the flux vector \mathbf{F} can also be obtained from the distribution function

$$\mathbf{F} = \int u f \varphi_\alpha d\Xi \quad (4.15)$$

It should be noted that Eq. (4.15) is the flux vector of x -direction in the Cartesian coordinate system. In the practical application such as curved boundary problems, we need to calculate the numerical flux in the normal direction of the interface \mathbf{F}_n

$$\mathbf{F}_n = (F_1, F_2, F_3, F_4, F_5)^T = \int u' f \varphi_\alpha d\Xi \quad (4.16)$$

where u' is the phase velocity in the normal direction of the interface. Suppose that

$\mathbf{n}_1 = n_{1x}, n_{1y}, n_{1z}$ is the unit vector in the normal direction of the interface and

$\mathbf{n}_2 = n_{2x}, n_{2y}, n_{2z}$, $\mathbf{n}_3 = n_{3x}, n_{3y}, n_{3z}$ are the unit vectors in the tangential directions.

Then, the relationship between the phase velocities in the normal and tangential directions (u', v', w') and the phase velocities in the Cartesian coordinate system

(u, v, w) are

$$u' = un_{1x} + vn_{1y} + wn_{1z}, \quad v' = un_{2x} + vn_{2y} + wn_{2z}, \quad w' = un_{3x} + vn_{3y} + wn_{3z} \quad (4.17)$$

and similarly

$$u = u'n_{1x} + v'n_{2x} + w'n_{3x}, \quad v = u'n_{1y} + v'n_{2y} + w'n_{3y}, \quad w = u'n_{1z} + v'n_{2z} + w'n_{3z} \quad (4.18)$$

Substituting Eq. (4.18) into Eq. (4.12), we have

$$\varphi_\alpha = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & n_{1x} & n_{2x} & n_{3x} & 0 \\ 0 & n_{1y} & n_{2y} & n_{3y} & 0 \\ 0 & n_{1z} & n_{2z} & n_{3z} & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \left(1, u', v', w', \frac{1}{2}(u'^2 + v'^2 + w'^2 + \xi^2) \right)^T \quad (4.19)$$

With the definition of a new moment

$$\varphi_\alpha^* = \left(1, u', v', w', \frac{1}{2}(u'^2 + v'^2 + w'^2 + \xi^2) \right)^T \quad (4.20)$$

and its corresponding flux vector

$$\mathbf{F}_n^* = (F_1^*, F_2^*, F_3^*, F_4^*, F_5^*)^T = \int u' f \varphi_\alpha^* d\Xi \quad (4.21)$$

the real flux vector \mathbf{F}_n can be obtained by substituting Eq. (4.19) into Eq. (4.16) and using Eq. (4.21)

$$\mathbf{F}_n = \int u' f \varphi_\alpha d\Xi = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & n_{1x} & n_{2x} & n_{3x} & 0 \\ 0 & n_{1y} & n_{2y} & n_{3y} & 0 \\ 0 & n_{1z} & n_{2z} & n_{3z} & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \mathbf{F}_n^* \quad (4.22)$$

The above equation (4.22) shows that the calculation of \mathbf{F}_n is equivalent to the evaluation of \mathbf{F}_n^* and the key issue is to obtain the gas distribution function f . In the next subsection, a 3D GKFS will be introduced to evaluate the gas distribution function f .

4.2 Three-dimensional Gas kinetic Flux Solver

As the flux vector \mathbf{F}_n^* is evaluated at local interfaces, a local coordinate system is applied in the derivation of distribution function f . It is known that the distribution function f can be separated into two parts, the equilibrium part f^{eq} and the non-equilibrium part f^{neq} with the relationship of

$$f = f^{eq} + f^{neq} \quad (4.23)$$

Here, the equilibrium part f^{eq} equals to

$$f^{eq} = g \quad (4.24)$$

With the C-E analysis, the non-equilibrium distribution function can be approximated as

$$f^{neq} = -\tau \left(\frac{\partial}{\partial t} + \mathbf{u} \cdot \nabla \right) f^{eq} = -\tau \left(\frac{\partial}{\partial t} + \mathbf{u} \cdot \nabla \right) g \quad (4.25)$$

Therefore, the gas distribution function in the local BGK equation becomes

$$f = f^{eq} + f^{neq} = g - \tau \left(\frac{\partial g}{\partial t} + u' \frac{\partial g}{\partial n_1} + v' \frac{\partial g}{\partial n_2} + w' \frac{\partial g}{\partial n_3} \right) \quad (4.26)$$

By applying the Taylor series expansion in time and physical space, the above equation can be simplified to

$$f(\mathbf{0}, t + \delta t) = g(\mathbf{0}, t + \delta t) - \frac{\tau}{\delta t} \left[g(\mathbf{0}, t + \delta t) - g(-\mathbf{u}'\delta t, t) \right] \quad (4.27)$$

where $f(\mathbf{0}, t + \delta t)$ is the gas distribution function at local interface; $g(\mathbf{0}, t + \delta t)$ and $g(-\mathbf{u}'\delta t, t)$ are the equilibrium distribution functions at local interface and its surrounding points, respectively. δt is the streaming time step. From Eq. (4.27), it can be seen that the non-equilibrium distribution f^{neq} is calculated by the difference

of equilibrium distribution functions between the interface and its surrounding points, which makes the current GKFS be much more straightforward.

In the present work, the conservative flow variables \mathbf{W}^* in Eq. (4.10) are defined at cell centers. With the conservative flow variables, the numerical flux in the normal direction of each cell interface \mathbf{F}_n^* should be evaluated first in order to solve Eq. (4.10) by marching in time. Suppose that the conservative flow variables at cell centers and their first order derivatives are already known, the conservative flow variables at left and right sides of an interface can be easily given by interpolation. Then, the equilibrium distribution functions at these two sides of the interface can be given via Eq. (4.2). After that, the second order approximation of $g(-\mathbf{u}'\delta t, t)$ at a time level t can be written as

$$g(-\mathbf{u}'\delta t, t) = \begin{cases} g_l - \frac{\partial g_l}{\partial n_1} u' \delta t - \frac{\partial g_l}{\partial n_2} v' \delta t - \frac{\partial g_l}{\partial n_3} w' \delta t, & u' \geq 0 \\ g_r - \frac{\partial g_r}{\partial n_1} u' \delta t - \frac{\partial g_r}{\partial n_2} v' \delta t - \frac{\partial g_r}{\partial n_3} w' \delta t, & u' < 0 \end{cases} \quad (4.28)$$

where g_l and g_r are the equilibrium distribution functions at left and right sides of the interface, respectively. Note that in Eq. (4.28), the equilibrium distribution functions at two sides of the interface are not necessarily the same, which means that a possible discontinuity has been taken into account in the form. By substituting Eq. (4.28) into Eq. (4.27), we have

$$\begin{aligned}
f(\boldsymbol{\theta}, t + \delta t) &= g(\boldsymbol{\theta}, t + \delta t) - \frac{\tau}{\delta t} \left[g(\boldsymbol{\theta}, t + \delta t) \right. \\
&\quad - \left(g_l - \frac{\partial g_l}{\partial n_1} u' \delta t - \frac{\partial g_l}{\partial n_2} v' \delta t - \frac{\partial g_l}{\partial n_3} w' \delta t \right) \\
&\quad \left. - \left(g_r - \frac{\partial g_r}{\partial n_1} u' \delta t - \frac{\partial g_r}{\partial n_2} v' \delta t - \frac{\partial g_r}{\partial n_3} w' \delta t \right) \right] \\
&= g(\boldsymbol{\theta}, t + \delta t) - \frac{\tau}{\delta t} \left[g(\boldsymbol{\theta}, t + \delta t) - g_l - g_r \right] \\
&\quad - \tau \left[u' \frac{\partial}{\partial n_1} (g_l + g_r) + v' \frac{\partial}{\partial n_2} (g_l + g_r) + w' \frac{\partial}{\partial n_3} (g_l + g_r) \right]
\end{aligned} \tag{4.29}$$

The above equation shows that the full information of distribution function at the interface can be decided once we have the equilibrium distribution function at the cell interface and its surrounding points.

4.2.1 Evaluation of conservative flow variables \mathbf{W}^* at cell interface

It is known that the non-equilibrium distribution has no influence on the computation of conservative flow variables, and thus Eq. (4.14) can be adopted to calculate the conservative flow variables \mathbf{W}^* at local interface

$$\mathbf{W}^* = (\rho, \rho U', \rho V', \rho W', \rho E)^T = \int g \varphi_\alpha^* d\Xi \tag{4.30}$$

Substituting Eq. (4.27) and Eq. (4.28) into Eq. (4.30), we have

$$\begin{aligned}
\mathbf{W}^* &= \int \varphi_\alpha^* g(\boldsymbol{\theta}, t + \delta t) d\Xi = \int \varphi_\alpha^* g(-\mathbf{u}' \delta t, t) d\Xi \\
&= \int \int_{u' \geq 0} \varphi_\alpha^* \left(g_l - \frac{\partial g_l}{\partial n_1} u' \delta t - \frac{\partial g_l}{\partial n_2} v' \delta t - \frac{\partial g_l}{\partial n_3} w' \delta t \right) d\Xi \\
&\quad + \int \int_{u' < 0} \varphi_\alpha^* \left(g_r - \frac{\partial g_r}{\partial n_1} u' \delta t - \frac{\partial g_r}{\partial n_2} v' \delta t - \frac{\partial g_r}{\partial n_3} w' \delta t \right) d\Xi
\end{aligned} \tag{4.31}$$

The above equation shows that the conservative flow variables at cell interface can be obtained by equilibrium distribution function of the surrounding points. However, it should be noted that the streaming time δt cannot be simply determined because the

velocities of particles in the phase space range from $-\infty$ to $+\infty$. In the work of Xu (2001), the conservative flow variables at cell interface are calculated by taking the limit $\delta t \rightarrow 0$

$$\mathbf{W}^* = \int \int_{u' \geq 0} \varphi_\alpha^* g_l d\Xi + \int \int_{u' < 0} \varphi_\alpha^* g_r d\Xi \quad (4.32)$$

The above equation means that the conservative flow variables at cell interface are simply computed by the reconstructed variables of left and right sides. Eq. (4.32) is adopted for the calculation of conservative flow variables on cell interface in this work. With parameters defined in Appendix A, the conservative flow variables \mathbf{W}^* at cell interface are given by

$$\rho = (\rho_l \cdot a'_l + \rho_r \cdot a'_r) \quad (4.33)$$

$$\rho U' = (\rho_l \cdot b'_l + \rho_r \cdot b'_r) \quad (4.34)$$

$$\rho V' = (\rho_l V'_l \cdot a'_l + \rho_r V'_r \cdot a'_r) \quad (4.35)$$

$$\rho W' = (\rho_l W'_l \cdot a'_l + \rho_r W'_r \cdot a'_r) \quad (4.36)$$

$$\begin{aligned} \rho E = & \frac{1}{2} \rho_l \left[c'_l + (V_l'^2 + W_l'^2 + (b-1)RT_l) \cdot a'_l \right] \\ & + \frac{1}{2} \rho_r \left[c'_r + (V_r'^2 + W_r'^2 + (b-1)RT_r) \cdot a'_r \right] \end{aligned} \quad (4.37)$$

where “ \cdot_l ” and “ \cdot_r ” (“ \cdot ” stands for any variable) denote the variables at the left and right side of interface, respectively.

4.2.2 Evaluation of numerical fluxes F_n^* at cell interface

As soon as the conservative flow variables at local interface \mathbf{W}^* are obtained, the equilibrium distribution function $g(\boldsymbol{\theta}, t + \delta t)$ can be known by Eq. (4.2). Then the numerical flux across the cell interface can be calculated via Eq. (4.29)

$$\begin{aligned}
F_n^* &= \int u' \varphi_\alpha^* f(\mathbf{0}, t + \delta t) d\Xi \\
&= \int u' \varphi_\alpha^* g(\mathbf{0}, t + \delta t) d\Xi - \frac{\tau}{\delta t} \left[\int u' \varphi_\alpha^* g(\mathbf{0}, t + \delta t) d\Xi \right. \\
&\quad \left. - \int \int_{u' \geq 0} u' \varphi_\alpha^* g_l d\Xi - \int \int_{u' < 0} u' \varphi_\alpha^* g_r d\Xi \right] \\
&\quad - \tau \left[\frac{\partial}{\partial n_1} \int \left(\int_{u' \geq 0} u'^2 \varphi_\alpha^* g_l + \int_{u' < 0} u'^2 \varphi_\alpha^* g_r \right) d\Xi \right. \\
&\quad \left. + \frac{\partial}{\partial n_2} \int \left(\int_{u' \geq 0} u' v' \varphi_\alpha^* g_l + \int_{u' < 0} u' v' \varphi_\alpha^* g_r \right) d\Xi \right. \\
&\quad \left. + \frac{\partial}{\partial n_3} \int \left(\int_{u' \geq 0} u' w' \varphi_\alpha^* g_l + \int_{u' < 0} u' w' \varphi_\alpha^* g_r \right) d\Xi \right] \tag{4.38}
\end{aligned}$$

As has been discussed above, the streaming time δt in the right side of Eq. (4.38) cannot be simply determined. Therefore, the above equation cannot be used to calculate the numerical flux at cell interface directly. Note that $g(\mathbf{0}, t + \delta t)$ is the equilibrium distribution function at the interface and time level $t + \delta t$, and g_l, g_r are the distribution functions at the left and right sides of the interface and the time level t .

In fact, the middle term of Eq. (4.38) can be approximated by

$$\begin{aligned}
& - \frac{\tau}{\delta t} \left[\int u' \varphi_\alpha^* g(\mathbf{0}, t + \delta t) d\Xi - \int \int_{u' \geq 0} u' \varphi_\alpha^* g_l d\Xi - \int \int_{u' < 0} u' \varphi_\alpha^* g_r d\Xi \right] \\
& = -\tau \int u' \varphi_\alpha^* \frac{\partial g(\mathbf{0}, t + \delta t)}{\partial t} d\Xi \tag{4.39}
\end{aligned}$$

According to the work of Xu (2001), $\partial g / \partial t$ can be approximated by

$$\frac{\partial g(\mathbf{0}, t + \delta t)}{\partial t} = g(\mathbf{0}, t + \delta t) (A_1 + A_2 u' + A_3 v' + A_4 w' + A_5 \varepsilon) \tag{4.40}$$

where A_1, A_2, A_3, A_4 and A_5 are the derivatives of macroscopic variables with respect to time, which will be determined from the compatibility condition,

$\varepsilon = \frac{1}{2} (u'^2 + v'^2 + w'^2 + \xi^2)$. Thus, the flux expression in Eq. (4.38) can be written as

$$\begin{aligned}
F_n^* &= \int u' \varphi_\alpha^* g(\mathbf{0}, t + \delta t) d\Xi \\
&- \tau \int u' \varphi_\alpha^* g(\mathbf{0}, t + \delta t) (A_1 + A_2 u' + A_3 v' + A_4 w' + A_5 \varepsilon) d\Xi \\
&- \tau \left[\frac{\partial}{\partial n_1} \int \left(\int_{u' \geq 0} u'^2 \varphi_\alpha^* g_l + \int_{u' < 0} u'^2 \varphi_\alpha^* g_r \right) d\Xi \right. \\
&\quad + \frac{\partial}{\partial n_2} \int \left(\int_{u' \geq 0} u' v' \varphi_\alpha^* g_l + \int_{u' < 0} u' v' \varphi_\alpha^* g_r \right) d\Xi \\
&\quad \left. + \frac{\partial}{\partial n_3} \int \left(\int_{u' \geq 0} u' w' \varphi_\alpha^* g_l + \int_{u' < 0} u' w' \varphi_\alpha^* g_r \right) d\Xi \right]
\end{aligned} \tag{4.41}$$

In the above equations, the effect of streaming time step is eliminated. The only undetermined variables in this scheme are the coefficients A_1 , A_2 , A_3 , A_4 and A_5 .

According to Eq. (4.31), we have

$$\begin{aligned}
W^* &= \int \varphi_\alpha^* g(\mathbf{0}, t + \delta t) d\Xi \\
&= \int \int_{u' \geq 0} \varphi_\alpha^* \left(g_l - \frac{\partial g_l}{\partial n_1} u' \delta t - \frac{\partial g_l}{\partial n_2} v' \delta t - \frac{\partial g_l}{\partial n_3} w' \delta t \right) d\Xi \\
&\quad + \int \int_{u' < 0} \varphi_\alpha^* \left(g_r - \frac{\partial g_r}{\partial n_1} u' \delta t - \frac{\partial g_r}{\partial n_2} v' \delta t - \frac{\partial g_r}{\partial n_3} w' \delta t \right) d\Xi
\end{aligned}$$

The above equation can be rewritten as

$$\begin{aligned}
&\frac{1}{\delta t} \int \varphi_\alpha^* \left[g(\mathbf{0}, t + \delta t) - \int_{u' \geq 0} g_l - \int_{u' < 0} g_r \right] d\Xi \\
&= - \left[\int \int_{u' \geq 0} \varphi_\alpha^* \left(\frac{\partial g_l}{\partial n_1} u' + \frac{\partial g_l}{\partial n_2} v' + \frac{\partial g_l}{\partial n_3} w' \right) d\Xi \right. \\
&\quad \left. + \int \int_{u' < 0} \varphi_\alpha^* \left(\frac{\partial g_r}{\partial n_1} u' + \frac{\partial g_r}{\partial n_2} v' + \frac{\partial g_r}{\partial n_3} w' \right) d\Xi \right]
\end{aligned} \tag{4.42}$$

Using Eqs. (4.39) and (4.40), the above equation can be written as

$$\begin{aligned}
&\int \varphi_\alpha^* g(\mathbf{0}, t + \delta t) (A_1 + A_2 u' + A_3 v' + A_4 w' + A_5 \varepsilon) d\Xi \\
&= - \left[\frac{\partial}{\partial n_1} \int \left(\int_{u' \geq 0} u' \varphi_\alpha^* g_l + \int_{u' < 0} u' \varphi_\alpha^* g_r \right) d\Xi + \frac{\partial}{\partial n_2} \int \left(\int_{u' \geq 0} v' \varphi_\alpha^* g_l + \int_{u' < 0} v' \varphi_\alpha^* g_r \right) d\Xi \right. \\
&\quad \left. + \frac{\partial}{\partial n_3} \int \left(\int_{u' \geq 0} w' \varphi_\alpha^* g_l + \int_{u' < 0} w' \varphi_\alpha^* g_r \right) d\Xi \right]
\end{aligned} \tag{4.43}$$

Defining

$$\begin{aligned} \frac{\partial}{\partial n_1} \int \left(\int_{u' \geq 0} u' \varphi_\alpha^* g_l + \int_{u' < 0} u' \varphi_\alpha^* g_r \right) d\Xi + \frac{\partial}{\partial n_2} \int \left(\int_{u' \geq 0} v' \varphi_\alpha^* g_l + \int_{u' < 0} v' \varphi_\alpha^* g_r \right) d\Xi \\ + \frac{\partial}{\partial n_3} \int \left(\int_{u' \geq 0} w' \varphi_\alpha^* g_l + \int_{u' < 0} w' \varphi_\alpha^* g_r \right) d\Xi = \begin{pmatrix} G_1 \\ G_2 \\ G_3 \\ G_4 \\ G_5 \end{pmatrix} \end{aligned} \quad (4.44)$$

Then Eq. (4.43) can be written as

$$\rho \begin{pmatrix} 1 & U' & V' & W' & \mathfrak{R}_1 \\ U' & U'^2 + \frac{1}{2\lambda} & UV' & UW' & \mathfrak{R}_2 \\ V' & UV' & V'^2 + \frac{1}{2\lambda} & VW' & \mathfrak{R}_3 \\ W' & UW' & VW' & W'^2 + \frac{1}{2\lambda} & \mathfrak{R}_4 \\ \mathfrak{R}_1 & \mathfrak{R}_2 & \mathfrak{R}_3 & \mathfrak{R}_4 & \mathfrak{R}_5 \end{pmatrix} \begin{pmatrix} A_1 \\ A_2 \\ A_3 \\ A_4 \\ A_5 \end{pmatrix} + \begin{pmatrix} G_1 \\ G_2 \\ G_3 \\ G_4 \\ G_5 \end{pmatrix} = 0 \quad (4.45)$$

where ρ , U' , V' , W' and λ are the macroscopic flow variables at the interface and

$$\mathfrak{R}_1 = \frac{1}{2} \left(U'^2 + V'^2 + W'^2 + \frac{K+3}{2\lambda} \right) \quad (4.46)$$

$$\mathfrak{R}_2 = \frac{1}{2} U' \left(U'^2 + V'^2 + W'^2 + \frac{K+5}{2\lambda} \right) \quad (4.47)$$

$$\mathfrak{R}_3 = \frac{1}{2} V' \left(U'^2 + V'^2 + W'^2 + \frac{K+5}{2\lambda} \right) \quad (4.48)$$

$$\mathfrak{R}_4 = \frac{1}{2} W' \left(U'^2 + V'^2 + W'^2 + \frac{K+5}{2\lambda} \right) \quad (4.49)$$

$$\mathfrak{R}_5 = \frac{1}{4} \left[(U'^2 + V'^2 + W'^2)^2 + (U'^2 + V'^2 + W'^2) \frac{K+5}{\lambda} + \frac{K^2 + 8K + 15}{4\lambda^2} \right] \quad (4.50)$$

From Eq. (4.45), the coefficients A_1 , A_2 , A_3 , A_4 and A_5 can be determined as

$$\begin{aligned} A_5 &= -\frac{8\lambda^2}{(K+3)\rho} \left[G_5 - U'G_2 - V'G_3 - W'G_4 - (\mathfrak{R}_1 - U'^2 - V'^2 - W'^2)G_1 \right] \\ &= -\frac{8\lambda^2}{(K+3)\rho} \left[G_5 - U'G_2 - V'G_3 - W'G_4 + \frac{1}{2} \left(U'^2 + V'^2 + W'^2 - \frac{K+3}{2\lambda} \right) G_1 \right] \end{aligned} \quad (4.51)$$

$$A_4 = -\frac{2\lambda}{\rho}(G_4 - W'G_1) - W'A_5 \quad (4.52)$$

$$A_3 = -\frac{2\lambda}{\rho}(G_3 - V'G_1) - V'A_5 \quad (4.53)$$

$$A_2 = -\frac{2\lambda}{\rho}(G_2 - U'G_1) - U'A_5 \quad (4.54)$$

$$A_1 = -\frac{1}{\rho}G_1 - U'A_2 - V'A_3 - W'A_4 - \mathfrak{R}_1 A_5 \quad (4.55)$$

Once the above coefficients are obtained, the numerical flux \mathbf{F}_n^* across the interface can be calculated via Eq. (4.41). Similar to conservative flow variables \mathbf{W}^* , the explicit expressions for numerical flux \mathbf{F}_n^* can also be given as

$$F_{n,1}^* = \rho U' \quad (4.56)$$

$$\begin{aligned} F_{n,2}^* = & (\rho U'^2 + p) - \tau \rho \left[A_1 \langle u'^2 \rangle + A_2 \langle u'^3 \rangle + A_3 \langle u'^2 \rangle \langle v'^1 \rangle + A_4 \langle u'^2 \rangle \langle w'^1 \rangle \right. \\ & \left. + \frac{1}{2} A_5 \left(\langle u'^4 \rangle + \langle u'^2 \rangle \langle v'^2 \rangle + \langle u'^2 \rangle \langle w'^2 \rangle + \langle u'^2 \rangle \langle \xi^2 \rangle \right) \right] \\ & - \tau \left[\frac{\partial(\rho_l \cdot d'_l + \rho_r \cdot d'_r)}{\partial n_1} + \frac{\partial(\rho_l V'_l \cdot c'_l + \rho_r V'_r \cdot c'_r)}{\partial n_2} + \frac{\partial(\rho_l W'_l \cdot c'_l + \rho_r W'_r \cdot c'_r)}{\partial n_3} \right] \end{aligned} \quad (4.57)$$

$$\begin{aligned} F_{n,3}^* = & \rho U' V' - \tau \rho \left[A_1 \langle u'^1 \rangle \langle v'^1 \rangle + A_2 \langle u'^2 \rangle \langle v'^1 \rangle + A_3 \langle u'^1 \rangle \langle v'^2 \rangle + A_4 \langle u'^1 \rangle \langle v'^1 \rangle \langle w'^1 \rangle \right. \\ & \left. + \frac{1}{2} A_5 \left(\langle u'^3 \rangle \langle v'^1 \rangle + \langle u'^1 \rangle \langle v'^3 \rangle + \langle u'^1 \rangle \langle v'^1 \rangle \langle w'^2 \rangle + \langle u'^1 \rangle \langle v'^1 \rangle \langle \xi^2 \rangle \right) \right] \\ & - \tau \left[\frac{\partial(\rho_l V'_l \cdot c'_l + \rho_r V'_r \cdot c'_r)}{\partial n_1} + \frac{\partial \left[(\rho_l V_l'^2 + p_l) \cdot b'_l + (\rho_r V_r'^2 + p_r) \cdot b'_r \right]}{\partial n_2} \right. \\ & \left. + \frac{\partial(\rho_l V'_l \cdot W'_l \cdot b'_l + \rho_r V'_r \cdot W'_r \cdot b'_r)}{\partial n_3} \right] \end{aligned} \quad (4.58)$$

$$\begin{aligned}
F_{n,4}^* = & \rho U W' - \tau \rho \left[A_1 \langle u'^1 \rangle \langle w'^1 \rangle + A_2 \langle u'^2 \rangle \langle w'^1 \rangle \right. \\
& + A_3 \langle u'^1 \rangle \langle v'^1 \rangle \langle w'^1 \rangle + A_4 \langle u'^1 \rangle \langle w'^2 \rangle \\
& + \left. \frac{1}{2} A_5 \left(\langle u'^3 \rangle \langle w'^1 \rangle + \langle u'^1 \rangle \langle v'^2 \rangle \langle w'^1 \rangle + \langle u'^1 \rangle \langle w'^3 \rangle + \langle u'^1 \rangle \langle w'^1 \rangle \langle \xi^2 \rangle \right) \right] \quad (4.59) \\
& - \tau \left[\frac{\partial (\rho_l V_l' \cdot c_l' + \rho_r W_r' \cdot c_r')}{\partial n_1} + \frac{\partial (\rho_l V_l' \cdot W_l' \cdot b_l' + \rho_r V_r' \cdot W_r' \cdot b_r')}{\partial n_2} \right. \\
& \left. + \frac{\partial \left[(\rho_l W_l'^2 + p_l) \cdot b_l' + (\rho_r W_r'^2 + p_r) \cdot b_r' \right]}{\partial n_3} \right]
\end{aligned}$$

$$\begin{aligned}
F_{n,5}^* = & (\rho E + p) U' - \frac{1}{2} \tau \rho \left\{ A_1 \left[\langle u'^3 \rangle + \langle u'^1 \rangle \langle v'^2 \rangle + \langle u'^1 \rangle \langle w'^2 \rangle + \langle u'^1 \rangle \langle \xi^2 \rangle \right] \right. \\
& + A_2 \left[\langle u'^4 \rangle + \langle u'^2 \rangle \langle v'^2 \rangle + \langle u'^2 \rangle \langle w'^2 \rangle + \langle u'^2 \rangle \langle \xi^2 \rangle \right] \\
& + A_3 \left[\langle u'^3 \rangle \langle v'^1 \rangle + \langle u'^1 \rangle \langle v'^3 \rangle + \langle u'^1 \rangle \langle v'^1 \rangle \langle w'^2 \rangle + \langle u'^1 \rangle \langle v'^1 \rangle \langle \xi^2 \rangle \right] \\
& + A_4 \left[\langle u'^3 \rangle \langle w'^1 \rangle + \langle u'^1 \rangle \langle v'^2 \rangle \langle w'^1 \rangle + \langle u'^1 \rangle \langle w'^3 \rangle + \langle u'^1 \rangle \langle w'^1 \rangle \langle \xi^2 \rangle \right] \\
& + \frac{1}{2} A_5 \left[\langle u'^5 \rangle + \langle u'^1 \rangle \langle v'^4 \rangle + \langle u'^1 \rangle \langle w'^4 \rangle + \langle u'^1 \rangle \langle \xi^4 \rangle + 2 \langle u'^3 \rangle \langle v'^2 \rangle \right. \\
& + 2 \langle u'^3 \rangle \langle w'^2 \rangle + 2 \langle u'^3 \rangle \langle \xi^2 \rangle + 2 \langle u'^1 \rangle \langle v'^2 \rangle \langle w'^2 \rangle \\
& \left. + 2 \langle u'^1 \rangle \langle v'^2 \rangle \langle \xi^2 \rangle + 2 \langle u'^1 \rangle \langle w'^2 \rangle \langle \xi^2 \rangle \right] \left. \right\} \\
& - \frac{1}{2} \tau \left\{ \frac{\partial}{\partial n_1} \left\{ \rho_l \left[e_l' + (V_l'^2 + W_l'^2 + (b-1)RT_l) \cdot c_l' \right] \right. \right. \\
& \quad \left. \left. + \rho_r \left[e_r' + (V_r'^2 + W_r'^2 + (b-1)RT_r) \cdot c_r' \right] \right\} \right. \\
& + \frac{\partial}{\partial n_2} \left\{ \rho_l V_l' \left[d_l' + (V_l'^2 + W_l'^2 + (b+1)RT_l) \cdot b_l' \right] \right. \\
& \quad \left. + \rho_r V_r' \left[d_r' + (V_r'^2 + W_r'^2 + (b+1)RT_r) \cdot b_r' \right] \right\} \quad (4.60) \\
& + \frac{\partial}{\partial n_3} \left\{ \rho_l W_l' \left[d_l' + (V_l'^2 + W_l'^2 + (b+1)RT_l) \cdot b_l' \right] \right. \\
& \quad \left. + \rho_r W_r' \left[d_r' + (V_r'^2 + W_r'^2 + (b+1)RT_r) \cdot b_r' \right] \right\} \left. \right\}
\end{aligned}$$

4.2.3 Collision time and Prandtl number fix

As has been discussed in the last chapter, the collision time τ is proportional to the physical viscosity

$$\tau = \mu/p \quad (4.61)$$

where μ is the dynamic viscosity and p is the pressure. Theoretically, the collision time τ should be determined by Eq. (4.61). On the one hand, numerical solution of N-S equations may encounter instability for some cases such as strong shock waves. Therefore, the effective viscosity should be a combination of both physical and numerical ones. A simple but effective way to incorporate numerical dissipation into the gas kinetic BGK scheme was presented by Xu (2001), and this way is also adopted in the present work

$$\tau = \frac{\mu}{p} + \frac{|p_L - p_R|}{p_L + p_R} \Delta t \quad (4.62)$$

where Δt is the time step in the solution of N-S equations, p_L and p_R are the pressure at the left and right sides of the cell interface, respectively. The second part corresponds to the numerical viscosity.

It is well known that the Prandtl number in the gas kinetic BGK scheme corresponds to unity (Xu, 2001). Several approaches are available to make the Prandtl number be consistent with the real problem. BGK-Shakhov model (Shakhov, 1968) is one of these attempts, which adjusts the heat flux in the relaxation term. In the Shakhov model, the Shakhov equilibrium distribution function is given by

$$g^s = g \left[1 + (1 - Pr) \mathbf{c} \cdot \mathbf{q} \left(\frac{c^2}{RT} - 5 \right) / (5pRT) \right] \quad (4.63)$$

where Pr is the Prandtl number, $\mathbf{c} = \mathbf{u} - \mathbf{U}$ is the peculiar velocity, R is the gas constant and \mathbf{q} is the heat flux

$$\mathbf{q} = \frac{1}{2} \int (\mathbf{u} - \mathbf{U}) \left((u - U)^2 + (v - V)^2 + (w - W)^2 + \xi^2 \right) f d\Xi \quad (4.64)$$

It can be seen from Eq. (4.63) that the Prandtl number can be changed to any realistic value easily. However, considerable work has to be devoted to extend the current GKFS to the above Shakhov model.

Another alternative approach is to make correction for heat flux, which has been presented in Xu (2001)

$$F_E^{new} = F_E + \left(\frac{1}{Pr} - 1 \right) q \quad (4.65)$$

where F_E is the energy flux and q is the heat flux defined in Eq. (4.64). Since almost all momentums in Eq. (4.65) have been obtained in the evaluation of energy flux F_E , there will not be much additional work in the above Prandtl number fix. Therefore, Eq. (4.65) is employed to adjust the Prandtl number in the present work.

4.2.4 Computational sequence

Overall, the basic solution procedure of the current 3D gas kinetic flux solver is summarized as follows:

- (1) At first, we need to calculate the derivatives of conservative flow variables and reconstruct the initial conservative flow variables at two sides of cell interface.
- (2) Compute the unit vector in the normal direction \mathbf{n}_1 and in the tangential directions \mathbf{n}_2 and \mathbf{n}_3 of the cell interface. Convert the velocities in the Cartesian coordinate system into the local coordinate system via Eq. (4.17).
- (3) Calculate the conservative flow variables at the cell interface \mathbf{W}^* by using Eqs. (4.33)-(4.37).

- (4) Calculate the vector $(G_1, G_2, G_3, G_4, G_5)^T$ by using Eq. (4.44) and further compute coefficients A_1, A_2, A_3, A_4, A_5 by Eqs. (4.51)-(4.55).
- (5) Calculate the numerical flux F_n^* by Eqs. (4.56)-(4.60).
- (6) Compute the heat flux q via Eq. (4.64), and make correction for energy flux by using Eq. (4.65).
- (7) Convert the numerical flux in the local coordinate system F_n^* to the global Cartesian coordinate system F_n by using Eq. (4.22).
- (8) Once the fluxes at all cell interfaces are obtained, solve ordinary differential equation (Eq. (4.10)) by using time marching method. This step gives the conservative flow variables at cell centers at new time step.
- (9) Repeat steps (1) to (8) until convergence criterion is satisfied.

4.3 Numerical Results and Discussions

In this section, the present 3D GKFS is validated through simulations of several 3D flow problems. First, the benchmark problem of 3D lid-driven cavity flow at two different Reynolds numbers are simulated. Another incompressible flow case of flow past a stationary sphere is also simulated to examine the capability of the present solver in solving curved boundary problems. Subsequently, simulations of 3D compressible flows are conducted. The first two cases are transonic flow past two different wings, which are ONERA M6 and DPW III. The turbulent effects in both two cases are taken into consideration by using the Spalart-Allmaras turbulence model (Spalart and Allmaras, 1992). The shock waves can be clearly observed at the wing surfaces. The last test case is the DLR-F6 wing-Body configuration, which is much more challenging because of its complicated geometry. The pressure coefficient

distributions and the force coefficients are compared with available data in literature to examine the proposed solver for 3D compressible turbulent flows. For temporal discretization to the governing equation (4.10), four-stages Runge-Kutta method is applied in the cases of 3D lid-driven cavity flow and flow past a stationary sphere. In compressible cases, the LU-SGS scheme (Yoon and Jameson, 1988) is adopted to accelerate the convergence and the Venkatakrishnan's limiter (Venkatakrishnan, 1995) is used to calculate the conservative flow variables at two sides of cell interface in the reconstruction process.

4.3.1 3D lid-driven cavity flow

The 3D lid-driven cavity flows in a cube are simulated to test the capability of the proposed explicit gas kinetic flux solver for simulating 3D incompressible viscous flows. The non-uniform mesh of $81 \times 81 \times 81$ is used for the cases of $Re = 100$ and 400. The mesh point in the x -direction is generated by

$$\begin{aligned} x_i &= 0.5 \left(1 - \eta \tan^{-1} \left((1 - \kappa_i) \cdot \tan(1/\eta) \right) \right) & i \leq (i_{\max} + 1)/2 \\ x_i &= 1.0 - x_{i_{\max} + 1 - i} & \text{else} \end{aligned} \quad (4.66)$$

where $\kappa_i = (i - 1) / ((i_{\max} - 1) / 2)$, i and i_{\max} are the mesh point index and total number of mesh points in the x direction; η is the parameter to control the mesh stretching and is selected as 1.1 in this study. Similarly, the mesh point in the y - and z -directions is generated in the same way.

In the current simulation, the fluid density is taken as $\rho = 1.0$ and the lid velocity is chosen as $u_\infty = 0.1$. Initially, the density inside the cavity is constant and the flow is static. The lid on the top boundary moves along the x -direction. The no-slip wall condition is imposed at all boundaries. To quantitatively examine the performance of

3D GKFS, the velocity profiles of x -direction component u along the vertical centerline and y -direction component v along the horizontal centerline for $Re=100$ and 400 are plotted in Figure 4.1. For comparison, the results of Shu et al. (2003) and Wu and Shu (2010) are also included in the figure. It can be found that all the velocity profiles by the current 3D GKFS agree very well with those of Shu et al. (2003) and Wu and Shu (2010), which demonstrates the capability of the present solver for the simulation of 3D incompressible flows on non-uniform grids. To further show the flow patterns of 3D lid-driven cavity flow, the streamlines for $Re=100$ and 400 at three orthogonal mid-planes located at $x=0.5$, $y=0.5$ and $z=0.5$ are displayed in Figure 4.2. The flow patterns along the mid-plane of $z=0.5$ in Figure 4.2 demonstrate that the primary vortices gradually shift toward the center position and the second vortices gradually moves to the lower bottom wall when the Reynolds number is increased. In this process, the strength of these vortices is also enhanced, which can also be proven by the flow patterns along other two mid-planes. All these observations match well with those in Shu et al. (2003).

4.3.2 Incompressible flow past a stationary sphere

The incompressible flow past a stationary sphere, which involves curved boundary, is a good benchmark problem for validating new numerical scheme. In this case, the flow is characterized by the Reynolds number defined by $Re = \rho U_\infty D / \mu$, where ρ and μ are the fluid density and dynamic viscosity, respectively. U_∞ is the free stream velocity and D is the sphere diameter. To simulate this test case with a simple Cartesian mesh, the implicit boundary condition-enforced immersed boundary method (Wu and Shu, 2009; Wang et al., 2015) is coupled with the present 3D GKFS. The computational domain is selected as a rectangular box of $30D \times 20D \times 20D$ in the x -,

y - and z - directions. The sphere is initially placed at $(10D, 10D, 10D)$, which is discretized by triangular elements with 1195 vertices. A non-uniform Cartesian mesh with mesh size of $137 \times 122 \times 122$ is used, in which a uniform mesh spacing of $0.02D$ is applied around the sphere. Here, laminar flows at low Reynolds numbers of 50, 100, 150, 200 and 250 are considered.

Firstly, the drag coefficients at $Re = 100, 200$ and 250 are computed and compared quantitatively in Table 4.1 to verify the accuracy of the present solver. The numerical results of previous studies (Johnson and Patel, 1999; Wu and Shu, 2010; Kim et al., 2001; Wang et al., 2008) are also included in the table for comparison. It can be clearly observed that the present results match well with those in the literature.

Then, for the steady axisymmetric flow, the streamlines of flow past a sphere at $Re \leq 200$ are depicted in Figure 4.3. Since the flow is axisymmetric, only the streamlines on the $x - y$ plane of symmetry are given. From the figure, a recirculation region is appeared behind the sphere and its length L_s increases with Reynolds number. Quantitative comparison between the present results of L_s and those of Johnson and Patel (1999) and Gilmanov et al. (2003) is made in Figure 4.4. Good agreement can be found in the figure. When the Reynolds number is increased to 250, the phenomenon of steady non-axisymmetric pattern shows up, which can be seen in Figure 4.5. In the figure, the streamlines on the $x - z$ plane remains symmetric. However, there are two asymmetric vortices on the $x - y$ plane, which implies that the symmetry is lost in this plane. These results are in good agreement with previous investigations (Johnson and Patel, 1999; Gilmanov et al., 2003).

4.3.3 Flow around an ONERA M6 wing

The ONERA M6 test case is chosen to validate the present solver for the simulation of compressible viscous flows with complex geometry. For numerical simulation, the free-stream Mach number is taken as $M_\infty = 0.8395$, the mean-chord based Reynolds number is chosen as $Re = 11.72 \times 10^6$ and the angle of attack is $\alpha = 3.06^\circ$. The computational mesh in the NASA website (Slater, 2002) is adopted in this work, which has 4 blocks and 316932 grid points. The mesh spacing of the first mesh point adjacent to the wing surface is 4.5×10^{-5} . Figure 4.6 shows the pressure contours at the wing surface obtained from the present solver, in which the “ λ ” shape shock wave on the upper surface is clearly presented. The above phenomenon matches well with the result from sphere function-based gas kinetic scheme (Yang et al., 2015). To further validate the present results, the pressure coefficient distributions at selected span-wise locations obtained from the present solver are displayed in Figure 4.7. The numerical results of WIND scheme (Slater, 2002) and Newton-Krylov algorithm (Wong and Zingg, 2008) as well as the experimental results (Schmitt and Charpin, 1979) are also included for comparison. As can be seen from the figure, the present results are much closer to the experimental data (Schmitt and Charpin, 1979) compared with the results of Wong and Zingg (2008). As a comparison with the WIND scheme (Slater, 2002), the current results show a good agreement and both of them compared well with the experimental data (Schmitt and Charpin, 1979). What is more, the pressure coefficient distributions at 65% and 80% spans show that the present results fit the experimental results (Schmitt and Charpin, 1979) better. It demonstrates that the present solver captures the shock wave more precisely and controls the numerical dissipation well.

4.3.4 Turbulent flow over the DPW Wings W1

In this case study, the baseline wing geometry (DPW-W1) from the third AIAA CFD drag prediction workshop (Vassberg et al., 2007) is considered. The flow over the DPW-W1 with turbulent boundary layers at transonic condition is tested to further validate the present solver. First, the free stream conditions of the present simulation are taken as: Reynolds number (Base on the reference chord) $Re_{c_{ref}} = 5 \times 10^6$, Mach number $M_\infty = 0.76$ and angle of attack $\alpha = 0.5^\circ$. The wing geometry, which is illustrated in Figure 4.8 with pressure contours, is a simple trapezoidal planform within modern supercritical airfoil. The reference values of the wing are that the planform area $S_{ref} = 290322 \text{ mm}^2$ and chord length $c_{ref} = 197.556 \text{ mm}$. The adiabatic no-slip boundary condition is imposed on the wing and the symmetry and free boundary conditions are applied at the wing root and the far field, respectively. The mesh consists of about 1511424 cell points and is provided by the DPW committee. The pressure contour at the wing surface is presented in Figure 4.8, from which a shock wave can be clearly observed. The surface pressure coefficient C_p distributions at eight different span locations are given in Figure 4.9. The results of Vassberg et al. (2007) are also included in the figure for comparison. It is clear that the present results are in good agreement with the reference data.

4.3.5 DLR-F6 wing-body configuration

The DLR-F6 wing-body configuration is a generic transport aircraft model from the second AIAA CFD drag prediction workshop (DPW II) (Laflin et al., 2005). Firstly, numerical simulations are conducted at a free-stream Mach number of $M_\infty = 0.75$, a mean-chord based Reynolds number of $Re = 3 \times 10^6$ and an angle of attack $\alpha = 0.49^\circ$.

For this selected free-stream condition, the corresponding experiment has been conducted in the S2MA wind tunnel facility of ONERA in France in 1990s and the experimental data are provided by the DPW-II organizing committee (Laflin et al., 2005). The geometry and computational mesh from the NASA website are utilized in the current work. Owing to the limitation of the computer's memory, only the coarse mesh with 26 blocks and 2298880 cells is used. Figure 4.10 is the pressure contour of DLR-F6 wing-body obtained by the present 3D GKFS. The separation bubble at the intersection of wing and body is clearly recognized in Figure 4.11, which is in line with the observations of Vassberg et al. (2007). To make a quantitative comparison, the pressure coefficient distributions at selected span-wise locations obtained by the present 3D GKFS are compared with the experimental results (Laflin et al., 2005) and numerical results of Vassberg et al. (2007) and Yang et al. (2014a) in Figure 4.12. It can be observed that the current results are close to the reference data (Vassberg et al., 2007; Yang et al., 2014a) and all of them basically agree well with the experimental measurement (Laflin et al., 2005).

To further verify the force coefficients of the current solver for the DLR-F6 wing-body, another test case is simulated with the free stream condition of Mach number $M_\infty = 0.75$, Reynolds number $Re = 5 \times 10^6$ and angle of attack $\alpha = 0^\circ$. Table 4.2 shows the present results of force coefficients, including lift coefficient C_l , pressure drag coefficient $C_{d,p}$, friction drag coefficient $C_{d,f}$, total drag coefficient C_d and moment coefficient C_M . The results of the present solver are close to the results of LBFS (Yang et al., 2014a) and can essentially match well with the reference data of Vassberg et al. (2007).

4.4 Concluding Remarks

In this chapter, a truly three-dimensional gas kinetic flux solver (3D GKFS) was presented for effective simulation of incompressible and compressible 3D flows. The 3D GKFS applied the finite volume method to solve the 3D N-S equations. Both the viscous and inviscid fluxes were evaluated in a simple and easy way at the cell interface by the local reconstruction of the continuous Boltzmann solutions. In the development, a local coordinate transformation was introduced to transform the velocities in the Cartesian coordinate system to the local normal and tangential directions at each cell interface. In this way, all the interfaces can be treated in the same manner. As an extension of two-dimensional work (Sun et al., 2015), the non-equilibrium function was evaluated by the difference of equilibrium distribution functions at the cell interface and its surrounding points. As a result, the full information of distribution function at the cell interface could be simply derived. The explicit formulations of the conservative flow variables and numerical fluxes at the cell interface were explicitly given, which was the first time in literature.

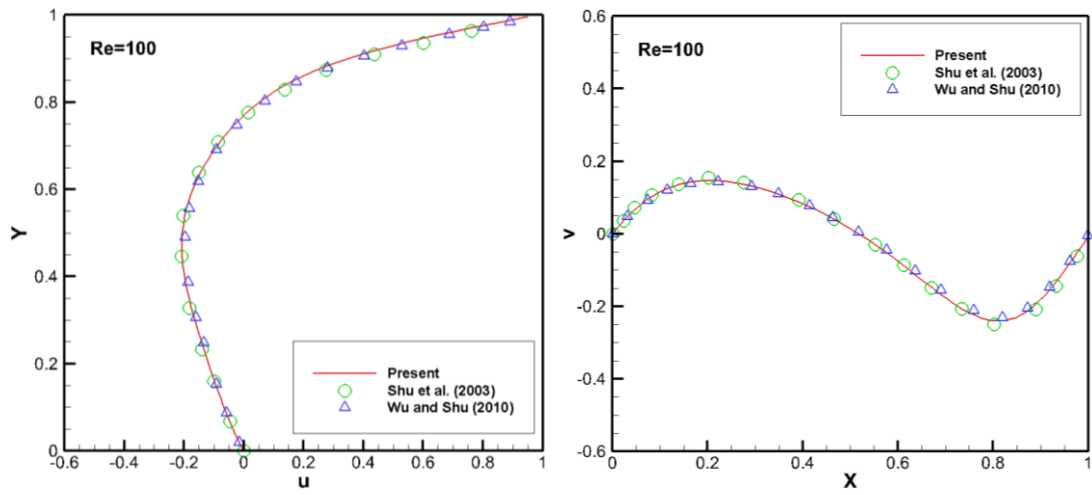
The 3D GKFS has been validated by simulating various 3D incompressible and compressible flows, such as 3D lid-driven cavity flow, incompressible flow past a stationary sphere, flow around an ONERA M6 wing, turbulent flow over the DPW-W1 and DLR-F6 wing-body configuration. Good agreements were achieved between the 3D GKFS solutions and those published in the literature. These good agreements successfully verified the reliability of the 3D GKFS for simulation of 3D flow with complex geometries and flow conditions. In the next chapter, the GKFS will be further extended to study flows with complex moving boundaries.

Table 4.1 Comparison of drag coefficient for flow past a stationary sphere

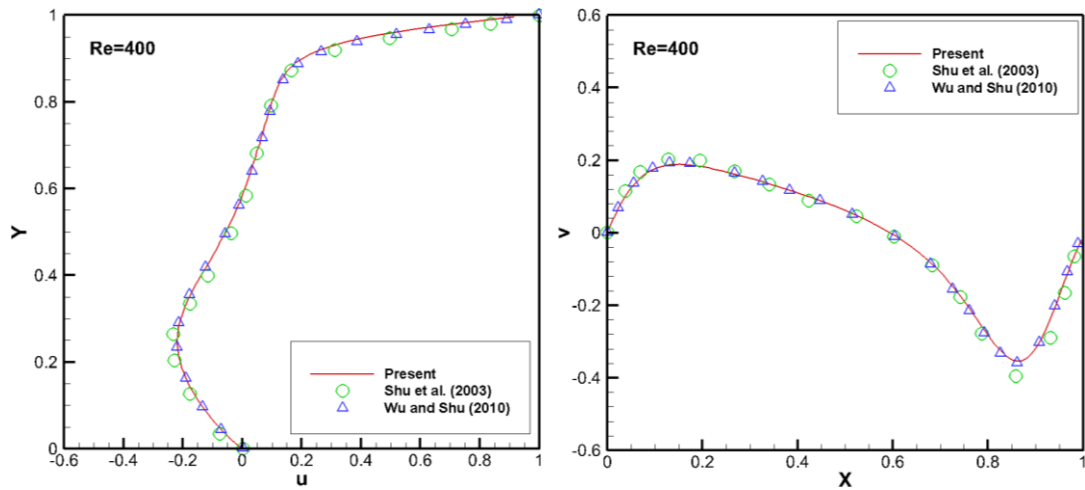
Re	References	C_d
100	Johnson and Patel (1999)	1.112
	Wu and Shu (2010)	1.128
	Present	1.116
200	Johnson and Patel (1999)	0.79
	Wu and Shu (2010)	0.8
	Present	0.791
250	Kim et al. (2001)	0.706
	Wang et al. (2008)	0.746
	Present	0.720

Table 4.2 Comparison of force coefficients for DLR-F6 wing-body configuration

Reference	C_l	$C_{d,p}$	$C_{d,f}$	C_d	C_M
Vassberg et al. (2007)	0.51600	0.01502	0.01229	0.02731	-0.15280
Yang et al. (2014a)	0.52312	0.01554	0.00979	0.02533	-0.14988
Present	0.52470	0.01549	0.00947	0.02496	-0.16230

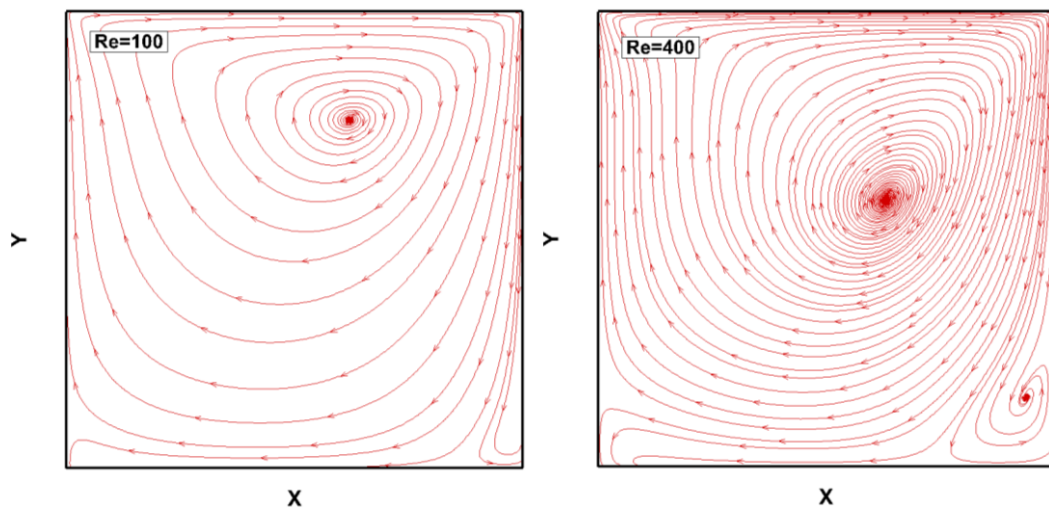


(a) $Re = 100$

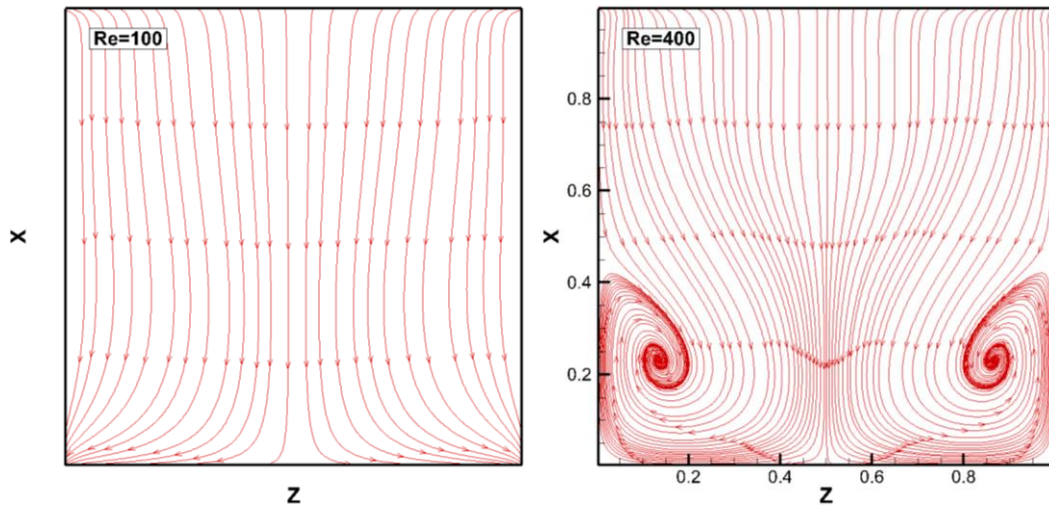


(b) $Re = 400$

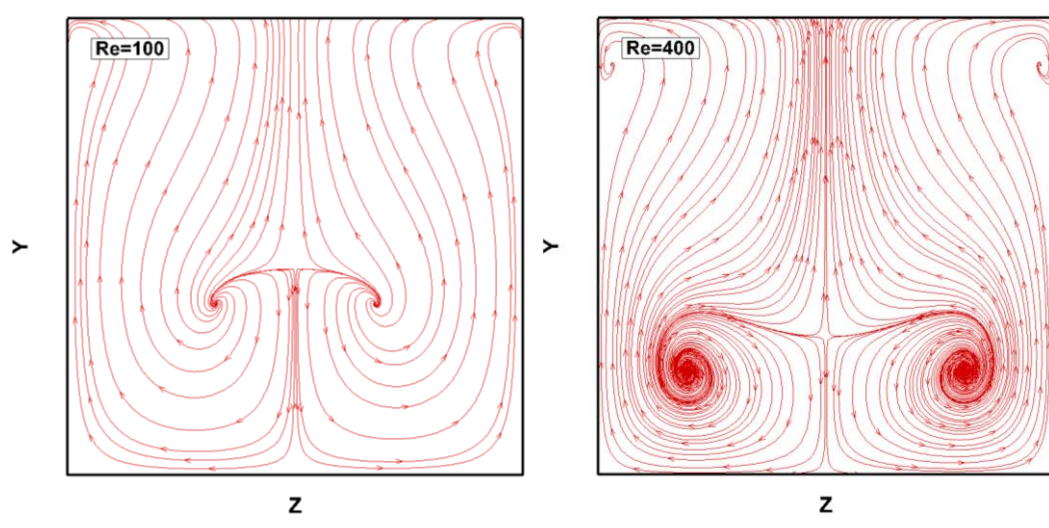
Figure 4.1 u and v velocity profiles on the plane of $z=0.5$ of cubic cavity for 3D lid-driven cavity flow at Reynolds numbers of 100 and 400



(a) mid-plane of $z = 0.5$



(b) mid-plane of $y = 0.5$



(c) mid-plane of $x = 0.5$

Figure 4.2 Streamlines on three mid-planes for 3D lid-driven cavity flow at Reynolds numbers of 100 (left) 400 (right)

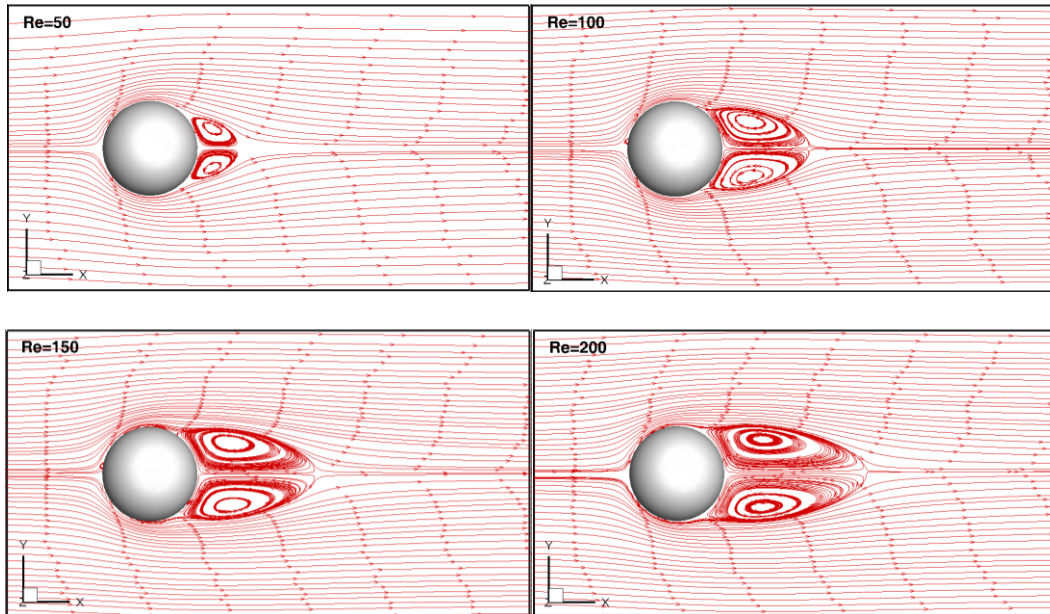


Figure 4.3 Streamlines at four different Reynolds numbers of 50, 100, 150 and 200 in the steady axisymmetric regime

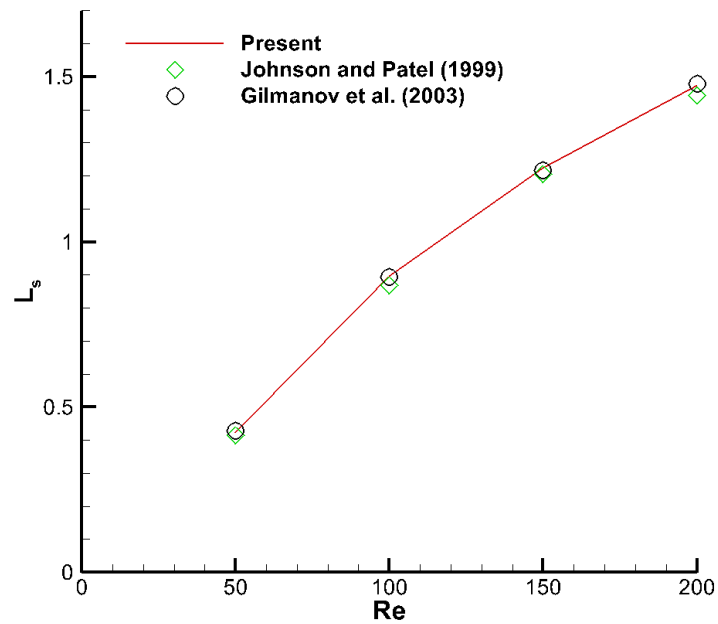


Figure 4.4 Comparison of recirculation length L_s at different Reynolds numbers

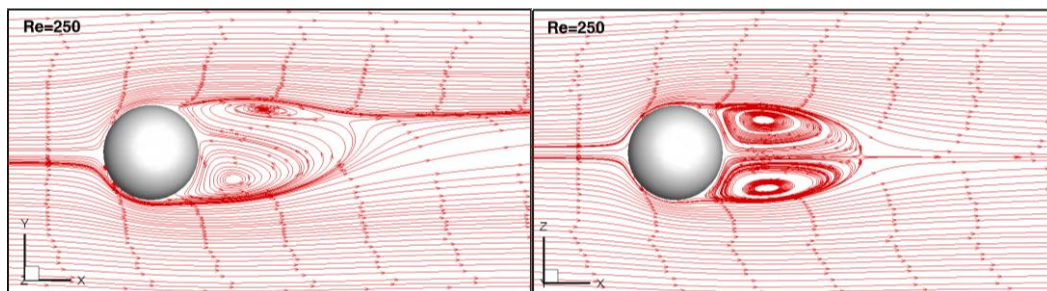


Figure 4.5 Streamlines for flow past a stationary sphere at $Re=250$ in the steady non-axisymmetric regime

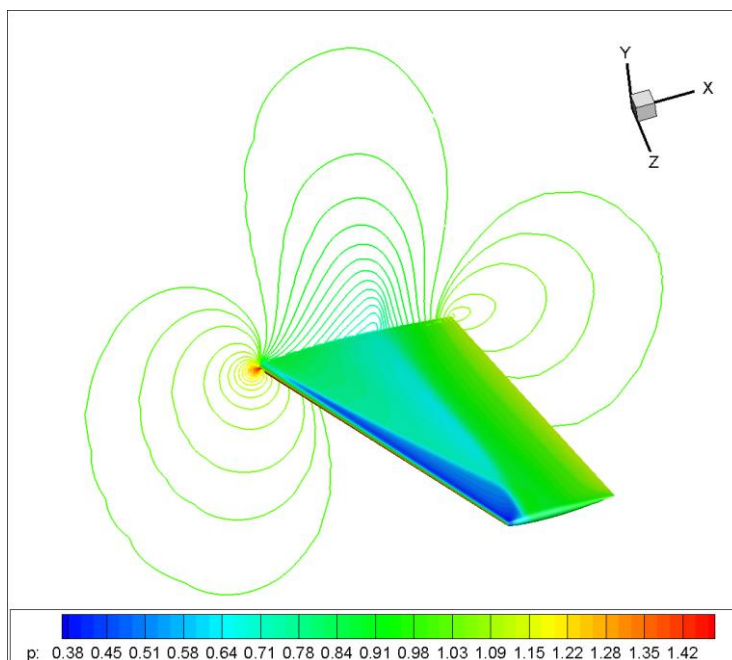


Figure 4.6 Pressure contour of flow around an ONERA M6 wing

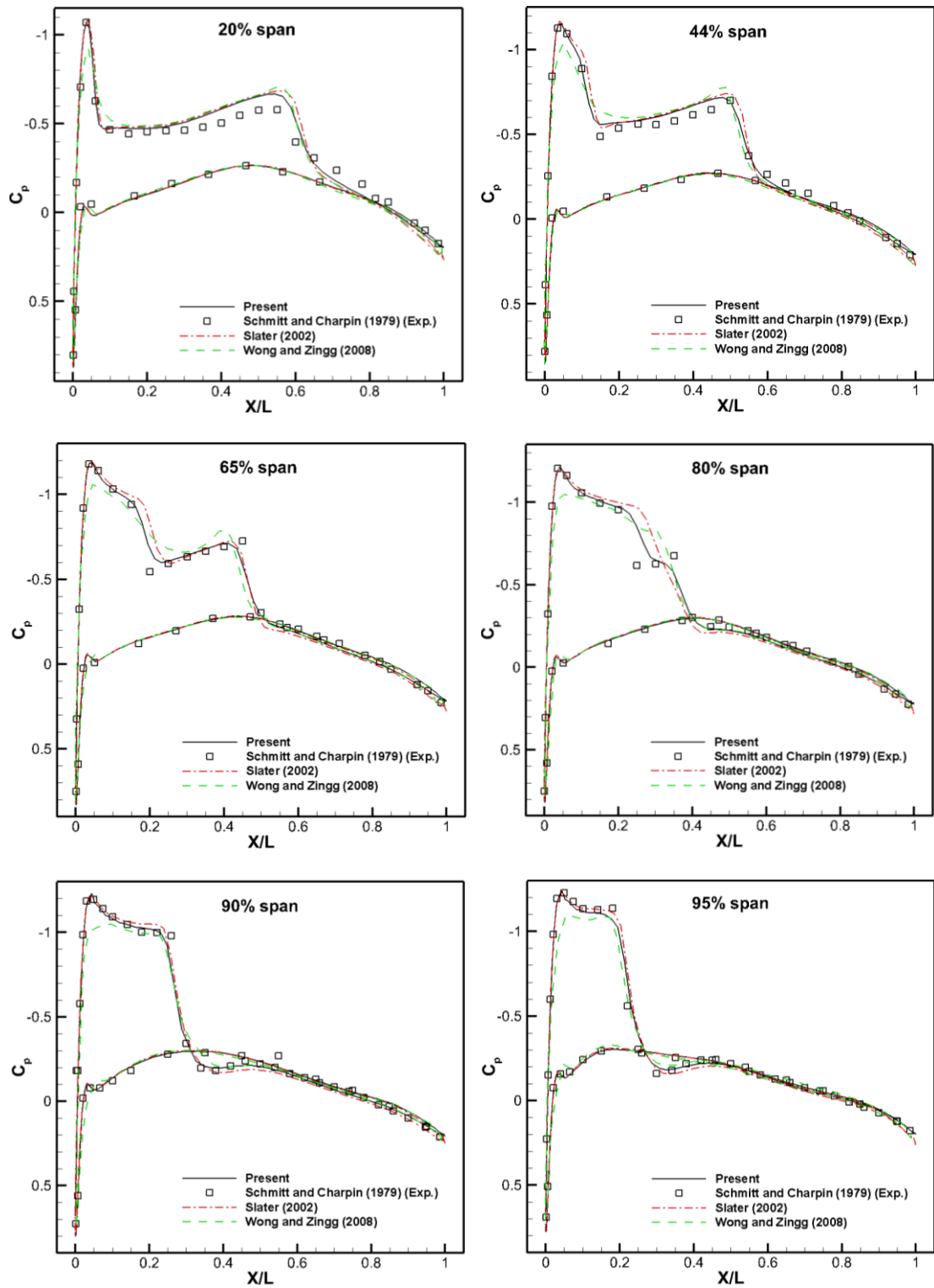


Figure 4.7 Comparison of pressure coefficient distribution at selected positions for ONERA M6 Wing

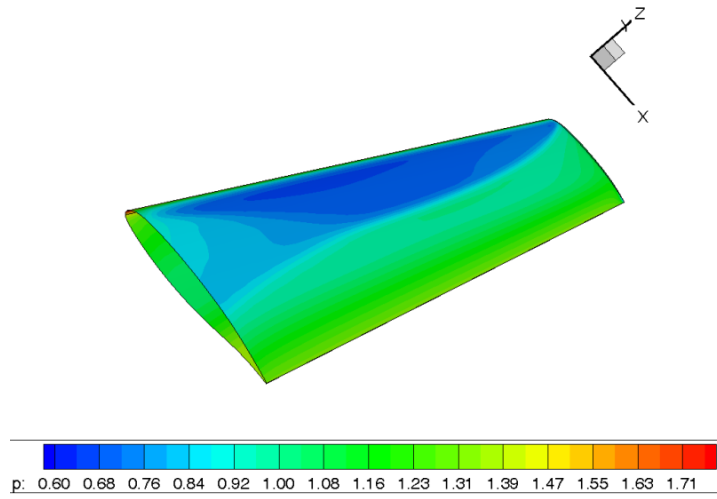
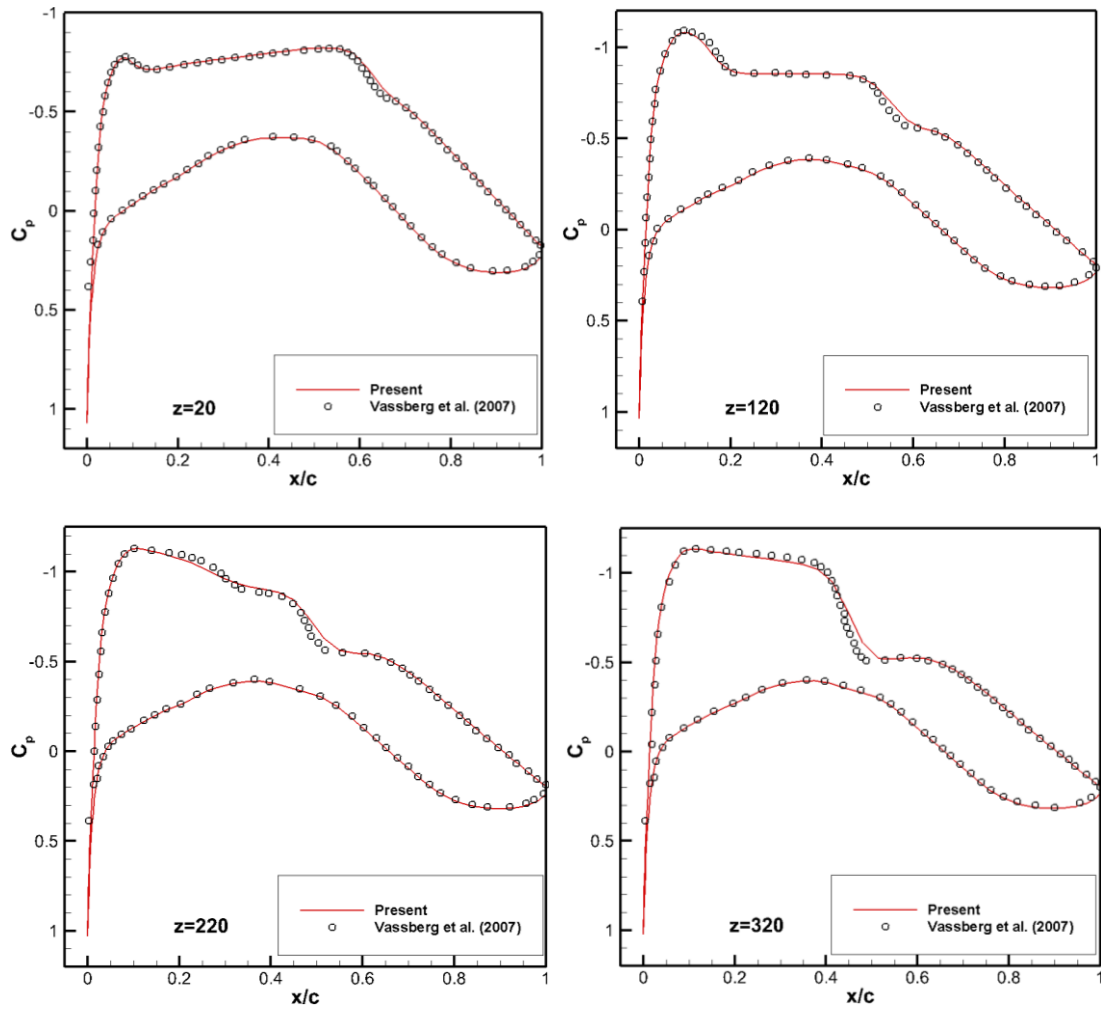


Figure 4.8 The DPW-W1 geometry with pressure contours



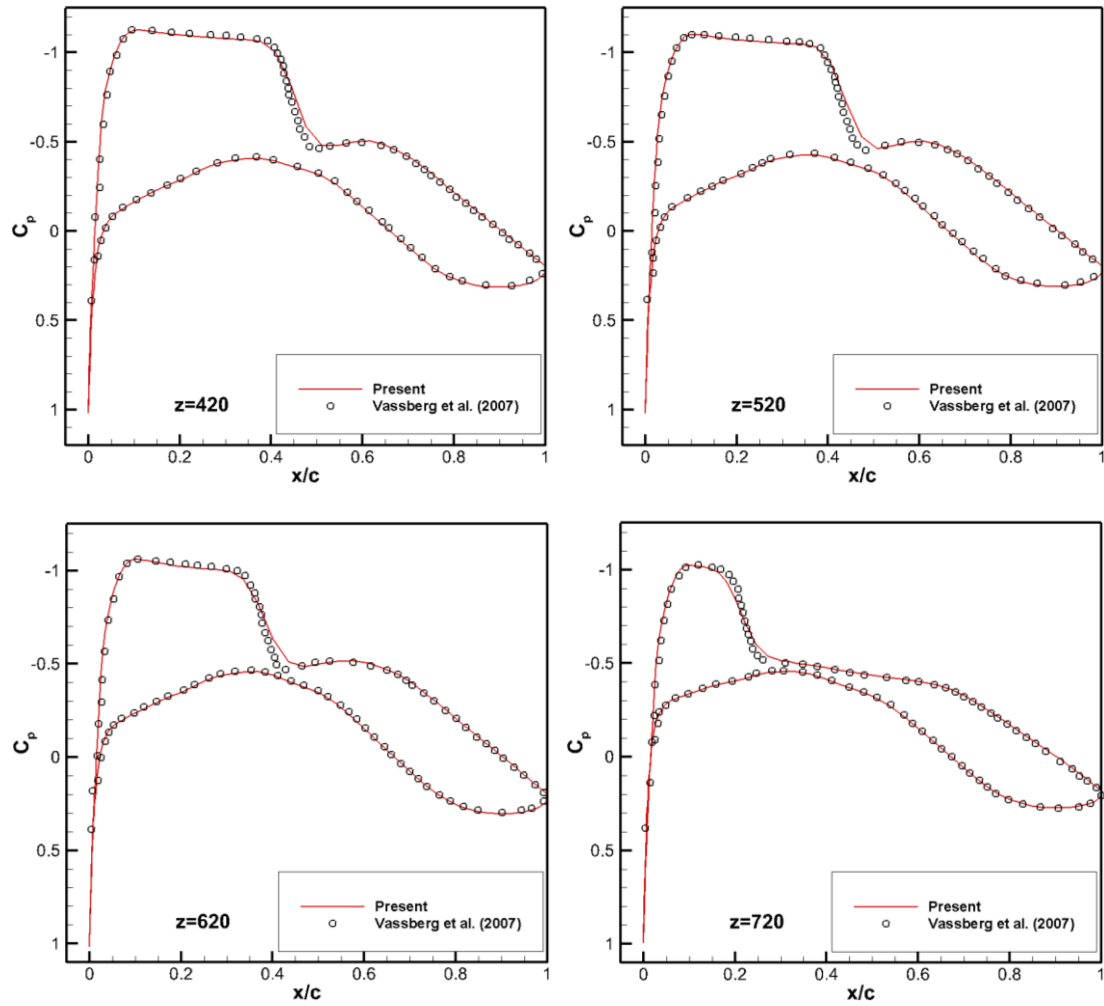


Figure 4.9 The Pressure Coefficient distribution along the DPW-W1

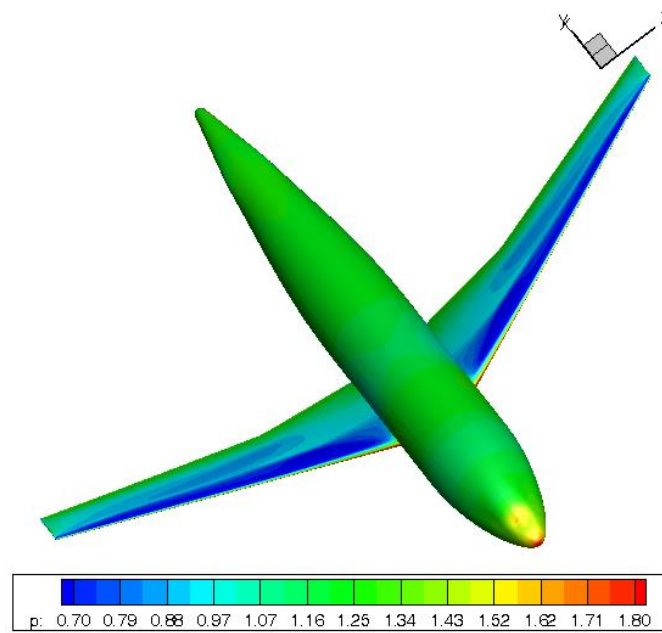


Figure 4.10 Pressure contours of DLR-F6 wing/body

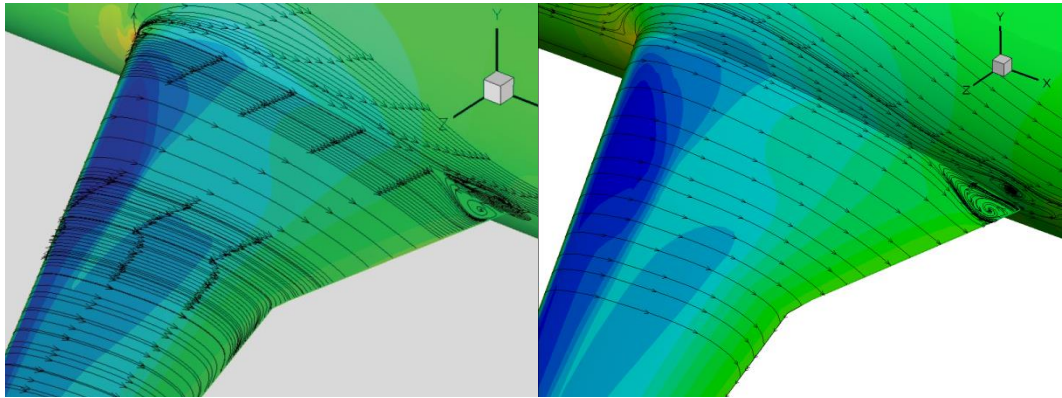
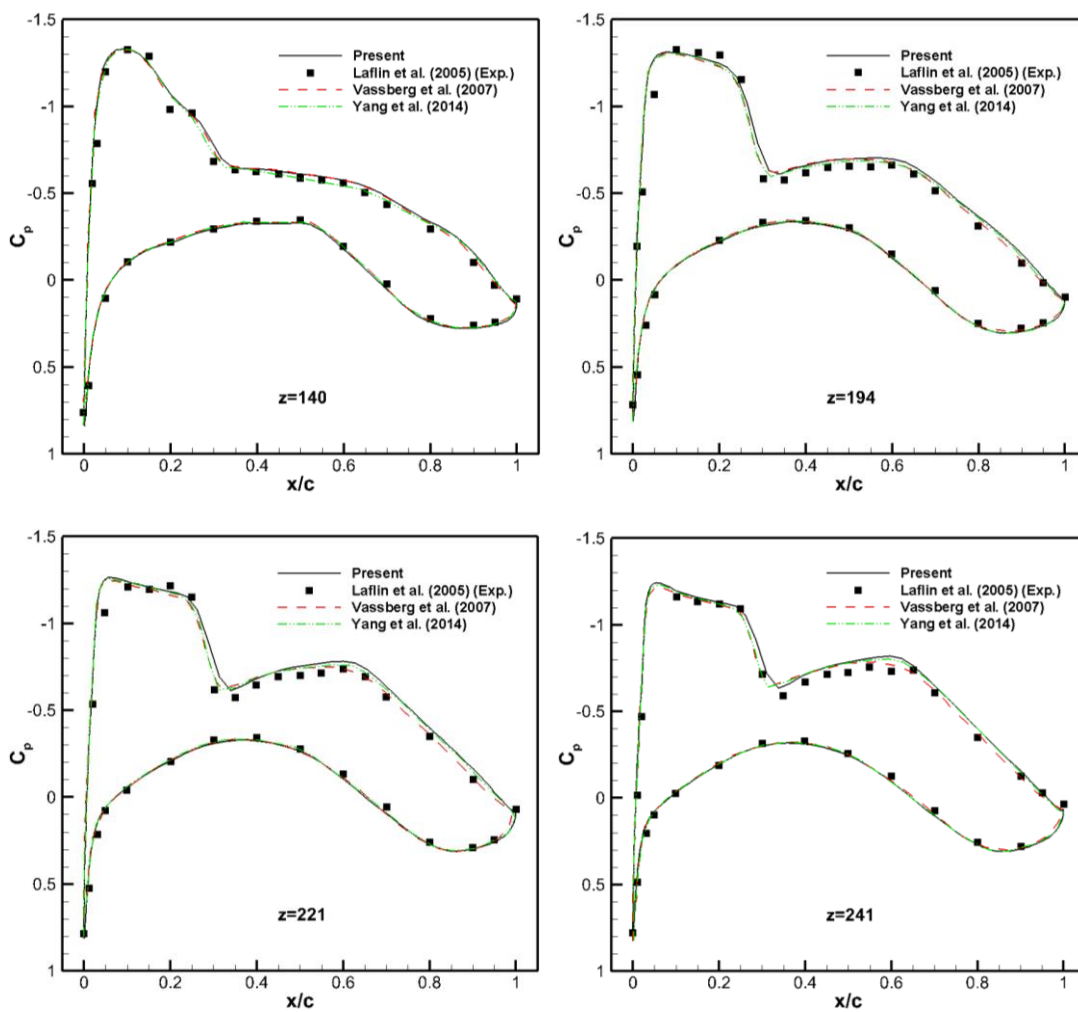


Figure 4.11 Separation bubble on the intersection of wing and body obtained from Vassberg et al. (2007) (Left) and present scheme (Right)



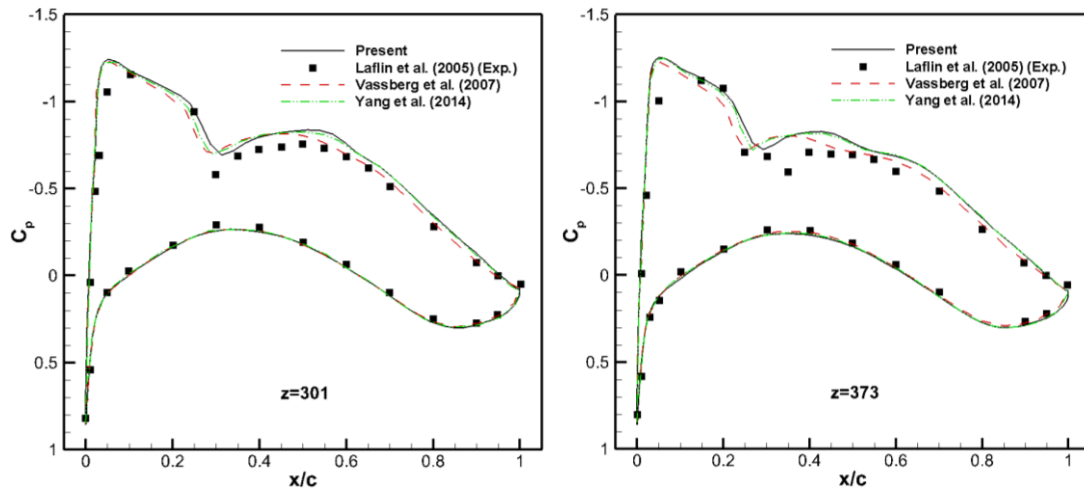


Figure 4.12 Comparison of pressure coefficient distribution of DLR-F6 wing/body at different locations

Chapter 5

Boundary condition-enforced Immersed Boundary-Gas Kinetic Flux Solver and its applications for Moving Boundary Flows

In the previous chapters, the gas kinetic flux solvers have been constructed for simulation of two- and three-dimensional viscous flows. These solvers not only inherit the intrinsic advantages of conventional gas kinetic scheme, such as robustness and applicability, but also effectively eliminate the drawbacks of GKS, including complexity and inefficiency. In this chapter, we further extend the GKFS to solve moving boundary flows, which are of great interest in both academic research and engineering applications. In the study of such flows, the immersed boundary method (IBM) is an efficient approach and has been popularly used. This chapter will first introduce the basic idea of conventional IBM and two different ways to compute the force density in conventional IBM. Since the force is usually pre-calculated, the no-slip boundary condition is approximately satisfied in conventional IBM. To eliminate this drawback, the boundary condition-enforced immersed boundary-gas kinetic flux solver (IB-GKFS) is proposed. By applying the fractional step technique, the solution process of the IB-GKFS can be separated into two steps, the predictor step and the velocity correction step. In the predictor step, the intermediate flow field is obtained by applying the GKFS. As the solid boundary is not considered in this step, there is no external force added in the gas distribution function during the evaluation of numerical flux at each cell interface. In the velocity correction step, no-slip boundary condition is imposed implicitly at all boundary points to make velocity correction on

the surrounding Eulerian points. In the current IB-GKFS, the no-slip boundary condition is accurately fulfilled and flow penetration is entirely avoided. With simple Cartesian mesh and flexible boundary condition treatment, the IB-GKFS can be conveniently applied to solve complex and moving boundary problems. The proposed IB-GKFS will be validated through numerical simulation of a variety of stationary and moving boundary flows.

5.1 Conventional immersed boundary method (IBM)

IBM is an efficient and flexible tool for solving stationary and moving boundary flow problems with complex geometries. In the method, the immersed body is represented in the form of a closed curve Γ in the computational domain Ω . There are two types of points, the Eulerian points \mathbf{x} to discretize the governing equations and the Lagrangian points \mathbf{X} to represent the immersed boundary. The essence of this approach is that the effects of the physical boundary are considered as forces acting on the fluid. These forces are distributed to the Eulerian points and then the governing equations with the distributed forcing terms can be solved in the whole computational domain. As a consequence, the overall governing equations (N-S equations) for the incompressible flow can be written as

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 \quad (5.1)$$

$$\frac{\partial}{\partial t} (\rho \mathbf{u}) + \nabla \cdot (\rho \mathbf{u} \mathbf{u}) = -\nabla p + \nabla \cdot [\mu (\nabla \mathbf{u} + \nabla \mathbf{u}^T)] + \mathbf{f} \quad (5.2)$$

$$\mathbf{f}(\mathbf{r}, t) = \int_{\Gamma} \mathbf{F}(s, t) \delta(\mathbf{r} - \mathbf{X}(s, t)) ds \quad (5.3)$$

where ρ , \mathbf{u} , p and μ are the density, velocities, pressure and dynamic viscosity, respectively. \mathbf{f} and \mathbf{F} are the restoring force acting on the Eulerian points \mathbf{r} and Lagrangian points \mathbf{X} , respectively. $\delta(\mathbf{r} - \mathbf{X}(s, t))$ is a delta function.

In order to solve the governing equations (5.1)-(5.3), the key issue is to evaluate the force density \mathbf{F} on Lagrangian points \mathbf{X} . After the evaluation of the force density \mathbf{F} , the forcing terms on the Eulerian points \mathbf{f} can be computed via delta function. The penalty force and direct forcing methods are two common methods to evaluate \mathbf{F} in conventional IBM. The penalty force model was first proposed by Peskin (1977). The basic idea of this method is the application of Hook's law. By taking the immersed boundary as an elastic fiber with stiffness, and then the boundary point \mathbf{X}_B will undergo a force density \mathbf{F} after a relative motion

$$\mathbf{F} = -k(\mathbf{X}_B - \mathbf{X}_B^o) \quad (5.4)$$

where k represents the spring constant and \mathbf{X}_B^o is the original position of the boundary point. It should be noted that the spring constant is a user-defined parameter. As a result, the accuracy of solution depends so much on the selection of the parameter.

To remove this restriction, the direct forcing method was proposed by Fadlun et al. (2000). As the Lagrangian points on the immersed boundary are also part of flow field, the momentum equation (5.2) can be applied to compute the force density \mathbf{f}

$$\mathbf{f} = \frac{\partial(\rho\mathbf{u})}{\partial t} + \nabla \cdot (\rho\mathbf{u}\mathbf{u}) + \nabla p - \nabla \cdot [\mu(\nabla\mathbf{u} + \nabla\mathbf{u}^T)] \quad (5.5)$$

Applying the flow field condition at the time level $t = t_n$ and the boundary condition at $t = t_{n+1}$, the force density at the Lagrangian points and $t = t_{n+1}$ can be given as follows

$$\mathbf{f}^{n+1} = \frac{\rho(\mathbf{U}_B^{n+1} - \mathbf{u}^n)}{\delta t} + \nabla \cdot (\rho \mathbf{u}^n \mathbf{u}^n) + \nabla p^n - \nabla \cdot [\mu(\nabla \mathbf{u}^n + \nabla \mathbf{u}^{n,T})] \quad (5.6)$$

The above equation is the direct forcing method, where no user-defined parameters are introduced. As compared with the penalty force method, both the efficiency and accuracy are improved in the direct forcing method. However, there is a major problem in the direct forcing method. Due to the pre-calculation of the force density, the no-slip boundary condition is approximately satisfied in the direct forcing method. Due to the approximate satisfaction, the flow penetration to the immersed boundary can be clearly observed. To overcome this drawback, a more accurately IBM together with the powerful GKFS can be combined for moving boundary problems. The development of GKFS-based IBM will be presented in the next section.

5.2 Boundary condition-enforced immersed boundary-gas kinetic flux solver (IB-GKFS)

In this section, the IB-GKFS is proposed for simulation of moving boundary flows with complex geometry. Firstly, the concept of a fractional step method is adopted to solve the governing equations (5.1) and (5.2). By using the fractional step method, the solution process of the governing equations can be divided into two steps: predictor step and corrector step. The resultant equations for two steps can be respectively written as:

Step 1 (predictor step): Solve the N-S equations without forcing term

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 \quad (5.7)$$

$$\frac{\partial}{\partial t} (\rho \mathbf{u}) + \nabla \cdot (\rho \mathbf{u} \mathbf{u}) = -\nabla p + \nabla \cdot [\mu (\nabla \mathbf{u} + \nabla \mathbf{u}^T)] \quad (5.8)$$

and obtain the density ρ^{n+1} and intermediate velocity \mathbf{u}^* at next time step.

Step 2 (corrector step): Correct the velocity field through

$$\frac{\rho^{n+1} (\mathbf{u}^{n+1} - \mathbf{u}^*)}{\Delta t} = \mathbf{f} \quad (5.9)$$

As shown above, the flow field is firstly predicted by GKFS via the finite volume discretization without consideration of the immersed boundary. After that, the boundary condition-enforced immersed boundary method (Wu and Shu, 2009) is applied to make corrections of the flow field to satisfy the no-slip boundary condition. Both of the two procedures will be introduced in detail in the following sections.

5.2.1 Gas kinetic flux solver for prediction of the flow field \mathbf{u}^*

To evaluate the intermediate flow field \mathbf{u}^* and ρ^{n+1} , GKFS is applied to solve the N-S equations without the forcing term in Eqs. (5.7)-(5.8). In Chapter 3, three schemes are proposed in the GKFS and Scheme II is more attractive in incompressible flows due to its high efficiency. As only the incompressible flows are considered in this chapter, Scheme II is adopted here.

By applying the finite volume method, Eqs. (5.7)-(5.8) can be rewritten over a control cell Ω_i ,

$$\frac{\mathbf{W}^* - \mathbf{W}^n}{\Delta t} + \frac{1}{\Delta V_i} \sum_{i=1}^N \mathbf{F}_i S_i = 0 \quad (5.10)$$

where $\mathbf{W}^* = (\rho^{n+1}, \rho^{n+1} \mathbf{u}^*)^T$ is the vector of intermediate conservative flow variables, ΔV_i and N are the volume and number of interfaces of the control volume, respectively. \mathbf{F}_i and S_i are the flux vector and length of interface i . From Eq. (5.10), it can be observed that with the conservative flow variables defined at cell centers, we need to evaluate the numerical flux at all cell interfaces. In the gas kinetic scheme, the connection between the distribution function f and flux vector is

$$\mathbf{F} = \int \mathbf{u} f \varphi_\alpha d\Xi \quad (5.11)$$

where $d\Xi = du dv d\xi$ is the volume element and φ_α is the moment given by

$$\varphi_\alpha = \left(1, \mathbf{u}, \frac{1}{2}(\mathbf{u}^2 + \xi^2) \right) \quad (5.12)$$

To evaluate the flux vector of N-S equations by Eq. (5.11), the distribution function f at the cell interface must be determined first. As is well known, the distribution function f consists of two parts: the equilibrium part and non-equilibrium part, which can be written as

$$f = f^{eq} + f^{neq} \quad (5.13)$$

In another word, to calculate the fluxes \mathbf{F}_i , f^{eq} and f^{neq} should be approximated at the interface between two adjacent control volumes.

Consider f^{eq} first. The equilibrium distribution function $f^{eq}(\mathbf{r}, t + \delta_t)$ is determined by flow variables at the cell interface, which is locally reconstructed by the Boltzmann solution

$$\mathbf{W}(\mathbf{r}, t + \delta_t) = \int f^{eq}(\mathbf{r}, t + \delta_t) \varphi_\alpha d\Xi = \int \int_{\mathbf{u} \cdot \mathbf{n} \geq 0} \varphi_\alpha g_l d\Xi + \int \int_{\mathbf{u} \cdot \mathbf{n} < 0} \varphi_\alpha g_r d\Xi \quad (5.14)$$

where g_l and g_r are the equilibrium distribution functions at left and right sides of the interface; \mathbf{n} is the unit outer normal vector of the cell interface. Once the flow variables are reconstructed by Eq. (5.14), $f^{eq}(\mathbf{r}, t + \delta_t)$ can be determined by using the Maxwellian distribution function.

For the non-equilibrium distribution function, f^{neq} is related to the equilibrium distribution functions at the cell interface and its surrounding points. To the order of N-S equations, f^{neq} can be written as (Xu and He, 2003)

$$f^{neq} = -\tau \left(\frac{\partial}{\partial t} + \mathbf{u} \cdot \nabla \right) f^{eq} \quad (5.15)$$

A second order approximation of f^{neq} by using Taylor series expansion in time and physical space gives

$$f^{neq}(\mathbf{r}, t + \delta_t) = -\frac{\tau}{\delta_t} \left[f^{eq}(\mathbf{r}, t + \delta_t) - f^{eq}(\mathbf{r} - \mathbf{u}\delta_t, t) \right] \quad (5.16)$$

where the equilibrium distribution function $f^{eq}(\mathbf{r} - \mathbf{u}\delta_t, t)$ can be computed by the properties of flow variables at the corresponding position. For the second order interpolation, the equilibrium distribution function can be expressed as

$$f^{eq}(\mathbf{r} - \mathbf{u}\delta_t, t) = \begin{cases} g_l + (\mathbf{r} - \mathbf{u}\delta_t) \cdot \nabla g_l, & \mathbf{u} \cdot \mathbf{n} \geq 0 \\ g_r + (\mathbf{r} - \mathbf{u}\delta_t) \cdot \nabla g_r, & \mathbf{u} \cdot \mathbf{n} < 0 \end{cases} \quad (5.17)$$

By substituting Eqs. (5.16) and (5.17) into Eq. (5.13), the distribution function can be given as

$$f(\mathbf{r}, t + \delta_t) = g(\mathbf{r}, t + \delta_t) - \frac{\tau}{\delta_t} \left[g(\mathbf{r}, t + \delta_t) - g_l \cdot H(\mathbf{u} \cdot \mathbf{n}) - g_r (1 - H(\mathbf{u} \cdot \mathbf{n})) \right] - \tau \mathbf{u} \cdot \nabla \left[g_l \cdot H(\mathbf{u} \cdot \mathbf{n}) + g_r (1 - H(\mathbf{u} \cdot \mathbf{n})) \right] \quad (5.18)$$

As the Cartesian mesh is used in the present IBM, the velocity vector \mathbf{u} becomes to u or v in the local interface, which greatly simplifies the scheme. Since the distribution function has been given in Eq. (5.18), the numerical fluxes across the interface can be computed by Eq. (5.11). The explicit formulations of the conservative variables and numerical fluxes at the cell interface can be referred to Chapter 3 and Appendix B.

5.2.2 Boundary condition-enforced IBM for velocity correction

After the evaluation of intermediate velocity \mathbf{u}^* , the boundary condition-enforced IBM (Wu and Shu, 2009; Wang et al., 2015) is applied to perform the velocity correction. From Eq. (5.9), the corrected velocity in the next time step can be expressed as

$$\mathbf{u}^{n+1} = \mathbf{u}^* + \delta\mathbf{u} \quad (5.19)$$

where the velocity correction $\delta\mathbf{u}$ is

$$\delta\mathbf{u} = \frac{1}{\rho^{n+1}} \mathbf{f} \delta t \quad (5.20)$$

In the conventional IBM, such as penalty force and direct forcing methods, \mathbf{f} is computed in advance. In these methods, the no-slip boundary condition is approximately satisfied and there might be penetration near the boundary. To overcome this drawback, the restoring force \mathbf{f} and correspondingly the velocity correction $\delta\mathbf{u}$ should be treated as unknown. As shown by Wu and Shu (2009), the velocity correction $\delta\mathbf{u}$ is obtained in an implicit way by accurately enforcing the no-slip boundary condition at the Lagrangian points.

In IBM, the boundary of immersed body is represented by a set of Lagrangian points \mathbf{X}_B^l . To guarantee the no-slip boundary condition, the velocity on the Lagrangian point $\mathbf{U}^{n+1}(\mathbf{X}_B^l)$ should be identical to the fluid velocity $\mathbf{u}^{n+1}(\mathbf{x}_{ij})$ at the same position,

$$\mathbf{U}^{n+1}(\mathbf{X}_B^l) = \sum_{ij} \mathbf{u}^{n+1}(\mathbf{r}_{ij}) D_{ij}(\mathbf{r}_{ij} - \mathbf{X}_B^l) h^2 \quad (l=1,2,\dots,m ; ij=1,2,\dots,n) \quad (5.21)$$

where m is the number of Lagrangian points and n is the number of surrounding Eulerian points used in the delta function interpolation. h is the grid size of Eulerian mesh, D_{ij} is a continuous kernel distribution

$$D_{ij}(\mathbf{r}_{ij} - \mathbf{X}_B^l) = \delta(r_{ij}^x - X_B^l) \delta(r_{ij}^y - Y_B^l) \quad (5.22)$$

where $\delta(r)$ was proposed by Peskin (2002)

$$\delta(r) = \begin{cases} \frac{1}{4} \left(1 + \cos\left(\frac{\pi|r|}{2}\right) \right), & |r| \leq 2 \\ 0, & |r| > 2 \end{cases} \quad (5.23)$$

In addition, by setting an unknown velocity correction vector $\delta\mathbf{u}_B^l$ at every boundary point, the Eulerian velocity correction $\delta\mathbf{u}$ can be interpolated via the Dirac delta function

$$\delta\mathbf{u}(\mathbf{r}_{ij}) = \sum_l \delta\mathbf{u}_B^l \Delta s^l D(\mathbf{r}_{ij} - \mathbf{X}_B^l) \quad (l=1,2,\dots,m) \quad (5.24)$$

Substituting Eq. (5.19) and Eq. (5.24) into Eq. (5.21), the following relationship can be obtained

$$\mathbf{U}^{n+1}(\mathbf{X}_B^l) = \sum_{ij} \mathbf{u}^*(\mathbf{r}_{ij}) D_{ij}(\mathbf{r}_{ij} - \mathbf{X}_B^l) h^2 + \sum_{ij} \sum_l \delta\mathbf{u}_B^l \Delta s^l D(\mathbf{r}_{ij} - \mathbf{X}_B^l) D(\mathbf{r}_{ij} - \mathbf{X}_B^l) h^2 \quad (5.25)$$

The above Eq. (5.25) can be written as the following matrix form

$$\mathbf{AX} = \mathbf{B} \quad (5.26)$$

where

$$\mathbf{X} = \{\delta \mathbf{u}_B^1 \Delta s^1, \delta \mathbf{u}_B^2 \Delta s^2, \dots, \delta \mathbf{u}_B^m \Delta s^m\}^T \quad (5.27)$$

$$\mathbf{A} = h^2 \begin{pmatrix} D_{11} & D_{12} & \cdots & D_{1n} \\ D_{21} & D_{22} & \cdots & D_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ D_{m1} & D_{m2} & \cdots & D_{mn} \end{pmatrix} \begin{pmatrix} D_{11} & D_{12} & \cdots & D_{1m} \\ D_{21} & D_{22} & \cdots & D_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ D_{n1} & D_{n2} & \cdots & D_{nm} \end{pmatrix} \quad (5.28)$$

$$\mathbf{B} = \begin{pmatrix} \mathbf{U}_B^1 \\ \mathbf{U}_B^2 \\ \vdots \\ \mathbf{U}_B^m \end{pmatrix}^{n+1} - \begin{pmatrix} D_{11} & D_{12} & \cdots & D_{1n} \\ D_{21} & D_{22} & \cdots & D_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ D_{m1} & D_{m2} & \cdots & D_{mn} \end{pmatrix} \begin{pmatrix} \mathbf{u}_1^* \\ \mathbf{u}_2^* \\ \vdots \\ \mathbf{u}_n^* \end{pmatrix} \quad (5.29)$$

It should be noted that the number of unknowns in Eq. (5.26) is the same as the number of Lagrangian points. By solving the equation system (5.26) with a direct method or iterative method, the unknown velocity correction $\delta \mathbf{u}_B^l \Delta s^l$ at all Lagrangian points can be obtained simultaneously. Then, the velocity correction $\delta \mathbf{u}$ and corrected velocity \mathbf{u}^{n+1} at the Eulerian points can be calculated by Eq. (5.24) and Eq. (5.19), respectively.

5.2.3 Computational sequence and force calculation

The basic solution procedure of the present boundary condition-enforced immersed boundary-gas kinetic flux solver can be summarized as follows:

- (1) Initially, the derivatives of conservative flow variables are calculated at each cell center and the initial reconstructions are conducted at two sides of cell interface.
- (2) The maximum velocity of particles in the phase velocity space U_{\max} is chosen properly. Then, the streaming time step δ_t is specified at each interface. The constraint for choosing δ_t is that the location of $g(\mathbf{r} - \mathbf{u}\delta_t, t)$ should be within

either the left or right cell of the interface.

- (3) Apply the Maxwellian distribution function to calculate g_l and g_r at the left and right sides of the cell interface.
- (4) Calculate the conservative flow variables at the cell interface W by Eq. (5.14), and then $f^{eq}(\mathbf{r}, t + \delta_t)$ can be determined.
- (5) Compute $f^{neq}(\mathbf{r}, t + \delta_t)$ by using Eq. (5.16).
- (6) Compute the distribution function $f(\mathbf{r}, t + \delta_t)$ by Eq. (5.18), and further compute the fluxes at each cell interface F by Eq. (5.11).
- (7) Once the fluxes at all cell interfaces are computed, predict the intermediate velocity \mathbf{u}^* by solving Eq. (5.10).
- (8) Solve linear system (5.26) to get the velocity correction at all Lagrangian points.
- (9) Use Eq. (5.19) and Eq. (5.24) to perform velocity correction. After this process, the flow variables in the next time step ρ^{n+1} , \mathbf{u}^{n+1} , p^{n+1} are obtained.
- (10) Repeat steps (1)-(9) until convergence criterion is satisfied.

It should be highlighted that the calculation of boundary force on the immersed boundary is quite convenient in the current method. According to Eq. (5.20), the force acting on the immersed boundary can be computed from velocity correction

$$\mathbf{F} = -\sum_l \frac{\rho_f \delta \mathbf{u}_B^l \Delta s^l}{\Delta t} \quad (5.30)$$

where ρ_f is the mean density of the fluid, $\delta \mathbf{u}_B^l \Delta s^l$ is the velocity correction obtained from Eq. (5.26). However, the above equation for calculation of force is not accurate for accelerating object. When the object undergoes an accelerating motion, additional terms related to the inertial effects or the internal mass effects of the moving object

should be taken into consideration (Suzuki and Inamuro, 2011). A more accurate expression to calculate the boundary force are given as

$$\mathbf{F}_f = -\sum_l \frac{\rho_f \delta \mathbf{u}_B^l \Delta s^l}{\Delta t} + \rho_f V_B \mathbf{a}_B \quad (5.31)$$

where V_B and \mathbf{a}_B are the area and the acceleration of the object, respectively. The second part of force in Eq. (5.31) is the internal mass effects caused by the fluid inside the object. It is set to zero in the stationary boundary problem.

5.3 Numerical example and discussion

In this section, the reliability and accuracy of the proposed IB-GKFS are tested by simulating several stationary and moving boundary flows. First, flows past stationary boundary problems are simulated, including flows over a circular cylinder and a NACA0012 airfoil. After that, more complex moving boundary problems, such as flows past a moving cylinder and a NACA0015 airfoil with fixed body trajectory, are solved. Then, a typical fluid-structure interaction (FSI) test case of one particle sedimentation in a rectangular box is simulated.

5.3.1 Flow past a stationary circular cylinder

The flow past a stationary circular cylinder is first chosen to validate the proposed method. As a benchmark case, this problem has been widely studied and there are plenty of numerical and experimental results available. The flow behaviors can be characterized by the Reynolds number, which is defined as $Re = \rho U_\infty D / \mu$, where ρ is the free stream density, U_∞ is the free stream velocity, D is the diameter of the cylinder and μ is the dynamic viscosity of the fluid. In the present simulation, these

variables are set as $\rho=1.0$, $U_\infty=0.1$ and $D=1.0$. The circular cylinder is represented by 150 Lagrangian points with a uniform distribution. Free stream flow properties are applied at the left boundary and natural boundary conditions are used on upper, lower and right boundaries. To obtain the high resolution near the boundary and save the computational effort in the meantime, a non-uniform mesh is adopted in the current work, in which a fine mesh is applied around the cylinder and a coarse mesh is used near the far field boundaries. The computational domain is $[-20D, 30D] \times [-20D, 20D]$ with the whole mesh size of 261×236 . The region around the cylinder is $1.2D \times 1.2D$ with a uniform mesh size of 97×97 .

Firstly, the steady state ($Re = 20$ and 40) is considered. When the flow reaches steady state, a pair of stationary recirculation vortices is developed behind the cylinder. Figure 5.1 shows the streamlines when the flow reaches the final steady state with $Re = 20$ and 40 . It can be clearly seen that there is no penetration near the boundary surface, which indicates that the no-slip boundary condition is accurately satisfied in the present method. The drag coefficient and the vortex length obtained by the present method are compared with previous computational results (Dennis and Chang, 1970; Shukla et al., 2007; Wu and Shu, 2009; Yuan et al., 2015) in Table 5.1. From the table, it shows that the present results agree well with those in the literature. When Reynolds number is increased to 100 and 200 , unsteady periodic flow occurs. The time evolutions of drag and lift coefficients on the cylinder for two cases are plotted in Figure 5.2. Table 5.2 quantitatively compares the lift and drag coefficients and the Strouhal number for two cases with those in the literature (Braza et al., 1986; Benson et al., 1989; Ding et al., 2004; Wang et al., 2015). Once again, good agreement can be found in the table.

To investigate the computational efficiency of the proposed IB-GKFS, an additional numerical test is carried out for flow past a stationary circular cylinder at $Re = 20$. The current results are compared with those of Wang et al. (2015), in which the flow field is obtained by a lattice Boltzmann flux solver (LBFS). The same computational mesh and Lagrangian points are adopted in both simulations. To compare the computational efficiency of two solvers, the CPU times required to get the same converged state are tested. The results show that the current IB-GKFS takes about 938.92 s with 47000 iteration numbers while the IB-LBFS takes about 775.47 s with 46200 iteration numbers. The present solver takes about 1.21 times of CPU time of IB-LBFS for the considered case. However, it should be noted that the IB-LBFS (Wang et al., 2015) is applicable only to incompressible flows due to incompressible limit of lattice Boltzmann method. In contrast, the current GKFS has a potential for a wide range of applications, including incompressible and compressible flows.

5.3.2 Flow over a NACA0012 airfoil

Apart from the cylinder, the present scheme can also be applied to more complicated geometries, such as airfoils. Here, incompressible flow over a NACA0012 airfoil is selected to further validate the present scheme. In this problem, the Reynolds number is taken as $Re = 500$ and the angle of attack is chosen as $\alpha = 0^\circ$. The free stream condition is given by fluid density $\rho = 1.0$ and velocity $U_\infty = 0.1$. The airfoil surface is represented by 160 Lagrangian points in a uniform distribution. Initially, the flow field is given by the free-stream density and velocity.

Figure 5.3 shows the pressure contours together with the streamlines around the airfoil, which shows that the no-slip boundary condition is accurately satisfied. Figure 5.4 indicates the velocity profiles of the boundary layer at $x=0.00$, 0.25 , 0.50 , 0.75 and 1.00 of the chord length, respectively. The results obtained by GILBM (Imamura et al., 2004) are also included for comparison and good agreement can be found between these two numerical results. Moreover, the drag and lift coefficients are also compared. In the current simulation, the drag coefficient is $C_d = 0.17442$ and lift coefficient is $C_l = 0.766 \times 10^{-10}$. They are close to the results of Lockard et al. (2002), in which the coefficients are $C_d = 0.17618$ and $C_l = 0.115 \times 10^{-6}$.

5.3.3 Flow past an in-line oscillating cylinder

The case of an in-line oscillating cylinder in a fluid at rest is one of the benchmark cases in the moving boundary problem and has been investigated both experimentally (DÜTsch et al., 1998) and numerically (Wang et al., 2009; DÜTsch et al., 1998; Yang and Balaras, 2006; Zhong et al., 2013) in many studies. Two key parameters describing this problem are the Reynolds number

$$Re = \frac{\rho U_{max} D}{\mu} \quad (5.32)$$

and the Keulegan-Carpenter number

$$KC = \frac{U_{max}}{f \cdot D} \quad (5.33)$$

where U_{max} is the maximum velocity of the oscillating cylinder, D is the diameter of the cylinder and f is the characteristic frequency of the oscillation. The above two parameters are set to $Re = 100$ and $KC = 5$ respectively, which correspond to the

LDA experiments and numerical simulations in the study of DÜTsch et al. (1998).

The cylinder's translational motion is given by a simple harmonic oscillation

$$x(t) = -A \sin(2\pi ft) \quad (5.34)$$

where A is the amplitude of the oscillation. The computational domain is set as $50D \times 30D$ and the total computational mesh size is 600×420 . The cylinder represented by 150 Lagrangian points with a uniform distribution is located at the center in the beginning. Natural boundary conditions are applied at all far-field boundaries.

Figure 5.5 shows the vorticity contours at four different phase angles ($0^\circ, 96^\circ, 192^\circ, 288^\circ$). When the phase angle is 0° , the cylinder moves to the left and a pair of counter-rotating vortices are formed. After the cylinder reaches the extreme left location, the vortex generation procedure stops. The same periodic vortex shedding process can be found on its right side. The above observations correspond to the studies of DÜTsch et al. (1998) and Yang and Balaras (2006). To make the quantitative comparison, the computed velocity profiles at four different x locations and three different phase angles are displayed in Figure 5.6. The experimental results of DÜTsch et al. (1998) and numerical results of Wang et al. (2009) are also displayed for comparison and good agreement can be found between them. Figure 5.7 depicts dimensionless in-line force $F_x(t)$ variation with time acting on the cylinder in a period of the oscillation. The good agreement between the present results and the reference data (DÜTsch et al., 1998, Wang et al., 2009) indicates that the present scheme can accurately predict the forces acting on the solid boundaries.

5.3.4 Flow past a transverse oscillating cylinder

To further validate the present scheme for moving boundary flows, the problem of flow past a transverse oscillating cylinder is considered. The transverse motion of the cylinder center is a harmonic oscillation given as

$$y(t) = A \sin(2\pi f_e t) \quad (5.35)$$

where $A = 0.2$ is the oscillating amplitude and f_e is the oscillating frequency. In this study, five different oscillating frequencies are considered, which are $f_e/f_o = 0.8, 0.9, 1.0, 1.1$ and 1.2 , respectively. Here, f_o is the natural vortex shedding frequency for flow past a stationary cylinder at $Re = 185$. The other parameters are chosen as: Reynolds number $Re = 185$, density of the fluid $\rho = 1.0$ and the diameter of the cylinder $D = 1.0$. Similar to the previous section, the flow domain is chosen as $[-20D, 30D] \times [-20D, 20D]$ with the whole mesh size of 493×460 . The cylinder is initially located at $(x = 0, y = 0)$ surrounded by a sub-domain of $[-0.8D, 0.8D] \times [-0.8D, 0.8D]$ with a uniform mesh size of 161×161 .

Figure 5.8 shows the time evolution of drag and lift coefficients of the present five different cases. It can be noted that drag and lift coefficients behave a simple harmonic oscillation once vortex shedding is established when f_e/f_o is equal to or less than 1.0. As the values of f_e/f_o increases, the magnitudes of force coefficients are enlarged. For values of f_e/f_o greater than 1.0, both the drag and lift coefficients exhibit modulation phenomenon. Figure 5.9 depicts the time-averaged drag coefficient $\overline{C_d}$, the root-mean-square values of the drag coefficient $C_{d,rms}$ and lift coefficient $C_{l,rms}$. The results of Guilmineau and Queutey (2002) and Wang et al.

(2015) are also displayed for comparison. Good agreement can be found between the present results and the reference data. The instantaneous streamlines when the oscillating cylinder is at the extreme upper position are shown in Figure 5.10. From the figure, no penetration is found around the cylinder, which shows the excellent satisfaction of the no-slip boundary condition. Furthermore, it is found that the streamline topologies are similar to each other when $f_e/f_o \leq 1.0$ and the saddle points will appear in the form of intersecting streamlines when $f_e/f_o > 1.0$. The above phenomenon is in correspondence with the observation in literature (Guilmineau and Queutey, 2002).

5.3.5 Laminar flow around a rapid pitching NACA0015 airfoil

Next, a more complex moving boundary problem of laminar flow around a rapidly pitching NACA0015 airfoil is simulated. In this case, the Reynolds number based on chord length and freestream velocity is $Re = 10,000$ and Mach number $M_\infty = 0.2$.

The pitch rate is

$$\Omega(t) = \Omega_0 (1 - \exp(-4.6t/t_0)) \text{ rad/s} \quad (5.36)$$

with $\Omega_0 c / u_\infty = 0.6$ and $t_0 u_\infty / c = 1$, where c and u_∞ are the chord length of airfoil and the freestream velocity, respectively. The pitch axis is located at the quarter chord. The flow field computed at zero-degree angle of attack is used as initial condition. As time goes on, the pitch rate will increase and reach 99% of its final rate Ω_0 at $t = t_0$ and thereafter the pitch rate is nearly constant.

The computational domain is selected as $50c \times 30c$ with a non-uniform mesh. A uniform mesh with mesh spacing $h = 0.001c$ is applied near the airfoil. 800

Lagrangian points are adopted to represent the airfoil surface. Figure 5.11 shows the present results of time-dependent lift and drag coefficients versus the angle of attack. The results of Visbal and Shang (1989) and Lomtev et al. (1999) are also included for comparison, both of which used a boundary-conforming mesh in their simulations. The present results are in good agreement with the reference data at angle of attack $\alpha \leq 40^\circ$. At $\alpha \approx 40^\circ$, there is some deviation between the present lift coefficient and the reference data. This can be explained by the strong dependence of the computed forces on the resolution of small flow structures at higher angles of attack. Figure 5.12 presents a description of the most crucial features of the flow past a pitching airfoil. At the initial condition ($\alpha = 0^\circ$), the flow is symmetric and displays a small trailing-edge separation region. As the pitching motion begins, the flow becomes fully attached along the lower surface of the airfoil. Counterclockwise vortices are formed and shed from the trailing-edge of lower surface into the wake, resulting in the increase of lift. On the other hand, on the upper surface of the airfoil, the near-wake experiences significant curvature due to the counterclockwise vortices ($\alpha = 22^\circ$). With the increase of airfoil incidence, the separation point on the upper surface moves upstream and eventually reaches the leading-edge area ($\alpha = 32^\circ$). There are two distinguishable vortical structures on the upper surface, e.g. the leading-edge vortex and the shear layer vortex ($\alpha = 32^\circ$). The leading-edge vortex grows in size and its center moves downstream ($\alpha = 44^\circ$). The shear layer vortex impinges on the airfoil surface due to the mutual influence of leading-edge and trailing-edge vortices ($\alpha = 44^\circ$). The leading-edge vortex detaches with the continuous increase in airfoil incidence ($\alpha = 52^\circ$). The basic flow structure is in qualitative agreement with experimental (Helin and Walker, 1985) and numerical (Visbal and Shang, 1989) observations.

5.3.6 One particle sedimentation in a rectangular domain

Another problem we select to further test the capability of the present method in solving moving boundary problems is the one particle sedimentation in a rectangular domain. This problem is a typical fluid-structure interaction problem (Zhong et al., 2013; Lee et al., 2013) and has been extensively studied (Feng and Michaelides, 2004; Wan and Turek, 2006; Wu and Shu, 2010; Ren et al., 2012; Wang et al., 2015). In the present simulation, the rectangular domain with 2 cm width and 6 cm height is selected. Inside the domain, the viscosity μ and density ρ_f of fluid are chosen as $\mu = 0.1\text{ g}/(\text{cm}\cdot\text{s})$ and $\rho_f = 1.0\text{ g}/\text{cm}^3$, respectively. The density of the rigid particle is $\rho_p = 1.25\text{ g}/\text{cm}^3$ and its radius is $r_p = 0.125\text{ cm}$. Initially, the particle is placed at the location of $(1\text{ cm}, 4\text{ cm})$ and both the fluid and particle are at the static state. A uniform grid of 201×601 is adopted in this simulation.

Once the particle is released, it will fall down due to the gravity force. Figure 5.13 displays the time evolutions of longitudinal coordinate of particle center y_p , longitudinal velocity of particle center v_p , Reynolds number of particle Re_p and translational kinetic energy E_t . Here, the Reynolds number Re_p and kinetic energy E_t are defined as

$$Re_p = \frac{2\rho_p r_p \sqrt{u_p^2 + v_p^2}}{\mu} \quad (5.37)$$

$$E_t = 0.5\rho_p \pi r_p^2 (u_p^2 + v_p^2) \quad (5.38)$$

where u_p and v_p are velocity components of particle center. Also included in the figure are the results of Wan and Turek (2006) and Wang et al. (2015), which used the finite element method and lattice Boltzmann flux solver, respectively. It is clear that good agreement can be found between the present results and the benchmark solutions. Figure 5.14 illustrates the time evolution of instantaneous vorticity contour around the falling particle. The temporal evolution of the particle and the vortex can be clearly seen. The above results show that the present solver can be effectively used to simulate moving boundary flow problems.

5.4 Concluding remarks

In this chapter, the boundary condition-enforced immersed boundary-gas kinetic flux solver (IB-GKFS) was proposed for the simulation of both stationary and moving boundary flows. In IB-GKFS, by applying the fractional-step technique, the solution process was decoupled into two steps: the predictor step and corrector step. Firstly, in the predictor step, the intermediate flow field was predicted by applying the GKFS, which reconstructs the fluxes using the continuous Boltzmann solutions. As the solid boundary was not taken into account in this step, the external forcing term was avoided during the evaluation of numerical flux at each cell interface, which greatly simplifies the implementation. Subsequently, to guarantee the no-slip boundary condition, the intermediate velocity field was corrected by using the implicit boundary condition-enforced immersed boundary method. The above procedure avoids the iterative process in the work of Yuan et al. (2015) when implementing the no-slip boundary condition and thus the numerical flux across the cell interface only needs to be calculated once at each time step. This makes the current scheme be straightforward and easy to be implemented. With simple Cartesian mesh and flexible

boundary condition treatment, the IB-GKFS can be conveniently applied to solve complex and moving boundary problems. Several numerical cases were simulated to validate the present scheme, including the flow past a stationary circular cylinder and a NACA0012 airfoil, flow past an in-line and transverse oscillating cylinder with a prescribed motion, laminar flow around a rapid pitching NACA0015 airfoil and one particle sedimentation in a rectangular domain. The present numerical results are in good agreement with available data in the literature, which demonstrates the good capability of the present scheme in simulating flows with both stationary and moving boundaries.

In the next chapter, the extension of IBM in the framework of GKFS to simulate compressible flows will be presented. The mechanisms to correct all the flow variables for the compressible flows will be introduced in detail.

Table 5.1 Comparison of drag coefficients, recirculation lengths and separation angles for steady flow past a circular cylinder at $Re=20$ and 40

<i>Re</i>	Authors	C_d	<i>L/D</i>
20	Dennis and Chang (1970)	2.05	0.94
	Shukla et al. (2007)	2.07	0.92
	Wu and Shu (2009)	2.05	0.94
	Yuan et al. (2015)	2.071	0.937
	Present	2.07	0.94
40	Dennis and Chang (1970)	1.52	2.35
	Shukla et al. (2007)	1.55	2.34
	Wu and Shu (2009)	1.554	2.31
	Yuan et al. (2015)	1.548	2.286
	Present	1.546	2.36

Table 5.2 Comparison of dynamic parameters for unsteady flow past a circular cylinder at $Re=100$ and 200

<i>Re</i>	Authors	C_l	C_d	<i>St</i>
100	Braza et al. (1986)	± 0.30	1.28 ± 0.02	0.16
	Benson et al. (1989)	± 0.38	1.46 ± 0.01	0.17
	Ding et al. (2004)	± 0.28	1.325 ± 0.008	0.164
	Wang et al. (2015)	± 0.37	1.334 ± 0.012	0.163
	Present	± 0.348	1.367 ± 0.01	0.164
200	Braza et al. (1986)	± 0.78	1.38 ± 0.07	0.19
	Benson et al. (1989)	± 0.65	1.45 ± 0.04	0.193
	Ding et al. (2004)	± 0.60	1.327 ± 0.045	0.196
	Wang et al. (2015)	± 0.75	1.43 ± 0.051	0.195
	Present	± 0.714	1.370 ± 0.049	0.195

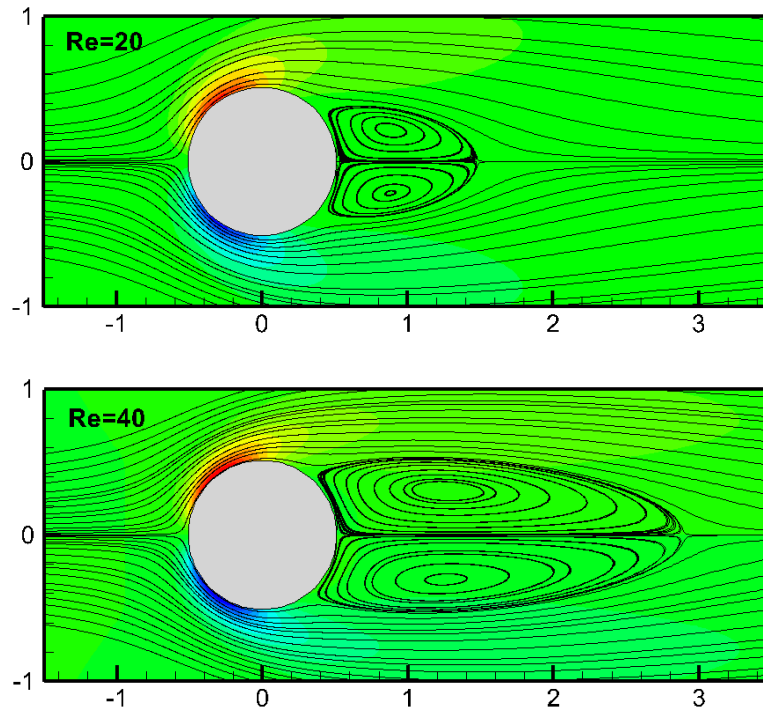


Figure 5.1 Streamlines for the flow over a circular cylinder at $Re=20$ and 40

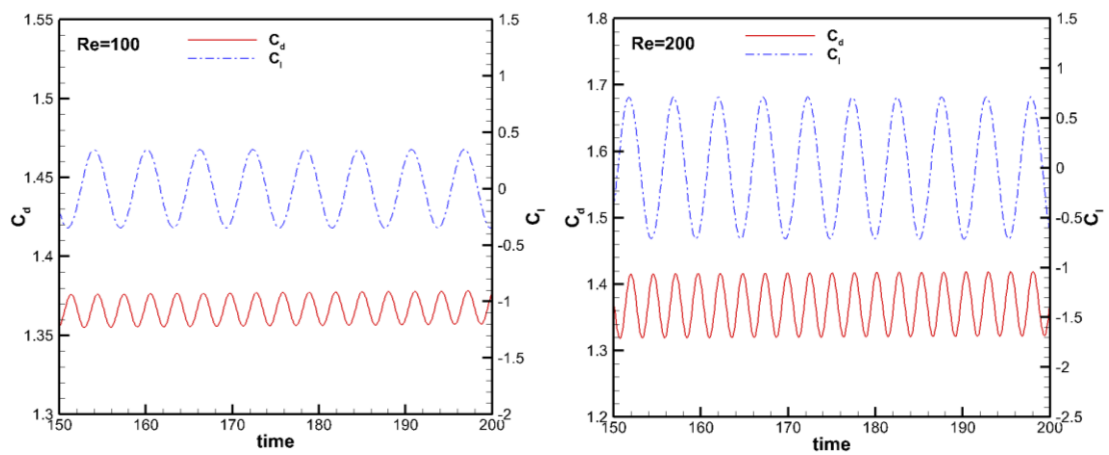


Figure 5.2 Evolution of drag and lift coefficients for flow over a cylinder at $Re=100$ and 200

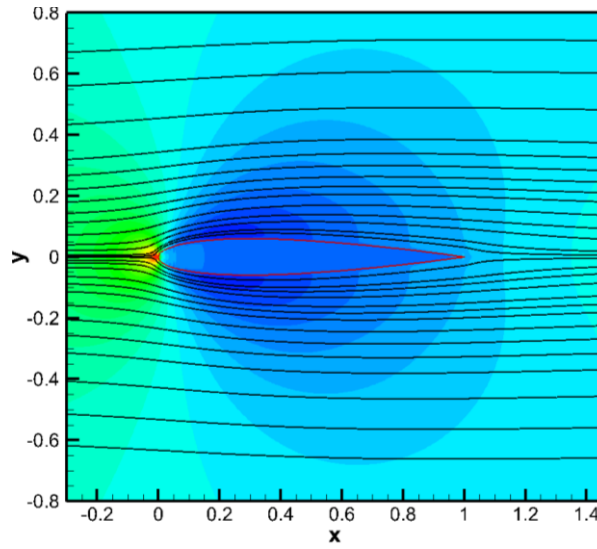
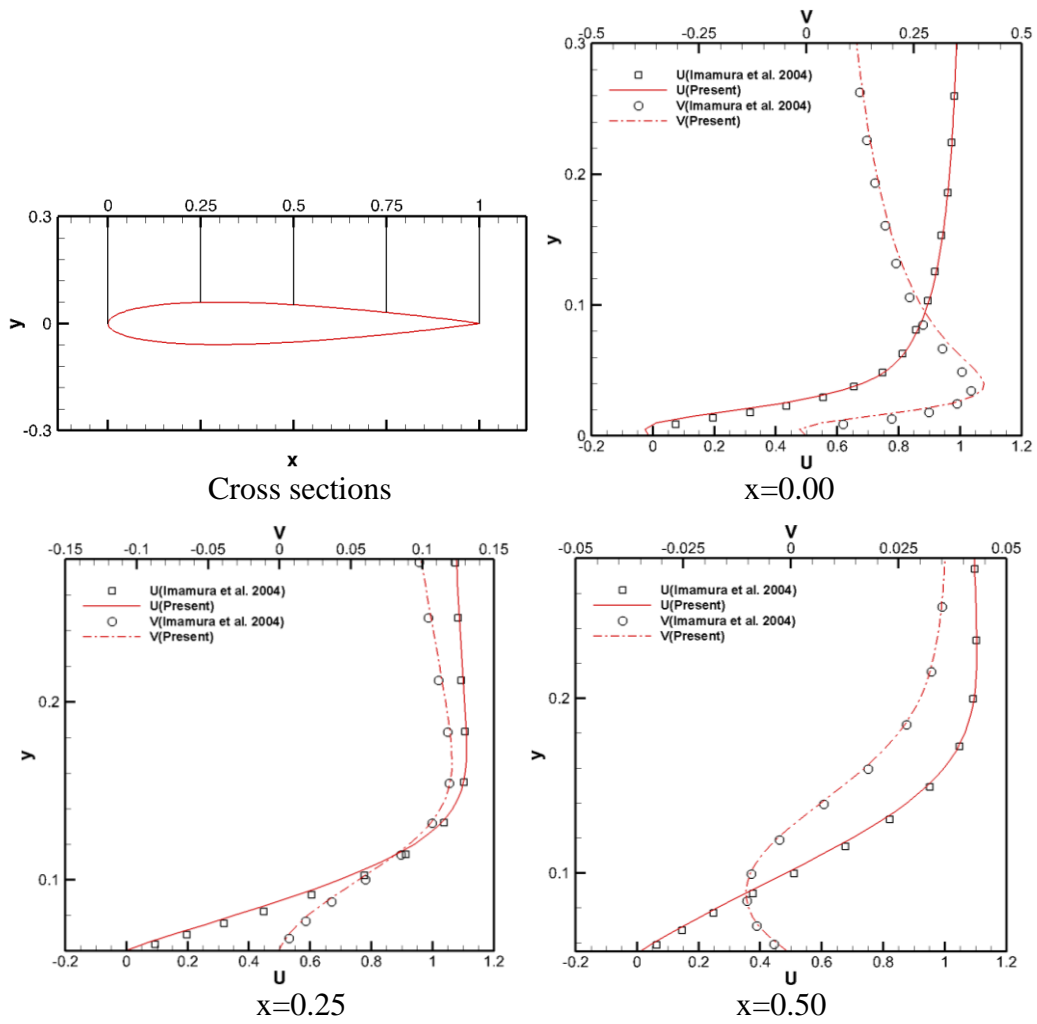


Figure 5.3 Pressure contours and streamlines for flow over a NACA0012 airfoil at $Re=500$ and $AoA=0^\circ$.



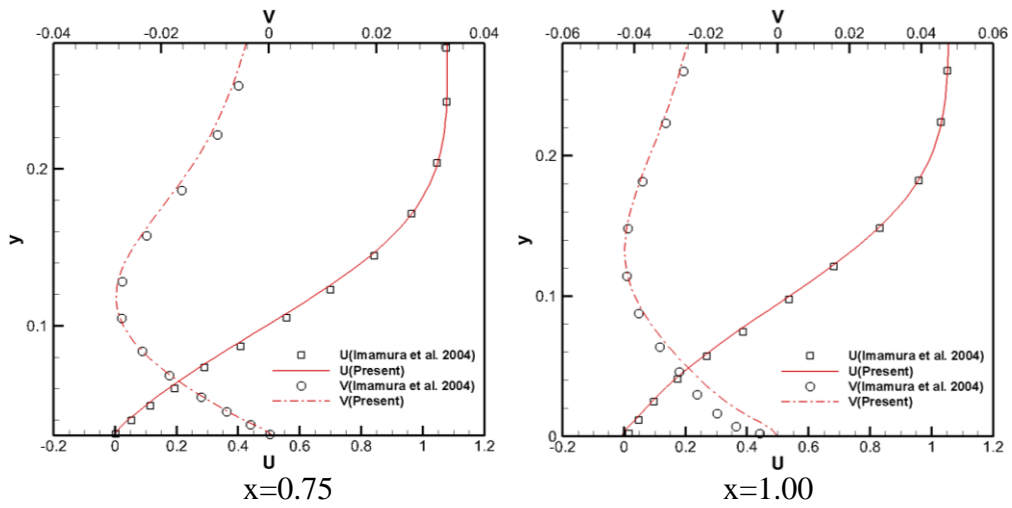


Figure 5.4 Velocity profiles at five cross sections of NACA0012 airfoil

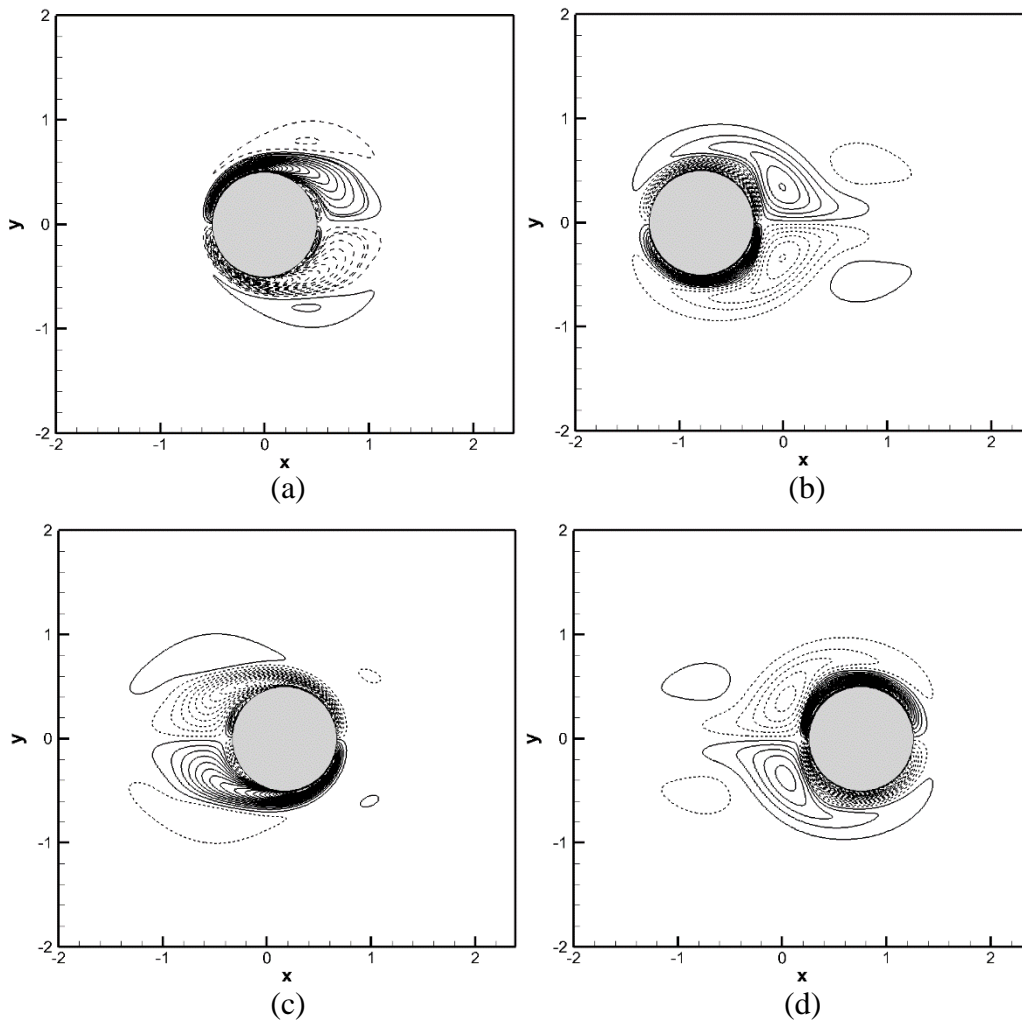
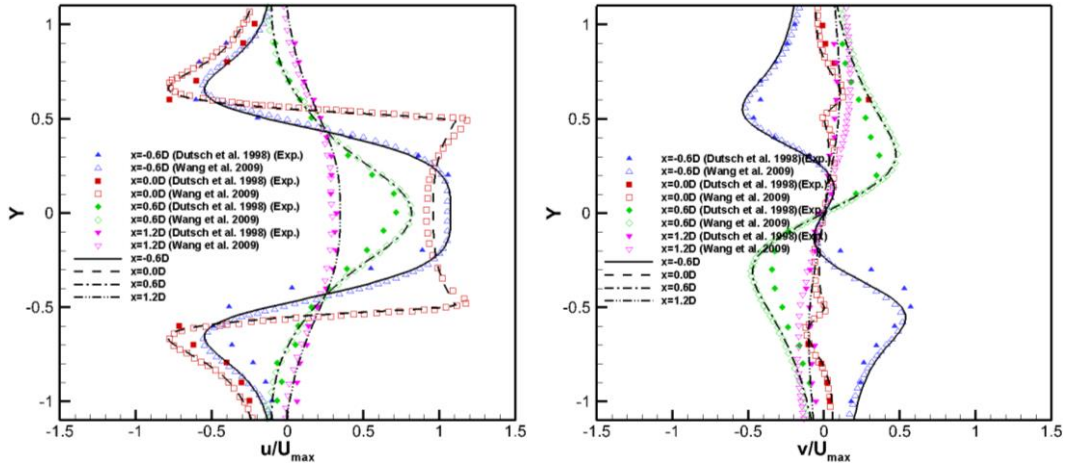
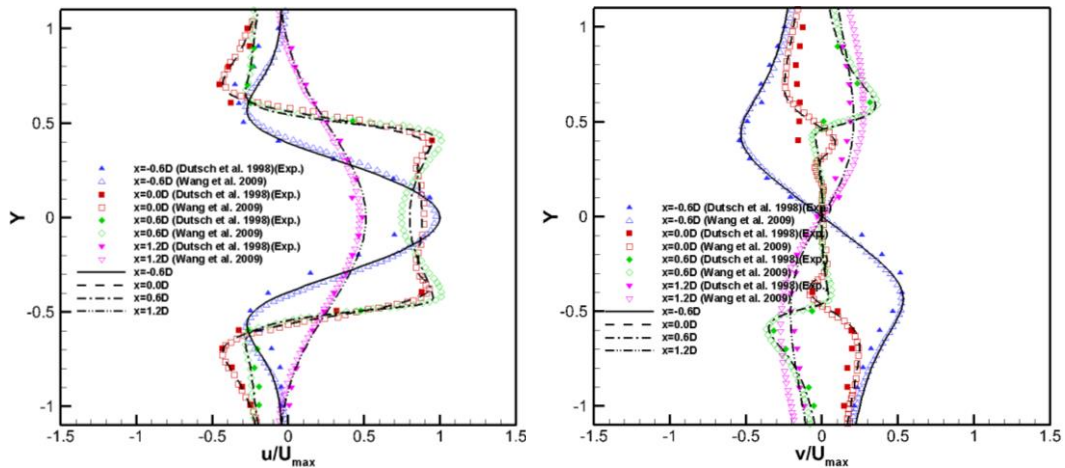


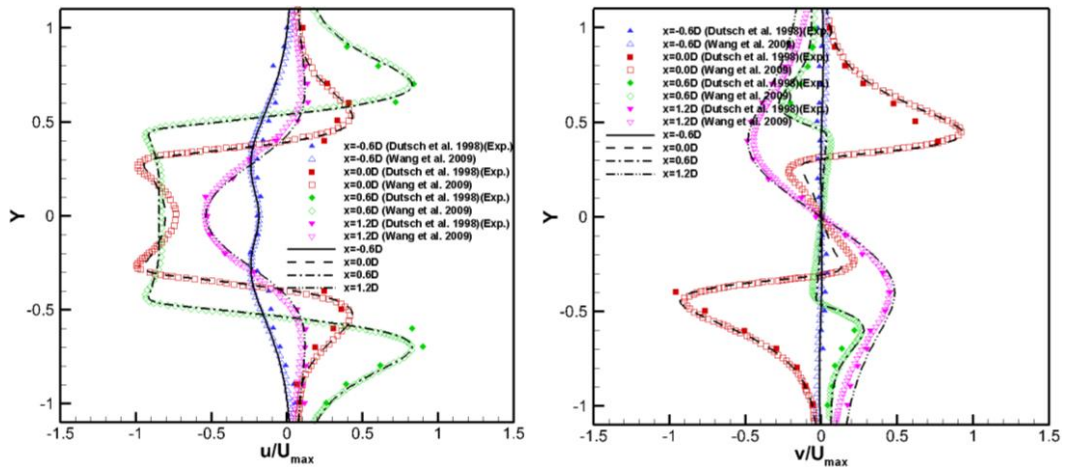
Figure 5.5 Vorticity contours at four different phase-angles for in-line oscillating cylinder in a fluid at rest. (a) 0° ; (b) 96° ; (c) 192° ; (d) 288°



(a) $\phi = 2\pi ft = 180^\circ$



(b) $\phi = 2\pi ft = 210^\circ$



(c) $\phi = 2\pi ft = 330^\circ$

Figure 5.6 Comparison of velocity profiles (u -component in the left column, v -component in the right column) at four different x locations and three different phase angles. (Lines are the present results, filled symbols are the experimental results of DÜTsch et al. (1998), and empty symbols represent numerical results of Wang et al. (2009))

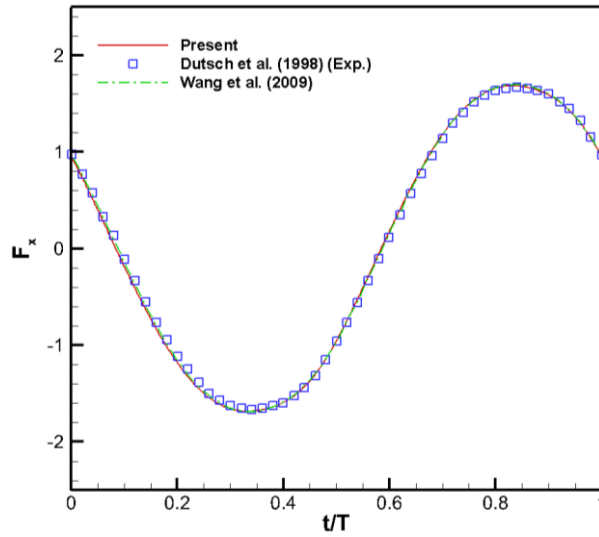
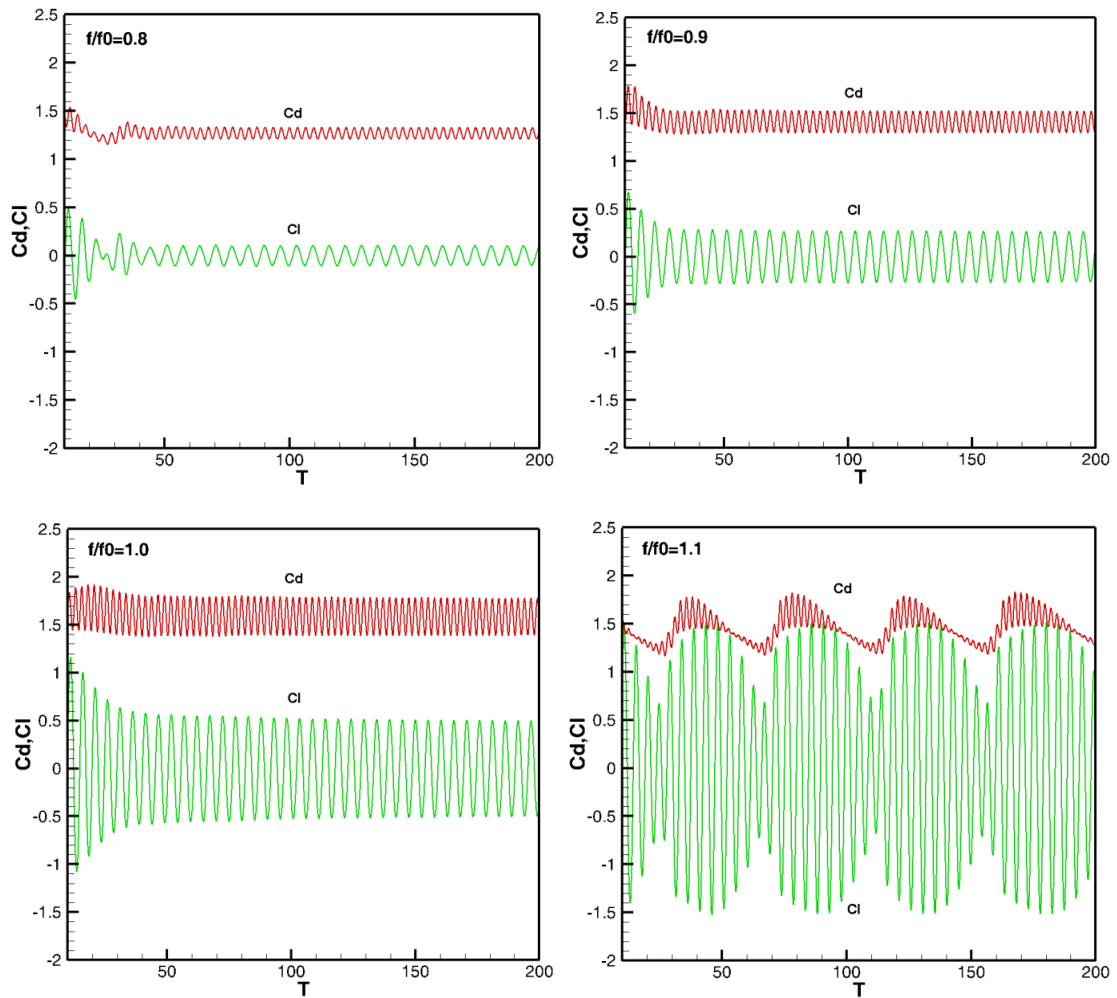


Figure 5.7 Comparison of time evolution of in-line force F_x in a period time T for flow past an in-line oscillating cylinder at $Re=100$ and $KC=5$.



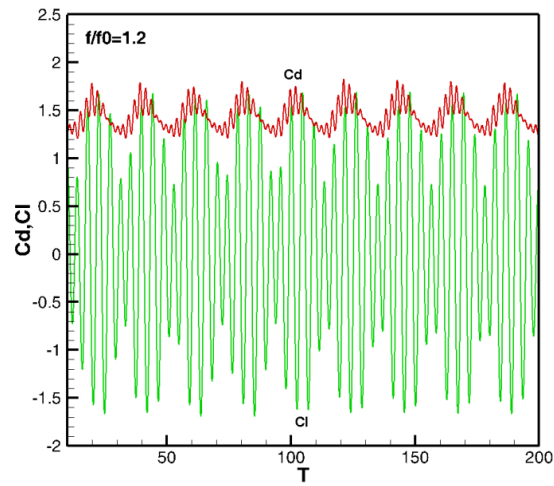


Figure 5.8 Lift and drag coefficients for flow past an oscillating cylinder at $Re=185$

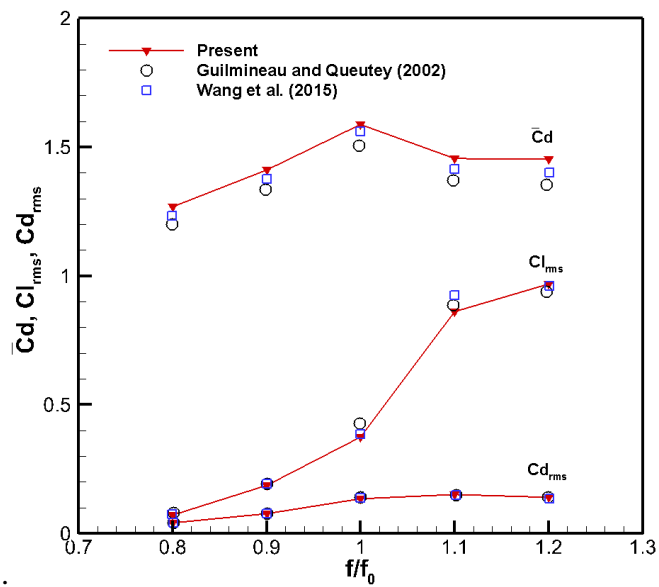
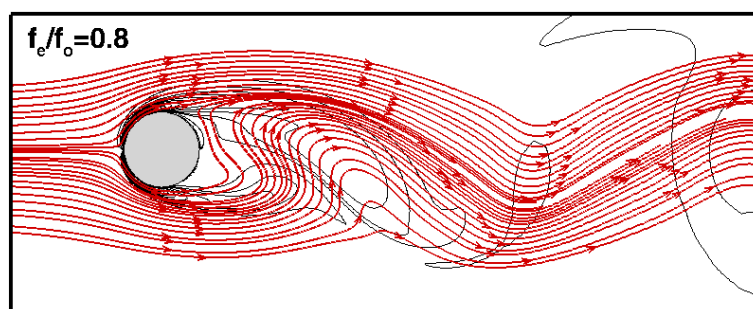


Figure 5.9 Comparison of $\overline{C_d}$, Cl_{RMS} and Cd_{RMS} for flow past a transverse oscillating cylinder at $Re=185$



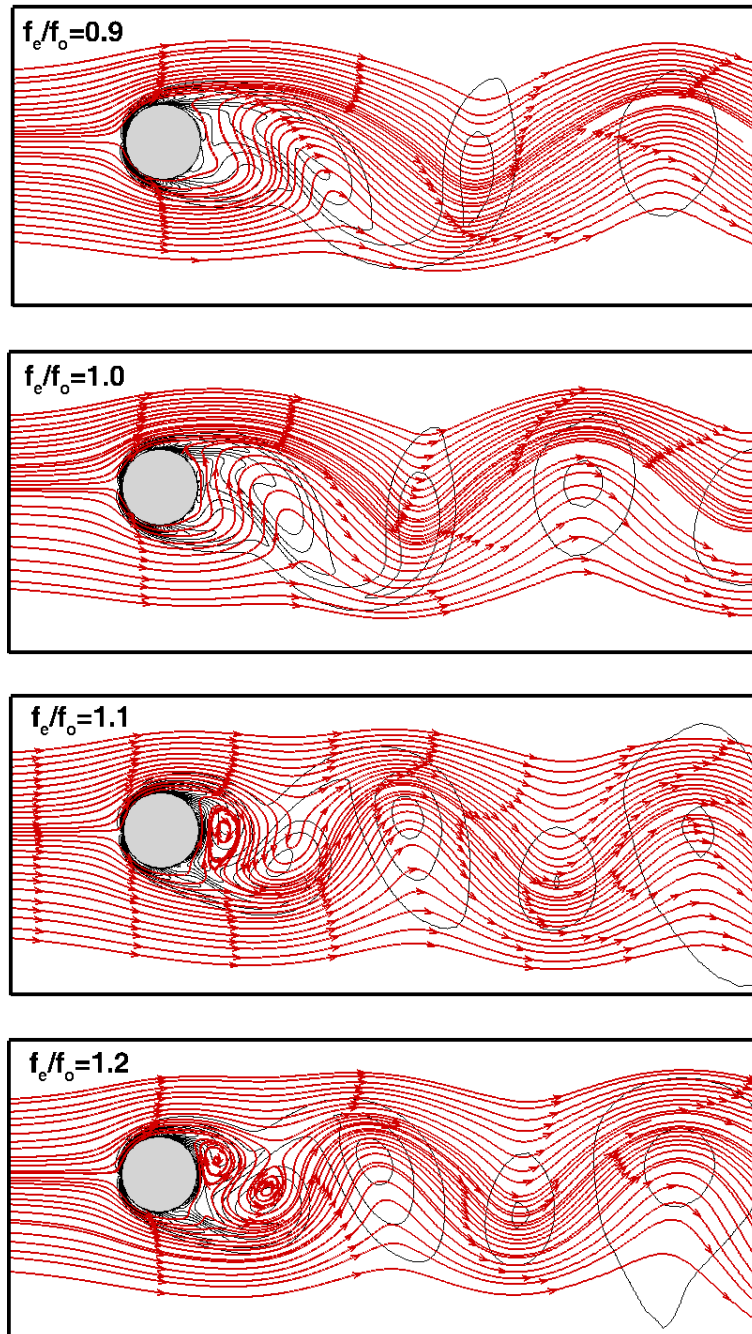


Figure 5.10 Streamlines and vorticity contours for flow past a transverse oscillating cylinder at $Re=185$

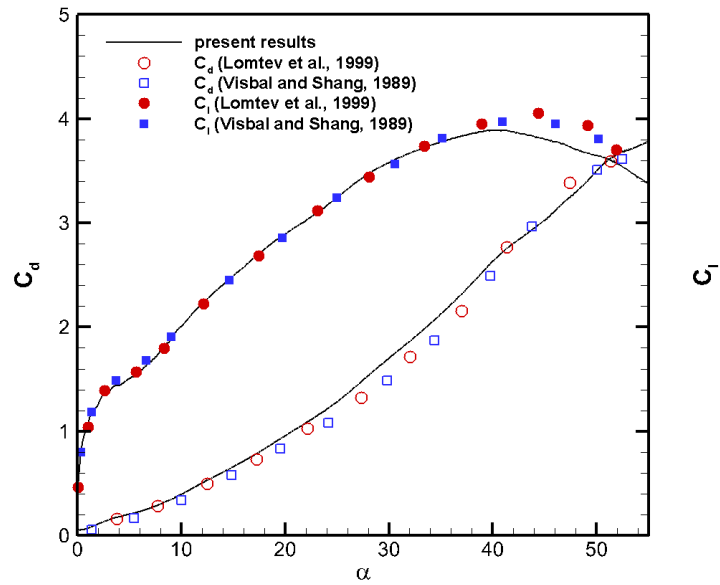


Figure 5.11 Lift (Upper curve) and drag (Lower curve) coefficients versus angle of attack in degrees.

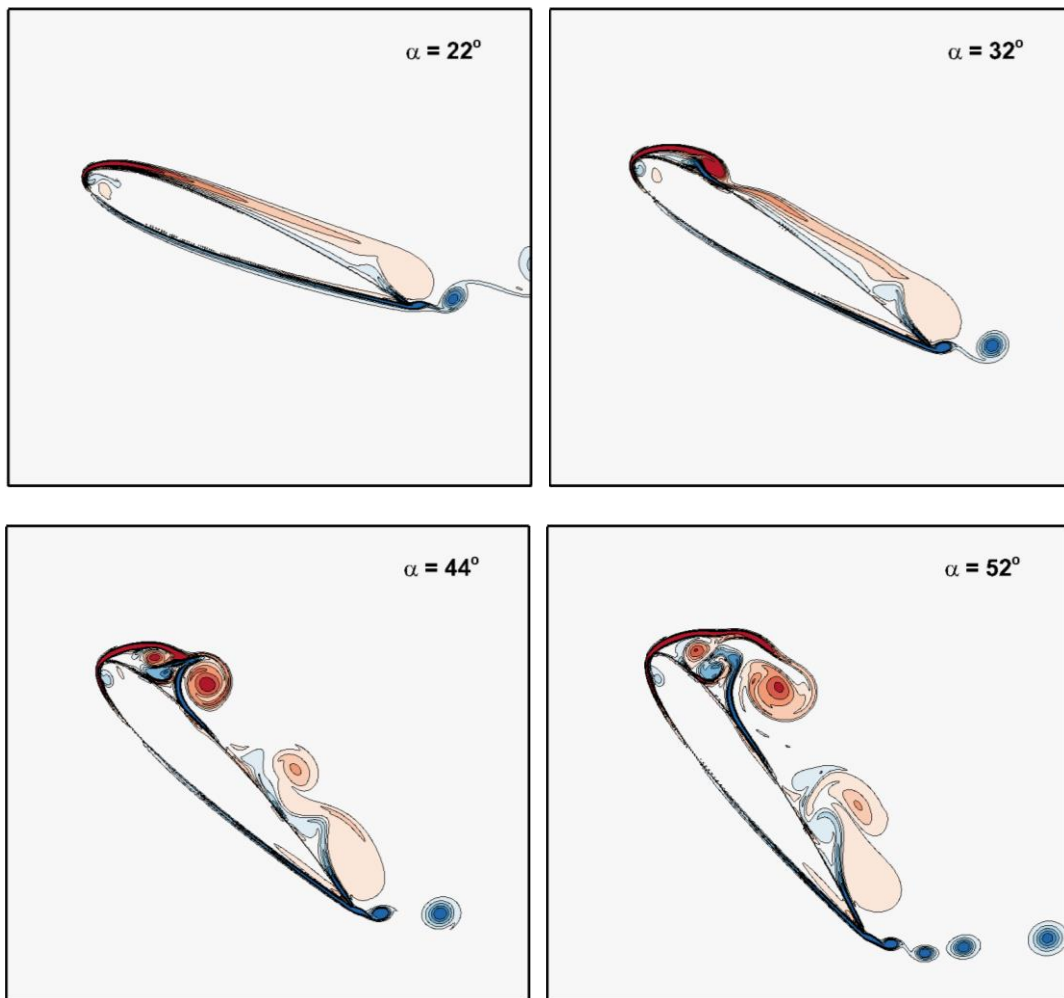


Figure 5.12 Evolution of vorticity field for flow around a rapid pitching NACA0015 airfoil

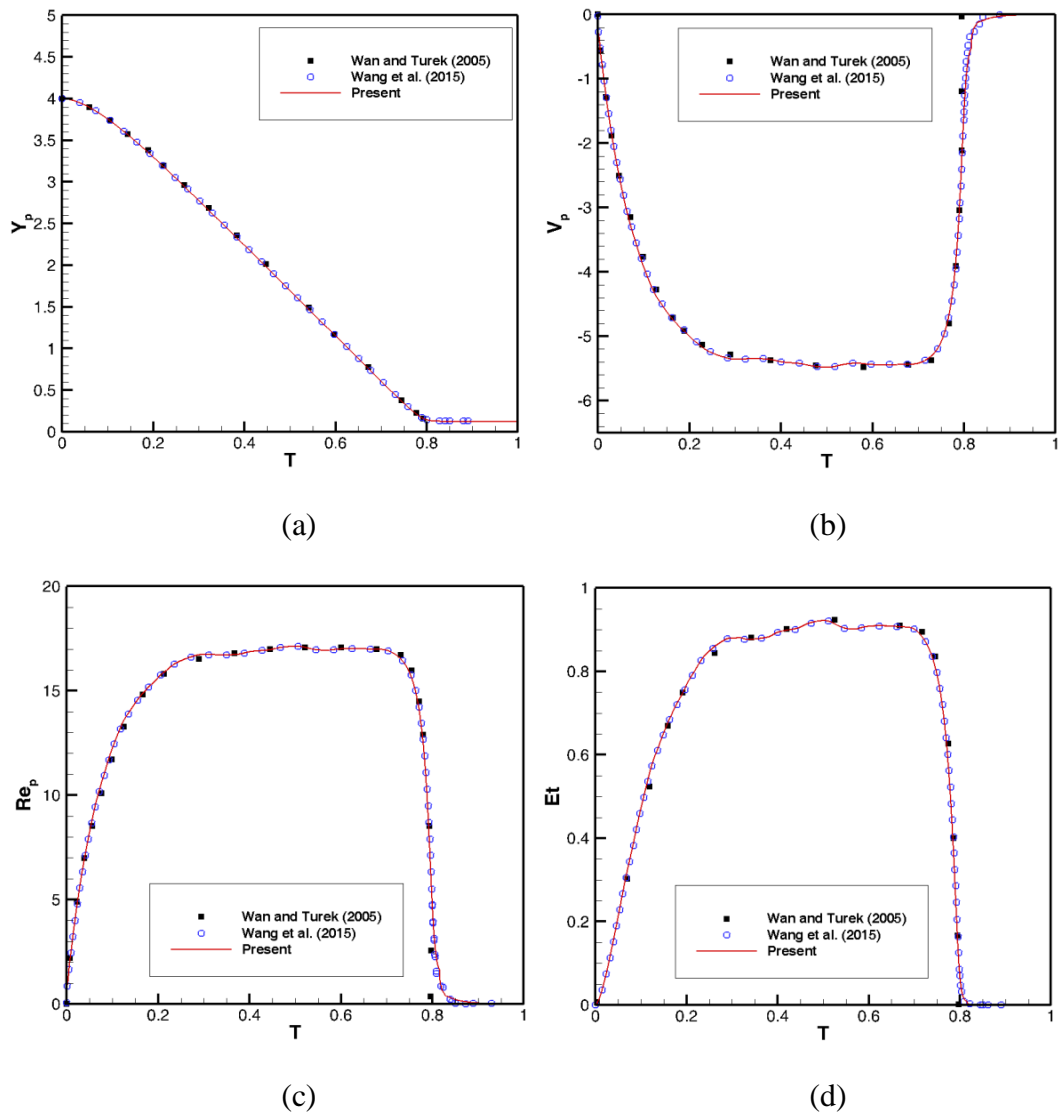


Figure 5.13 Comparison of time evolution of four representative quantities for freely falling particle in a rectangular domain. (a) Longitudinal coordinate. (b) Longitudinal velocity. (c) Particle Reynolds number. (d) Translational kinetic energy.

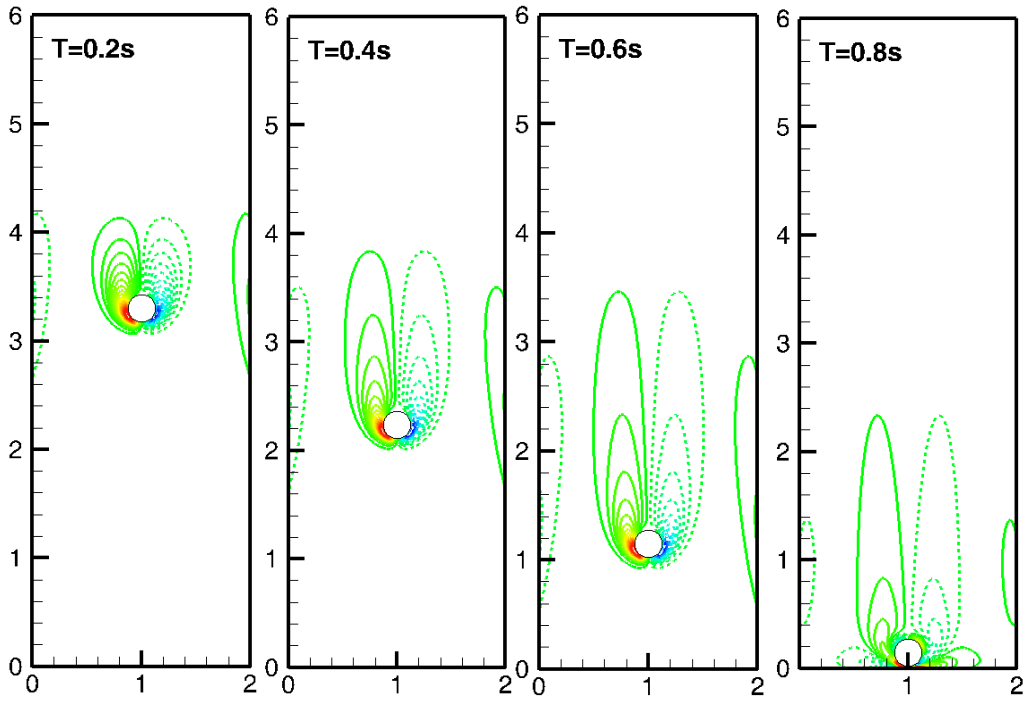


Figure 5.14 Instantaneous vorticity contours for the freely falling particle in a rectangular domain at different times

Chapter 6

A diffuse interface immersed boundary method for simulation of compressible viscous flows around stationary and moving boundaries

In the previous chapter, the boundary condition-enforced immersed boundary-gas kinetic flux solver (IB-GKFS) has been developed and successfully applied for simulating incompressible moving boundary flows. In IB-GKFS, no-slip boundary condition is implicitly imposed and the velocity field can be accurately corrected. As a result, the flow penetration, which is a drawback of conventional immersed boundary method (IBM), is entirely avoided. In IB-GKFS, only the velocity field is corrected because the immersed boundary has negligible effect on the density and pressure fields for incompressible flows. However, this method cannot be directly applied to simulate compressible flows. The reason is that for compressible flows, the immersed boundary exerts huge influence not only on velocity field but also on density, pressure and temperature fields. However, the mechanisms to correct the density, pressure and temperature fields are deficient in previous work. To overcome this difficulty, a diffuse interface IBM for compressible viscous flows around stationary and moving boundaries is developed in this chapter. Two specific schemes are proposed for stationary and moving boundaries, respectively. For stationary boundaries, the momentum field is firstly corrected by converting the no-slip velocity condition to the momentum condition. After that, the density correction can be made from the momentum correction by applying the continuity equation. While for moving boundaries, the momentum condition on the boundaries is unknown.

Therefore, the density field has to be corrected by an iterative process rather than by using the momentum correction. To validate the proposed method, several compressible stationary and moving boundary problems are simulated. The obtained results are examined by comparing with reference data published in the literature.

6.1 Governing equations and GKFS for prediction of intermediate flow field

Similar to the work of incompressible IBM in Chapter 5, the effect of the immersed boundary can be considered as a source term added in the governing equation for the compressible flows. To introduce the present scheme, the N-S equations with source term is discretized by finite volume method firstly. Then, a fractional step technique is adopted to solve these equations in two steps: the predictor step and the corrector step. The GKFS, which has been introduced in previous chapters, will be presented briefly in the predictor step. In the corrector step, a novel diffuse interface IBM for compressible viscous flows will be presented in detail in next sections.

6.1.1 Governing equations and fractional step method

After including the source term generated by the immersed boundary, the governing equations for two-dimensional case can be written as

$$\frac{d\mathbf{W}}{dt} = -\frac{1}{\Delta V_i} \sum_{i=1}^N \mathbf{F}_i S_i + \mathbf{Q}_s \quad (6.1)$$

where $\mathbf{W} = (\rho, \rho U, \rho V, \rho E)^T$ is the vector of conservative flow variables, ΔV and N are the volume and number of interfaces of the control volume, \mathbf{F}_i and S_i are the

flux vector and length of interface i , Q_s is the source term to reflect the effect of immersed boundary.

Similar to the procedure for incompressible flow problems, introducing the source term Q_s into the governing equations is equivalent to making corrections to the flow field. The main difference compared with the incompressible IBM is that in the current scheme, all the flow fields, including density, velocity, temperature and pressure, have to be corrected. By applying the fractional step method, the solution process of governing equations (6.1) can be separated as the following two steps:

(1) Predictor step for W^*

In this step, the governing equations without source term Q_s is solved:

$$\frac{W^* - W^n}{\Delta t} + \frac{1}{\Delta V_i} \sum_{i=1}^N F_i S_i = 0 \quad (6.2)$$

where W^* is the intermediate conservative flow variables at the time step $n+1$. As shown in Eq. (6.2), the flow field is firstly predicted by GKFS without consideration of the immersed boundary. After the predictor step, the flow field is rectified by the corrector step.

(2) Corrector step by the novel IBM for W

$$\frac{W^{n+1} - W^*}{\Delta t} - Q_s = 0 \quad (6.3)$$

In this step, the corrections of the flow field are conducted in such a way that the physical boundary conditions are accurately satisfied. In the following, details of these two procedures will be introduced.

6.1.2 GKFS for prediction of intermediate flow variables

To evaluate the intermediate flow field \mathbf{W}^* , the GKFS is applied. By applying a cell-centered finite volume method to solve Eq. (6.2) over a control cell Ω_i , the main task is to evaluate the numerical flux \mathbf{F} at all cell interfaces. Unlike the traditional N-S solver, in GKFS, a gas distribution function is used to describe the dynamics of the particles, which are regarded as the carriers of mass, momentum and energy. The numerical flux at the cell interface \mathbf{F} can be calculated by the integration of the distribution function f . Let the subscript $i+1/2, j$ denote the cell interface, the relation between the interface flux vector $\mathbf{F}_{i+1/2, j}$ and the distribution function f is

$$\mathbf{F}_{i+1/2, j} = \int u_n f(\mathbf{x}_{i+1/2, j}, u, v, \xi, t) \varphi_\alpha d\Xi \quad (6.4)$$

where φ_α is the vector of moments

$$\varphi_\alpha = \left(1, u, v, \frac{1}{2}(u^2 + v^2 + \xi^2) \right) \quad (6.5)$$

$d\Xi = dudvd\xi$ is the volume element in the phase space, (u, v) is the particle velocity and u_n is its normal component to the interface, ξ is the internal energy with the degree of freedom K , which is equal to $(4-2\gamma)/(\gamma-1)$ in 2D flows.

From Eq. (6.4), to obtain the numerical flux $\mathbf{F}_{i+1/2, j}$, the main issue is to calculate the gas distribution function f at the cell interface. Here, the scheme III of GKFS, which has been presented in Chapter 3, is utilized. The time-dependent gas distribution function at the cell interface can be written as

$$\begin{aligned}
& f(\mathbf{x}_{i+1/2,j}, u, v, \xi, t + \delta t) \\
& = g_0 - \tau g_0 (A_1 + A_2 u_n + A_3 u_\tau + A_4 \varepsilon) \\
& \quad - \tau \left[\left(\frac{\partial u_n g_l}{\partial n} + \frac{\partial u_\tau g_l}{\partial \tau} \right) H(u_n) + \left(\frac{\partial u_n g_r}{\partial n} + \frac{\partial u_\tau g_r}{\partial \tau} \right) (1 - H(u_n)) \right]
\end{aligned} \tag{6.6}$$

where $H(u_n)$ is the Heaviside function defined as

$$H(u_n) = \begin{cases} 0, & u_n < 0 \\ 1, & u_n \geq 0 \end{cases} \tag{6.7}$$

In Eq. (6.6), g_0 is the Maxwellian distribution function at the local interface and g_l , g_r are the Maxwellian distribution function at left and right sides of the interface, respectively. These equilibrium distribution functions can be determined from the conservative flow variables and their slopes. A_1, A_2, A_3, A_4 are the coefficients, which can be calculated explicitly. The details of the calculation of these coefficients can be referred to Chapter 3. Since the gas distribution function f at the interface has been obtained, the numerical flux across the interface can be computed by Eq. (6.4). After the flux at the interface is determined, at the projection stage, the intermediate macroscopic variables in all cells can be updated through Eq. (6.2).

6.2 Diffuse interface IBM for compressible flow around stationary boundaries

After solving Eq. (6.2) to get the predicted flow variables \mathbf{W}^* without considering the immersed boundary, we have to perform the corrector step to impose the boundary conditions. In the corrector step, the ordinary differential equations have been shown in Eq. (6.3)

$$\frac{W^{n+1} - W^*}{\Delta t} - Q_s = 0$$

As has been discussed in the work of Shu et al. (2007), in the incompressible flows, the effect of the source term in the governing equations is equivalent to making velocity corrections in the flow field. This idea can also be extended to compressible flows. That is, introducing the source term Q_s is equivalent to correcting the conservative flow variables W . This can be seen clearly in Eq. (6.3). The requirement for this correction is that the modified conservative flow variables have to satisfy the physical conditions on the immersed boundary. As a result, in this work, the conservative flow variables are corrected not by Eq. (6.3) but by physical conditions directly. For two-dimensional compressible viscous flows, the physical boundary conditions at the immersed boundary are usually the given velocity and temperature or heat flux. In this work, both the stationary and moving boundary problems are investigated. For the temperature field, only the isothermal boundary condition is considered. Following the idea of Wu and Shu (2009), the velocity and temperature field can be accurately corrected by applying the implicit boundary condition-enforced IBM. However, there is no mechanism to correct the density and pressure field in the work of Wu and Shu (2009). To overcome this difficulty, a novel method to correct the density field is proposed by employing the continuity equation in this work. As the ways to handle the density correction in stationary and moving boundary problems are different, they will be introduced separately hereinafter. Finally, the pressure can be calculated by the equation of state by the corrected temperature and density. In this way, all the flow variables are corrected to accurately satisfy the physical boundary conditions. Details of the corrections for density, velocity,

temperature and pressure will be shown below. The overall computational sequence will also be outlined.

6.2.1 For stationary boundary problems

The continuity equation for 2D compressible flows can be written as

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho U)}{\partial x} + \frac{\partial(\rho V)}{\partial y} = 0 \quad (6.8)$$

From Eq. (6.8), it can be seen that the time derivative of density depends on the spatial gradients of momentum components ρU and ρV . This connection provides the important insight that the density field might be corrected by the momentum correction. For the stationary boundary problems, the no-slip boundary condition can be easily converted to the momentum condition. The details of correction processes are shown below.

Correction of momentum field

After solving the governing equation (6.2), the intermediate momentum on Eulerian point is given as $(\rho U)^*$. We will perform the momentum correction to obtain $(\rho U)^{n+1}$ by applying the diffuse interface IBM. Suppose that corrected momentum on the Eulerian point is $\Delta(\rho U)$, therefore, the final momentum on the Eulerian point can be given as

$$(\rho U)^{n+1} = (\rho U)^* + \Delta(\rho U) \quad (6.9)$$

At the immersed boundary points, the physical condition for momentums is $(\rho U)_B^{n+1}$, where B represents the immersed boundary. To guarantee physical boundary condition, the momentums $(\rho U)_B^{n+1}$ on the Lagrangian points X_B^l must be identical

to those fluid momentums at the same positions, which is equivalent to imposing the boundary condition. Mathematically, the fluid momentums can be interpolated from the corrected momentums $(\rho\mathbf{U})^{n+1}$ at the Eulerian points by using a discrete delta function

$$(\rho\mathbf{U})_B^{n+1}(\mathbf{X}_B^l) = \sum_{ij} (\rho\mathbf{U})^{n+1}(\mathbf{x}_{ij}) D_{ij}(\mathbf{x}_{ij} - \mathbf{X}_B^l) h^2, \quad l=1,2,\dots,m, \quad j=1,2,\dots,n, \quad (6.10)$$

where h is the grid size of Eulerian mesh, m and n are the numbers of Lagrangian and Eulerian points. D_{ij} is the discrete delta function given by

$$D_{ij}(\mathbf{x}_{ij} - \mathbf{X}_B^l) = \delta\left(\frac{x_{ij} - X_B^l}{h}\right) \delta\left(\frac{y_{ij} - Y_B^l}{h}\right) \quad (6.11)$$

$$\delta(r) = \begin{cases} \frac{1}{8}(3 - 2|r| + \sqrt{1 + 4|r| - 4r^2}) & |r| \leq 1 \\ \frac{1}{8}(5 - 2|r| + \sqrt{-7 + 12|r| - 4r^2}) & 1 < |r| \leq 2 \\ 0 & |r| > 2 \end{cases} \quad (6.12)$$

Substituting Eq. (6.9) into Eq. (6.10) and using the relation

$$\Delta(\rho\mathbf{U})(\mathbf{x}_{ij}) = \sum_l \delta(\rho\mathbf{U})_B^l(\mathbf{X}_B^l) D_{ij}(\mathbf{x}_{ij} - \mathbf{X}_B^l) \Delta s^l, \quad l=1,2,\dots,m \quad (6.13)$$

we have

$$\begin{aligned} (\rho\mathbf{U})_B^{n+1}(\mathbf{X}_B^l) &= \sum_{i,j} (\rho\mathbf{U})^*(\mathbf{x}_{ij}) D_{ij}(\mathbf{x}_{ij} - \mathbf{X}_B^l) h^2 \\ &+ \sum_{ij} \left(\sum_l \delta(\rho\mathbf{U})_B^l(\mathbf{X}_B^l) D_{ij}(\mathbf{x}_{ij} - \mathbf{X}_B^l) \Delta s^l \right) D_{ij}(\mathbf{x}_{ij} - \mathbf{X}_B^l) h^2 \end{aligned} \quad (6.14)$$

The above equation can be further written as a matrix form

$$\mathbf{AX} = \mathbf{B} \quad (6.15)$$

where

$$\mathbf{X} = \left\{ \delta(\rho\mathbf{U})_B^1 \Delta s^1, \delta(\rho\mathbf{U})_B^2 \Delta s^2, \dots, \delta(\rho\mathbf{U})_B^m \Delta s^m \right\}^T \quad (6.16)$$

$$\mathbf{A} = h^2 \begin{pmatrix} D_{11} & D_{12} & \cdots & D_{1n} \\ D_{21} & D_{22} & \cdots & D_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ D_{m1} & D_{m2} & \cdots & D_{mn} \end{pmatrix} \begin{pmatrix} D_{11} & D_{12} & \cdots & D_{1m} \\ D_{21} & D_{22} & \cdots & D_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ D_{n1} & D_{n2} & \cdots & D_{nm} \end{pmatrix} \quad (6.17)$$

$$\mathbf{B} = \begin{pmatrix} (\rho\mathbf{U})_B^1 \\ (\rho\mathbf{U})_B^2 \\ \vdots \\ (\rho\mathbf{U})_B^m \end{pmatrix}^{n+1} - \begin{pmatrix} D_{11} & D_{12} & \cdots & D_{1n} \\ D_{21} & D_{22} & \cdots & D_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ D_{m1} & D_{m2} & \cdots & D_{mn} \end{pmatrix} \begin{pmatrix} (\rho\mathbf{U})_1^* \\ (\rho\mathbf{U})_2^* \\ \vdots \\ (\rho\mathbf{U})_n^* \end{pmatrix} \quad (6.18)$$

It should be noted that in Eq. (6.18), the momentum on the Lagrangian points $(\rho\mathbf{U})_B^{n+1}$ are unknown in most cases. This is because the no-slip boundary condition cannot be directly converted to momentum boundary condition as the density on the Lagrangian points is unknown. Therefore, linear system of Eq. (6.15) cannot be solved directly in moving boundary problems. On the other hand, for stationary boundary problems, the velocity on the Lagrangian points $\mathbf{U}_B = 0$. As a result, the momentum value $(\rho\mathbf{U})_B = 0$ in this special case. By solving linear system of Eq. (6.15), the unknown $\delta(\rho\mathbf{U})_B^l \Delta s^l$ at all Lagrangian points can be obtained simultaneously. After that, the momentum correction on Lagrangian points are distributed to surrounding Eulerian points by using Eq. (6.13). Finally, the corrected momentum on Eulerian points can be obtained by Eq. (6.9).

Correction of density field

As has been introduced before, the density correction is conducted by using the continuity equation with the momentum correction values. Firstly, the continuity equation for compressible flow can be written as (Eq. (6.8))

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho U)}{\partial x} + \frac{\partial(\rho V)}{\partial y} = 0$$

Setting

$$\rho = \rho^* + \Delta\rho, \quad \rho U = (\rho U)^* + \Delta(\rho U), \quad \rho V = (\rho V)^* + \Delta(\rho V) \quad (6.19)$$

As the intermediate values ρ^* , $(\rho U)^*$ and $(\rho V)^*$ also satisfy the continuity equation, by substituting Eq. (6.19) into Eq. (6.8), we can get the following formulation

$$\frac{\partial \Delta\rho}{\partial t} + \frac{\partial \Delta(\rho U)}{\partial x} + \frac{\partial \Delta(\rho V)}{\partial y} = 0 \quad (6.20)$$

In this work, the simple Cartesian mesh is adopted for the flow field. As a result, the spatial derivatives of the corrected momentum components $\Delta(\rho U)$ and $\Delta(\rho V)$, which have been obtained in the previous section, can be simply approximated by a central difference method. Therefore, the discretization of Eq. (6.20) at a mesh point (i, j) can be written as

$$(\Delta\rho)_{i,j} = -\Delta t \left[\frac{\Delta(\rho U)_{i+1,j} - \Delta(\rho U)_{i-1,j}}{2\Delta h} + \frac{\Delta(\rho V)_{i,j+1} - \Delta(\rho V)_{i,j-1}}{2\Delta h} \right] \quad (6.21)$$

From the above equation, it can be clearly seen that the momentum correction can be directly used to modify the density field. This is the main reason why the no-slip boundary condition is converted to momentum condition in the previous section. By using Eq. (6.21), the density correction can be easily obtained. And then, the corrected density can be obtained by Eq. (6.19).

Correction of temperature and calculation of pressure

In this work, only the isothermal boundary condition is considered. For simplicity, the temperature on the boundary point is denoted as T_B . After the predictor step, the

intermediate value of temperature on Eulerian point can be given as T^* . In this corrector step, suppose that the corrected temperature is ΔT . Therefore, the corrected temperature can be written as

$$T = T^* + \Delta T \quad (6.22)$$

Similar to the requirement in the momentum correction, the temperature on the Lagrangian point T_B has to be identical to the fluid temperature at the same positions.

This requirement can be expressed as

$$T_B(\mathbf{X}_B^l) = \sum_{ij} T(\mathbf{x}_{ij}) D_{ij}(\mathbf{x}_{ij} - \mathbf{X}_B^l) h^2 \quad (6.23)$$

Substituting Eq. (6.22) into Eq. (6.23) and using the relation that the temperature correction ΔT is distributed from the temperature correction δT_B at the Lagrangian points

$$\Delta T(\mathbf{x}_{ij}) = \sum_l \delta T_B(\mathbf{X}_B^l) D_{ij}(\mathbf{x}_{ij} - \mathbf{X}_B^l) \Delta s^l \quad (6.24)$$

we have

$$\begin{aligned} T_B^{n+1}(\mathbf{X}_B^l) &= \sum_{i,j} T^*(\mathbf{x}_{ij}) D_{ij}(\mathbf{x}_{ij} - \mathbf{X}_B^l) h^2 \\ &+ \sum_{ij} \left(\sum_k \delta T_B^k(\mathbf{X}_B^k) D_{ij}(\mathbf{x}_{ij} - \mathbf{X}_B^k) \Delta s^k \right) D_{ij}(\mathbf{x}_{ij} - \mathbf{X}_B^l) h^2 \end{aligned} \quad (6.25)$$

Take $\delta T_B^k \Delta s^k$ as unknowns, the above equation systems can be rewritten in a matrix form

$$\mathbf{A} \mathbf{X}_T = \mathbf{B}_T \quad (6.26)$$

where the matrix

$$\mathbf{X}_T = \left\{ \delta T_B^1 \Delta s^1, \delta T_B^2 \Delta s^2, \dots, \delta T_B^m \Delta s^m \right\}^T \quad (6.27)$$

$$\mathbf{B}_T = \begin{pmatrix} T_B^1 \\ T_B^2 \\ \vdots \\ T_B^m \end{pmatrix}^{n+1} - \begin{pmatrix} D_{11} & D_{12} & \cdots & D_{1n} \\ D_{21} & D_{22} & \cdots & D_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ D_{m1} & D_{m2} & \cdots & D_{mn} \end{pmatrix} \begin{pmatrix} T_1^* \\ T_2^* \\ \vdots \\ T_n^* \end{pmatrix} \quad (6.28)$$

Note that the matrix \mathbf{A} in Eq. (6.26) is the same as that in Eq. (6.17). By solving the linear system of Eq. (6.26), the unknowns $\delta T_B^k \Delta s^k$ can be obtained. After that, the temperature correction on Eulerian points can be re-distributed by Eq. (6.24). Consequently, the corrected temperature on the Eulerian points can be obtained by using Eq. (6.22).

Once the density and temperature fields are modified, the pressure on each Eulerian point can be simply computed by using the equation of state,

$$p(\mathbf{x}_{ij}) = \rho(\mathbf{x}_{ij})RT(\mathbf{x}_{ij}) \quad (6.29)$$

6.2.2 For moving boundary problems

The above section presents the correction procedures for stationary boundary problems. This method is straightforward and efficient because the no-slip boundary condition can be easily transferred to momentum condition in stationary boundary problems. However, for moving boundary problems, this transformation is not feasible because the density on the Lagrangian points is not given. As a result, a new correction method should be proposed for moving boundary problems. In the following, the velocity field is firstly corrected by using the no-slip boundary condition. Then, an iterative method is adopted to correct the density field by using the continuity equation. The ways to correct the temperature and pressure fields are identical with those in the previous section.

Correction of velocity field

Following the idea of Wu and Shu (2009), the velocity field is corrected by using the boundary condition-enforced IBM. Suppose that the velocity correction is $\Delta\mathbf{U}$, the corrected velocity can be expressed as,

$$\mathbf{U}^{n+1} = \mathbf{U}^* + \Delta\mathbf{U} \quad (6.30)$$

To guarantee the no-slip boundary condition, the velocity on the Lagrangian point $\mathbf{U}_B^{n+1}(\mathbf{X}_B^l)$ should be identical to the fluid velocity $\mathbf{U}^{n+1}(\mathbf{x}_{ij})$ at the same position,

$$\mathbf{U}_B^{n+1}(\mathbf{X}_B^l) = \sum_{ij} \mathbf{U}^{n+1}(\mathbf{r}_{ij}) D_{ij}(\mathbf{r}_{ij} - \mathbf{X}_B^l) h^2 \quad (l=1,2,\dots,m ; ij=1,2,\dots,n) \quad (6.31)$$

By substituting Eq. (6.30) into Eq. (6.31) and using the relation that the Eulerian velocity correction $\Delta\mathbf{U}$ can be interpolated via the Dirac delta function from the velocity correction vector $\delta\mathbf{U}_B^l$ at the boundary point, which is

$$\Delta\mathbf{U}(\mathbf{r}_{ij}) = \sum_l \delta\mathbf{U}_B^l \Delta s^l D(\mathbf{r}_{ij} - \mathbf{X}_B^l) \quad (l=1,2,\dots,m) \quad (6.32)$$

we can obtain

$$\mathbf{U}_B^{n+1}(\mathbf{X}_B^l) = \sum_{ij} \mathbf{U}^*(\mathbf{r}_{ij}) D_{ij}(\mathbf{r}_{ij} - \mathbf{X}_B^l) h^2 + \sum_{ij} \sum_l \delta\mathbf{U}_B^l \Delta s^l D(\mathbf{r}_{ij} - \mathbf{X}_B^l) D(\mathbf{r}_{ij} - \mathbf{X}_B^l) h^2 \quad (6.33)$$

The above Eq. (6.33) can be written as the following matrix form

$$\mathbf{A}\mathbf{X} = \mathbf{B} \quad (6.34)$$

where

$$\mathbf{X} = \{\delta\mathbf{U}_B^1 \Delta s^1, \delta\mathbf{U}_B^2 \Delta s^2, \dots, \delta\mathbf{U}_B^m \Delta s^m\}^T \quad (6.35)$$

$$\mathbf{A} = h^2 \begin{pmatrix} D_{11} & D_{12} & \cdots & D_{1n} \\ D_{21} & D_{22} & \cdots & D_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ D_{m1} & D_{m2} & \cdots & D_{mn} \end{pmatrix} \begin{pmatrix} D_{11} & D_{12} & \cdots & D_{1m} \\ D_{21} & D_{22} & \cdots & D_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ D_{n1} & D_{n2} & \cdots & D_{nm} \end{pmatrix} \quad (6.36)$$

$$\mathbf{B} = \begin{pmatrix} \mathbf{U}_B^1 \\ \mathbf{U}_B^2 \\ \vdots \\ \mathbf{U}_B^m \end{pmatrix}^{n+1} - \begin{pmatrix} D_{11} & D_{12} & \cdots & D_{1n} \\ D_{21} & D_{22} & \cdots & D_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ D_{m1} & D_{m2} & \cdots & D_{mn} \end{pmatrix} \begin{pmatrix} \mathbf{U}_1^* \\ \mathbf{U}_2^* \\ \vdots \\ \mathbf{U}_n^* \end{pmatrix} \quad (6.37)$$

By solving the equation system (6.34) with a direct method, the velocity correction on the Lagrangian points $\delta \mathbf{U}_B^l \Delta s^l$ can be calculated. After that, the velocity correction $\delta \mathbf{U}$ and the corrected velocity \mathbf{U}^{n+1} on all Eulerian points can be obtained by Eq. (6.32) and Eq. (6.30), respectively.

Correction of density field

In moving boundary problems, the momentum correction cannot be calculated. As a result, the correction of density field cannot be made by Eq. (6.21). For fractional step method, the density and velocity on Eulerian points can be written as

$$\rho = \rho^* + \Delta \rho, \quad U = U^* + \Delta U, \quad V = V^* + \Delta V \quad (6.38)$$

Substituting Eq. (6.38) into continuity equation,

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho U)}{\partial x} + \frac{\partial(\rho V)}{\partial y} = 0$$

we have

$$\begin{aligned} & \frac{\partial(\rho^* + \Delta \rho)}{\partial t} + \frac{\partial(\rho^* + \Delta \rho)(U^* + \Delta U)}{\partial x} + \frac{\partial(\rho^* + \Delta \rho)(V^* + \Delta V)}{\partial y} = 0 \\ \Rightarrow & \frac{\partial \rho^*}{\partial t} + \frac{\partial(\rho^* U^*)}{\partial x} + \frac{\partial(\rho^* V^*)}{\partial y} \\ & + \frac{\partial \Delta \rho}{\partial t} + \frac{\partial(\rho^* \Delta U + U^* \Delta \rho + \Delta U \Delta \rho)}{\partial x} + \frac{\partial(\rho^* \Delta V + V^* \Delta \rho + \Delta V \Delta \rho)}{\partial y} = 0 \end{aligned} \quad (6.39)$$

As the intermediate values ρ^* , U^* , V^* also satisfy the continuity equation, the above equation can be simplified as

$$\frac{\partial \Delta \rho}{\partial t} + \frac{\partial(\rho^* \Delta U + U^* \Delta \rho + \Delta \rho \Delta U)}{\partial x} + \frac{\partial(\rho^* \Delta V + V^* \Delta \rho + \Delta \rho \Delta V)}{\partial y} = 0 \quad (6.40)$$

In Eq. (6.40), all the values except for the density correction $\Delta \rho$ are known.

Therefore, the above equation can be written as

$$\frac{\partial \Delta \rho}{\partial t} + \frac{\partial(U \Delta \rho)}{\partial x} + \frac{\partial(V \Delta \rho)}{\partial y} + S = 0 \quad (6.41)$$

where $S = \frac{\partial(\rho^* \Delta U)}{\partial x} + \frac{\partial(\rho^* \Delta V)}{\partial y}$ can be viewed as the source term. Applying the

central difference discretization to Eq. (6.41),

$$(\Delta \rho)_{i,j} = -\Delta t \left[\frac{(U \Delta \rho)_{i+1,j} - (U \Delta \rho)_{i-1,j}}{2\Delta h} + \frac{(V \Delta \rho)_{i,j+1} - (V \Delta \rho)_{i,j-1}}{2\Delta h} + S_{i,j} \right] \quad (6.42)$$

$$\text{where } S_{i,j} = \frac{(\rho^* \Delta U)_{i+1,j} - (\rho^* \Delta U)_{i-1,j}}{2\Delta h} + \frac{(\rho^* \Delta V)_{i,j+1} - (\rho^* \Delta V)_{i,j-1}}{2\Delta h}$$

Eq. (6.42) shows that the calculation of $(\Delta \rho)_{i,j}$ needs the information of adjacent density correction, which is unknown. To solve Eq. (6.42), an iterative procedure is required. At time level n , it is known that after the predictor step and velocity correction, the intermediate density ρ^* , the velocity correction ΔU and the corresponding corrected velocity U at all Eulerian points are already obtained. Then, the following iteration procedure is conducted for NF times

$$\Delta \rho^{NF+1} = -\Delta t \left[\frac{(U \Delta \rho^{NF})_{i+1,j} - (U \Delta \rho^{NF})_{i-1,j}}{2\Delta h} + \frac{(V \Delta \rho^{NF})_{i,j+1} - (V \Delta \rho^{NF})_{i,j-1}}{2\Delta h} + S_{i,j} \right] \quad (6.43)$$

until the following convergence criterion is satisfied

$$\frac{\sum_{i,j} |\Delta \rho^{NF+1} - \Delta \rho^{NF}|}{\sum_{i,j} |\Delta \rho^{NF+1}|} \leq 10^{-7} \quad (6.44)$$

After the iteration procedure, the density correction at time level n can be derived. Finally, the accurate density can be calculated by the sum of intermediate value and correction value.

Correction of temperature and calculation of pressure

As only the Dirichlet boundary condition is considered here, the way to correct the temperature field is the same as that in previous section. By solving the linear system of Eq. (6.26), the temperature correction at all Eulerian points can be computed simultaneously and the accurate temperature field is obtained afterwards. Once all the corrections are made, we can simply compute the pressure by using the equation of state in Eq. (6.29).

6.2.3 Force calculation and solution procedure

It is worth mentioning that in the above processes, the correction of flow variables is based on the physical boundary conditions, rather than by using explicit form of source term \mathbf{Q}_s . Following the governing equation (6.3) in the corrector step, the explicit expression of \mathbf{Q}_s can be given as

$$\mathbf{Q}_s = \left[\frac{\delta p}{\delta t}, \frac{\delta(\rho U)}{\delta t}, \frac{\delta(\rho V)}{\delta t}, \frac{\delta(\rho E)}{\delta t} \right]^T \quad (6.45)$$

Following the idea of boundary condition-enforced IBM (Wu and Shu, 2009), the momentum change on fluid is due to the force exerted by solid boundary. Therefore, by using Newton's third law, the force on the rigid body generated by the surrounding fluid flow can be simply calculated as

$$\mathbf{F} = -\sum_{i,j} \frac{\Delta(\rho \mathbf{U})_{i,j}}{\Delta t} \quad (6.46)$$

In the present scheme, the ways to deal with stationary and moving boundaries are different. The computational sequence of the current diffuse interface IBM for stationary boundary problems can be summarized as follows:

- (1) Use Eq. (6.6) and Eq. (6.4) to calculate the flux vector \mathbf{F} on all cell interfaces.
- (2) Predict the intermediate flow field \mathbf{W}^* by solving Eq. (6.2).
- (3) Solve the linear system of Eq. (6.15) to obtain the momentum force density $\delta(\rho\mathbf{U})_B^k \Delta s^k$ at all Lagrangian points.
- (4) Apply Eq. (6.9) and Eq. (6.13) to perform momentum corrections and get $(\rho\mathbf{U})^{n+1}$.
- (5) Compute the density correction $\Delta\rho$ by using Eq. (6.21) and get ρ^{n+1} by Eq. (6.19).
- (6) Solve the linear system of Eq. (6.26) to obtain the temperature correction on Lagrangian points $\delta T_B^k \Delta s^k$.
- (7) Calculate the temperature correction ΔT by Eq. (6.24) and update the temperature field by using Eq. (6.22).
- (8) Apply the equation of state (6.29) to calculate the pressure.
- (9) Repeat steps (1) - (8) until final solutions are obtained.

For moving boundary problems, the solution procedure is slightly different from the above one because the boundary condition for momentum on the immersed boundary points is unpredictable. The basic solution procedure is:

- (1) Use Eq. (6.6) and Eq. (6.4) to calculate the flux vector \mathbf{F} on all cell interfaces.

- (2) Predict the intermediate flow field \mathbf{W}^* by solving Eq. (6.2).
- (3) Solve the linear system of Eq. (6.34) to obtain the force density $\delta \mathbf{U}_B^k \Delta s^k$ at all Lagrangian points.
- (4) Apply Eq. (6.32) and Eq. (6.30) to perform velocity corrections and get \mathbf{U}^{n+1} .
- (5) Calculate the density correction $\Delta \rho$ for NF times by Eq. (6.43) until the convergence of Eq. (6.44) is satisfied.
- (6) Get the corrected density ρ^{n+1} by Eq. (6.38).
- (7) Solve the linear system of Eq. (6.26) to obtain the temperature correction on Lagrangian points $\delta T_B^k \Delta s^k$.
- (8) Calculate the temperature correction ΔT by Eq. (6.24) and update the temperature field by using Eq. (6.22).
- (9) Apply the equation of state (6.29) to calculate the pressure.
- (10) Repeat steps (1) - (9) until final solutions are obtained.

6.3 Numerical validation and discussion

In this section, the present diffuse interface IBM is validated through simulations of compressible flows over both stationary and moving boundary problems. First, compressible flow around a circular cylinder is simulated to validate the two methods. Here, the stationary test case can be viewed as a special case to validate the method for moving boundaries. The obtained numerical results are compared with the reference data in literature. Subsequently, the method for stationary boundaries is further verified by simulating more complex supersonic flows over a NACA0012 airfoil with two different free stream conditions. After that, numerical experiments of

three moving boundary problems are carried out to validate the flexibility and capability of the present method for moving boundaries.

6.3.1 Flow around a circular cylinder

The flow around a circular cylinder, which is a good benchmark problem, is simulated to validate the current compressible IBM. In the present simulation, the computational domain is selected as $40D \times 30D$, where D is the diameter of the cylinder. The circular cylinder discretized by 400 Lagrangian points is placed at $(15D, 15D)$. A non-uniform mesh of 504×500 points is adopted in the computational domain and a uniform mesh with the mesh spacing of $0.005D$ is applied around the circular cylinder. Dirichlet conditions are imposed on all flow variables at the inflow boundary and natural boundary condition is imposed at the outflow boundary. The Reynolds number based on the diameter of the cylinder is 300 for all cases. Two difference Mach numbers of 1.2 and 2.0 are considered in the present simulation.

Figure 6.1 and Figure 6.2 show the density, pressure and velocity contours in the near field of the cylinder at Mach number $M_\infty = 1.2$ and 2.0. In these figures, the solid lines are the results of the present method for stationary boundary and the contours and dash lines represent the results obtained by DVM (Yang et al., 2016), which uses a body-fitted grid. The results of the method for moving boundaries are not included because they are almost identical to those from the method for stationary boundaries. From the results, the flow fields are symmetric with a bow shock before the circular cylinder. It can be clearly seen that excellent agreements are found between the two results in both two figures. Figure 6.3 displays the pressure coefficient distributions on the cylinder surface computed by two methods. Also included in this figure are the

numerical data of Takahashi et al. (2014) for comparison. As can be seen, good agreement has been achieved. The comparison of drag coefficients of the present results with those of Takahashi et al. (2014) is provided in Table 6.1. From the table, the present results are close to those in reference and the maximum relative error is below 3.8%. In both two cases, the lift coefficients are equal to zero. It should also be noted that for the method of moving boundary, it only takes 4 iterative steps to meet convergence criterion in Eq. (6.44) at each time step. The above observations show that both two methods give almost the same results in the present test case and the accuracy and capability of these methods are fully validated.

6.3.2 Compressible flow over a NACA0012 airfoil

In this simulation, a compressible laminar flow over a NACA0012 airfoil is tested. The schematic of the current problem is depicted in Figure 6.4. As can be seen, the distance between the leading edge and the free stream boundary is around 11 chord lengths and the distance between the trailing edge and the outflow boundary is around 16 chord lengths. The total grid size for the whole non-uniform mesh is 918×454 . There are 1000 Lagrangian points on the airfoil and the mesh spacing around the airfoil is $0.002c$, where $c = 1$ is the chord length of the airfoil. Two different cases are selected for study here, which are: (1) $M_\infty = 0.5, \alpha = 0^\circ, Re = 5000$ and (2) $M_\infty = 2.0, \alpha = 10^\circ, Re = 1000$.

Firstly, the subsonic case of $M_\infty = 0.5$ is tested. In this case, the Reynolds number is near the upper limit for steady laminar flow. In Figure 6.5, the results of solid lines show the density, pressure and velocity contours around the NACA0012 airfoil for the present simulation. Also included in this figure are the body-fitted results obtained by

GKFS, which are represented by flood and dashed lines. Excellent agreement is found in Figure 6.5. Figure 6.6 shows the streamlines around the airfoil. We can see that no penetration is found on the airfoil surface and a pair of vortices appear near the trailing edge. This observation is similar to those in literature (Bassi and Rebay, 1997; Ivan and Groth, 2014). In addition, Figure 6.7 shows the present results of pressure coefficient distribution along the airfoil surface. It compares well with those obtained by Bassi and Rebay (1997) and Jawahar and Kamath (2000).

In order to examine the capability of proposed methodology in solving supersonic flow problems with complex geometry, the laminar supersonic flow over a NACA0012 airfoil is further tested. The free stream condition is Mach number $M_\infty = 2.0$, angle of attack $\alpha = 10^\circ$ and Reynolds number based on the chord length $Re = 1000$. The contours of flow variables, including density, pressure and velocities, are plotted in Figure 6.8. Also included in this figure are the results of body-fitted grid. Once again, excellent agreement is obtained. Moreover, the pressure contour shows that the shock is computed monotonically. The pressure coefficient distribution along the airfoil surface is plotted in Figure 6.9. It compares well with that obtained by De Palma et al. (2006). To further test the capability and accuracy of the present scheme in calculating the forces, the lift and drag coefficients for both two cases are shown in Table 6.2. It can be seen that the drag coefficients are close to those in the literature (Mavriplis and Jamesone, 1990; Crumpton et al., 1993; Jawahar and Kamath, 2000; Bristeau, 2013; De Palma et al., 2006). Meanwhile, the lift coefficient in the second case is slightly different from those in literature (Bristeau, 2013; De Palma et al., 2006). A possible reason could be that at the trailing edge, there are overlapping Eulerian points for a few of upper and lower Lagrangian points. As a result, the flow

variables in these Eulerian points will be corrected twice and thus bring error to the calculation of forces.

6.3.3 Flow over a high-frequency plunging SD7003 airfoil

Flows over SD7003 airfoil with different motions have been experimentally studied by McGowan et al. (2008, 2011) and numerically studied by Visbal et al. (2009). In this case, the detail of the flow field around the airfoil is investigated with a prescribed motion at low-Reynolds number condition. The Reynolds number based on the chord length of the airfoil is 10,000. For this low Reynolds number, transition does not occur over the airfoil at moderate angles of attack prior to stall. The SD7003 airfoil is set at a static angle of attack $\alpha_0 = 4^\circ$. The plunging motion is given as

$$h(t) = h_0 \sin(2kU_\infty t/c) \quad (6.47)$$

where non-dimensional amplitude $h_0 = 0.05$; reduced frequency $k = 3.93$; U_∞ is the freestream velocity which is parallel to x - axis; $c = 1$ is the chord length of the airfoil. Here, the Mach number is set as $M_\infty = 0.2$ with specific heat ratio $\gamma = 5/3$. The computational domain is $[-20c, 30c] \times [-20c, 20c]$ with the grid size of 942×520 . The number of Lagrangian points on the airfoil is 667. A uniform mesh with mesh spacing $\Delta h = 0.002$ is selected near the airfoil. The flow field computed at $h = 0.0$ is used as initial condition.

Figure 6.10 shows the time evolution of lift and drag coefficients. It shows that the present results match well with the results of Visbal et al. (2009). Figure 6.11 shows the instantaneous flow structure at selected phases of the plunging motion. It is worth noting that the selected phases correspond to positions of maximum upward velocity

($t=0T$), maximum upward displacement ($t=1/4T$), maximum downward velocity ($t=1/2T$) and maximum downward displacement ($t=3/4T$). At the maximum downward displacement, two visible vortices at the leading edge are formed (Figure 6.11(d)). These two vortices propagate at the vicinity of the airfoil surface during the upstroke (Figure 6.11(a)) until reaching the airfoil trailing edge. During the downstroke, a single dynamic-stall vortex is also observed at the lower surface of the airfoil because of the large negative angle of attack (Figure 6.11(b) and Figure 6.11(c)). The present results show reasonable agreement with the observations in Visbal et al. (2009).

6.3.4 Compressible flow over a rotating cylinder

Flow past moving bluff bodies, especially in high speed flows, is an interesting topic in aeronautics. Due to the complexity and deficiency in generating body-fitted grids, this flow problem is a great challenge for the traditional body-fitted numerical methods. Fortunately, this difficulty can be effectively resolved since no grid transformation or regeneration is involved in IBM. In this section, we will demonstrate the capability of the present diffuse interface IBM by solving the compressible flow over a rotating cylinder.

In this problem, a rotating cylinder with the normalized angular velocity $\Omega = \omega D / (2U_\infty)$ in the clockwise direction is placed in a free stream of velocity U_∞ , where D is the diameter of the cylinder. The free-stream conditions for this test case are: Mach number $M_\infty = 2.0$, Reynolds number $Re = 1000$, angle of attack $\alpha = 0^\circ$. In the present simulation, the computational domain is selected as $40D \times 40D$. The

circular cylinder, which is discretized by 400 Lagrangian points, is placed at $(15D, 15D)$. A uniform mesh with the mesh spacing of $0.004D$ is applied around the circular cylinder. The flow field computed around the stationary cylinder is used as initial condition.

Figure 6.12 shows the streamlines and pressure contours around the rotating cylinder. Also included are the results obtained by DVM (Yang et al., 2016) using a body-fitted grids. In both simulations, the no-slip and isothermal boundary conditions are imposed on the cylinder surfaces. It can be seen from the figure that due to the cylinder rotation, the fluid passing over the upper side of the cylinder continues to move downward and then enters the wake. Figure 6.13 shows the density, pressure and velocity contours in the near field of the cylinder. In this figure, the solid lines are the results of the present method and the contours and floods and dash lines represent the results obtained from DVM (Yang et al., 2016). Similar to the stationary problem, a bow shock is found in front of the circular cylinder. However, the contours behind the cylinder are quite different from those of stationary problem. Figure 6.14 displays the pressure coefficient distributions along the cylinder surface. The numerical results of body-fitted grids obtained by DVM (Yang et al., 2016) are also included in this figure for comparison. As can be seen, good agreement has been achieved.

6.3.5 Laminar flow over a harmonic oscillating NACA0012 airfoil

In this test case, the flow around a harmonic oscillating NACA0012 airfoil at laminar flow conditions are numerically investigated. The schematic diagram of this problem is depicted in Figure 6.15. The airfoil pitching center locates at the quarter-chord axis,

i.e., $c/4$ behind the leading edge, where c is the chord length of the airfoil. The time-dependent angle between the airfoil and the horizontal axis is given by

$$\alpha(t) = 10^\circ [1 - \cos(\omega t)] \quad (6.48)$$

The reduced frequency of the oscillation, which is usually used to characterize the unsteady motion, is defined by

$$K_s = \frac{\omega c}{2U_\infty} \quad (6.49)$$

where U_∞ is the free-stream velocity. In the present simulation, the Reynolds number $Re = \rho U_\infty c / \mu$ is 5000 and the freestream Mach number is 0.4. The reduced frequency of oscillation is 0.5. The computational domain is a rectangle with the size of $40c \times 30c$ in the x - and y - directions, respectively. A grid size of 743×500 is applied to discretize the domain. The steady flow at $\alpha = 0^\circ$ is chosen as the initial condition.

Figure 6.16 shows the density contours in the second cycle of motion for the case of $Re = 5000$, $M_\infty = 0.4$, $K_s = 0.5$. The results of Guo et al. (1994) are also included for comparison in this figure. From Figure 6.16(a) and Figure 6.16(i) or any other pairs of figures with the same angle of attack, it is observed that the flow structures vary dramatically in the upstroke and downstroke processes. During the upstroke process (Figure 6.16(a)-(c)), the trailing edge vortices are first formed at a lower angle of attack. After that, a leading-edge vortex appears at a higher angle of attack, which is between 16° (Figure 6.16(d)) and 20° (Figure 6.16(e)). During the downstroke process (Figure 6.16(f)-(g)), this leading-edge vortex grows very quickly. In addition, a second counterclockwise vortex is formed at the upper airfoil surface, which might

be caused by the combined effect of adverse gradient and the leading-edge vortex. It is worth noting that in contrast with high-Reynolds-number turbulence flows, even a small adverse gradient in laminar flows can produce a recirculation region which affects the local flow field. With the movement of the airfoil downstroke, these vortices still exist and grow in size at the upper surface of the airfoil even through the local angle of attack reaches the minimum. The above observations are in line with those in the work of Guo et al. (1994). The present computation is limited to low-Reynolds-number laminar flows, but is of value since they can illustrate important dynamic stall features and trends.

6.4 Concluding remarks

In this chapter, a diffuse interface IBM was proposed for simulation of compressible stationary and moving boundary flows. In the method, the solution of the flow field and implementation of boundary condition were decoupled into two steps by applying the fractional step technique. The intermediate flow field was first predicted by implementing the developed GKFS and the correction of flow variables was conducted subsequently by using the current diffuse interface IBM. Two different schemes were proposed for stationary and moving boundaries, respectively. For stationary boundary, the momentum field was firstly corrected by converting the no-slip velocity condition to the momentum condition. After that, the density correction was made from the momentum correction by using the continuity equation, while for the moving boundary, the momentum condition on the boundary is unknown. As a result, the density field has to be corrected by an iterative process rather than by using the momentum correction. It showed that only a small number of iterations were needed to get the correct density in each time step. The present diffuse interface IBM

was validated to be a much simpler and more flexible way to simulate the compressible moving boundary problems. Moreover, the proposed correction procedure makes the hydrodynamic forces calculation on the immersed boundary quite convenient and accurate.

The proposed two methods of diffuse interface IBM were firstly validated by simulating flow around a stationary circular cylinder. The obtained results of two methods were almost identical with each other and achieved good agreements with the data available in the literature. After that, several stationary and moving boundary flow problems were simulated, such as compressible flow over a NACA0012 airfoil, flow over a high-frequency plunging SD7003 airfoil, compressible flow over a rotating cylinder and laminar flow over a harmonic oscillating NACA0012 airfoil. Once again, good agreements between the present results and the published/computed data were achieved, which validated the capability of the present diffuse interface IBM for simulation of complex compressible moving boundary flows.

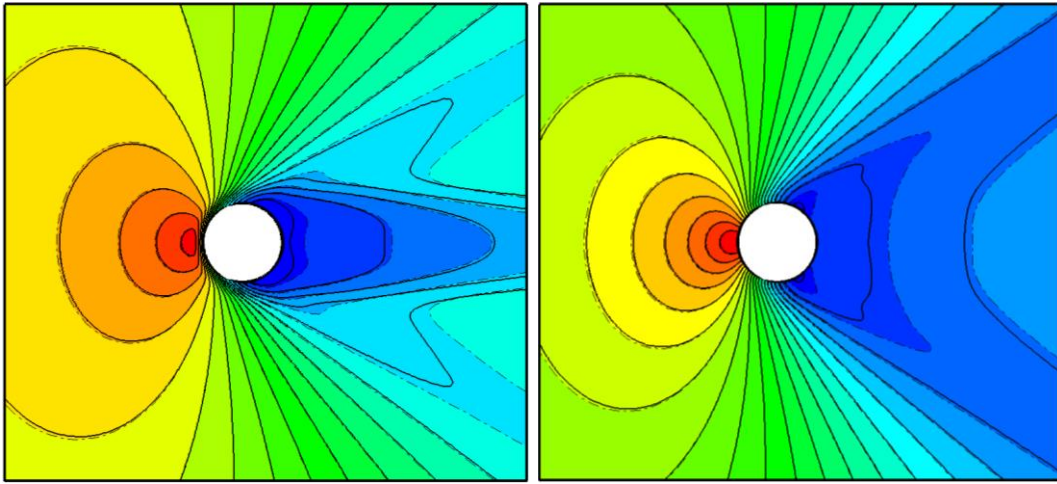
Table 6.1 Comparison of drag coefficient for supersonic flow around a circular cylinder

Cases	Reference	C_d
$M_\infty = 1.2, Re = 300$	Takahashi et al. (2014)	1.6221
	Present¹	1.6831
	Present²	1.6805
$M_\infty = 2.0, Re = 300$	Takahashi et al. (2014)	1.5480
	Present¹	1.5880
	Present²	1.5905

Present¹: Method for stationary boundary
Present²: Method for moving boundary

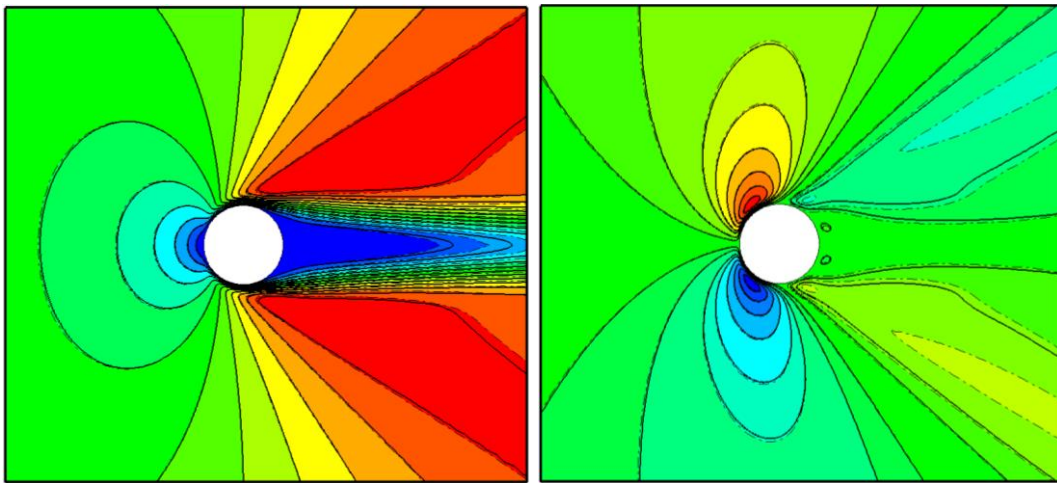
Table 6.2 Lift and drag coefficients for viscous flow over a NACA0012 airfoil

Cases	References	C_d	C_l
$M_\infty = 0.5$ $Re = 5000$ $\alpha = 0^\circ$	Mavriplis and Jameson (1990)	0.05610	--
	Crumpton et al. (1993)	0.05610	--
	Jawahar and Kamath (2000)	0.05557	0.00
	Present	0.05813	0.00
$M_\infty = 2.0$ $Re = 1000$ $\alpha = 10^\circ$	GAMM workshop (Bristeau, 2013)	0.2515-0.2535	0.3388-0.3427
	De Palma et al. (2006)	0.2515	0.3400
	Present	0.25317	0.30319



(a) Density

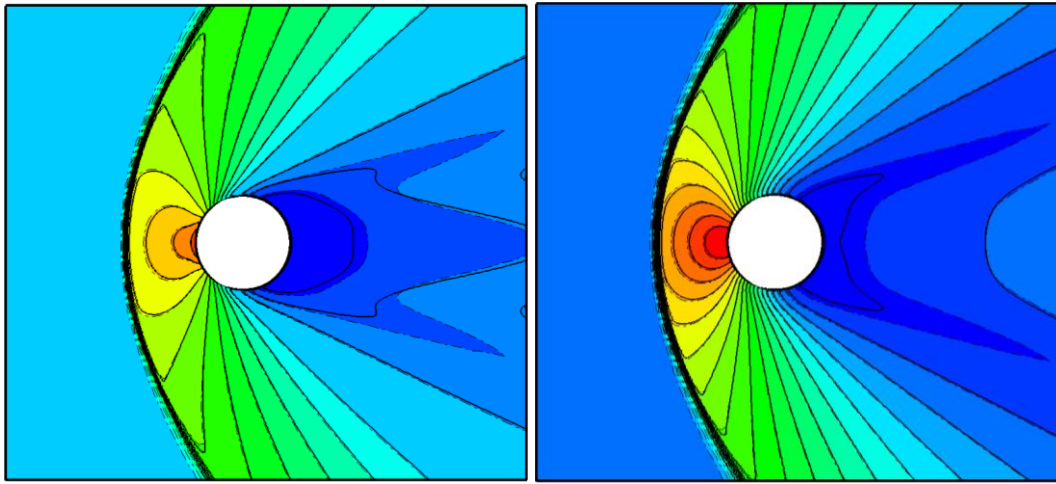
(b) Pressure



(c) U-velocity

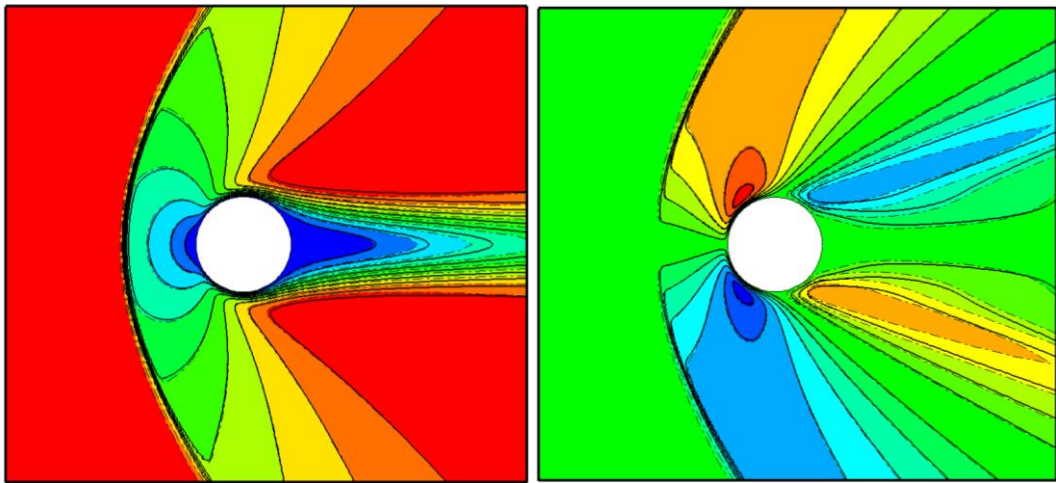
(d) V-velocity

Figure 6.1 Comparison of contours for flow over cylinder at $M_\infty = 1.2$, $Re = 300$ obtained from present scheme (Solid lines) and DVM (Flood and dashed lines)



(a) Density

(b) Pressure



(c) U-velocity

(d) V-velocity

Figure 6.2 Comparison of contours for flow over cylinder at $M_\infty = 2.0$, $Re = 300$ obtained from present scheme (Solid lines) and DVM (Flood and dashed lines)

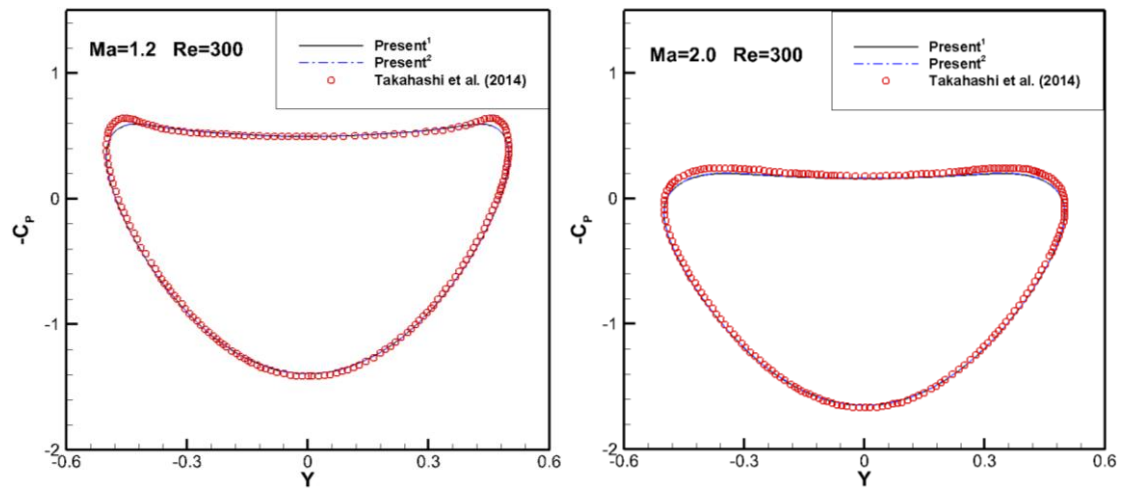


Figure 6.3 Pressure coefficient distribution on the cylinder surface

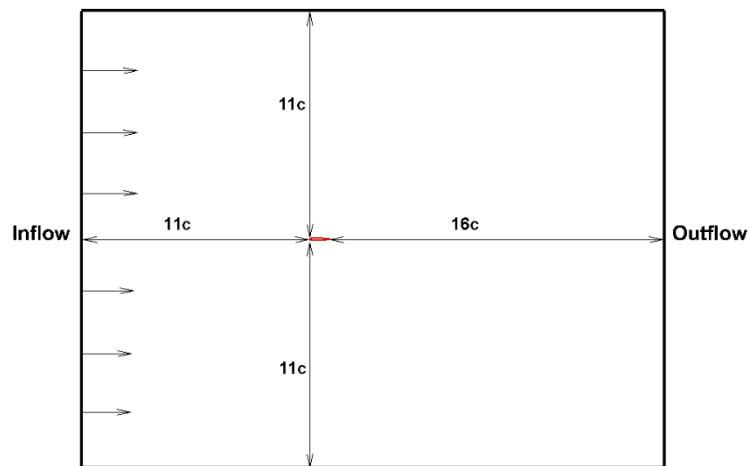
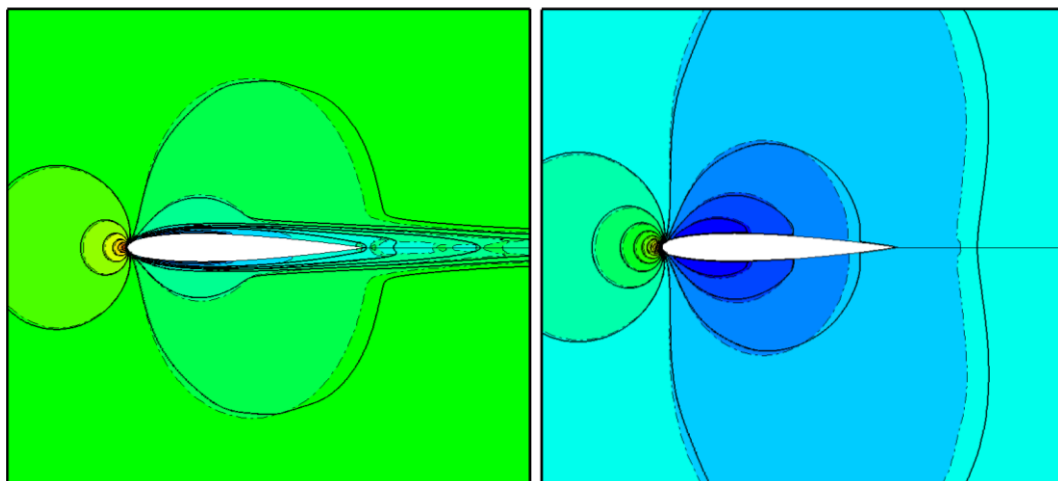


Figure 6.4 Schematic diagram for flow over a NACA0012 airfoil



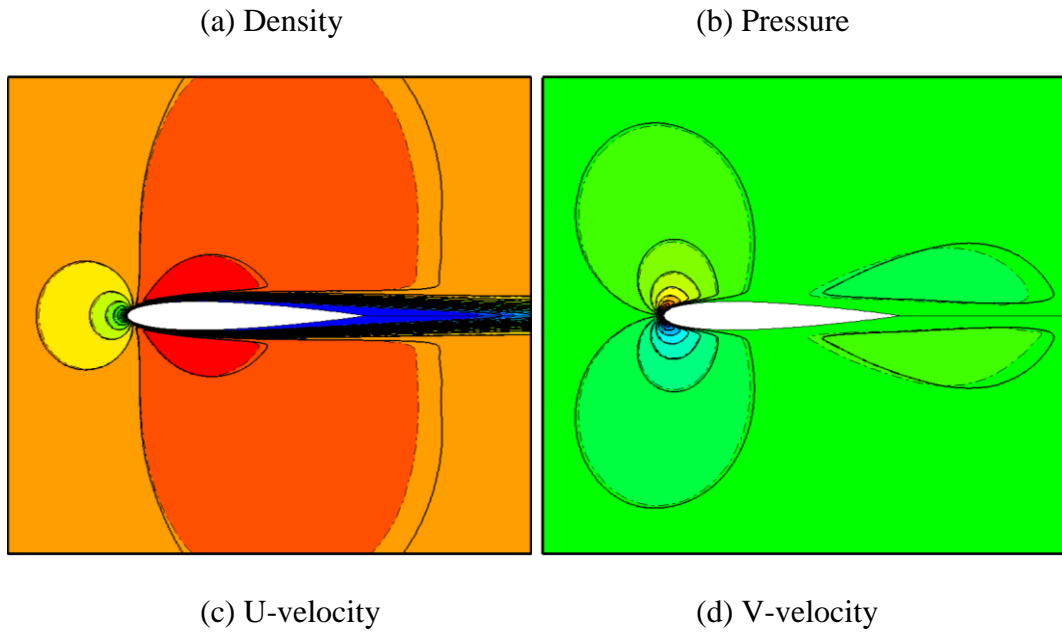


Figure 6.5 Comparison of contours for NACA0012 airfoil at $M_\infty = 0.5$, $\alpha = 0^\circ$, $Re = 5000$ obtained from present scheme (Solid lines) and body-fitted grids (Flood and dashed lines)

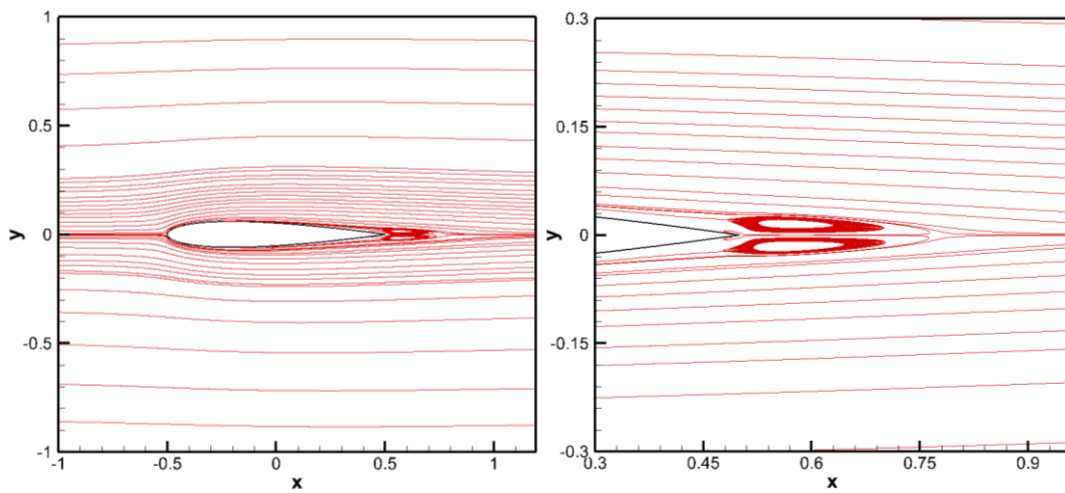


Figure 6.6 Streamlines for flow over a NACA0012 airfoil at $M_\infty = 0.5$, $\alpha = 0^\circ$, $Re = 5000$

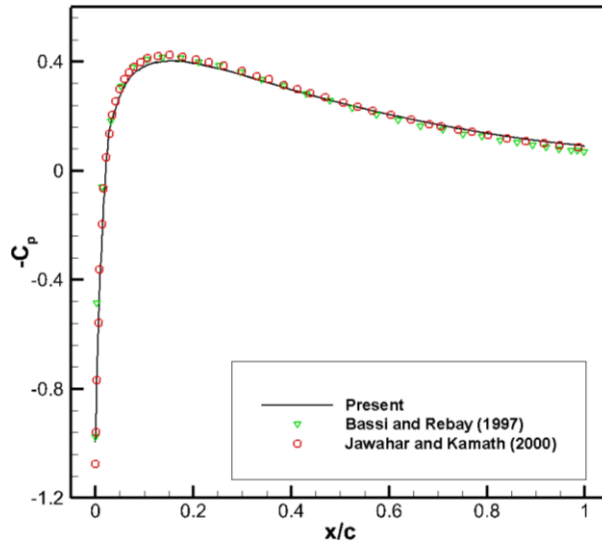
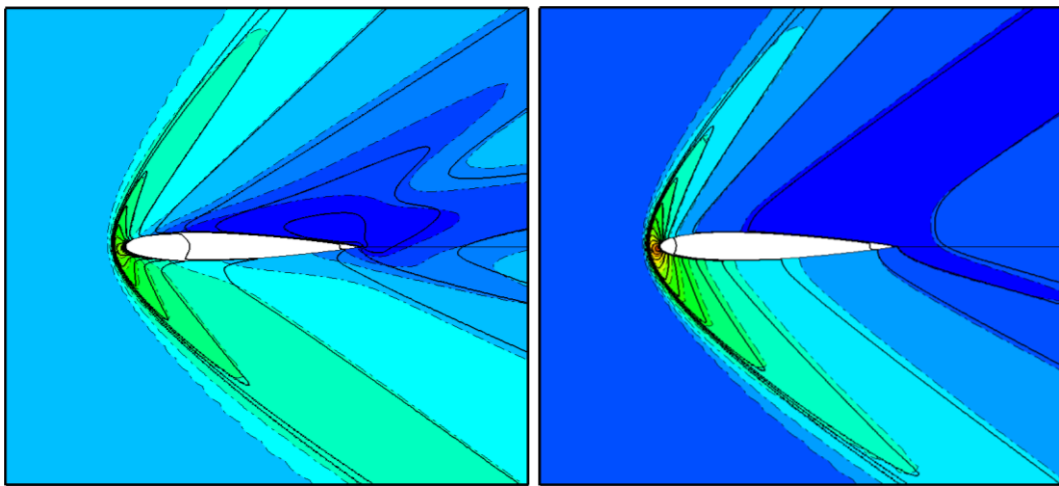


Figure 6.7 Pressure coefficient distribution for NACA0012 airfoil at $M_\infty = 0.5$, $\alpha = 0^\circ$, $Re = 5000$



(a) Density

(b) Pressure

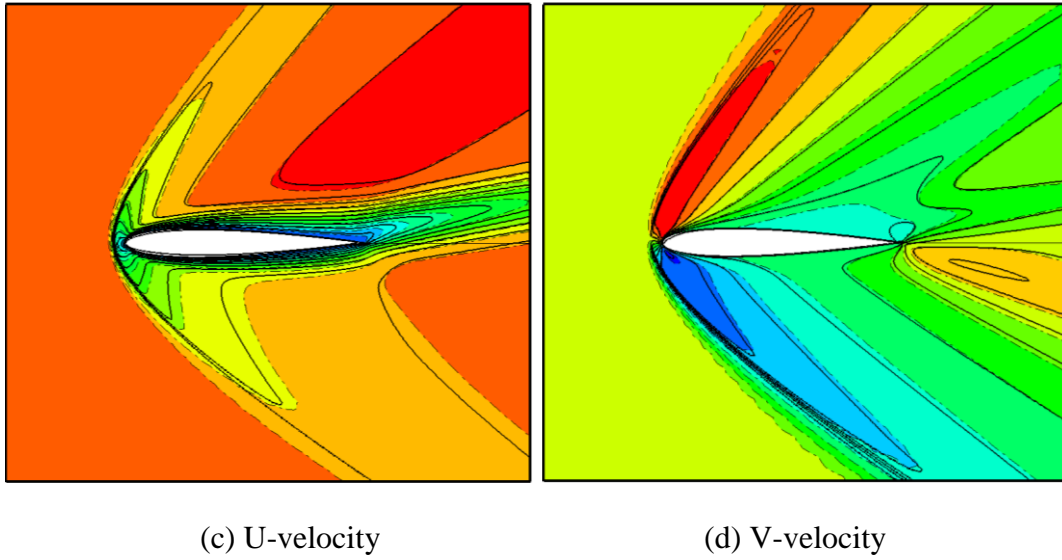


Figure 6.8 Comparison of contours for NACA0012 airfoil at $M_\infty = 2.0$, $\alpha = 10^\circ$, $Re = 1000$ obtained from present scheme (Solid lines) and body-fitted grids (Flood and dashed lines)

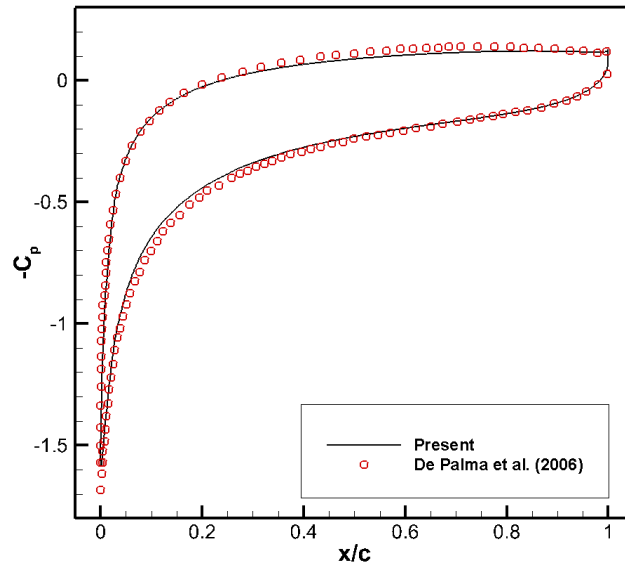


Figure 6.9 Pressure coefficient distribution for NACA0012 airfoil at $M_\infty = 2.0$, $\alpha = 10^\circ$, $Re = 1000$

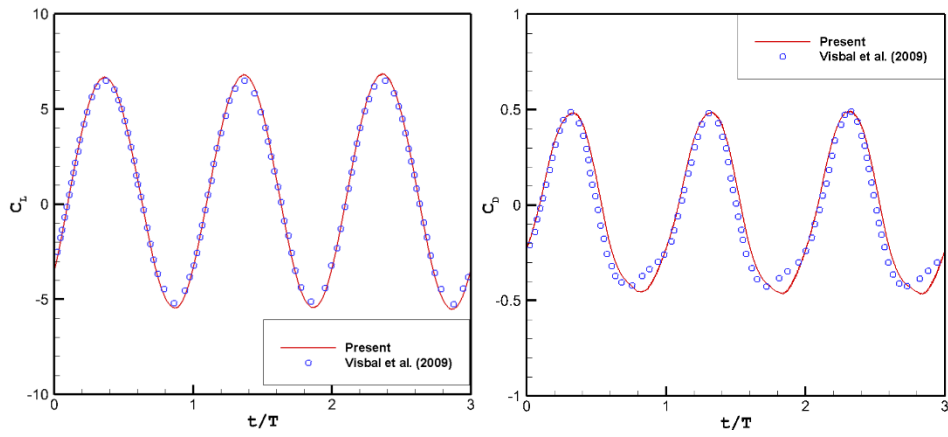
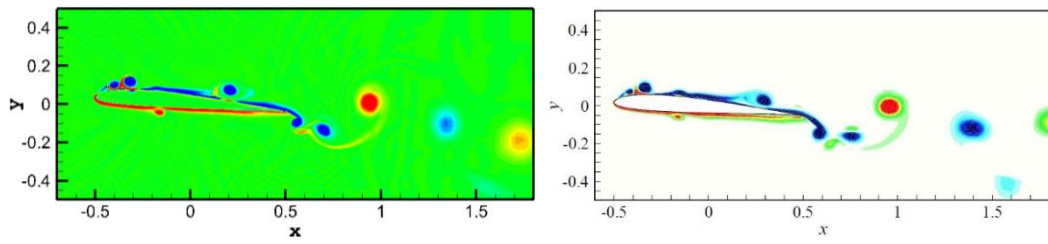
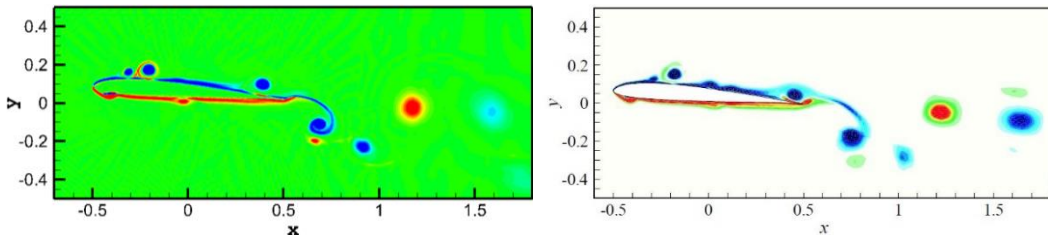


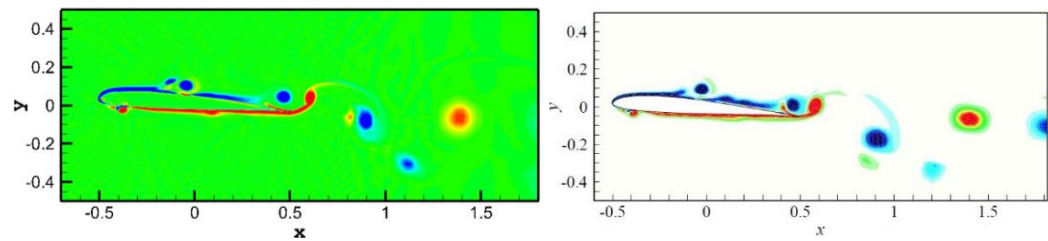
Figure 6.10 Lift and drag coefficients versus normalized time t/T



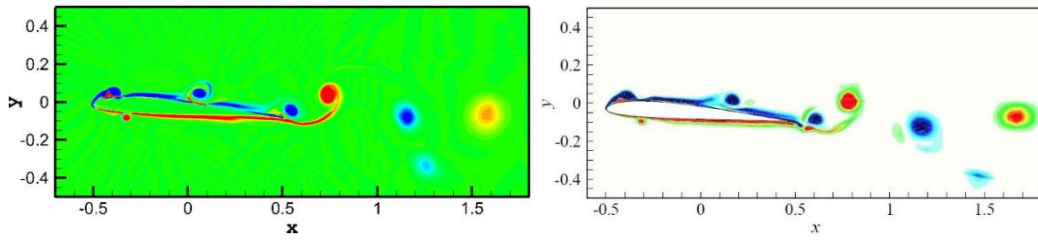
(a) $t = 0T$



(b) $t = 0.25T$



(c) $t = 0.5T$



(d) $t = 0.75 T$

Figure 6.11 Comparison of the vorticities of present results (Left) and Visbal et al. (2009) (Right)

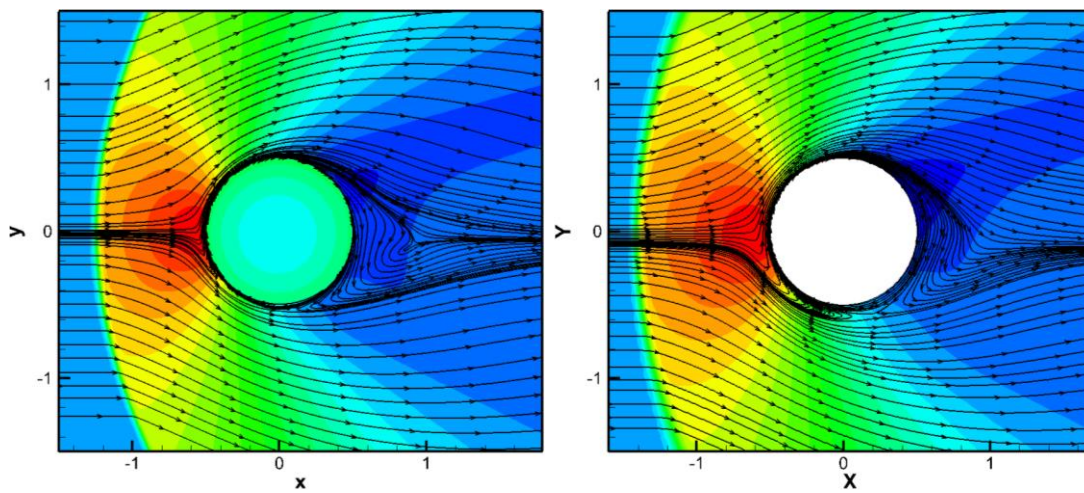
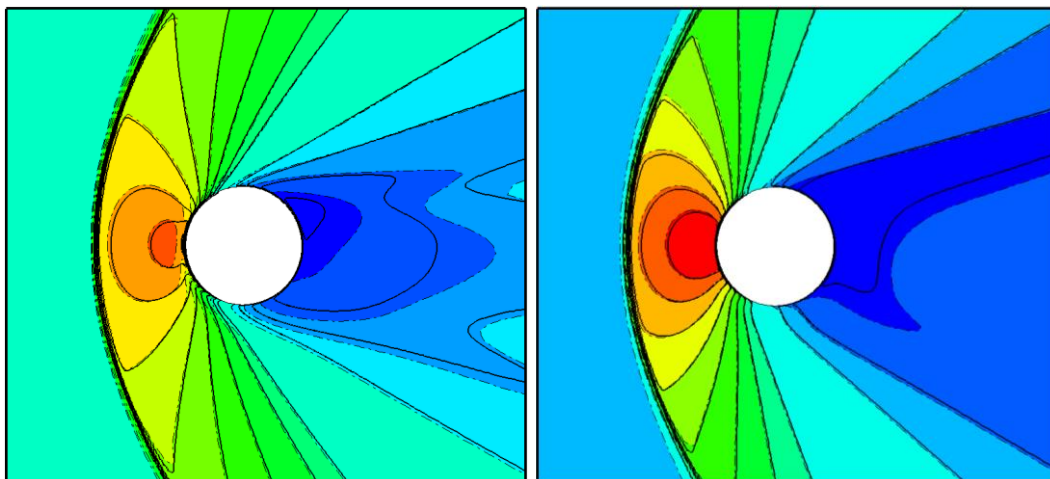
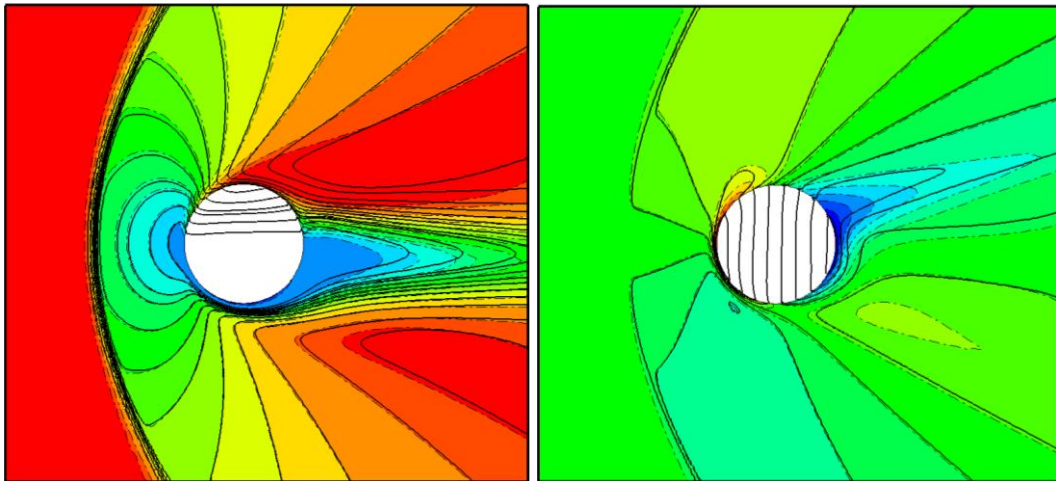


Figure 6.12 The streamlines near the cylinder with pressure contours obtained from present scheme (Left) and DVM (Right)



(a) Density

(b) Pressure



(c) U-velocity

(d) V-velocity

Figure 6.13 Comparison of contours for flow past a rotational cylinder obtained from present scheme (Solid lines) and DVM (Flood and dashed lines)

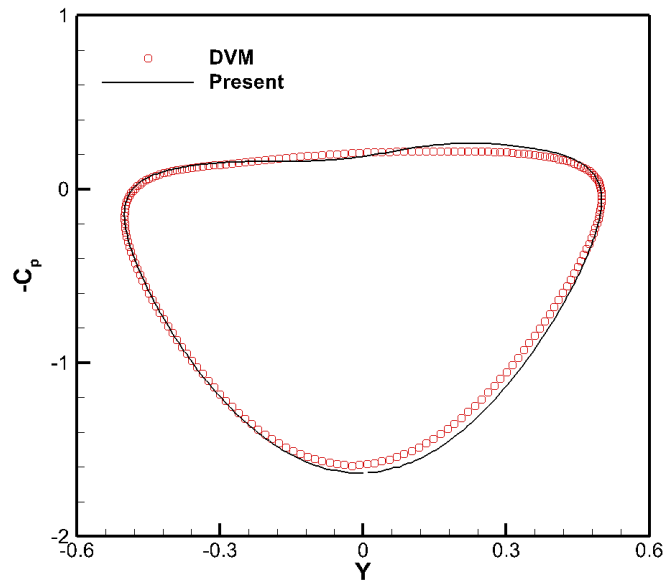


Figure 6.14 Pressure coefficient distributions along the y-direction

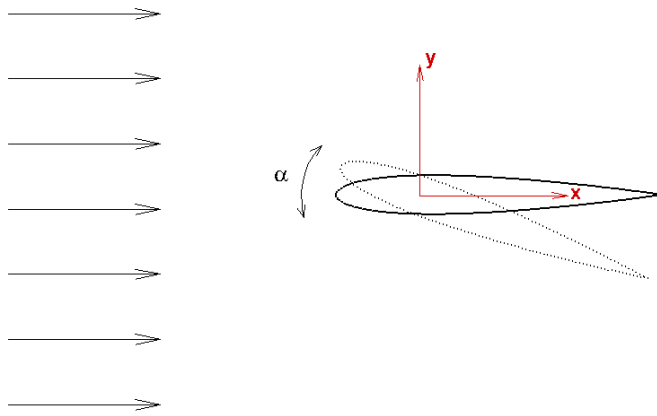
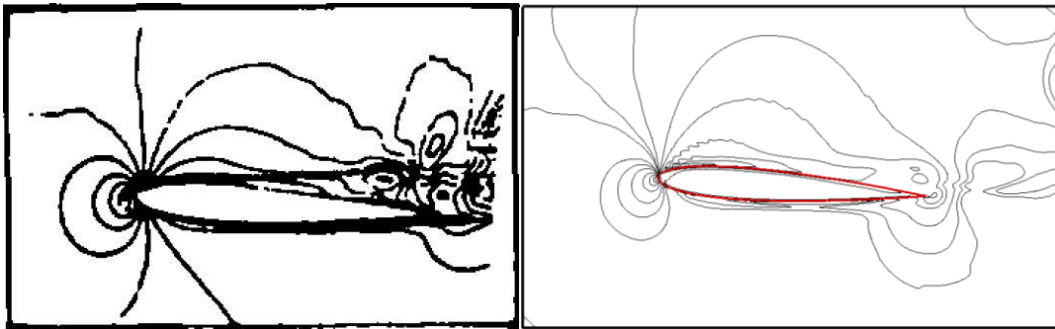
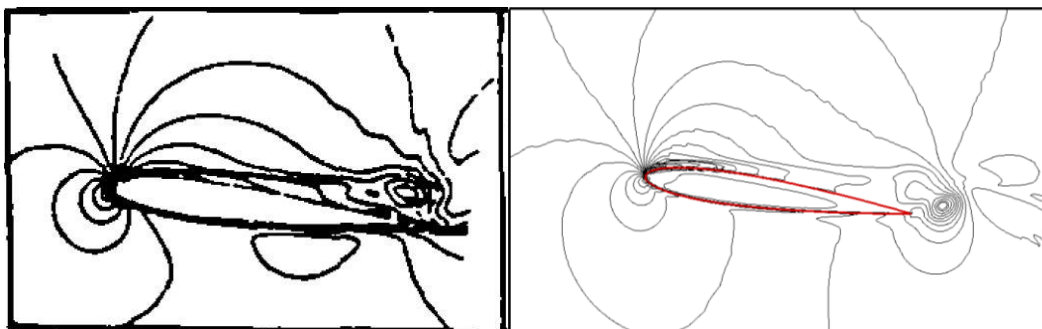


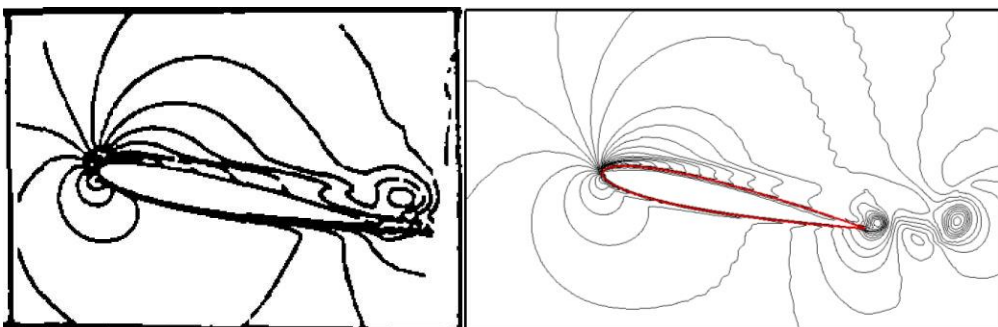
Figure 6.15 Schematic diagram for flow over a harmonic oscillating cylinder



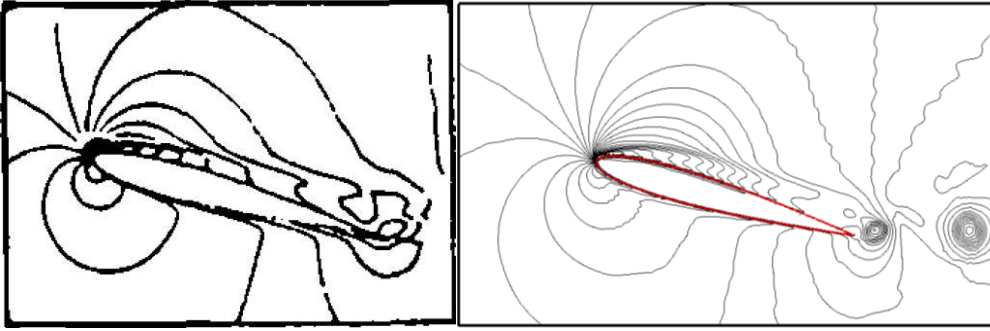
(a) up $\alpha = 4^\circ$



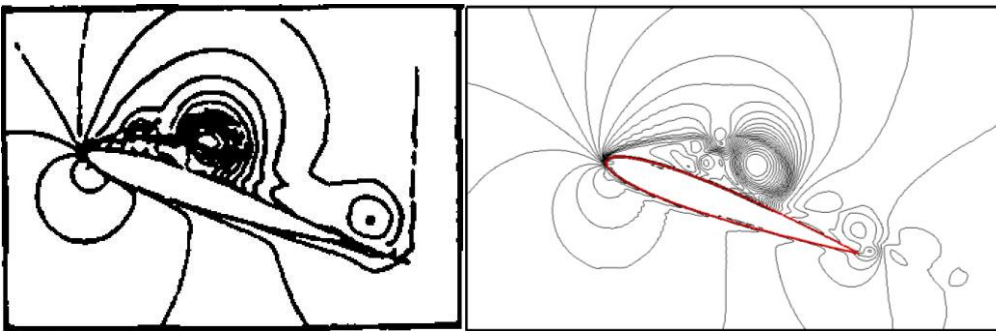
(b) up $\alpha = 8^\circ$



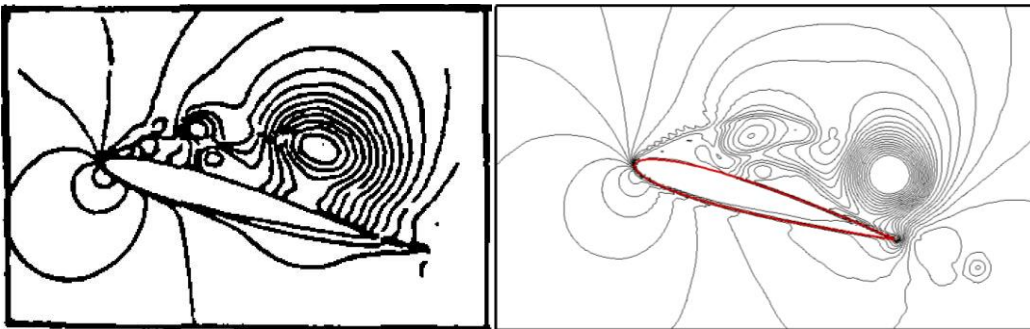
(c) up $\alpha = 12^\circ$



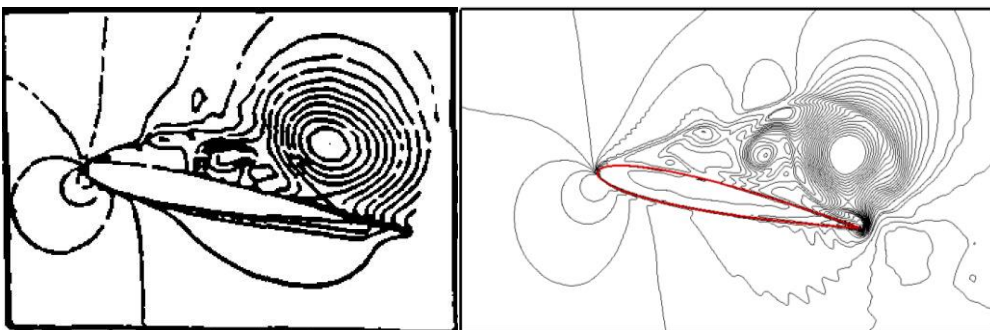
(d) up $\alpha = 16^\circ$



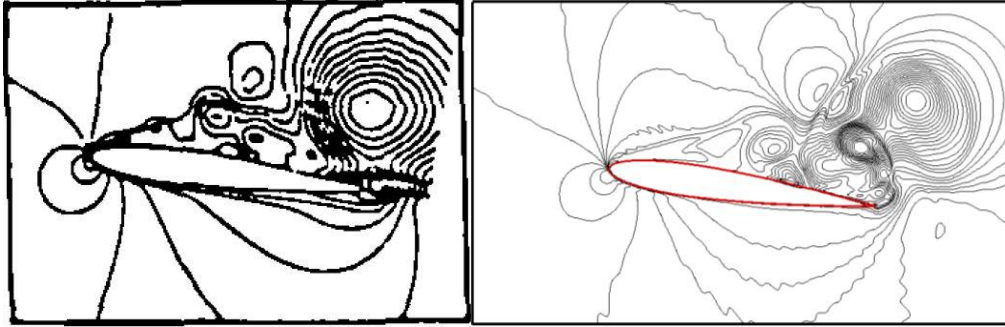
(e) up $\alpha = 20^\circ$



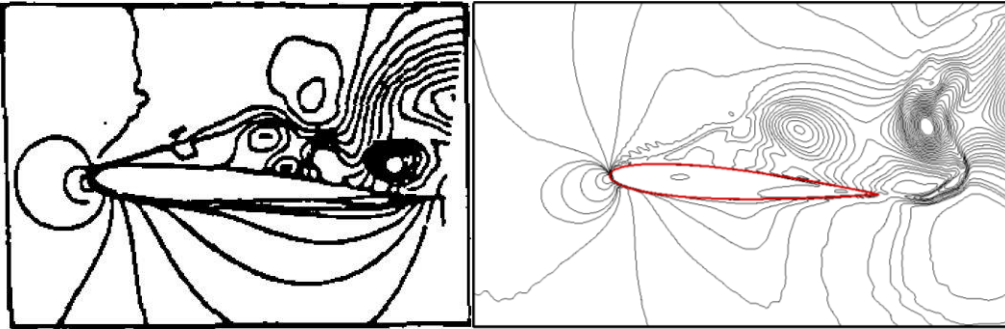
(f) down $\alpha = 16^\circ$



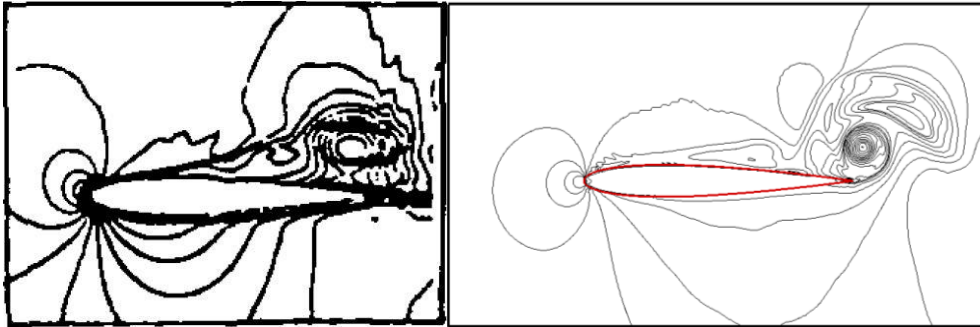
(g) down $\alpha = 12^\circ$



(h) down $\alpha = 8^\circ$



(i) down $\alpha = 4^\circ$



(j) down $\alpha = 0^\circ$

Figure 6.16 Instantaneous density contours for the flow around an oscillating NACA0012 airfoil at $M_\infty = 0.4$, $Re = 5000$, $K_s = 0.5$ of Guo et al. (1994) (Left) and present results (Right)

Chapter 7

Conclusions and Recommendations

7.1 Conclusions

In this thesis, a series of efficient and consistent gas kinetic flux solvers (GKFSs) have been proposed for simulating a variety of 1D to 3D flow problems, including incompressible and compressible flows, inviscid and viscous flows, and complex moving boundary problems. Compared with conventional kinetic flux vector scheme (KFVS) or gas kinetic BGK scheme, the proposed GKFSs have been shown to be versatile and enjoy many attractive advantages, i.e., (1) good control of the numerical dissipation, (2) eliminating the drawbacks of gas kinetic scheme, such as improving the computational efficiency and reducing the complexity by using fewer coefficients, (3) capable of providing explicit formulations for conservative flow variables and numerical fluxes, (4) effectiveness in solving complex moving boundary problems involving incompressible and compressible flows. A detailed summary of the GKFSs will be provided next.

As a type of gas kinetic schemes, the KFVS is only applicable to inviscid flows because of its defect in controlling the numerical dissipation. To remove this drawback, a switch function-based gas kinetic scheme (SF-GKS) was proposed. An easy and efficient way was developed to control the numerical dissipation by introducing a switch function. As a result, the proposed method can not only well capture the strong shock waves but also resolve thin boundary layers. The proposed SF-GKS was firstly validated by simulating inviscid flows, including 1D Euler shock

tube, regular shock reflection and double Mach reflection. It showed that the results of SF-GKS were slightly better than those of KFVS probably because of less numerical dissipation introduced. In addition, the capability of SF-GKS was further extended from simulating inviscid flows to viscous flows. Numerical experiments, including compressible turbulent flow around a RAE2822 airfoil and hypersonic flow around one half of a cylinder, showed that the compressible viscous flows can be well simulated by the developed SF-GKS. Moreover, the computational accuracy and numerical stability were also verified. It is worth mentioning that the SA turbulence model applied in the Reynolds-averaged Navier-Stokes equations performs stably and accurately in the above turbulence flow simulations.

To develop a more general and efficient flux solver based on another type of gas kinetic schemes, a gas kinetic flux solver was proposed. In this solver, the governing differential equations were discretized by the finite volume method and the macroscopic flow variables were directly updated at cell centers. Through the local reconstruction of the Boltzmann equation with BGK approximation, the inviscid and viscous fluxes at cell interface were computed simultaneously by using the gas distribution function, which is evaluated by a much simpler and efficient way in this work. Moreover, the explicit formulations of conservative flow variables and numerical fluxes at cell interface can be explicitly given. Similar to conventional gas kinetic BGK scheme, the BGK collision model was adopted to control the numerical dissipation by a real collision time. The proposed GKFS was successfully validated by simulating several 2D numerical examples. Firstly, three specific schemes of GKFS proposed in this work were applied to simulate incompressible flows, including decaying vortex flow, lid-driven cavity flow, flow over a circular cylinder and

Couette flow with a temperature gradient. Numerical results of decaying vortex flow showed that all three schemes roughly had the second order of accuracy in space. Among them, Scheme II was more attractive due to its high efficiency from the results of lid-driven cavity flow. However, for simulation of compressible flows, Scheme I and Scheme II encountered numerical instability. In contrast, Scheme III performed equally well for simulation of both incompressible and compressible flows. Several numerical examples of compressible flows, such as shock-boundary layer interaction, laminar flow over a NACA0012 airfoil and hypersonic flow over a cylinder, were simulated to verify the capability of GKFS. It showed that GKFS only takes about 57.1% of the CPU time of gas kinetic BGK scheme for shock-boundary layer interaction on the same non-uniform grids. Numerical results also showed the good agreements with the published data in the literature.

To extend the GKFS to solve 3D flows, a truly 3D flux solver was presented for effective simulation of incompressible and compressible 3D flows. The 3D GKFS applied the finite volume method to solve three-dimensional N-S equations. The viscous and inviscid fluxes were evaluated in a simple and easy way at the cell interface by the local reconstruction of the continuous Boltzmann solutions. In the development, a local coordinate transformation was introduced to transform the velocities in the Cartesian coordinate system to the local normal and tangential directions at each cell interface. In this way, all the interfaces can be treated in the same manner. The 3D GKFS has been validated by simulating several 3D incompressible and compressible flows, such as 3D lid-driven cavity flow, incompressible flow past a stationary sphere, flow around an ONERA M6 wing, turbulent flow over the DPW-W1 and DLR-F6 wing-body configuration. The

obtained numerical results compared well with the experimental and/or numerical data in the literature. Through numerical validation, it is verified that the proposed GKFS is capable of simulating challenging flow problems with complex geometries.

Another important contribution of this thesis was the extension of immersed boundary method (IBM) in the framework of GKFS for the simulation of stationary and moving boundary problems. The IBM may be the simplest and efficient way to deal with complex as well as moving solid boundaries. Firstly, the boundary condition-enforced immersed boundary-gas kinetic flux solver (IB-GKFS) was proposed for simulating incompressible flows. In IB-GKFS, a fractional step approach was applied to split the overall solution process into two steps: the predictor step and the corrector step. In the predictor step, the intermediate flow field was predicted by applying the GKFS, which reconstructs the fluxes using the continuous Boltzmann solutions. As the solid boundary was not taken into account in this step, the external forcing term was avoided during the evaluation of numerical flux at each cell interface, which greatly simplifies the implementation. Subsequently, to guarantee the no-slip boundary condition, the intermediate velocity field was corrected by using the implicit boundary condition-enforced immersed boundary method. The proposed IB-GKFS was successfully validated by various stationary and moving boundary flows, such as flow past a stationary circular cylinder and the NACA0012 airfoil, flow past an in-line and transverse oscillating cylinder with a prescribed motion, laminar flow around a rapid pitching NACA0015 airfoil and one particle sedimentation in a rectangular domain. The obtained results achieved good agreements with available data in the literature. Through numerical simulation, it was successfully verified that the proposed IB-GKFS could be effectively applied for complex and moving boundary problems

Although the proposed IB-GKFS was successfully developed for simulating incompressible moving boundary problems, it is not applicable for compressible flows. This is because in IB-GKFS, only the velocity field is corrected and the influence of the immersed boundary on the density, pressure and temperature fields is neglected. There is no mechanism to correct these flow variables in IB-GKFS. To overcome this difficulty, a diffuse interface immersed boundary method was further developed for the simulation of compressible viscous flows around stationary and moving boundaries. In this method, the solution of the flow field and implementation of boundary condition were decoupled into two steps by applying the fractional step technique. The intermediate flow field was first predicted by implementing the proposed GKFS and the correction of flow variables was conducted subsequently by using the current diffuse interface IBM. Two different schemes were proposed for stationary and moving boundaries, respectively. The proposed two methods of diffuse interface IBM were firstly validated by simulating flow around a stationary circular cylinder. The obtained results of two methods were identical with each other and achieved good agreements with the data available in the literature. After that, several stationary and moving boundary flow cases were tested, such as compressible flow over a NACA0012 airfoil, flow over a high-frequency plunging SD7003 airfoil, compressible flow over a rotating cylinder and laminar flow over a harmonic oscillating NACA0012 airfoil. Once again, good agreements between the present results and the published/computed data were achieved, which validated the capability of the present diffuse interface IBM for simulation of complex compressible moving boundary flows. The present diffuse interface IBM was proven to be a much simpler and more flexible way to simulate the compressible moving boundary problems.

Moreover, the proposed correction procedure makes the calculation of hydrodynamic forces on the immersed boundary quite convenient and accurate.

7.2 Recommendations

In the above section, the distinctive features and excellent performance of the present GKFSs have been summarized. In the following, the limitations of the GKFSs will be presented and the corresponding recommendations will be addressed. Firstly, although the explicit formulations of numerical flux can be given in GKFS, it is still not easy to be implemented directly. In this regard, simplification of these formulations should be considered in the future, especially for the incompressible flows. By applying the continuity assumption for flow variable distributions at a cell interface, it is believed that simplification can be made to improve the numerical efficiency. Secondly, only the isothermal boundary condition, which is subject to Dirichlet-type boundary condition, is considered in the current compressible IBM. In the aspect of broader range of application, there is a need to extend this method to Neumann-type boundary condition, such as adiabatic wall. Another possible avenue of future work is to extend the current methods from the second-order to high-order schemes by using high-order schemes to discretize the N-S equations and adding high order derivatives in approximation of distribution functions.

Appendix

Appendix A: Moments of Maxwellian Distribution Function

In this thesis, some notations are taken to simplify the formulations. In this appendix, the notations for the moments of Maxwellian distribution function are introduced.

Firstly, the Maxwellian distribution function for 2D flows is given as

$$g = \rho \left(\frac{\lambda}{\pi} \right)^{\frac{K+2}{2}} e^{-\lambda((u-U)^2+(v-V)^2+\xi^2)} \quad (\text{A.1})$$

For 3D flows, the Maxwellian distribution function becomes

$$g = \rho \left(\frac{\lambda}{\pi} \right)^{\frac{K+3}{2}} e^{-\lambda((u-U)^2+(v-V)^2+(w-W)^2+\xi^2)} \quad (\text{A.2})$$

Following the idea of Xu (2001), the notation for the moments of g is defined as:

$$\rho \langle \dots \rangle = \int (\dots) g du dv dw d\xi \quad (\text{A.3})$$

Then the general moment formulation becomes

$$\langle u^n v^m w^l \xi^p \rangle = \langle u^n \rangle \langle v^m \rangle \langle w^l \rangle \langle \xi^p \rangle \quad (\text{A.4})$$

In general, the phase velocities (u, v, w) are based on the Cartesian coordinate system.

As a result, in a local cell interface, the phase velocities (u, v, w) are not necessarily in the normal or tangential directions of the interface and a local coordinate transformation might be needed. Here, to make the notations general, only the moments of u are presented for phase velocities. When the integral of velocity is from $-\infty$ to $+\infty$, the moments of u^n and ξ^n are

$$\langle u^0 \rangle = 1 \quad (\text{A.5})$$

$$\langle u^1 \rangle = U \quad (\text{A.6})$$

$$\langle u^2 \rangle = U^2 + \frac{1}{2\lambda} \quad (\text{A.7})$$

$$\langle u^3 \rangle = U^3 + \frac{3}{2\lambda} U \quad (\text{A.8})$$

$$\langle u^4 \rangle = U^4 + \frac{3}{\lambda} U^2 + \frac{3}{4\lambda^2} \quad (\text{A.9})$$

$$\langle u^5 \rangle = U^5 + \frac{5}{\lambda} U^3 + \frac{15}{4\lambda^2} U \quad (\text{A.10})$$

and

$$\langle \xi^0 \rangle = 1 \quad (\text{A.11})$$

$$\langle \xi^2 \rangle = \frac{K}{2\lambda} \quad (\text{A.12})$$

$$\langle \xi^4 \rangle = \frac{K^2 + 2K}{4\lambda^2} \quad (\text{A.13})$$

When the moments for u^n are calculated in the half space, the error function and the complementary error function appear in the formulation. Take notation of the integral from 0 to $+\infty$ as $\langle \dots \rangle_{>0}$ and integral from $-\infty$ to 0 as $\langle \dots \rangle_{<0}$, the moments become

$$a_l = \langle u^0 \rangle_{>0} = \frac{1}{2} \operatorname{erfc}(-\sqrt{\lambda_l} U_l) \quad (\text{A.14})$$

$$b_l = \langle u^1 \rangle_{>0} = U_l \cdot a_l + \frac{1}{2} \frac{e^{-\lambda_l U_l^2}}{\sqrt{\pi \lambda_l}} \quad (\text{A.15})$$

$$c_l = \langle u^2 \rangle_{>0} = U_l \cdot b_l + \frac{1}{2\lambda_l} \cdot a_l \quad (\text{A.16})$$

$$d_l = \langle u^3 \rangle_{>0} = U_l \cdot c_l + \frac{1}{\lambda_l} \cdot b_l \quad (\text{A.17})$$

$$e_l = \langle u^4 \rangle_{>0} = U_l \cdot d_l + \frac{3}{2\lambda_l} \cdot c_l \quad (\text{A.18})$$

and

$$a_r = \langle u^0 \rangle_{<0} = \frac{1}{2} \operatorname{erfc}(\sqrt{\lambda_r} U_r) \quad (\text{A.19})$$

$$b_r = \langle u^1 \rangle_{<0} = U_r \cdot a_r - \frac{1}{2} \frac{e^{-\lambda_r U_r^2}}{\sqrt{\pi \lambda_r}} \quad (\text{A.20})$$

$$c_r = \langle u^2 \rangle_{<0} = U_r \cdot b_r + \frac{1}{2 \lambda_r} \cdot a_r \quad (\text{A.21})$$

$$d_r = \langle u^3 \rangle_{<0} = U_r \cdot c_r + \frac{1}{\lambda_r} \cdot b_r \quad (\text{A.22})$$

$$e_r = \langle u^4 \rangle_{<0} = U_r \cdot d_r + \frac{3}{2 \lambda_r} \cdot c_r \quad (\text{A.23})$$

The same formulation can be obtained for $\langle v^m \rangle$ and $\langle w^l \rangle$ by changing U to V and W in the above moments of $\langle u^n \rangle$.

Appendix B: Expressions of Conservative Flow Variables and Fluxes in y Direction

In Chapter 3, the detailed derivation of conservative flow variables and fluxes at the interface in the x direction is presented. When evaluating conservative flow variables and fluxes in the y direction, the integral domain of velocity v is separated to $(-\infty, 0)$ and $(0, +\infty)$. Therefore, the moments in Appendix A should be changed to

$$a_i^v = \langle v^0 \rangle_{>0} = \frac{1}{2} \operatorname{erfc}(-\sqrt{\lambda_i} V_i) \quad (\text{B.1})$$

$$b_i^v = \langle v^1 \rangle_{>0} = V_i \cdot a_i^v + \frac{1}{2} \frac{e^{-\lambda_i V_i^2}}{\sqrt{\pi \lambda_i}} \quad (\text{B.2})$$

$$c_l^v = \langle v^2 \rangle_{>0} = V_l \cdot b_l^v + \frac{1}{2\lambda_l} \cdot a_l^v \quad (\text{B.3})$$

$$d_l^v = \langle v^3 \rangle_{>0} = V_l \cdot c_l^v + \frac{1}{\lambda_l} \cdot b_l^v \quad (\text{B.4})$$

$$e_l^v = \langle v^4 \rangle_{>0} = V_l \cdot d_l^v + \frac{3}{2\lambda_l} \cdot c_l^v \quad (\text{B.5})$$

and similarly

$$a_r^v = \langle v^0 \rangle_{<0} = \frac{1}{2} \operatorname{erfc}(\sqrt{\lambda_r} V_r) \quad (\text{B.6})$$

$$b_r^v = \langle v^1 \rangle_{<0} = V_r \cdot a_r^v - \frac{1}{2} \frac{e^{-\lambda_r V_r^2}}{\sqrt{\pi \lambda_r}} \quad (\text{B.7})$$

$$c_r^v = \langle v^2 \rangle_{<0} = V_r \cdot b_r^v + \frac{1}{2\lambda_r} \cdot a_r^v \quad (\text{B.8})$$

$$d_r^v = \langle v^3 \rangle_{<0} = V_r \cdot c_r^v + \frac{1}{\lambda_r} \cdot b_r^v \quad (\text{B.9})$$

$$e_r^v = \langle v^4 \rangle_{<0} = V_r \cdot d_r^v + \frac{3}{2\lambda_r} \cdot c_r^v \quad (\text{B.10})$$

The values of moments whose integrals are from $-\infty$ to $+\infty$ are the same as those in Appendix A.

With parameters defined above, the conservative flow variables \mathbf{W} at the interface in the y direction are given by

$$\rho = (\rho_l \cdot a_l^v + \rho_r \cdot a_r^v) - \left[\frac{\partial(\rho_l U_l \cdot a_l^v + \rho_r U_r \cdot a_r^v)}{\partial x} + \frac{\partial(\rho_l \cdot b_l^v + \rho_r \cdot b_r^v)}{\partial y} \right] \delta t \quad (\text{B.11})$$

$$\begin{aligned} \rho U &= (\rho_l U_l \cdot a_l^v + \rho_r U_r \cdot a_r^v) \\ &\quad - \left[\frac{\partial [(\rho_l U_l^2 + p_l) \cdot a_l^v + (\rho_r U_r^2 + p_r) \cdot a_r^v]}{\partial x} + \frac{\partial (\rho_l U_l \cdot b_l^v + \rho_r U_r \cdot b_r^v)}{\partial y} \right] \delta t \end{aligned} \quad (\text{B.12})$$

$$\rho V = (\rho_l \cdot b_l^v + \rho_r \cdot b_r^v) - \left[\frac{\partial (\rho_l U_l \cdot b_l^v + \rho_r U_r \cdot b_r^v)}{\partial x} + \frac{\partial (\rho_l \cdot c_l^v + \rho_r \cdot c_r^v)}{\partial y} \right] \delta t \quad (\text{B.13})$$

$$\begin{aligned} \rho E &= \frac{1}{2} \rho_l [c_l^v + (U_l^2 + (b-1)RT_l) \cdot a_l^v] + \frac{1}{2} \rho_r [c_r^v + (U_r^2 + (b-1)RT_r) \cdot a_r^v] \\ &\quad - \frac{1}{2} \left\{ \frac{\partial}{\partial x} \left[\rho_l U_l [c_l^v + (U_l^2 + (b+1)RT_l) \cdot a_l^v] + \rho_r U_r [c_r^v + (U_r^2 + (b+1)RT_r) \cdot a_r^v] \right] \right. \\ &\quad \left. + \frac{\partial}{\partial y} \left[\rho_l [d_l^v + (U_l^2 + (b-1)RT_l) \cdot b_l^v] + \rho_r [d_r^v + (U_r^2 + (b-1)RT_r) \cdot b_r^v] \right] \right\} \delta t \end{aligned} \quad (\text{B.14})$$

Eqs. (B.11)-(B.14) are the explicit expressions of conservative flow variables for Scheme I. For Scheme II and Scheme III, the expressions are simplified as

$$\rho = (\rho_l \cdot a_l^v + \rho_r \cdot a_r^v) \quad (\text{B.15})$$

$$\rho U = (\rho_l U_l \cdot a_l^v + \rho_r U_r \cdot a_r^v) \quad (\text{B.16})$$

$$\rho V = (\rho_l \cdot b_l^v + \rho_r \cdot b_r^v) \quad (\text{B.17})$$

$$\rho E = \frac{1}{2} \rho_l [c_l^v + (U_l^2 + (b-1)RT_l) \cdot a_l^v] + \frac{1}{2} \rho_r [c_r^v + (U_r^2 + (b-1)RT_r) \cdot a_r^v] \quad (\text{B.18})$$

After the evaluation of conservative flow variables, the explicit expressions for numerical fluxes are given by

$$\mathbf{F}^y = \mathbf{F}^{y(0)} + \mathbf{F}^{y(1)} \quad (\text{B.19})$$

where

$$\mathbf{F}^{y(0)} = \int v \varphi_\alpha g(0, 0, t + \delta t) d\Xi = \begin{bmatrix} \rho V \\ \rho UV \\ \rho V^2 + p \\ (\rho E + p)V \end{bmatrix} \quad (\text{B.20})$$

and

$$F_1^{y(1)} = -\frac{\tau}{\delta t} (\rho V - \rho_l \cdot b_l^v - \rho_r \cdot b_r^v) - \tau \left[\frac{\partial (\rho_l U_l \cdot b_l^v + \rho_r U_r \cdot b_r^v)}{\partial x} + \frac{\partial (\rho_l \cdot c_l^v + \rho_r \cdot c_r^v)}{\partial y} \right] \quad (\text{B.21})$$

$$F_2^{y(1)} = -\frac{\tau}{\delta t} [\rho UV - \rho_l U_l \cdot b_l^v - \rho_r U_r \cdot b_r^v] - \tau \left[\frac{\partial [(\rho_l U_l^2 + p_l) \cdot b_l^v + (\rho_r U_r^2 + p_r) \cdot b_r^v]}{\partial x} + \frac{\partial (\rho_l U_l \cdot c_l^v + \rho_r U_r \cdot c_r^v)}{\partial y} \right] \quad (\text{B.22})$$

$$F_3^{y(1)} = -\frac{\tau}{\delta t} [(\rho V^2 + p) - \rho_l \cdot c_l^v - \rho_r \cdot c_r^v] - \tau \left[\frac{\partial (\rho_l U_l \cdot c_l^v + \rho_r U_r \cdot c_r^v)}{\partial x} + \frac{\partial [\rho_l \cdot d_l^v + \rho_r \cdot d_r^v]}{\partial y} \right] \quad (\text{B.23})$$

$$F_4^{y(1)} = -\frac{1}{2} \frac{\tau}{\delta t} \left\{ \rho V [(U^2 + V^2) + (b+2)RT] - \rho_l [d_l^v + (U_l^2 + (b-1)RT_l) \cdot b_l^v] - \rho_r [d_r^v + (U_r^2 + (b-1)RT_r) \cdot b_r^v] \right\} - \frac{1}{2} \tau \left\{ \frac{\partial}{\partial x} [\rho_l U_l [d_l^v + (U_l^2 + (b+1)RT_l) \cdot b_l^v] + \rho_r U_r [d_r^v + (U_r^2 + (b+1)RT_r) \cdot b_r^v]] \right\} + \frac{\partial}{\partial y} \left\{ \rho_l [e_l^v + (U_l^2 + (b-1)RT_l) \cdot c_l^v] + \rho_r [e_r^v + (U_r^2 + (b-1)RT_r) \cdot c_r^v] \right\} \quad (\text{B.24})$$

Eqs. (B.21)-(B.24) are for Scheme I and Scheme II, in which the streaming time step

δt is involved in the expressions. For Scheme III, $\mathbf{F}^{y(1)}$ should be changed to

$$F_1^{y(1)} = -\tau \rho \left[A_1 \langle v^1 \rangle + A_2 \langle u^1 \rangle \langle v^1 \rangle + A_3 \langle v^2 \rangle + \frac{1}{2} A_4 (\langle u^2 \rangle \langle v^1 \rangle + \langle v^3 \rangle + \langle v^1 \rangle \langle \xi^2 \rangle) \right] - \tau \left[\frac{\partial (\rho_l U_l \cdot b_l^v + \rho_r U_r \cdot b_r^v)}{\partial x} + \frac{\partial (\rho_l \cdot c_l^v + \rho_r \cdot c_r^v)}{\partial y} \right] \quad (\text{B.25})$$

$$\begin{aligned}
F_2^{y(1)} = & -\tau\rho \left[A_1 \langle u^1 \rangle \langle v^1 \rangle + A_2 \langle u^2 \rangle \langle v^1 \rangle + A_3 \langle u^1 \rangle \langle v^2 \rangle \right. \\
& \left. + \frac{1}{2} A_4 \left(\langle u^3 \rangle \langle v^1 \rangle + \langle u^1 \rangle \langle v^3 \rangle + \langle u^1 \rangle \langle v^1 \rangle \langle \xi^2 \rangle \right) \right] \\
& - \tau \left[\frac{\partial \left[(\rho_l U_l^2 + p_l) \cdot b_l^v + (\rho_r U_r^2 + p_r) \cdot b_r^v \right]}{\partial x} + \frac{\partial (\rho_l U_l \cdot c_l^v + \rho_r U_r \cdot c_r^v)}{\partial y} \right]
\end{aligned} \tag{B.26}$$

$$\begin{aligned}
F_3^{y(1)} = & -\tau\rho \left[A_1 \langle v^2 \rangle + A_2 \langle u^1 \rangle \langle v^2 \rangle + A_3 \langle v^3 \rangle + \frac{1}{2} A_4 \left(\langle u^2 \rangle \langle v^2 \rangle + \langle v^4 \rangle + \langle v^2 \rangle \langle \xi^2 \rangle \right) \right] \\
& - \tau \left[\frac{\partial (\rho_l U_l \cdot c_l^v + \rho_r U_r \cdot c_r^v)}{\partial x} + \frac{\partial (\rho_l \cdot d_l^v + \rho_r \cdot d_r^v)}{\partial y} \right]
\end{aligned} \tag{B.27}$$

$$\begin{aligned}
F_4^{y(1)} = & -\frac{1}{2} \tau\rho \left\{ A_1 \left[\langle u^2 \rangle \langle v^1 \rangle + \langle v^3 \rangle + \langle v^1 \rangle \langle \xi^2 \rangle \right] \right. \\
& + A_2 \left[\langle u^3 \rangle \langle v^1 \rangle + \langle u^1 \rangle \langle v^3 \rangle + \langle u^1 \rangle \langle v^1 \rangle \langle \xi^2 \rangle \right] \\
& + A_3 \left[\langle u^2 \rangle \langle v^2 \rangle + \langle v^4 \rangle + \langle v^2 \rangle \langle \xi^2 \rangle \right] \\
& + \frac{1}{2} A_4 \left[\langle u^4 \rangle \langle v^1 \rangle + \langle v^5 \rangle + \langle v^1 \rangle \langle \xi^4 \rangle + 2 \langle u^2 \rangle \langle v^3 \rangle + 2 \langle u^2 \rangle \langle v^1 \rangle \langle \xi^2 \rangle + 2 \langle v^3 \rangle \langle \xi^2 \rangle \right] \left. \right\} \\
& - \frac{1}{2} \tau \left\{ \frac{\partial}{\partial x} \left\{ \rho_l U_l \left[d_l^v + (U_l^2 + (b+1)RT_l) \cdot b_l^v \right] + \rho_r U_r \left[d_r^v + (U_r^2 + (b+1)RT_r) \cdot b_r^v \right] \right\} \right. \\
& \left. + \frac{\partial}{\partial y} \left\{ \rho_l \left[e_l^v + (U_l^2 + (b-1)RT_l) \cdot c_l^v \right] + \rho_r \left[e_r^v + (U_r^2 + (b-1)RT_r) \cdot c_r^v \right] \right\} \right\}
\end{aligned} \tag{B.28}$$

Appendix C: Expression for heat flux q

As is well known, when the Maxwellian distribution function is adopted as the equilibrium state, the Prandtl number is fixed to unity. In order to adjust the Prandtl number to any realistic value, the modification of the heat flux with a variable Prandtl number can be selected (Eq. (3. 80)). The heat flux q is evaluated by Eq. (3.81), which is expressed in the integral form

$$q = \frac{1}{2} \int (u-U) \left((u-U)^2 + (v-V)^2 + \xi^2 \right) f d\Xi \tag{C.1}$$

In this appendix, the expression for the heat flux q is presented. The above equation can be expanded to

$$\begin{aligned} q &= \frac{1}{2} \int (u-U) \left[(u-U)^2 + (v-V)^2 + \xi^2 \right] f d\Xi \\ &= \int (u-U) \left[\frac{1}{2} (u^2 + v^2 + \xi^2) - uU - vV + \frac{1}{2} (U^2 + V^2) \right] f d\Xi \end{aligned} \quad (\text{C.2})$$

As $\int u f d\Xi = U \int f d\Xi$, Eq. (C.1) can be rewritten as

$$\begin{aligned} q &= \int (u-U) \left[\frac{1}{2} (u^2 + v^2 + \xi^2) - uU - vV \right] f d\Xi \\ &= \int u \frac{1}{2} (u^2 + v^2 + \xi^2) f d\Xi - U \int u^2 f d\Xi - V \int u v f d\Xi \\ &\quad - U \int \frac{1}{2} (u^2 + v^2 + \xi^2) f d\Xi + U^2 \int u f d\Xi + UV \int v f d\Xi \end{aligned} \quad (\text{C.3})$$

It can be found that the numerical fluxes for momentum and energy equations in the x direction are

$$F_{\rho U}^x = \int u v f d\Xi \quad (\text{C.4})$$

$$F_{\rho V}^x = \int u^2 f d\Xi \quad (\text{C.5})$$

$$F_{\rho E}^x = \int u \frac{1}{2} (u^2 + v^2 + \xi^2) f d\Xi \quad (\text{C.6})$$

where $F_{\rho U}^x$ and $F_{\rho V}^x$ are the fluxes of momentum equations and $F_{\rho E}^x$ is the flux of energy equation. On the other hand, the conservative flow variables can be expressed as

$$\rho U = \int u f d\Xi \quad (\text{C.7})$$

$$\rho V = \int v f d\Xi \quad (\text{C.8})$$

$$\rho E = \int \frac{1}{2} (u^2 + v^2 + \xi^2) f d\Xi \quad (\text{C.9})$$

where ρ, U, V, E are the flow variables on the interface.

Substituting Eqs. (C.3)-(C.9) into Eq. (C.2), the heat flux q can be expressed as

$$q = F_{\rho E}^x - U \cdot F_{\rho U}^x - V \cdot F_{\rho V}^x - U \left(\rho E - \rho U^2 - \rho V^2 \right) \quad (\text{C.10})$$

The above Eq. (C.10) means that the heat flux q can be obtained after the evaluation of the conservative flow variables and numerical fluxes at the cell interface. Not much additional computational effort is required by taking this modification. In addition, for three-dimensional theory, the heat flux q can be written similarly

$$q = F_{\rho E}^x - U \cdot F_{\rho U}^x - V \cdot F_{\rho V}^x - W \cdot F_{\rho W}^x - U \left(\rho E - \rho U^2 - \rho V^2 - \rho W^2 \right) \quad (\text{C.11})$$

References

- Anderson J. D., Wendt J., Computational fluid dynamics, *Springer*, 1995.
- Bassi F., Rebay S., A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier–Stokes equations, *Journal of computational physics* 131:267-279, 1997.
- Benson M., Bellamy-Knights P., Gerrard J., Gladwell I., A viscous splitting algorithm applied to low Reynolds number flows round a circular cylinder, *Journal of Fluids and Structures* 3:439-479, 1989.
- Bhatnagar P. L., Gross E. P., Krook M., A model for collision processes in gases. I. Small amplitude processes in charged and neutral one-component systems, *Physical Review* 94:511, 1954.
- Braza M., Chassaing P., Minh H. H., Numerical study and physical analysis of the pressure and velocity fields in the near wake of a circular cylinder, *Journal of fluid mechanics* 165:79-130, 1986.
- Bristeau M. O., Numerical simulation of compressible Navier-Stokes flows: a GAMM workshop, *Springer Science & Business Media*, 2013.
- Bristeau M. O., Glowinski R., Periaux J., Viviand H., Numerical simulation of compressible Navier-Stokes flows, *Springer*, 1987.
- Chae D., Kim C., Rho O.-H., Development of an improved gas-kinetic BGK scheme for inviscid and viscous flows, *Journal of Computational Physics* 158:1-27, 2000.
- Chapman S., Cowling T. G., The mathematical theory of non-uniform gases: an account of the kinetic theory of viscosity, thermal conduction and diffusion in gases, *Cambridge university press*, 1970.

- Chen S., Doolen G. D., Lattice Boltzmann method for fluid flows, *Annual review of fluid mechanics* 30:329-364, 1998.
- Chen S., Jin C., Li C., Cai Q., Gas-kinetic scheme with discontinuous derivative for low speed flow computation, *Journal of Computational Physics* 230:2045-2059, 2011.
- Chen S., Xu K., Lee C., Cai Q., A unified gas kinetic scheme with moving mesh and velocity space adaptation, *Journal of Computational Physics* 231:6643-6664, 2012.
- Chou S.-Y., Baganoff D., Kinetic flux–vector splitting for the Navier–Stokes equations, *Journal of Computational Physics* 130:217-230, 1997.
- Cockburn B., Shu C.-W., TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws. II. General framework, *Mathematics of Computation* 52:411-435, 1989.
- Cook P. H., McDonald M. A., Firmin M. C. P., Aerofoil RAE2822-Pressure Distributions, and Boundary Layer and Wake Measurements, *AGARD Report AR 138*, 1979.
- Crumpton P., Mackenzie J., Morton K., Cell vertex algorithms for the compressible Navier-Stokes equations, *Journal of Computational Physics* 109:1-15, 1993.
- De Palma P., de Tullio M. D., Pascazio G., Napolitano M., An immersed-boundary method for compressible viscous flows, *Computers & Fluids* 35:693-702, 2006.
- Dennis S., Chang G.-Z., Numerical solutions for steady flow past a circular cylinder at Reynolds numbers up to 100, *Journal of Fluid Mechanics* 42:471-489, 1970.
- Deshpande S. M., A second-order accurate kinetic-theory-based method for inviscid compressible flows, *NASA STI/Recon Technical Report N 87:18783*, 1986.

- Ding H., Shu C., Cai Q., Applications of stencil-adaptive finite difference method to incompressible viscous flows with curved boundary, *Computers & fluids* 36:786-793, 2007.
- Ding H., Shu C., Yeo K., Xu D., Simulation of incompressible viscous flows past a circular cylinder by hybrid FD scheme and meshless least square-based finite difference method, *Computer Methods in Applied Mechanics and Engineering* 193:727-744, 2004.
- Donea J., Huerta A., Finite element methods for flow problems, *John Wiley & Sons*, 2003.
- DÜTsch H., Durst F., Becker S., Lienhart H., Low-Reynolds-number flow around an oscillating circular cylinder at low Keulegan-Carpenter numbers, *Journal of Fluid Mechanics* 360:249-271, 1998.
- Elizarova T. G. e., Chetverushkin B. N., Kinetic algorithms for calculating gas dynamic flows, *USSR Computational Mathematics and Mathematical Physics* 25:164-169, 1985.
- Engquist B., Osher S., One-sided difference approximations for nonlinear conservation laws, *Mathematics of Computation* 36:321-351, 1981.
- Fadlun E., Verzicco R., Orlandi P., Mohd-Yusof J., Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations, *Journal of Computational Physics* 161:35-60, 2000.
- Feng Z.-G., Michaelides E. E., The immersed boundary-lattice Boltzmann method for solving fluid-particles interaction problems, *Journal of Computational Physics* 195:602-628, 2004.

- Ghia U., Ghia K. N., Shin C. T., High-Re solutions for incompressible flow using the Navier-Stokes equations and a multigrid method, *Journal of Computational Physics* 48:387-411, 1982.
- Gilmanov A., Sotiropoulos F., Balaras E., A general reconstruction algorithm for simulating flows with complex 3D immersed boundaries on Cartesian grids, *Journal of Computational Physics* 191:660-669, 2003.
- Godunov S. K., A difference method for numerical calculation of discontinuous solutions of the equations of hydrodynamics, *Matematicheskii Sbornik* 89:271-306, 1959.
- Guilmineau E., Queutey P., A numerical simulation of vortex shedding from an oscillating circular cylinder, *Journal of Fluids and Structures* 16:773-794, 2002.
- Guo W., Fu D., Ma Y., Numerical investigation of dynamic stall of an oscillating aerofoil, *International journal for numerical methods in fluids* 19:723-734, 1994.
- Guo Z., Liu H., Luo L.-S., Xu K., A comparative study of the LBE and GKS methods for 2D near incompressible laminar flows, *Journal of Computational Physics* 227:4955-4976, 2008.
- Guo Z., Wang R., Xu K., Discrete unified gas kinetic scheme for all Knudsen number flows. II. Thermal compressible case, *Physical Review E* 91:033313, 2015.
- Guo Z., Xu K., Wang R., Discrete unified gas kinetic scheme for all Knudsen number flows: Low-speed isothermal case, *Physical Review E* 88:033305, 2013.
- Hakkinen R. J., Greber I., Trilling L., Abarbanel S., The interaction of an oblique shock wave with a laminar boundary layer, *NASA*, 1959.

- Hardy J., Pomeau Y., de Pazzis O., Time Evolution of a Two-Dimensional Classical Lattice System, *Physical Review Letters* 31:276-279, 1973.
- Harten A., High resolution schemes for hyperbolic conservation laws, *Journal of computational physics* 49:357-393, 1983.
- Harten A., Engquist B., Osher S., Chakravarthy S. R., Uniformly high order accurate essentially non-oscillatory schemes, III, *Journal of computational physics* 71:231-303, 1987.
- Helin H., Walker J., Interrelated effects of pitch rate and pivot point on airfoil dynamic stall, *AIAA* 85-0130, 1985.
- Higuera F. J., Jiménez J., Boltzmann Approach to Lattice Gas Simulations, *EPL (Europhysics Letters)* 9:663, 1989.
- Imamura T., Suzuki K., Nakamura T., Yoshida M., Flow simulation around an airfoil using lattice Boltzmann method on generalized coordinates, *AIAA* 244, 2004.
- Ivan L., Groth C. P. T., High-order solution-adaptive central essentially non-oscillatory (CENO) method for viscous flows, *Journal of Computational Physics* 257:830-862, 2014.
- Jawahar P., Kamath H., A high-resolution procedure for Euler and Navier–Stokes computations on unstructured grids, *Journal of Computational Physics* 164:165-203, 2000.
- Jiang G.-S., Shu C.-W., Efficient Implementation of Weighted ENO Schemes, *Journal of Computational Physics* 126:202-228, 1996.
- Jiang S., Ni G., A γ -model BGK scheme for compressible mult fluids, *International journal for numerical methods in fluids* 46:163-182, 2004.
- Jiang S., Ni G., A second-order γ -model BGK scheme for multimaterial compressible flows, *Applied numerical mathematics* 57:597-608, 2007.

- Johnson T., Patel V., Flow past a sphere up to a Reynolds number of 300, *Journal of Fluid Mechanics* 378:19-70, 1999.
- Katz A. J., Meshless methods for computational fluid dynamics, *ProQuest*, 2009.
- Kim J., Kim D., Choi H., An Immersed-Boundary Finite-Volume Method for Simulations of Flow in Complex Geometries, *Journal of Computational Physics* 171:132-150, 2001.
- Kim K., Kim C., Rho O.-H., Hong S., Cure for shock instability: development of an improved Roe scheme, *AIAA* 548:2002, 2002.
- Laflin K. R., Klausmeyer S. M., Zickuhr T., Vassberg J. C., Wahls R. A., Morrison J. H., Brodersen O. P., Rakowitz M. E., Tinoco E. N., Godard J.-L., Data Summary from Second AIAA Computational Fluid Dynamics Drag Prediction Workshop, *Journal of Aircraft* 42:1165-1178, 2005.
- Lee C., Su Z., Zhong H., Chen S., Zhou M., Wu J., Experimental investigation of freely falling thin disks. Part 2. Transition of three-dimensional motion from zigzag to spiral, *Journal of Fluid Mechanics* 732:77-104, 2013.
- Lee C. B., Hong Z. X., Kachanov Y. S., Borodulin V. I., Gaponenko V. V., A study in transitional flat plate boundary layers: measurement and visualization, *Experiments in Fluids* 28:243-251, 2000.
- Lee C. B., Wang S., Study of the shock motion in a hypersonic shock system/turbulent boundary layer interaction, *Experiments in Fluids* 19:143-149, 1995.
- Lee C. B., Wu J. Z., Transition in Wall-Bounded Flows, *Applied Mechanics Reviews* 61:030802-030802, 2008.
- LeVeque R. J., Numerical methods for conservation laws, *Springer*, 1992.

- Li Q., Fu S. Application of Gas-Kinetic BGK Scheme in Three-Dimensional Flow, *49th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, 2011.
- Li Q., Fu S., Xu K., A compressible Navier–Stokes flow solver with scalar transport, *Journal of Computational Physics* 204:692-714, 2005.
- Liu C., Zheng X., Sung C., Preconditioned multigrid methods for unsteady incompressible flows, *Journal of Computational Physics* 139:35-57, 1998.
- Liu H., Gas-Kinetic Methods for Viscous Fluid Flows, *Ph.D. Thesis, The Hong Kong University of Science and Technology*, 2007.
- Liu X.-D., Osher S., Chan T., Weighted essentially non-oscillatory schemes, *Journal of computational physics* 115:200-212, 1994.
- Liu Y., Vinokur M., Wang Z., Spectral difference method for unstructured grids I: basic formulation, *Journal of Computational Physics* 216:780-801, 2006a.
- Liu Y., Vinokur M., Wang Z. J., Spectral (finite) volume method for conservation laws on unstructured grids V: extension to three-dimensional systems, *Journal of Computational Physics* 212:454-472, 2006b.
- Lockard D. P., Luo L.-S., Milder S. D., Singer B. A., Evaluation of PowerFLOW for aerodynamic applications, *Journal of Statistical Physics* 107:423-478, 2002.
- Lomtev I., Kirby R., Karniadakis G., A discontinuous Galerkin ALE method for compressible viscous flows in moving domains, *Journal of Computational Physics* 155:128-159, 1999.
- Macrossan M. N., The equilibrium flux method for the calculation of flows with non-equilibrium chemical reactions, *Journal of Computational Physics* 80:204-231, 1989.

- Mandal J., Deshpande S., Kinetic flux vector splitting for Euler equations, *Computers & Fluids* 23:447-478, 1994.
- Mavriplis D. J., Jameson A., Multigrid solution of the Navier-Stokes equations on triangular meshes, *AIAA* 28:1415-1425, 1990.
- May G., Srinivasan B., Jameson A., An improved gas-kinetic BGK finite-volume method for three-dimensional transonic flow, *Journal of Computational Physics* 220:856-878, 2007.
- McGowan G., Gopalarathnam A., Ol M., Edwards J., Fredberg D., Computation vs. experiment for high-frequency low-Reynolds number airfoil pitch and plunge, *AIAA* 653, 2008.
- McGowan G. Z., Granlund K., Ol M. V., Gopalarathnam A., Edwards J. R., Investigations of lift-based pitch-plunge equivalence for airfoils at low Reynolds numbers, *AIAA* 49:1511-1524, 2011.
- McNamara G. R., Zanetti G., Use of the Boltzmann equation to simulate lattice-gas automata, *Physical Review Letters* 61:2332, 1988.
- Mittal S., Tezduyar T., A unified finite element formulation for compressible and incompressible flows using augmented conservation variables, *Computer Methods in Applied Mechanics and Engineering* 161:229-243, 1998.
- Moschetta J.-M., Pullin D. I., A Robust Low Diffusive Kinetic Scheme for the Navier–Stokes/Euler Equations, *Journal of Computational Physics* 133:193-204, 1997.
- Ni G., Jiang S., Xu X., A moving mesh BGK scheme for multi-material flows, *International Journal for Numerical Methods in Fluids* 69:878-896, 2012.
- Ohwada T., Xu K., The kinetic scheme for the full-Burnett equations, *Journal of Computational Physics* 201:315-332, 2004.

- Pan L., Xu K., A third-order gas-kinetic scheme for three-dimensional inviscid and viscous flow computations, *Computers & Fluids* 119:250-260, 2015.
- Pan L., Zhao G., Tian B., Wang S., A gas kinetic scheme for the Baer–Nunziato two-phase flow model, *Journal of Computational Physics* 231:7518-7536, 2012.
- Park J., Kwon K., Choi H., Numerical solutions of flow past a circular cylinder at Reynolds numbers up to 160, *KSME International Journal*, 12 (1998) 1200-1205.
- Perthame B., Second-order Boltzmann schemes for compressible Euler equations in one and two space dimensions, *SIAM Journal on Numerical Analysis* 29:1-19, 1992.
- Peskin C. S., Numerical analysis of blood flow in the heart, *Journal of Computational Physics* 25:220-252, 1977.
- Peskin C. S., The immersed boundary method, *Acta numerica* 11:479-517, 2002.
- Prendergast K. H., Xu K., Numerical hydrodynamics from gas-kinetic theory, *Journal of Computational Physics* 109:53-66, 1993.
- Pullin D., Direct simulation methods for compressible inviscid ideal-gas flow, *Journal of Computational Physics* 34:231-244, 1980.
- Qian Y., d'Humières D., Lallemand P., Lattice BGK models for Navier-Stokes equation, *EPL (Europhysics Letters)* 17:479, 1992.
- Ren W., Wu J., Shu C., Yang W., A stream function–vorticity formulation-based immersed boundary method and its applications, *International Journal for Numerical Methods in Fluids* 70:627-645, 2012.
- Roache P. J., Computational fluid dynamics, *Hermosa publishers*, 1972.
- Roe P. L., Approximate Riemann solvers, parameter vectors, and difference schemes, *Journal of computational physics* 43:357-372, 1981.

- Ruan Y., Jameson A., Gas-kinetic BGK method for three-dimensional compressible flows, *AIAA 550*, 2002.
- Schmitt V., Charpin F., Pressure distributions on the ONERA-M6-Wing at transonic Mach numbers, *Experimental data base for computer program assessment 4*, 1979.
- Shakhov E. M., Generalization of the Krook kinetic relaxation equation, *Fluid Dyn* 3:95-96, 1968.
- Shu C.-W., Osher S., Efficient implementation of essentially non-oscillatory shock-capturing schemes, *Journal of Computational Physics* 77:439-471, 1988.
- Shu C., Liu N., Chew Y. T., A novel immersed boundary velocity correction–lattice Boltzmann method and its application to simulate flow past a circular cylinder, *Journal of Computational Physics* 226:1607-1622, 2007.
- Shu C., Niu X., Chew Y., Taylor series expansion and least squares-based lattice Boltzmann method: three-dimensional formulation and its applications, *International Journal of Modern Physics C* 14:925-944, 2003.
- Shu C., Wang Y., Teo C. J., Wu J., Development of Lattice Boltzmann Flux Solver for Simulation of Incompressible Flows, *Adv Appl Math Mech* 6:436-460, 2014.
- Shukla R. K., Tatineni M., Zhong X., Very high-order compact finite difference schemes on non-uniform grids for incompressible Navier–Stokes equations, *Journal of Computational Physics* 224:1064-1094, 2007.
- Slater J. W., ONERA M6 Wing, <http://www.grc.nasa.gov/WWW/wind/valid/m6wing/m6wing01/m6wing01.html>.
- Spalart P. R., Allmaras S. R., A one-equation turbulence model for aerodynamic flows, *AIAA 92-0439*, 1992.

- Steger J. L., Warming R., Flux vector splitting of the inviscid gasdynamic equations with application to finite-difference methods, *Journal of computational physics* 40:263-293, 1981.
- Su M., Xu K., Ghidaoui M., Low-speed flow simulation by the gas-kinetic scheme, *Journal of Computational Physics* 150:17-39, 1999.
- Sun Y., Shu C., Teo C. J., Wang Y., Yang L. M., Explicit formulations of gas-kinetic flux solver for simulation of incompressible and compressible viscous flows, *Journal of Computational Physics* 300:492-519, 2015.
- Suzuki K., Inamuro T., Effect of internal mass in the simulation of a moving body by the immersed boundary method, *Computers & Fluids* 49:173-187, 2011.
- Takahashi S., Nonomura T., Fukuda K., A numerical scheme based on an immersed boundary method for compressible turbulent flows with shocks: application to two-dimensional flows around cylinders, *Journal of Applied Mathematics* 10:1155-1175, 2014.
- Tang L., Progress in gas-kinetic upwind schemes for the solution of Euler /Navier–Stokes equations–I: Overview, *Computers & Fluids* 56:39-48, 2012.
- Tian C., Xu K., Chan K., Deng L., A three-dimensional multidimensional gas-kinetic scheme for the Navier–Stokes equations under gravitational fields, *Journal of Computational Physics* 226:2003-2027, 2007.
- Toro E. F., Riemann solvers and numerical methods for fluid dynamics: a practical introduction, *Springer Science & Business Media*, 2009.
- Turkel E., Radespiel R., Kroll N., Assessment of preconditioning methods for multidimensional aerodynamics, *Computers & Fluids*, 26 (1997) 613-634.

- Vassberg J. C., Tinoco E. N., Mani M., Brodersen O. P., Eisfeld B., Wahls R., Morrison J. H., Zickuhr T., Laflin K. R., Mavriplis D., Summary of the Third AIAA CFD Drag Prediction Workshop, *AIAA 0260*, 2007.
- Venkatakrishnan V., Convergence to steady state solutions of the Euler equations on unstructured grids with limiters, *Journal of Computational Physics* 118:120-130, 1995.
- Versteeg H. K., Malalasekera W., An introduction to computational fluid dynamics: the finite volume method, *Pearson Education*, 2007.
- Visbal M. R., Gordnier R. E., Galbraith M. C., High-fidelity simulations of moving and flexible airfoils at low Reynolds numbers, *Experiments in Fluids* 46:903-922, 2009.
- Visbal M. R., Shang J., Investigation of the flow structure around a rapidly pitching airfoil, *AIAA 27*:1044-1051, 1989.
- Wan D., Turek S., Direct numerical simulation of particulate flow via multigrid FEM techniques and the fictitious boundary method, *International journal for numerical methods in fluids* 51:531-566, 2006.
- Wang X., Yeo K., Chew C., Khoo B., A SVD-GFD scheme for computing 3D incompressible viscous fluid flows, *Computers & Fluids* 37:733-746, 2008.
- Wang Y., Shu C., Teo C., Development of LBGK and incompressible LBGK-based lattice Boltzmann flux solvers for simulation of incompressible flows, *International Journal for Numerical Methods in Fluids* 75:344-364, 2014.
- Wang Y., Shu C., Teo C., Wu J., An immersed boundary-lattice Boltzmann flux solver and its applications to fluid–structure interaction problems, *Journal of Fluids and Structures* 54:440-465, 2015.

- Wang Z., Fan J., Cen K., Immersed boundary method for the simulation of 2D viscous flow based on vorticity–velocity formulations, *Journal of Computational Physics* 228:1504-1520, 2009.
- Wang Z. J., Spectral (finite) volume method for conservation laws on unstructured grids. basic formulation: Basic formulation, *Journal of Computational Physics* 178:210-251, 2002.
- Wieting A., Holden M., Experimental Study of Shock Wave Interference Heating on a Cylinder Leading Edge, *AIAA* 87-1511, 1988.
- Wong P., Zingg D. W., Three-dimensional aerodynamic computations on unstructured grids using a Newton–Krylov approach, *Computers & Fluids* 37:107-120, 2008.
- Woodward P., Colella P., The numerical simulation of two-dimensional fluid flow with strong shocks, *Journal of Computational Physics* 54:115-173, 1984.
- Wu J., Shu C., Implicit velocity correction-based immersed boundary-lattice Boltzmann method and its applications, *Journal of Computational Physics* 228:1963-1979, 2009.
- Wu J., Shu C., An improved immersed boundary-lattice Boltzmann method for simulating three-dimensional incompressible flows, *Journal of Computational Physics* 229:5022-5042, 2010.
- Xu K., Gas-kinetic schemes for the unsteady compressible flow simulations, *VKI for Fluid Dynamics Lecture Series* 1998-03, 1998.
- Xu K., A gas-kinetic BGK scheme for the Navier–Stokes equations and its connection with artificial dissipation and Godunov method, *Journal of Computational Physics* 171:289-335, 2001.

- Xu K., He X., Lattice Boltzmann method and gas-kinetic BGK scheme in the low-Mach number viscous flow simulations, *Journal of Computational Physics* 190:100-117, 2003.
- Xu K., Huang J.-C., A unified gas-kinetic scheme for continuum and rarefied flows, *Journal of Computational Physics* 229:7747-7764, 2010.
- Xu K., Mao M., Tang L., A multidimensional gas-kinetic BGK scheme for hypersonic viscous flow, *Journal of Computational Physics* 203:405-421, 2005.
- Xu K., Martinelli L., Jameson A., Gas-kinetic finite volume methods, flux-vector splitting, and artificial diffusion, *Journal of Computational Physics* 120:48-65, 1995.
- Xu K., Sun Q., Yu P., Valid physical processes from numerical discontinuities in computational fluid dynamics, *International Journal of Hypersonics* 1:157-172, 2010.
- Yang J., Balaras E., An embedded-boundary formulation for large-eddy simulation of turbulent flows interacting with moving boundaries, *Journal of Computational Physics* 215:12-40, 2006.
- Yang L., Shu C., Wu J., A Hybrid Lattice Boltzmann Flux Solver for Simulation of 3D Compressible Viscous Flows, *Eighth International Conference on Computational Fluid Dynamics*, ICCFD8-2014-0228, 2014a.
- Yang L., Shu C., Wu J., A simple distribution function-based gas-kinetic scheme for simulation of viscous incompressible and compressible flows, *Journal of Computational Physics* 274:611-632, 2014b.
- Yang L., Shu C., Wu J., Zhao N., Lu Z., Circular function-based gas-kinetic scheme for simulation of inviscid compressible flows, *Journal of Computational Physics* 255:540-557, 2013.

- Yang L., Shu C., Wu J., A three-dimensional explicit sphere function-based gas-kinetic flux solver for simulation of inviscid compressible flows, *Journal of Computational Physics* 295:322-339, 2015.
- Yang L., Shu C., Wu J., Wang Y., Numerical simulation of flows from free molecular regime to continuum regime by a DVM with streaming and collision processes, *Journal of Computational Physics* 306:291-310, 2016.
- Yoon S., Jameson A., Lower-upper symmetric-Gauss-Seidel method for the Euler and Navier-Stokes equations, *AIAA* 26:1025-1026, 1988.
- Yuan R., Zhong C., Zhang H., An immersed-boundary method based on the gas kinetic BGK scheme for incompressible viscous flow, *Journal of Computational Physics* 296:184-208, 2015.
- Zhong H., Lee C., Su Z., Chen S., Zhou M., Wu J., Experimental investigation of freely falling thin disks. Part 1. The flow structures and Reynolds number effects on the zigzag motion, *Journal of Fluid Mechanics* 716:228-250, 2013.

Vita

NAME: SUN YU

DATE OF BIRTH: 9 MARCH, 1990

I was born in Zhejiang Province of China in 1990. I obtained my *B.Eng.* Degree in 2012 from Zhejiang University. In July of 2012, I came to Singapore and studied as a *Ph.D.* candidate in Department of Mechanical Engineering in National University of Singapore. The following are the publications related to my *Ph.D.* work:

1. **Y. Sun**, C. Shu, C.J. Teo, Y. Wang, L.M. Yang, Explicit formulations of gas-kinetic flux solver for simulation of incompressible and compressible viscous flows, *Journal of Computational Physics*, 300: 492-519 (2015).
2. **Y. Sun**, C. Shu, Y. Wang, C.J. Teo, Z. Chen, An immersed boundary-based gas-kinetic scheme for simulation of inviscid and viscous compressible flows, *Computers and Fluids*, In Press (2016).
3. **Y. Sun**, C. Shu, C.J. Teo, A switch function-based gas-kinetic scheme for simulation of inviscid and viscous compressible flows, *Advances in Applied Mathematics and Mechanics*, 8: 1-19 (2016).
4. L.M. Yang, C. Shu, Y. Wang, **Y. Sun**, Development of discrete gas kinetic scheme for simulation of 3D viscous incompressible and compressible flows, *Journal of Computational Physics*, 319: 129-144 (2016).

5. Y.L. Qiu, C. Shu, J. Wu, Y. Sun, L.M. Yang, T.Q. Guo, A boundary condition-enforced immersed boundary method for compressible viscous flows, *Computers and Fluids*, 136: 104-113 (2016).

6. Y. Wang, C. Shu, L.M. Yang, Y. Sun, On the immersed boundary-lattice Boltzmann simulations of incompressible flows with freely moving objects, *International Journal for Numerical Methods in Fluids*, In Press (2016).